Nancy-Université
Université Henri Poincaré

AVERTISSEMENT

Ce document numérisé est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.
Il est soumis à la propriété intellectuelle de l'auteur au même titre que sa version papier. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.
D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.
La Bibliothèque a pris soin d'adresser un courrier à l'auteur dans lequel elle l'informe de la mise en ligne de son travail. Celui-ci peut en suspendre la diffusion en prenant contact avec notre service.

➢ Contact SCD Nancy 1 : theses.sciences@scd.uhp-nancy.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

SC N2003 110B

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LÈS-NANCY CEDEX

Département de formation doctorale en informatique
UFR STMIA

École doctorale IAE+M Lorraine

Modèle multi-agent réactif pour la fusion de percepts : application à la localisation et au suivi en robotique mobile

THÈSE

présentée et soutenue publiquement le 29 octobre 2003

pour l'obtention du

Doctorat de l'université Henri Poincaré – Nancy 1
(spécialité informatique)

par

Franck Gechter

Composition du jury

Rapporteurs :	Marc Acheroy	PR Ecole Royale Militaire	Bruxelles
	Abder Koukam	PR UTBM	Belfort
Rapporteur interne :	Didier Galmiche	PR UHP Nancy I, LORIA	Nancy
Directeurs de thèse :	François Charpillet	DR INRIA, LORIA	Nancy
Examinateurs :		MC HDR UHP Nancy I, LORIA	Nancy
		IR, HDR IRIT	Toulouse

BIBLIOTHEQUE SCIENCES NANCY 1
D 095 158477 6

Laboratoire Lorrain de Recherche en Informatique et ses Applications - INRIA Lorraine - UMR 7503

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

Monsieur **GECHTER Franck**

DOCTORAT de l'UNIVERSITE HENRI POINCARE, NANCY 1

en INFORMATIQUE

VU, APPROUVÉ ET PERMIS D'IMPRIMER N° 867

Nancy, le 12 Novembre 2003

Le Président de l'Université

Cl. BURLET

Université Henri Poincaré, Nancy I
24-30 rue Lionnois - B.P. 3069 - 54013 NANCY Cédex
Tél. : 03 83 68 20 00 - Fax : 03 83 68 21 00

REMERCIEMENTS

Je tiens, en premier lieu, à remercier mes deux codirecteurs de thèse, François CHARPILLET, directeur de recherche et responsable de l'équipe MAIA, et Vincent CHEVRIER, maître de conférence, pour leur soutien et leur disponibilité, leurs conseils précieux, et surtout pour leur encadrement actif. Je leur suis, de plus, reconnaissant pour m'avoir donné la chance de réaliser ma thèse au sein de l'équipe MAIA.

Je remercie Abder KOUKAM, professeur à l'UTBM, et Marc ACHEROY, professeur à l'école royale militaire de Bruxelles, mes rapporteurs pour leur travail de relecture et leurs remarques pertinentes qui ont fait progresser ce document.

Je souhaite également remercier Pierre GLIZE, ingénieur de recherche CNRS, et Didier GALMICHE, professeur à l'UHP NANCY I, pour leurs conseils et leurs commentaires constructifs.

Mes remerciements vont également à Jean-Paul HATON, Professeur d'Université à l'UHP Nancy I, qui, le premier, m'a donné l'opportunité d'effectuer un stage au LORIA et d'intégrer par ce biais une équipe dynamique et ouverte.

Je tiens à remercier Didier FASS pour ses conseils avisés et ses encouragements lors de mes périodes de doute.

Je suis également reconnaissant envers Jean-Marie PIERREL, professeur d'Université à l'UHP Nancy I, Suzanne COLLIN, maître de conférence à l'ESIAL, et Jean-Pierre JACQUOT, maître de conférence à l'Université Nancy I, Isabelle DEBLED, maître de conférence IUFM de Lorraine pour leur encadrement et leurs conseils pédagogiques lors de mes années d'enseignement.

Je souhaite aussi remercier tous les permanents de l'équipe MAIA, pour leurs conseils et leur soutien.

Les compétences techniques d'Eric LUCCHESE, ingénieur expert INRIA, m'ont été précieuses et son aide m'a, en particulier, permis d'exploiter mes algorithmes et de les mettre en place sur les robots.

Je suis reconnaissant à Alain DUTECH, Laurent JEANPIERRE (et à tous les membres de l'équipe MAIA et à tous les stagiaires présents et passés) travaillant dans le bureau B 105 de m'avoir supporté ces trois dernières années, en particulier lors des interludes musicaux qui ont ponctué cette thèse.

En dernier lieu, je me dois de féliciter ma femme Céline et mes parents d'avoir su s'accommoder de mon caractère pendant toutes ces années d'étude.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cedex

Résumé

Ce travail s'inscrit dans un mouvement récent, les Systèmes Multi-Agents dont l'un des buts est de montrer que des systèmes intelligents complexes peuvent être élaborés à partir d'éléments réactifs simples et autonomes interagissant entre eux et avec leur environnement.

L'objectif de cette thèse est de mettre en oeuvre ce principe de conception de systèmes intelligents à un cadre applicatif complexe. Ainsi, nous voulons montrer qu'il est possible de construire un système multi-agents situés et réactifs devant résoudre un problème complexe concret inscrit dans une démarche d'interprétation du monde réel à travers ses caractéristiques et sa dynamique.

En ce qui concerne le cadre applicatif, nous avons choisi de traiter un problème récurrent en robotique mobile : la localisation. Nous voulons proposer une architecture permettant de localiser et de suivre une population hétérogène ou homogène de robots mobiles à l'aide de percepts fournis par des capteurs dont le type, le nombre et la position ne sont pas définis *a priori*.

L'enjeu d'une telle approche est la conception d'un dispositif intelligent d'un point de vue global à partir d'agents aux comportements réactifs simples. En effet, l'intelligence d'une tel système peut être considérée comme un phénomène émergent dû à l'organisation spatio-temporelle issue des interactions. Le point clef est alors la conception et la mise au point des comportements élémentaires afin d'obtenir un résultat émergent correspondant aux caractéristiques désirées.

Notre travail porte à la fois sur la conception d'une architecture adaptée : formalisation du problème via l'environnement, définition du processus de résolution, interprétation des résultats et sur son évaluation en simulation et dans un cadre réel.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

Abstract

The aim of this work is to show how a complex system, that has to resolve a real task, can be designed from single entities with small decision abilities. This approach stems from the multi-agent systems and, more particularly, from the reactive-agent systems.

As for the real task, we choose to deal with the localization and tracking problem. The localization is a tough, wide spread task in mobile robotics. It can be defined as the action to determine the position of an object considering a reference coordinate system.

By contrast with the classical algorithms, which usually stem from the signal or the image processing, we propose a reactive architecture based on interaction phenomenon inspired by Physics.

The aim is to design a localization and tracking device able to focus on a variable number of targets that are moving in a structured and partially known environment. Furthermore, the device has to be able to accomplish its task by using a variable number of sensors, static or mobile in the environment.

Our works deals with both the design of the reactive-agent architecture (choice of the world representation, definition of the resolution process, interpretation of the resulting spatio-temporal organization) and its application and evaluation with real robots.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

Table des Matières

CHAPITRE 1 : INTRODUCTION 19

1.1. Contexte général 21

1.2. Principes fondateurs de l'approche 23

1.3. Objectifs 24

1.4. L'architecture du système proposé 25

1.5. Plan du mémoire 27

CHAPITRE 2 : LE CADRE APPLICATIF 29

2.1. Généralités. 31
2.1.1. La localisation 31
2.1.2. Le suivi 32

2.2. Combien faut-il de capteurs pour localiser et suivre une cible ? 32
2.2.1. Un seul capteur est-il suffisant ? 33
2.2.2. Plusieurs capteurs pour une meilleure précision ? 35
2.2.3. Discussion 36

2.3. Deux points de vue différents pour la localisation 37
2.3.1. La localisation avec des capteurs embarqués 37
2.3.2. La localisation avec des capteurs fixes situés dans l'environnement 39

2.4. Le suivi ou la prise en compte du facteur temporel 40
2.4.1. Intérêt 40
2.4.2. Les méthodes usuelles 40
2.4.2.1. Le filtre de Kalman 41
2.4.2.2. Le flot optique 42

2.5. Conclusion 43

CHAPITRE 3 : LES SYSTEMES MULTI-AGENTS 45

3.1. Généralités 47

3.2. Les agents 48
3.2.1. Définition d'un agent 48

3.2.2. De l'agent vers le système d'agents 50
3.2.3. Société d'agents intelligents ou société intelligente d'agents 51
3.2.3.1. Agents cognitifs 52
3.2.3.2. Agents réactifs 52

3.3. L'environnement 53

3.4. Les interactions. 54

3.5. Organisation et auto-organisation 54

3.6. Quelques exemples de systèmes réactifs 55
3.6.1. Fourragement et tri de couvain (brood sorting) chez les fourmis 55
3.6.2. Construction de toile chez les araignées sociales 59
3.6.3. Les boids et le déplacement en essaim (flocking) 61
3.6.4. Algorithmes réactifs d'évitement de collision appliqués à la gestion du trafic aérien ... 63
3.6.5. The Particle Swarm Optimization 64

3.7. Discussion 65

CHAPITRE 4 : PROPOSITION D'UN PRINCIPE DE RESOLUTION POUR LA LOCALISATION 67

4.1. Principes 69
4.1.1. Généralités 69
4.1.2. Les différents éléments du système 70
L'environnement 71
Les agents 71
L'organisation 71
Les interactions 72

4.2. Description détaillée de l'architecture 73
4.2.1. Généralités 73
4.2.2. L'environnement 74
4.2.2.1. Représentation 74
4.2.2.2. Dynamique 76
4.2.3. Description des agents, de leurs comportements et des interactions 79
4.2.3.1. Etat interne 79
4.2.3.2. Les comportements des agents 82
4.2.4. Résolution des équations dynamiques 91

4.3. Résultats et fonctionnement 93

4.4. Discussion 94

CHAPITRE 5 : TESTS SIMULES : MISE EN ŒUVRE ET RESULTATS 97

5.1. Mise en œuvre **99**
5.1.1. Premiers essais avec Starlogo® 99
5.1.2. Utilisation de Madkit® 100

5.2. Tests et résultats **102**
5.2.1.1. Tests préliminaires 103
5.2.1.1.1. Tests de fonctionnement 103
5.2.1.1.2. Tests avec plusieurs générateurs de plots et avec plusieurs cibles 106
5.2.1.2. Tests de localisation 106
5.2.1.3. Tests de séparation 112
5.2.1.4. Tests de robustesse 116
5.2.1.5. Tests de suivi 118

5.3. Discussion et Commentaires **123**

5.4. Bilan **125**

CHAPITRE 6 : PASSAGE AU MONDE REEL : COMMENT INTRODUIRE LA PERCEPTION ?.....127

6.1. Concept d'unité perceptive **130**
6.1.1. Les capteurs physiques. 130
6.1.2. L'unité perceptive 131
6.1.2.1. Principe général 131
6.1.2.2. Intérêts pour notre architecture 132

6.2. Construction de deux unités perceptives simples **133**
6.2.1. L'analyse en composantes principales 133
6.2.2. Utilisation d'un filtre de Kalman 136

6.3. Conclusion **137**
6.3.1. Bilan 137
6.3.2. Discussion 137

CHAPITRE 7 : APPLICATION A LA LOCALISATION EN ROBOTIQUE MOBILE.....139

7.1. Le cadre expérimental **141**

7.2. Tests préliminaires **142**
7.2.1. Tests de continuité de la trajectoire 142
7.2.2. Test de suivi avec un seul robot 145
7.2.3. Tests avec plusieurs robots 146
7.2.4. Tests avec plusieurs unités perceptives 147

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

7.3. Tests comparatifs avec un filtre de Kalman 149
7.3.1. Objectifs et mise en œuvre 149
7.3.2. Le filtre de Kalman utilisé 149
7.3.3. Résultats à partir d'une source d'informations de référence 151
7.3.4. Résultats avec une source d'informations bruitée 152
7.3.5. Bilan 155

7.4. Tests de Fusion 156
7.4.1. Objectifs et mise en œuvre 156
7.4.2. A partir de plusieurs sources bruitées 160
7.4.3. A partir de trois filtres de Kalman différents 162

7.5. Conclusion 166

CHAPITRE 8 : CONCLUSION 169

8.1. Rappel du sujet 171

8.2. Bilan et Apports 171
8.2.1. Modèle 172
8.2.2. Application 172

8.3. Problèmes non encore résolus 174

8.4. Améliorations et perspectives 175
8.4.1. Du point de vue applicatif 175
8.4.2. Du point de vue du modèle 176

CHAPITRE 9 : BIBLIOGRAPHIE 179

CHAPITRE 10 : ANNEXES 195

10.1. Construire une représentation du monde : Utilisation des capteurs 197
10.1.1. But du chapitre 197
10.1.2. Les différents types de capteurs 198
10.1.2.1. L'odomètre 198
10.1.2.2. Les capteurs de proximité ou de contact 200
10.1.2.3. Les capteurs de distance 201
10.1.2.3.1. Le télémètre laser 204
10.1.2.3.2. Les sonars 206
10.1.2.3.3. Les capteurs infrarouges 206
10.1.2.4. Les dispositifs imageurs 207

10.2. L'exploitation des capteurs 209
10.2.1. But de ce chapitre 209
10.2.2. Cas mono-capteur 210
10.2.2.1. Méthodes couramment utilisées 210
10.2.2.1.1. Localisation d'objets dans une scène 211
Capteurs embarqués 211
Capteurs fixes dans l'environnement 213
10.2.2.1.2. Reconstruction d'une position dans l'espace 214
10.2.3. Cas multi-capteurs 216
10.2.3.1. Fusion de données 216
10.2.3.2. Algorithmes classiques en fusion de données 217
10.2.3.2.1. Fusion statistique 217
10.2.3.2.2. Fusion physique 217
Principe général 217
Un exemple de fusion physique:la stéréovision 218
10.2.4. Conclusion 222

10.3. La détermination des paramètres d'acquisition d'un dispositif imageur dans le cas d'un modèle simple. 223
10.3.1. Le modèle classique 223
10.3.2. Un modèle amélioré 227

10.4. L'intégration temporelle : exemples de la littérature 233
10.4.1. Introduction 233
10.4.2. Méthodes utilisées en traitement du signal 233
10.4.2.1. Filtre de Kalman 234
10.4.2.2. Flot Optique 234
10.4.2.3. Les approches Markoviennes 236

10.5. Le robot « Simplet » 238

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

Liste des illustrations

figure 1: Principe général de l'architecture 27
figure 2 : Deux exemples de télémètres laser couramment utilisés. A gauche un télémètre à laser à diode (Infra-Rouge) couplé à une caméra CCD (Nomadics Technologies). A droite, un télémètre à balayage (Sick) 33
figure 3 : Un exemple de caméra miniature (avec transmetteur vidéo HF) utilisée sur les robots de l'équipe MAIA 34
figure 4 : Extraction de primitives (arrêtes verticales) dans le but d'effectuer une reconstruction par stéréovision (d'après [Gechter99]) 38
figure 5 : Détection de la frontière de la rétine (d'après [Morris99]) 38
figure 6 : Exemples de mires utilisées [Baerveldt01] à gauche et [Horaud93b] au centre et [Gechter99] à droite. 40
figure 7 : Représentation de carte de mouvement de flot optique pour des mouvements de visage (d'après [DeCarlo00]). *A* : Translation vers l'avant, *B* : Translation à droite, *C* :Rotation vers la droite, *D* :Ouverture de la bouche. 42
figure 8 : Représentation d'un agent cognitif 52
figure 9 : Représentation d'un agent réactif 53
figure 10 : Recherche du plus court chemin à l'aide d'algorithme fourmis 57
figure 11 : Adaptation de la recherche du plus court chemin 58
figure 12 : Exemple de construction de toile. L'image de gauche représente le modèle biologique tel qu'il peut être observé dans la nature. L'image de droite représente le modèle informatique reconstruit à partir des comportements individuels proposés par les biologistes. 60
figure 13 : Détection de régions radiométriquement homogènes utilisant un algorithme inspiré des comportements des araignées sociales [Thomas01] (Chaque couleur correspond à une araignée, toutes les araignées se trouvant sur une toile unique). 61
figure 14 : Représentation des comportements utilisés pour le flocking. Séparation (A), Alignement (B) et Cohésion (C). Le disque centré sur l'agent correspond à sa limite de perception. D'après [Reynolds87]. 62
figure 15 : Les limitations de la perception d'un boid 62
figure 16 : Exemple de comportement d'évitement. (d'après [Zeghal94]) 64
figure 17 : Architecture de notre système multi-agents 73
figure 18 : Agents dans leur environnement 76
figure 19 : Prise en compte du déplacement des cibles. (A) Présence d'un plot dans l'environnement correspondant à une cible. (B) Sous le phénomène d'évaporation la hauteur du plot diminue. (C) La cible s'est déplacée formant ainsi un nouveau plot dans l'environnement (phénomène d'accumulation) L'altitude du premier plot continue à diminuer. (D) Le premier plot a complètement disparu, le nouveau commence à s'évaporer. 78
figure 20 : Espace-temps à deux dimensions incurvé par la présence d'une masse 80
figure 21 : schéma du comportement de répulsion 84
figure 22 : Schéma du comportement d'attraction 86
figure 23 : Les différentes zones du comportement d'attraction 87

figure 24 : Profil de la force d'attraction. 89
figure 25: Le générateur de plots 100
figure 26 : Structure de l'architecture de test 101
figure 27 : l'état stable fondamental 103
figure 28 : Convergence du système vers un état stable après l'apparition d'un percept 104
figure 29 : Retour du système dans l'état stable fondamental 104
figure 30 : Influence du nombre d'agents sur l'équilibre du système 105
figure 31 : Tableau montrant l'erreur et l'écart type de la localisation statique pour un monde de taille 200 x 200 et pour 150 agents. 108
figure 32 : Suivi de trajectoire rectiligne uniforme avec une vitesse relative égale à 0,5. La droite représente la trajectoire du plot et les points représentent les positions détectées des groupes. 109
figure 33 : Tableau montrant l'erreur et l'écart type de la localisation dynamique sur une trajectoire rectiligne pour un monde de taille 200 x 200 et pour 150 agents. 110
figure 34 : Suivi de trajectoire sinusoïdale avec une vitesse relative inférieure ou égale à 0,5 111
figure 35 : Mesures de l'erreur moyenne et de l'écart type pour une trajectoire sinusoïdale pour différentes portions du parcourt 111
figure 36 : Fusion de deux trajectoires et mesure de la distance de discrimination pour une distance de perception des agents de 25 pixels. (distance de discrimination mesurée : 19,41 pixels) 113
figure 37 : Evolution de la distance de discrimination en fonction de la distance d'attraction des plots 114
figure 38 : Séparation de deux trajectoires et distance de séparation pour une distance de perception de 45 (distance de séparation mesurée : 38,12) 115
figure 39 : Courbe évolution du nombre de plots détectés en fonction du coefficient d'évaporation. Le générateur utilisé produit un plot toutes les secondes. 117
figure 40 : Un exemple de suivi de trajectoire. En pointillés la trajectoire idéale suivie par les plots et en continu la trajectoire détectée par les agents de localisation. L'axe des abscisses et celui des ordonnées représentent les coordonnées des positions successives. 119
figure 41 : Erreur temporelle moyenne et écart type pour différentes vitesses relatives 120
figure 42 : Rapport de détection (nombre de groupes formés/ nombre plots générés) en fonction de la vitesse relative 120
figure 43 : Suivis de deux plots 122
figure 44 : Représentation schématique d'un capteur physique. 131
figure 45 : Représentation schématique d'une unité perceptive (ou capteur logique) . 132
figure 46 : Le modèle d'unité perceptive que nous utilisons. 133
figure 47 : Les minis robots de tests 142
figure 48 : Percepts discontinus fournis par l'ACP (position en X en ordonnée) au cours du temps lors d'une phase de tests réels 143
figure 49 : Ensemble des positions fournies par les dispositif réactif en fonction des percepts issus de l'ACP 144

figure 50 : Erreur et écart type sur la localisation en situation réelle 145
figure 51 : Schéma de l'architecture de tests pour des environnements de grande taille 148
figure 52 : Exemple de trajectoire de référence et de filtre de Kalman..... 150
figure 53 : Exemple de trajectoire de référence et de suivi avec les agents 151
figure 54 : Résultats comparatifs entre Kalman et le système multi-agent 152
figure 55 : Résultats obtenus avec un filtre de Kalman avec une trajectoire de référence bruitée 153
figure 56 : Résultats obtenus avec le dispositif multi-agent avec une trajectoire de référence bruitée 154
figure 57 : tableau comparatif entre le filtre de Kalman et le dispositf multi-agent pour une référence bruitée 154
figure 58 : tableau récapitulatif des résultats comparatifs entre notre approche et un filtre de Kalman..... 155
figure 59 : Les trois sources bruitées utilisées pour les tests de fusion 157
figure 60 : le premier filtre de Kalman utilisé 158
figure 61 : le second filtre de Kalman utilisé..... 159
figure 62 : le troisième filtre de Kalman utilisé..... 159
figure 63 : Résultats obtenus par le dispositif multi-agent à partir de trois sources d'informations bruitées 161
figure 64 : erreur moyenne et écart type obtenus à partir de trois sources bruitées 161
figure 65 : Résultats obtenus avec plusieurs filtres de Kalman comme sources d'informations (pondération uniforme)..... 162
figure 66 : Erreur moyenne et écart type avec trois filtres de Kalman pondérés de façon uniforme 163
figure 67 : Résultats obtenus avec plusieurs filtres de Kalman comme sources d'informations (pondération non uniforme)..... 164
figure 68 : Erreur moyenne et écart type avec trois filtres de Kalman pondérés de façon non uniforme 164
figure 69 : Résultats obtenus avec une fusion de trois filtres de Kalman et de la trajectoire de référence. 165
figure 70 : tableau récapitulatif des résultats obtenus en fusionnant différent percepts. 165
figure 71 : Exemples de commande de moteur par PWM, moteur à 100% (en haut) et moteur à 30% (en bas) 199
figure 72 : Principe de fonctionnement de l'odomètre à disque 199
figure 73 : Quelques exemples de roues d'odomètre 200
figure 74 : Principe de fonctionnement d'un capteur de distance 201
figure 75 : La spécularité 202
figure 76 : Phénomène de réflexions multiples 203
figure 77 : Phénomène de moyennage..... 204
figure 78 : Un modèle de télémètre laser utilisant une association caméra CCD/laser à diode (Nomadic Technologies) 205
figure 79 : Un modèle de télémètre laser à miroir (SICK) 205

figure 80 : La ceinture de sonars du robot "Gaston" (Nomad200 Nomadic Techologies) 206
figure 81 : Un exemple de caméra miniature (avec transmetteur vidéo HF) utilisée sur les robots de l'équipe MAIA 207
figure 82 : Localisation et identification d'essieux sur une chaîne de montage (Ford) 212
figure 83 : Extraction de primitives (arrêtes verticales) 212
figure 84 : Détection d'un visage dans une image (d'après [Cootes95]) 213
figure 85 : Détection de la frontière de la rétine (d'après [Morris99]) 213
figure 86 : Reconstruction à l'aide d'une seule caméra 214
figure 87 : Variation de le surface au sol correspondant à chaque pixel en fonction de l'éloignement par rapport à l'axe optique (en pointillé) 215
figure 88 : Exemples de mires utilisées [Baerveldt01] à gauche et [Horaud93b] au centre et [Gechter99] à droite. 221
figure 89 : Le modèle à trou d'épingle 224
figure 90 : Le modèle à sténopé (d'après [Horaud93b]) 225
figure 91 : La condition de parallélisme de la méthode Tsai. 227
figure 92 : Représentation de carte de mouvement de flot optique pour des mouvements de visage (d'après [DeCarlo00]). *A* : Translation vers l'avant, *B* : Translation à droite, *C* :Rotation vers la droite, *D* :Ouverture de la bouche. 235
figure 93 : Exemple de carte de mouvement β dans le cas où deux mobiles se croisent 236
figure 94 : Le robot "Simplet" (Premier Prototype) 238
figure 95 : Structure du processeur XA-S3 239
figure 96 : La caméra miniature et son transmetteur 240
figure 97 : Contrôle et perception à travers le réseau du LORIA 240
figure 98 : Architecture complète permettant le contrôle du robot "Simplet" 241
figure 99 : "Simplet" version 2.0 241

Chapitre 1 : Introduction

1.1. Contexte général

Ce travail s'inscrit dans un mouvement récent qui est le fondement principal de l'équipe MAIA (MAchines Intelligentes Autonomes) dont le but est de prouver qu'il est possible de réaliser des systèmes intelligents sans avoir obligatoirement recours à une manipulation de représentations symboliques complexes. Au contraire, ce mouvement prône le fait que des systèmes intelligents complexes peuvent être élaborés à partir d'éléments réactifs simples et autonomes interagissant entre eux et avec leur environnement. Ce type d'approche s'oppose par essence aux méthodes de conception de systèmes de type mono-agent ou de type systèmes répartis fortement supervisés qui peuvent parfois s'avérer trop rigides.

L'objectif de cette thèse est d'appliquer ce principe de conception de systèmes intelligents à un cadre applicatif complexe. Ainsi, nous voulons montrer qu'il est possible de construire un SMA (système multi-agents) constitués d'agents situés et réactifs devant résoudre un problème complexe concret inscrit dans une démarche d'interprétation du monde réel à travers ses caractéristiques et sa dynamique. Autrement dit, il s'agit de mettre en évidence la possibilité de construire un système fournissant une réponse intelligente à un problème complexe réel à partir d'entités simples, disposant de mécanismes de prise de décision limités.

L'application et le mode de conception que nous avons choisis se trouvent à la frontière entre plusieurs domaines scientifiques. Ainsi, la démarche conceptuelle s'inscrit au confluent de l'informatique, de la biologie du comportement, et de la mécanique classique. En outre, le fait que l'application concrète se place dans le domaine de la robotique mobile implique des apports scientifiques supplémentaires en provenance de la physique, de l'automatique ou encore du traitement du signal.

En ce qui concerne le cadre applicatif, nous avons choisi de traiter un problème délicat et récurrent en robotique mobile : la localisation. Ainsi, l'objectif que nous nous sommes fixé est de proposer une architecture permettant de localiser et de suivre une population hétérogène ou homogène de robots mobiles à l'aide de percepts fournis par des capteurs dont le type, le nombre et la position ne sont pas définis *a priori*.

Etant donné l'ambition de ce projet, il est impossible d'envisager la réalisation complète de l'ensemble de celui-ci dans le cadre d'une thèse. Ainsi, le travail qui est abordé dans ce document s'inscrit dans un projet scientifique à plus long terme au sein de l'équipe de recherche. Nous ne voulons aucunement proposer une nième méthode de localisation pour un robot mobile. Notre objectif est de fournir une approche innovante de conception d'un système pouvant être qualifié d'intelligent tout en étant basé sur des éléments simples.

Le principe général de la localisation est la comparaison entre une représentation immédiate (locale ou globale) de la configuration spatiale d'un environnement donné et une représentation de référence. La localisation peut être séparée en deux familles d'approches dont la différence se situe au niveau du point de vue adopté pour observer l'environnement. En effet, la localisation peut s'effectuer soit avec des capteurs externes non liés aux objets à localiser, soit avec des capteurs situés sur les entités mobiles. Dans

ce dernier cas, on parle d'auto-localisation. En ce qui concerne notre approche, nous avons choisi de traiter à la fois les deux situations. C'est-à-dire que notre proposition doit prendre en compte la possibilité d'utiliser des percepts fournis par des capteurs dont la position ou le point de vue sur l'environnement peut être quelconque.

Une fois la tâche de localisation accomplie, il faut pouvoir être en mesure de suivre les différentes cibles, c'est-à-dire de pouvoir construire un lien logique entre deux phases de localisation successives. Il s'agit donc d'effectuer une intégration temporelle des perceptions afin de construire une trajectoire spatiale pour toutes les cibles détectées.

De manière générale en robotique mobile, la localisation et le suivi concernent essentiellement des situations dans lesquelles un robot unique évolue dans un environnement dont on connaît les caractéristiques principales. S'il paraît difficile de s'affranchir, dans un premier temps, des connaissances sur l'environnement d'évolution des objets mobiles, la contrainte d'unicité de la cible semble délicate à conserver dans la majorité des applications. Aussi, nous avons décidé de ne pas nous limiter à la localisation et au suivi d'une cible unique, mais de rendre possible l'utilisation de notre proposition dans le cadre d'une application mettant en œuvre plusieurs robots ou cibles aux capacités perceptives et motrices différentes. En outre, notre proposition devra également prendre en compte la possibilité d'intervention d'objets non contrôlables dans l'environnement de test. Ainsi, notre approche doit permettre de localiser et de suivre des cibles dont on ne connaît pas le modèle dynamique *a priori*.

1.2. Principes fondateurs de l'approche

La démarche que nous proposons s'inscrit dans une approche de type écosystèmes synthétiques [Parunak99] ou intelligence en essaim [Bonabeau99] qui s'inspire en particulier des systèmes sociaux biologiques. Ce type d'approche peut être considéré comme une partie des Systèmes Multi-Agents réactifs [Ferber95]. Un SMA réactif est, en général, défini comme un système composé d'un ensemble d'entités simples, appelées agents, en interaction entre-elles et avec leur environnement qui possède une dynamique propre. Les agents d'un tel système sont dits réactifs, ils possèdent des comportements simples de type Stimulus/Réponse, et situés puisqu'ils possèdent une perception et une potentialité d'action limitée sur leur environnement proche.

L'intérêt de tels systèmes est leur capacité à posséder un comportement collectif complexe, qui peut être qualifié d'intelligent, au-delà de la simplicité individuelle de chacun des agents et des mécanismes d'interactions mis en place. Ce comportement collectif peut alors être considéré, du point de vue du système, comme une réponse à un problème posé par l'environnement. En effet, en l'absence de variation de l'environnement, le système va évoluer de façon auto-organisée vers un état stable représentant une solution au problème, conditionnée par la configuration de l'environnement.

La matérialisation d'une réponse est obtenue lorsqu'il y a équilibre entre les dynamiques du système : la dynamique propre de l'environnement rendant compte de l'évolution du problème d'une part et la dynamique comportementale des agents d'autre part. Les variations de l'environnement étant difficilement maîtrisables, l'obtention de la solution dépend essentiellement de l'évolution spatio-temporelle des agents.

Les SMA constitués d'agents réactifs possèdent un certain nombre d'avantages. Ils sont caractérisés par une fiabilité du système dans son ensemble. Ainsi, la perte d'un agent ne remet pas en cause l'accomplissement du processus collectif. En outre, ils possèdent une grande flexibilité et une faculté d'adaptation face aux variations des conditions de fonctionnement du système.

L'enjeu de notre approche est donc d'élaborer un système permettant de répondre à un problème complexe donné tout en respectant les contraintes de simplicité des comportements des agents et de leurs interactions tout en prohibant l'utilisation d'une représentation du but global du système dans chacun des agents. Ainsi, résoudre un problème donné revient à définir:

- l'environnement et sa dynamique de façon à ce qu'il rende compte de tous les paramètres utiles du problème.
- le processus de résolution sous la forme des comportements des agents et de leurs interactions de sorte que l'ensemble fournisse une solution au problème posé.

- l'interprétation du résultat collectif obtenu en fonction de la nature du problème à résoudre.

D'autre part, étant donné que nous voulons inscrire notre démarche dans un cadre applicatif réel, il faut ajouter une étape à la résolution du problème. En effet, il faut effectuer le lien entre le système de résolution et le monde réel. Ce lien s'effectuant par l'intermédiaire de l'environnement des agents, il s'agit donc de construire un mécanisme permettant de modifier celui-ci en fonction des variations du monde réel.

1.3. Objectifs

Même si nous avons choisi un cadre applicatif qui a induit un grand nombre de travaux dans la communauté scientifique, nous voulons montrer la faisabilité d'une application réelle à partir de principes tirés de l'intelligence artificielle collective et présenter les avantages d'un tel modèle non pas en terme de performances mais en terme de propriétés. Ainsi, notre objectif est de voir comment il est possible de concevoir un système réactif simple permettant de résoudre le problème de la localisation et du suivi et, de façon plus générale, comment aborder le problème de la fusion de données avec des agents réactifs.

Nous n'avons pas trouvé dans la littérature de dispositif de localisation utilisant des systèmes multi-agents constitués d'agents réactifs. Cependant, il existe des approches connexes, tels que la cartographie d'environnement [Grabowski99], [Sellem00] ou la fusion de données [Chong98] qui utilisent des approches s'inspirant des SMA mais à partir d'agents cognitifs spécialisés. Il est donc très intéressant d'évaluer les possibilités des SMA réactifs à traiter ce type de problèmes.

Par ailleurs, nous désirons que notre système possède des caractéristiques et des libertés par rapport à la configuration du dispositif physique. Nous voulons donc que :

- L'environnement réel d'évolution des cibles soit partiellement connu.
- Le système puisse utiliser des capteurs internes et/ou embarqués possédant une perception partielle de du monde.
- Le dispositif puisse disposer de plusieurs capteurs, ce qui implique la mise en œuvre d'un mécanisme de fusion d'informations.
- Les cibles ou leur modèle ne soient pas connus à l'avance : pas de connaissances *a priori* sur le modèle de déplacement (vitesse limite, inertie,...).
- Le système soit en mesure de localiser et de suivre plusieurs cibles de façon simultanée.

Ce dispositif doit, en outre, posséder certaines propriétés telles que :

- Le nombre de sources d'informations sur le problème doit être variable pendant l'exécution pourvu que celui –ci soit supérieur à un.
- Le système doit être robuste au niveau de la qualité et du nombre de sources d'informations. Par exemple, il doit pouvoir gérer les pannes totales ou partielles des sources sous la forme d'informations parasites, d'absence momentanée d'informations ou d'imprécisions. En outre, on s'attend à ce qu'il puisse fournir des trajectoires de cibles continues même en cas de panne.
- Le dispositif doit être capable de gérer un nombre de cibles quelconque et variable en cours de fonctionnement. Bien évidemment, le nombre de cibles maximal est directement lié aux ressources disponibles telles que le nombre d'agents présents dans le système. Il est naturel que le nombre de cibles à suivre ne puisse pas être supérieur au nombre d'agents présents.

Les contraintes que nous venons d'énoncer sont très fortes. Cependant, ces propriétés sont compatibles avec celles habituellement mises en évidence dans les SMA constitués d'agents réactifs. En effet, nous voulons que notre dispositif soit fiable (gestion de panne et des imprécisions des capteurs), flexible (pas de connaissance a priori sur les cibles et utilisation de capteurs de natures différentes) et adaptatif (nombre de sources d'informations et de cibles variable). Par conséquent, l'utilisation de SMA réactifs semble être particulièrement adaptée aux exigences que nous nous sommes fixées.

1.4. L'architecture du système proposé

L'architecture proposée peut être décrite selon trois étapes fondamentales : l'environnement représentatif du problème à résoudre, le processus de résolution réactif et l'organisation résultante dont l'interprétation fournit une solution au problème.

Dans notre cas, l'environnement est une représentation discrète du monde réel sous la forme d'un ensemble d'états représentants chacun une position possible d'une cible dans l'environnement réel. La dynamique de l'environnement doit rendre compte des propriétés de celle du monde extérieur. Le lien entre les deux dynamiques se fait par accumulation de percepts dans les différents états. Un percept est un couple probabilité/position fourni par une unité perceptive, entité utilisant un ensemble de capteurs et de traitements adaptés pour construire une représentation du monde à partir d'une observation. Chacun de ces percepts est donc intégré dans l'environnement des agents sous la forme d'un plot, point localisé dans l'une des cases de l'environnement discrétisé et possédant une valeur de confiance que l'on peut comparer à une altitude. Si plusieurs percepts correspondent au même état discret ceux-ci y sont alors accumulés.

L'addition de ces percepts se traduit par une augmentation de l'altitude (valeur de confiance) du plot.

Le processus de résolution est directement lié aux comportements individuels des agents et aux interactions de ceux-ci, entre eux et avec leur environnement.

Le modèle comportemental des agents est inspiré à la fois du vol en essaim (« flocking ») .[Reynolds87] et du comportement de poursuite de la reine chez les abeilles. L'ensemble des comportements dont on a doté nos agents a pour but une focalisation de ceux-ci sur les plots se trouvant à proximité et une répartition spatiale homogène dans les régions où il n'y a pas de percept. Comme dans tous les systèmes auto-organisés, pour obtenir un résultat cohérent, il faut la conjonction de deux principes comportementaux antagonistes [Müller98], l'un participant à l'amplification du phénomène d'organisation et l'autre participant à sa régulation (amplification et feedback négatif). Dans notre système, les agents subissent des interactions impliquant à la fois un comportement d'attraction pour les plots de leur environnement et un comportement de répulsion entre eux.

Conformément à ce que l'on trouve dans la plupart des approches de ce type, le système possède deux dynamiques. La première est la dynamique de l'environnement qui est liée à la fois à celle du monde réel et à la fréquence des percepts fournis par les ensembles capteurs/traitements. La seconde dynamique est celle des agents contrôlés par les items comportementaux précédemment cités.

L'organisation qui émerge de cet ensemble est la focalisation des agents sur les plots formant ainsi des groupes. Cette formation de groupe matérialise la présence d'une cible sur une position du monde réel correspondant à l'état sur lequel les agents se sont focalisés.

Dans notre approche, le suivi peut être considéré comme une succession de localisations temporellement cohérentes. Il est obtenu de façon indirecte grâce à la dynamique des percepts dans l'environnement. En effet, le déplacement des percepts induit un déplacement des groupes d'agents. Les localisations successives de ces groupes permettent de construire une trajectoire supposée pour la cible réelle.

L'architecture globale que nous proposons est donc la suivante :

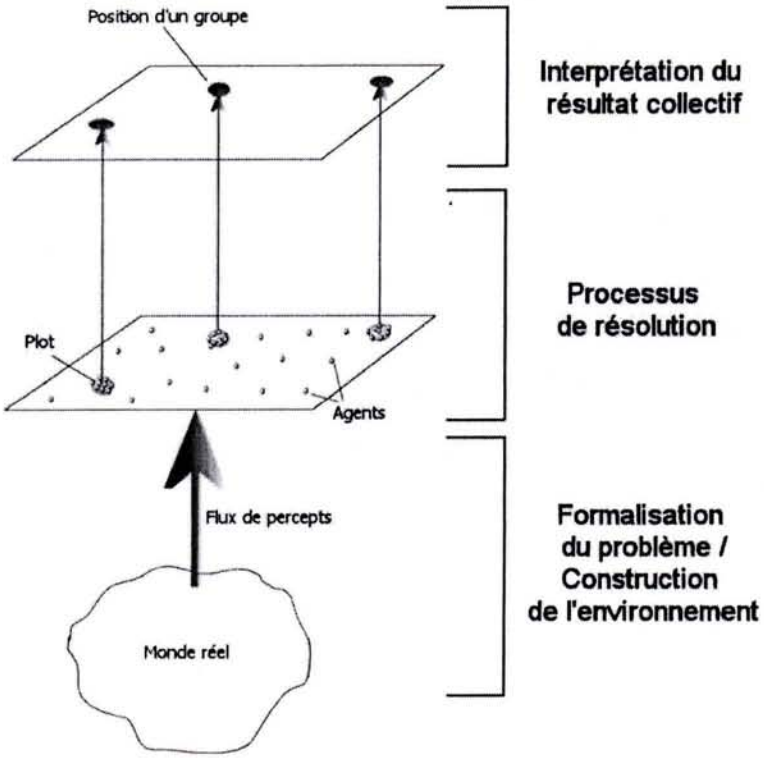

figure 1: Principe général de l'architecture

Ainsi, les points clefs de notre approche sont les suivants :

- Construction d'un environnement pour les agents de façon à ce qu'il représente le plus fidèlement possible le monde réel ainsi que sa dynamique propre.
- Elaboration des mécanismes comportementaux des agents, de façon à ce que l'ensemble du système fournisse une réponse au problème posé par l'environnement.

1.5. Plan du mémoire

Notre document va se composer essentiellement de deux parties fournissant chacune une réponse à l'un des enjeux de notre approche.

Après une présentation du cadre applicatif que nous envisageons, nous allons exposer les bases du contexte scientifique dans lequel nous inscrivons notre approche. Ainsi, dans ce second chapitre, nous allons expliquer les notions d'agents, de systèmes multi-agents et d'organisation. Nous allons également profiter de ce chapitre pour

exposer brièvement quelques unes des inspirations qui nous ont aidées dans la définition de notre architecture.

Une fois ces concepts mis en place, nous allons détailler notre proposition tant au niveau du modèle choisi qu'à la lumière des objectifs que nous nous sommes fixés. Cette description va ensuite déboucher sur une série de tests simulés permettant de mettre en évidence les caractéristiques et les limitations de notre proposition.

Etant donné, que notre objectif est non seulement de construire une architecture à base d'entités réactives simples mais également de tester celle-ci dans un cadre applicatif réel. Nous allons nous concentrer dans les chapitres suivants sur la mise en œuvre de notre proposition avec des robots. Pour cela, il s'agit, tout d'abord, de voir comment il est possible d'intégrer des processus perceptifs du monde réel dans notre architecture de localisation. Enfin, une fois le problème de la prise en compte des perceptions résolu, nous allons procéder à quelques tests réels non seulement pour évaluer les performances de notre système en situation concrète mais également pour évaluer la viabilité d'une telle approche.

Enfin, nous conclurons sur notre proposition en détaillant, dans un premier temps, les objectifs atteints par rapport à ceux que nous nous sommes fixés. Dans un second temps, nous exposerons les limitations de notre approche. Ensuite, nous aborderons les améliorations possibles et les perspectives induites par notre démarche conceptuelle.

Chapitre 2 : Le Cadre Applicatif

Le but de ce chapitre est de définir le cadre applicatif que nous avons choisi tout en effectuant un rapide aperçu des méthodes généralement employées pour accomplir une tâche de localisation et de suivi. Notre but ici n'est pas de faire un état de l'art exhaustif mais de montrer quelles sont les différentes manières d'aborder le problème dans le cas où les capteurs sont liés à l'environnement et liés à l'objet à localiser.

2.1. Généralités.

2.1.1. La localisation

La localisation est un terme générique très employé dans de nombreux domaines de la recherche tels que la médecine, l'astrophysique ou la géologie. Aussi, la définition de ce terme est étroitement liée au champ d'application dans lequel il est utilisé. Les domaines qui nous intéressent plus particulièrement, dans le cadre de cette thèse, sont la robotique mobile et la surveillance de locaux structurés. Il nous faut donc une définition de la localisation suffisamment large pouvant être applicable à ces deux domaines mais aussi suffisamment précise afin de servir de base à notre travail. Ainsi, dans la suite de ce document, lorsque nous parlerons de localisation, nous nous référerons à la définition suivante :

Définition : La localisation correspond à l'action de trouver la position d'un objet, mobile ou fixe, par rapport à un référentiel connu.

Cette définition, très générale, s'applique à la fois à la robotique mobile et à la surveillance de locaux. Les objets mobiles ou fixes seront les robots, les individus et/ou les obstacles présents dans l'environnement. Le référentiel pour la localisation pourra être lié soit à l'environnement, la position d'un objet pourra alors être donnée sous forme de coordonnées 2D, ou à l'un des objets présents dans l'environnement. Dans le cadre de la robotique mobile, par exemple, il peut être judicieux d'utiliser un référentiel lié à l'un des robots pour faciliter l'exécution d'algorithmes d'évitement d'obstacles. En effet, il est souvent plus aisé de déterminer une trajectoire d'évitement locale liée à la position relative de l'objet, par exemple, plutôt que de construire une trajectoire globale dans le référentiel de l'environnement et de la "convertir" de façon à donner un plan local au robot qui doit éviter l'obstacle.

Dans le contexte posé par la définition choisie, les applications possibles de la localisation sont très nombreuses. Outre la robotique mobile et la surveillance d'individus, que nous allons développer dans la suite de ce document, nous pouvons noter, par exemple, les applications concernant le transport et les applications militaires

de recherche de cibles. Parmi celles qui concernent le domaine des transports, l'application qui se rapproche le plus du cadre de cette thèse est la localisation de véhicules automobiles par rapport au référentiel terrestre pour effectuer une tâche d'aide à la navigation.

2.1.2. Le suivi

Le principe général du suivi est de pouvoir intégrer, au cours du temps, l'évolution des localisations d'une même cible. Cela revient donc à prendre en compte, en plus de la dimension spatiale présente dans la localisation, la dimension temporelle du processus observé. Ainsi, il est donc possible de définir le suivi de la façon suivante :

Définition : Le suivi peut être considéré comme l'action de construire une trajectoire cohérente dans l'espace et dans le temps à partir d'une succession de localisations.

Ainsi, suivre un objet revient à élaborer des liens temporels entre plusieurs localisations successives proches les unes des autres. En général, la construction d'une trajectoire nécessite une connaissance, même partielle, de la dynamique du processus observé. Néanmoins, il est possible de se passer d'une partie de cette connaissance en effectuant des hypothèses logiques concernant l'évolution entre deux localisations successives.

2.2. Combien faut-il de capteurs pour localiser et suivre une cible ?

Lorsque l'on veut construire un dispositif devant interpréter les observations d'un phénomène réel, il est nécessaire de se demander combien de sources d'informations sont indispensables pour obtenir une interprétation aussi précise que possible tout en limitant le coût d'exploitation des sources d'informations supplémentaires

D'une façon générale, plus l'on dispose d'informations sur le déroulement d'un processus donné, plus il est aisé d'en trouver les caractéristiques essentielles. Cependant, même s'il est toujours intéressant de disposer d'un grand nombre d'informations, il est souvent difficile d'intégrer celles-ci de manière pertinente tout en respectant des contraintes de temps raisonnables vis-à-vis de la nature du processus observé. En outre, un très grand nombre de sources d'informations n'est pas forcément synonyme d'un résultat de meilleure qualité. En effet, certaines sources d'informations peuvent être

redondantes. Aussi, il est judicieux de choisir correctement les capteurs et les traitements utilisés de façon à obtenir des observations complémentaires au sens mathématique du terme.

Dans le cas de la localisation et du suivi, il a donc fallu d'abord étudier la possibilité de n'utiliser qu'un seul capteur.

2.2.1. Un seul capteur est-il suffisant ?

Afin d'obtenir des résultats pertinents et précis lors de l'utilisation d'un capteur unique, il faut que ce dernier soit suffisamment riche pour en extraire le maximum d'informations. Ce principe fondamental écarte de ce type d'applications les capteurs très bruités et perceptivement limités tels que les sonars ou encore les capteurs infrarouges. Les seuls capteurs envisageables dans le cadre d'une localisation à capteurs unique sont le télémètre laser et la caméra. Nous remarquons au passage, que ces dispositifs sont en fait des capteurs multiples puisque le télémètre laser effectue plusieurs mesures d'une même scène en utilisant des dispositifs mécaniques alors que la caméra est en fait une structure composée de plusieurs photosites pouvant chacun être considéré comme un capteur à part entière.

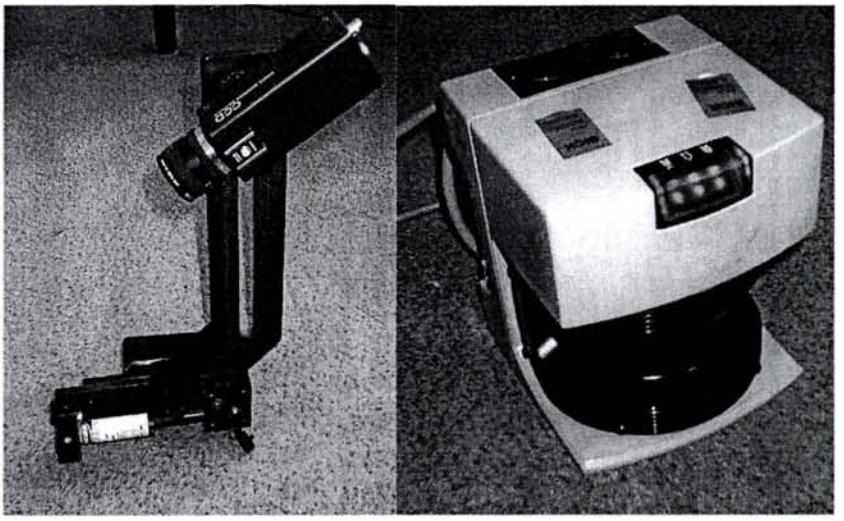

figure 2 : Deux exemples de télémètres laser couramment utilisés. A gauche un télémètre à laser à diode (Infra-Rouge) couplé à une caméra CCD (Nomadics Technologies). A droite, un télémètre à balayage (Sick)

Les télémètres (cf. figure 2) fournissent en général des informations de bonne qualité, cependant leur coût et leur masse ne les prédisposent pas à toutes les plates-formes robotiques mais plus particulièrement aux structures les plus imposantes. Les

informations qu'ils fournissent sont beaucoup plus précises que celles issues des capteurs de type sonars ou infrarouges.

Les applications utilisant un télémètre laser sont souvent basées sur une cartographie de l'environnement effectuée par accumulation des acquisitions lors des déplacements du capteur [Wallner97], [Crowley98]. D'autre part, un télémètre laser fixe peut également être utilisé pour la localisation en comparant une acquisition dépourvue de présence (acquisition de référence) avec les acquisitions courantes. Dans tout les cas, une localisation précise ne peut s'effectuer qu'en fournissant une métrique aux acquisitions qui est étroitement liée à la métrique du monde réel. Ce lien entre les différentes métriques s'appelle l'étalonnage (ou calibration) et se retrouve également lors de l'utilisation de caméras.

L'utilisation d'une caméra pour la localisation est beaucoup plus courante. Ceci s'explique non seulement par le faible coût d'achat d'une caméra par rapport à un télémètre, mais également par la plus grande facilité d'interprétation des informations fournies.

Depuis quelques années, la taille (cf. figure 3) et surtout les prix des dispositifs imageurs ont fortement diminués, ce qui tend à imposer les caméras comme capteur essentiel à toute application robotique ou domotique. Bien qu'il soit fort intéressant de disposer d'une caméra, l'apport d'informations important qu'elle fournit va de pair avec une augmentation indispensable des ressources de traitement. En outre, bien qu'il soit naturel pour l'être humain d'interpréter les informations visuelles, il n'est pas aussi facile de formaliser cette interprétation sous forme algorithmique. Néanmoins, la richesse des informations fournies par les dispositifs imageurs en font un capteur indispensable dès lors que l'on veut accomplir des tâches complexes et précises. C'est donc naturellement que nous nous sommes concentrés sur l'utilisation des caméras dans le cadre de notre application.

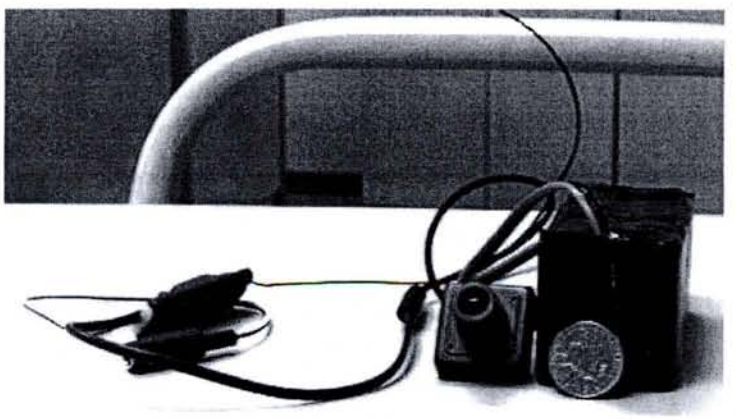

figure 3 : Un exemple de caméra miniature (avec transmetteur vidéo HF) utilisée sur les robots de l'équipe MAIA

Il reste maintenant à détailler les méthodes destinées à l'utilisation d'une ou de plusieurs caméras (cf. paragraphe 2.3)

Les problèmes inhérents à l'utilisation d'un capteur unique peuvent être nombreux. Parmi ceux-ci, nous pouvons citer:

- L'impossibilité de pallier une panne ou une déficience éventuelle du capteur.
- La difficulté à construire un dispositif de raisonnement indépendant de la nature du capteur utilisé. Ceci se traduit souvent par un manque de généricité de l'algorithme d'interprétation construit.
- L'impossibilité de trouver un capteur pertinent dans toutes les plages d'utilisation désirées. En effet, il n'existe pas de capteur universellement idéal. Aussi, le choix d'utiliser un capteur unique limite obligatoirement l'application à une plage de fonctionnement restreinte.

Suite à ces constatations, l'utilisation de plusieurs capteurs semble plus adaptée aux contraintes que nous nous sommes fixées.

2.2.2. Plusieurs capteurs pour une meilleure précision ?

Dès lors que l'on doit utiliser plusieurs capteurs, il est nécessaire d'avoir recours à une fusion de l'information fournie. Le choix de la méthode de fusion à employer est directement lié à la nature des sources d'informations (information numérique, information symbolique,...) et à la forme des résultats que l'on veut obtenir.

La fusion de données peut être définie, de façon générale, comme étant l'intégration de données provenant de sources multiples (capteurs, modules de décision, modules d'interprétation,...) dans le but de construire une interprétation pertinente et spécifique sur une entité, une activité ou un événement. Cette interprétation doit en particulier prendre en compte les incohérences entre les différentes sources et fournir un résultat plus précis et plus riche par rapport à ce que l'on pourrait obtenir en exploitant séparément les informations initiales.

D'un point de vue formel, le but de la localisation est de construire, à partir de plusieurs sources d'information {S1,S2,...,Sn}, une source de connaissance Sc. Cette source doit rassembler l'information contenue dans les sources initiales de façon à être "meilleure" du point de vue quantitatif, qualitatif, confiance ou sémantique (information d'un plus haut niveau d'abstraction). Les sources d'informations peuvent être concurrentielles ou complémentaires et fournir des informations qui, une fois combinées, vont permettre de construire une connaissance meilleure.

Il existe bien évidemment de nombreuses méthodes pour réaliser de la fusion de données. Parmi ces méthodes nous pouvons citer entre autres :

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

- Les méthodes de fusion statistique dont le principe est de construire une hypothèse de décision à partir de données, fournies par des sources d'information différentes, en prenant en compte les connaissances *a priori* sur le processus d'observation. Nous pouvons classer dans cette catégorie des méthodes telles que l'inférence classique, la méthode de Dempster-Shafer [Bellot02] et l'approche Bayesienne [Lopez-Orozco98].
- Les méthodes de fusion physique dont le but est d'assembler un ensemble de sources d'informations ou d'observation dont les caractéristiques sont définies à partir d'un modèle précis. En d'autres termes, il s'agit d'adapter l'information issue des sources initiales pour qu'elles correspondent à un modèle physique prédéfini. La stéréovision rentre dans le cadre de ce type de fusion. En effet, il s'agit alors de fusionner la position d'amers (points dans les images représentant la même source initiale) dans des vues différentes de la même scène dans le but de reconstruire une position dans l'espace ([Ayache89]).
- Les méthodes de fusion à base d'agents telles que la méthode du tableau noir ([Mees00]) dont le principe est d'utiliser des agents, en général cognitifs, pour traiter de façon séquentielle ou de façon parallèle les informations disponibles.

2.2.3. Discussion

Etant donné les limitations induites par l'utilisation d'un capteur unique, il est impossible, à la vue des contraintes que nous nous sommes fixées, d'envisager une telle solution pour notre architecture. Il est indispensable d'envisager l'utilisation de plusieurs capteurs. Dans la mesure où nous ne connaissons pas de dispositif de fusion utilisant des SMA réactifs, il va falloir construire des mécanismes afin de réaliser ce processus.

Les méthodes de fusion à base d'agents n'utilisent, en général, que des agents cognitifs spécialisés ([Sellem00], [Grabowski99], [Chong98]). Le choix d'utiliser des agents réactifs est donc un véritable défi puisqu'il faut réussir à construire un mécanisme de fusion simple devant manipuler des sources d'informations parfois complexes dans le but de construire une connaissance suffisamment riche pour traiter du problème de la localisation. L'élaboration d'un mécanisme de fusion réactif est un des points clefs de notre approche et ceci indépendamment du cadre applicatif fixé.

2.3. Deux points de vue différents pour la localisation

La localisation, telle que nous la concevons, peut être scindée en deux catégories distinctes : la localisation utilisant des capteurs liés à l'environnement et la localisation utilisant des capteurs embarqués. En effet, les méthodes usuellement employées pour résoudre ce problème différent sensiblement en fonction de la position des capteurs. Il est bien évident que la localisation ne peut pas être abordée de la même façon si les capteurs sont mobiles et se trouvent sur l'un des objets à localiser ou si ceux-ci sont fixes par rapport au référentiel de l'environnement.

Etant donné la nature des principales applications que nous abordons, nous devons nous intéresser aux deux catégories de localisation. Dans le cadre de la robotique mobile, il est tout à fait légitime d'utiliser les capteurs embarqués disponibles pour réaliser la localisation alors que, dans le cadre de la surveillance de locaux, il paraît difficile de placer un capteur sur les personnes à localiser.

Dans la suite, nous allons détailler les méthodes inhérentes à chacune des deux approches.

2.3.1. La localisation avec des capteurs embarqués.

La localisation avec des capteurs embarqués ou ***l'auto-localisation*** peut à son tour se scinder en deux familles d'approches. Soit l'objectif à atteindre est de déterminer la position de l'objet mobile équipé de capteurs par rapport au référentiel lié à l'environnement dans lequel il évolue, soit il s'agit de déterminer la position de ce même objet par rapport à un référentiel lié aux obstacles observés. Nous allons appeler la première approche la ***localisation absolue*** et la seconde la ***localisation relative***.

Le principe général de l'***auto-localisation absolue*** est de trouver, à partir d'une représentation locale de l'environnement proche de l'entité mobile fournie par les capteurs embarqués, la position de la carte globale de l'environnement qui correspond le mieux à cette représentation locale. Cette recherche de la meilleure position possible de l'environnement peut se faire de façon systématique par des méthodes adaptées telles que l'Analyse en Composantes Principales (ACP) ([Pourraz99], [Gechter00a]), les réseaux de neurones ([Rizzi01]), ou de façon orientée notamment lorsque que l'on connaît la position de départ a priori de l'entité mobile et nécessite dans tous les cas un modèle, sous la forme d'une base d'images par exemple, de l'environnement de référence. En outre, il est possible d'améliorer considérablement les résultats de la localisation en prenant en considération des informations sur la dynamique propre du robot à localiser. Pour cela, on utilise en général des méthodes de prédiction de mouvement à base de Processus de Markov ([Simmons95], [Fox98], [Laroche99], [Gechter00a], [Thrun99]) , ou de voisinages dynamiques ([Gechter00b]).

La ***localisation relative*** possède une différence importante par rapport à la localisation absolue : l'absence de représentation de référence permettant une triangulation dans le référentiel lié à l'environnement. Dans ce cadre, le but est de trouver la position des objets visibles de l'environnement par rapport au référentiel de l'entité à localiser. Pour accomplir cette tâche, il existe de nombreux algorithmes, en général extraits du monde du traitement d'images. D'une manière générale, les dispositifs de localisation relative sont composés de deux étapes. La première consiste en l'extraction de primitives, formes géométriques simples, représentatives des objets présents dans l'image. Pour cela, on dispose de nombreux algorithmes tels que la détection de contours ou d'arrêtes par des méthodes de Sobel, de Gradient ou de Laplaciens ([Baerveldt01], [Gechter99]) (cf. figure 4) ou encore à l'aide de contours actifs ([Cootes95], [Morris99]) (cf. figure 5) prenant en compte des caractéristiques géométriques des objets à localiser.

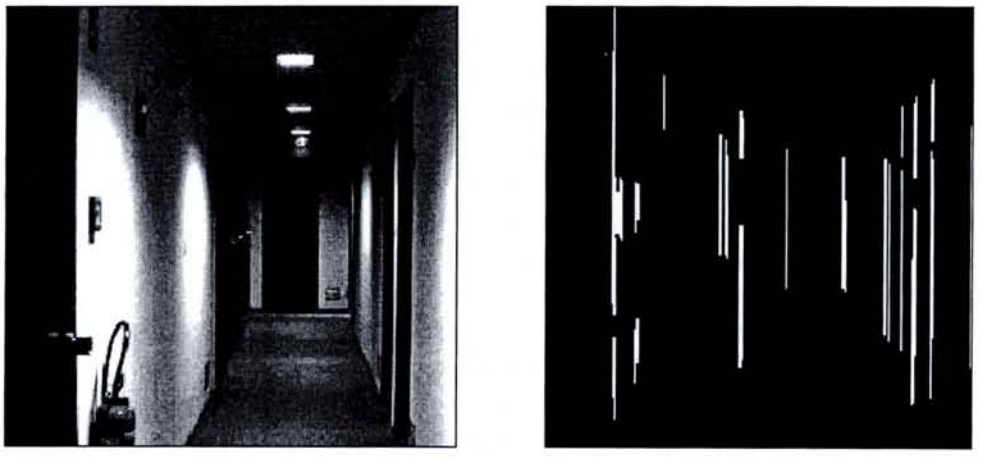

figure 4 : Extraction de primitives (arrêtes verticales) dans le but d'effectuer une reconstruction par stéréovision (d'après [Gechter99])

figure 5 : Détection de la frontière de la rétine (d'après [Morris99])

La seconde étape consiste en la reconstruction spatiale de la position de différents objets. Parmi les algorithmes les plus répandus nous pouvons citer, les algorithmes de stéréovision ([Ayache89], [Baerveldt01], [Zingaretti98], [Laubach99], [Murray96a], [Murray96b], [Olson97a], [Olson97b], [Gechter99] ou [Henkel00]) ou ceux utilisant des télémètres laser comme outils de perception ([Wallner97], [Crowley98]).

Le but d'une telle localisation est en général la construction de plans locaux permettant d'éviter les obstacles éventuels.

2.3.2. La localisation avec des capteurs fixes situés dans l'environnement.

Le problème de la localisation effectuée avec des capteurs fixes situés dans l'environnement est généralement résolu avec des algorithmes similaires à ceux employés en localisation relative tels que la Stéréovision ou le Pattern Matching. La principale difficulté rencontrée dans le cadre de ces méthodes est la caractérisation du lien existant entre la scène réelle et sa représentante observée par un capteur. Le problème du lien entre le monde réel et l'observation qui en est faite se retrouve également dans la localisation relative. Par contre, ce problème n'est pas directement explicite en auto-localisation de par la nature des méthodes utilisées. En effet, la plupart d'entre elles fonctionnent sur le principe d'une association entre une acquisition à un instant **t** et un ensemble d'acquisitions de référence effectuées avec le même capteur. Le lien ne se fait donc pas directement avec le monde réel mais avec une de ses représentations. Les deux représentations étant construites de la même façon et avec les mêmes paramètres d'acquisition, le lien se fait de façon implicite. Toute la difficulté de la localisation à l'aide de capteurs externes est d'extraire des informations issues des observations pouvant être interprétées dans le monde réel.

Certains types de capteurs disposent d'une interface permettant d'effectuer, directement ou indirectement, le lien entre les observations et le monde réel. C'est le cas notamment des télémètres laser qui fournissent une évaluation des distances les séparant des objets présents dans la scène, des sonars et des capteurs infra-rouges qui fournissent également une estimation de la distance de l'obstacle le plus proche. La tâche de localisation est alors une tâche d'interprétation des distances fournies connaissant la position des capteurs dans l'environnement.

Dans le cadre de capteurs plus généraux comme les caméras, il est nécessaire de connaître parfaitement le processus d'acquisition afin de pouvoir l'inverser et ainsi retrouver la position de l'objet observé dans le monde à partir d'une acquisition. La connaissance du processus d'acquisition s'obtient en paramétrant les différentes étapes de passage entre le réel et le percept. Cette paramétrisation est, dans la majorité des situations, appelée étalonnage ou calibration. Pour réaliser cette calibration, on a en général recours à une mire, agencement périodique de balises facilement identifiables (cf. figure 6) (pour plus de détails sur les principes de l'étalonnage d'une caméra le lecteur pourra se reporter aux annexes).

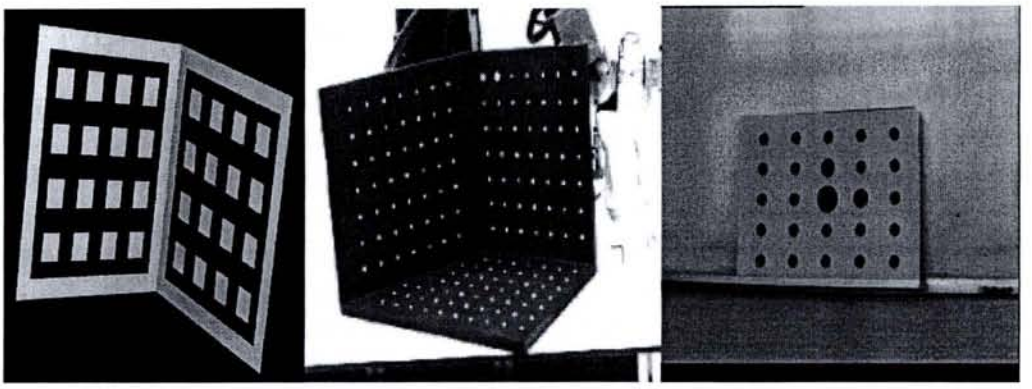

figure 6 : Exemples de mires utilisées [Baerveldt01] à gauche et [Horaud93b] au centre et [Gechter99] à droite.

De la même façon qu'en localisation relative, les algorithmes utilisés sont essentiellement issus du traitement d'image ([Horaud93a], [Pratt78]).

2.4. Le suivi ou la prise en compte du facteur temporel

2.4.1. Intérêt

L'intégration du temps dans les paramètres de perception est indispensable à la fois pour prédire les déplacements des cibles à localiser et suivre, et pour assurer la continuité dans les perceptions.

L'intérêt de connaître les déplacements du robot entre deux positions à localiser est double. Il s'agit, d'une part, de limiter les recherches dans les positions possibles du robot en prenant en compte le déplacement et la ou les positions précédemment évaluées. D'autre part, ceci permet de conserver un historique des déplacements et donc de construire une trajectoire dans l'environnement.

2.4.2. Les méthodes usuelles

L'ajout de la prise en compte des informations sur le déplacement du robot peut être réalisé au travers de plusieurs méthodes.

Dans un premier temps, il est envisageable d'utiliser des capteurs supplémentaires pour estimer les déplacements réels entre deux actions de localisation successives. Dans

ce cadre, les capteurs odométriques ou gyroscopiques semblent être les plus adaptés puisqu'ils sont destinés à évaluer le nombre de rotations des roues pour les premiers et à estimer les accélérations du robot pour les seconds.

Une autre solution consiste à modéliser la cinématique du robot mobile, soit de façon continue en posant les équations différentielles du mouvement et en les intégrant, soit de façon discrète en liant les actions du robot aux changements de position possibles.

Enfin, dans ces deux cas, il est possible d'intégrer les aléas de déplacement en introduisant des incertitudes probabilistes, soit dans l'expression d'équations différentielles, soit dans les transitions entre les positions possibles.

Les approches continues sont généralement employées dans la détection, par traitement systématique, d'entités mobiles dans une succession d'acquisitions, suivi d'un véhicule dans une séquence d'images par exemple ([Koller93], [Koller94] et [Marchand01]). Les approches discrètes sont, quant à elles, réservées aux approches dont la localisation consiste en la détermination d'un couple position/orientation parmi un ensemble fini de possibilités ([Laroche99]).

Nous allons maintenant décrire un peu plus en détail deux approches continues classiques que sont le filtrage de Kalman et le flot optique. Ces approches sont les plus représentatives de ce qui est généralement utilisé en traitement du signal pour effectuer du suivi de cibles. En outre, celles-ci peuvent servir de base de comparaison à notre proposition.

2.4.2.1. Le filtre de Kalman

Le principe général du filtre de Kalman est de prédire la position d'un objet dans une séquence de perception en utilisant une position initiale, déterminée par des méthodes classiques analogues à celle vues dans le chapitre précédent, et un modèle de déplacement paramètré et modifiable. Un objet à suivre est caractérisé par un vecteur d'état, comprenant, entre autre, la position et les paramètres du mouvement, et par une modélisation des processus aléatoires pouvant influer sur le déplacement. Ces derniers sont en général modélisés par un bruit blanc gaussien. Le suivi s'effectue par estimation des vecteurs d'états successifs de façon récursive.

La méthode du filtre de Kalman est en général décomposé en 5 phases [Giai95] :

- Assignation d'un modèle cinématique à chaque primitive
- Prédiction de la position des primitives dans l'image suivante en utilisant les paramètres du modèle cinématique avec une certaine incertitude.
- Utilisation de l'incertitude sur la prédiction pour déterminer une zone de recherche.
- Recherche du meilleur appariement possible en utilisant une distance normalisée de type distance de Mahalanobis.
- Utilisation des nouvelles mesures pour mettre à jour les paramètres du modèle cinématique.

Le filtre de Kalman est très utilisé en traitement de séquences d'images et plus particulièrement en robotique. ([Roumeliotis99]) Son principal inconvénient est qu'il nécessite une connaissance importante du processus à observer pour fournir de bons résultats. En outre, avec ce type d'approche se pose le problème de l'initialisation des cibles ; problème qui n'est pas encore totalement résolu.

2.4.2.2. Le flot optique

Le flot optique est une méthode plus précise mais également plus coûteuse en temps de calcul que le filtre de Kalman classique. La principale différence entre les deux approches est la granularité des objets suivis. En effet, plutôt que de chercher un modèle cinématique adapté à chacune des objets à suivre, il s'agit ici de paramétrer le déplacement de chacun des pixels de l'image 2D. Ces paramètres sont en général représentés sous la forme d'une carte de déplacement ([DeCarlo00]) ou carte de mouvement sur laquelle est représenté un vecteur de déplacement pour chacun des pixels en mouvement. (cf. figure 7)

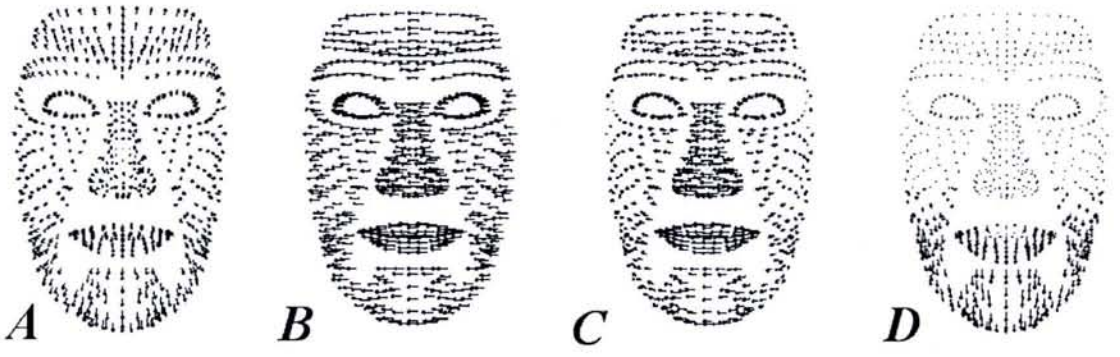

figure 7 : Représentation de carte de mouvement de flot optique pour des mouvements de visage (d'après [DeCarlo00]). *A* : Translation vers l'avant, *B* : Translation à droite, *C* :Rotation vers la droite, *D* :Ouverture de la bouche.

Le flot optique est couramment utilisé dans de nombreuses applications, en particulier des applications de surveillance de flux de circulation ([Mück00], [Nagel98]). Cette méthode est également utilisée dans la détermination des déplacements du dispositif d'acquisition en particulier lorsque celui-ci est placé sur un robot mobile. [Dev97]

Le filtrage par flot optique nécessite nettement moins de connaissance sur la cible à suivre, néanmoins les temps de calcul sont en général très importants ce qui limite son utilisation dans des applications temps réel.

2.5. Conclusion

En ce qui concerne le choix des capteurs et surtout de leur nombre, il semble plus intéressant d'avoir recours à plusieurs dispositifs d'observations différents. Néanmoins, il est nécessaire de doter notre processus de localisation/suivi d'un mécanisme de fusion d'informations simple pouvant gérer un nombre variable de sources de données.

La localisation d'objets, que ce soit avec des capteurs fixes ou mobiles, est une tâche assez délicate et essentielle pour les applications de robotique mobile ou de télésurveillance. Néanmoins parmi les approches qu'il est possible de trouver dans la littérature, il n'existe pas de méthode permettant de traiter à la fois les deux types de localisation absolue. En outre, toutes les méthodes précitées sont éminemment dépendantes du type des capteurs utilisés. En effet, étant donné que la plupart des algorithmes sont issus du traitement du signal, les méthodes d'élaboration des résultats ont été conçues à partir d'un capteur bien connu. Aucune de ces méthodes ne permet donc de gérer les capteurs défectueux ou totalement en panne, et encore moins la modification du nombre de ceux-ci en cours de fonctionnement, objectif que nous nous sommes fixé pour notre architecture. Néanmoins, ces méthodes de localisation vont nous servir de point de départ pour la construction des sources d'informations nécessaires à notre proposition.

Le suivi apporte une richesse supplémentaire concernant la connaissance des objets observés. En effet, la connaissance de l'évolution temporelle de la position des cibles permet d'émettre des hypothèses sur la dynamique propre de celles-ci et donc sur leur nature. Le principal problème des méthodes de suivi usuelles est la nécessité de posséder une connaissance *a priori* sur la dynamique des cibles. Pour le filtre de Kalman, par exemple, il est nécessaire de connaître la position initiale de la cible, l'équation régissant l'évolution de la cible entre deux positions successives, l'erreur sur l'observation du phénomène,.... Il en va de même pour les autres méthodes telles que les approches Markoviennes ([Simmons95], [Fox98], [Thrun99]) ou encore le filtrage particulaire [Schulz01]. Pour cette raison, il semble délicat de construire un système de suivi avec un système réactif. Néanmoins, il est possible de contourner le problème de la connaissance du processus à observer en dotant nos agents de propriétés dynamiques cohérentes, par l'intermédiaire des paramètres des interactions, par rapport à la nature des cibles que l'on veut suivre.

Suite à l'analyse des différents dispositifs existants pour réaliser une tâche de localisation et de suivi, deux éléments clefs ressortent :

Notre dispositif de localisation doit pouvoir gérer plusieurs types de sources d'informations, sous la forme de couples capteurs/traitements, en effectuant une fusion des informations par l'intermédiaire d'un mécanisme simple compatible avec l'utilisation d'un SMA constitués d'agents réactifs.

La dynamique du processus de résolution que nous voulons mettre en place, doit pouvoir effectuer un suivi des cibles en limitant les connaissances *a priori* sur les lois de

déplacements des cibles potentielles. La traduction de la connaissance de cette dynamique peut s'effectuer par l'intermédiaire des paramètres régissant les interactions entre les agents et avec leur environnement.

Chapitre 3 : Les Systèmes Multi-Agents

Le but de ce chapitre est de mettre en place les différents concepts liés à la construction ou à l'analyse des SMA. En particulier, nous allons définir ce que sont un agent et un système multi-agents. Nous verrons également comment il est possible de décrire un SMA à partir de ces composantes qui sont :

- Ses Agents (A)
- Son Environnement (E)
- Ses Interactions (I)
- Son Organisation (O)

Une fois tous ces éléments définis, nous allons, à partir d'exemples tirés de la bibliographie, montrer comment il est possible d'aborder les problèmes de la construction des agents, du choix des modes d'interactions, de la représentation de l'environnement et de son rôle, et de la caractérisation de l'organisation.

Etant donné que nous avons choisi d'explorer l'approche réactive pour la résolution du problème de la localisation et du suivi, les exemples que nous allons développer sont principalement issus de ce mouvement.

3.1. Généralités

Les SMA sont à la base d'un grand nombre d'applications telles que la simulation de systèmes biologiques complexes, la construction d'agents Web, ou encore la robotique mobile. Cependant, malgré le nombre important de domaines abordés et l'engouement croissant de la communauté scientifique pour ce type d'approches, il n'existe pas de formalisation théorique complète permettant de définir, concevoir et évaluer de tels systèmes hormis pour des cas particuliers. Cette absence de formalisation est essentiellement due à leur complexité naturelle et à la jeunesse du domaine. Cependant, on peut distinguer deux approches distinctes : l'une fondée sur l'analyse de systèmes réels biologiques ou physiques [Reynolds87], et l'autre, appelée approche constructiviste, qui a pour objectif l'élaboration et la construction de systèmes multi-agents pour la résolution de problèmes [Sellem00].

Les SMA peuvent être décrits à partir des caractéristiques communes se dégageant des différentes applications abordées [Weis99]. Ainsi, les SMA peuvent être considérés comme des systèmes ***intrinsèquement distribués***, tant du point de vue spatial ou temporel que du point de vue fonctionnel, et ***naturellement complexes*** de par l'impossibilité d'exercer un contrôle unique sur l'ensemble de ses constituants [Ferber95].

La complexité est due à la fois aux interactions qui impliquent un très grand nombre d'agents et aux paramètres qui agissent sur le processus d'organisation. Le comportement global du système, qui peut être qualifié d'intelligent, provient de l'assemblage des comportements élémentaires des agents et de leurs interactions entre eux et avec leur environnement.

Dans ce chapitre, nous allons définir les concepts inhérents aux systèmes multi-agents et essentiels à la compréhension de notre travail. En outre, nous allons nous focaliser sur les caractéristiques importantes pour la conception de tels systèmes

3.2. Les agents

3.2.1. Définition d'un agent

Avant de définir ce qu'est un SMA, nous allons dans un premier temps préciser la définition d'un agent. Jacques Ferber donne une définition très complète de ce que doit être un agent dans [Ferber95] :

Définition : Un agent est une entité physique ou virtuelle

- *qui est capable d'agir dans un environnement.*
- *qui peut communiquer directement avec d'autres agents.*
- *qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser)*
- *qui possède des ressources propres.*
- *qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune).*
- *qui possède des compétences et offre des services.*
- *qui peut éventuellement se reproduire.*
- *dont le comportement tend à satisfaire ses objectifs en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.*

Un agent possède tout ou partie de ces caractéristiques.

Cette définition est assez générale et regroupe quasiment tous les types d'agents que l'on peut rencontrer. Néanmoins, certains points peuvent prêter à discussion. En effet, la communication entre agents n'est pas obligatoirement directe. Dans certains cas, celle-ci se fait indirectement par l'intermédiaire de l'environnement, c'est notamment le cas dans certains systèmes d'inspiration biologique tels que les fourmis [Parunak97] ou les termites [Bonabeau99].

D'autre part, Makram Bouzid donne, dans [Bouzid01], une définition qui correspond à la notion d'agent telle qu'elle est appréhendée par l'équipe MAIA :

Définition : L'agent est une entité qui perçoit son environnement à travers ses capteurs et sur lequel il agit via ses effecteurs. Il doit aussi vérifier les propriétés suivantes :

- *L'autonomie, i.e. la capacité à agir sans intervention humaine.*
- *La réactivité, i.e. la capacité à répondre aux événements extérieurs.*
- *La possession d'un ou de plusieurs objectifs*
- *La situation, i.e. l'agent perçoit partiellement son environnement via ses capteurs, et peut en changer la configuration en agissant dessus localement via ses effecteurs.*

Un agent peut donc être physique (insecte, robot, véhicule,...), dans ce cas l'environnement dans lequel il évolue est le monde réel, ou il peut être virtuel, il s'agit alors de composants logiciels.

L'agent agit sur son environnement proche, c'est-à-dire que ses actions ont une portée locale. Ceci est à mettre en association avec la perception limitée de l'agent, on parle alors d'agents ***situés***. En effet, un agent ne peut agir sur des éléments qui échappent à sa perception. Toutes les actions, du déplacement à la modification de l'environnement, sont envisageables pour un agent. En fait, une action est un comportement qui modifie l'état des perceptions d'au moins un des agents. De cette façon, on peut également définir une action comme étant un comportement qui modifie l'environnement.

Ainsi, la communication entre agents peut être considérée comme une action puisqu'elle modifie la perception d'au moins un agent. S'il s'agit d'une communication directe, elle modifie la perception du ou des agents à qui est destiné le message. Par contre, s'il s'agit d'une communication indirecte la modification de la perception se fera par l'intermédiaire de l'environnement.

Le concept d'agent ainsi énoncé se rapproche de la définition de l'agent purement situé proposée par Ferber [Ferber95].

Dans le cadre de cette thèse, nous allons utiliser une définition plus générale et inspirée à la fois de la définition de Jacques Ferber et de celle de Makram Bouzid :

Définition : Un agent est une entité physique ou virtuelle

- *qui agit sur son environnement proche via ses effecteurs*
- *qui possède une perception limitée de son environnement*
- *qui se comporte de façon autonome pour son propre intérêt*

Cette définition représente, à mes yeux, les caractéristiques essentielles d'un agent. D'autres caractéristiques peuvent bien évidemment être rajoutées telles que la possibilité de posséder des ressources propres ou de se reproduire, mais celles-ci ne sont pas essentielles à la définition. Ainsi, j'ai choisi de ne pas faire apparaître la représentation interne du monde car celle-ci n'est pas toujours présente dans les agents, en particulier lorsque ceux-ci sont réactifs. ([Parunak97])

3.2.2. De l'agent vers le système d'agents

Maintenant que nous avons défini le concept d'agent, nous allons pouvoir définir ce qu'est un système multi-agents. Pour cela, appuyons-nous sur la définition de Ferber [Ferber95] :

Définition : On appelle système multi-agents (ou SMA) un système composé des éléments suivants :

- *Un environnement E, c'est-à-dire d'un espace disposant généralement d'une métrique.*
- *Un ensemble d'objet O. Ces objets sont situés, c'est-à-dire que, pour tout objet, il est possible, à un moment donné, d'associer une position dans E. Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.*
- *Un ensemble A d'agents, qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système.*
- *Un ensemble de relations R qui unissent des objets (et donc les agents) entre eux.*
- *Un ensemble d'opérations Op permettant aux agents A de percevoir produire, consommer, transformer et manipuler des objets de O.*
- *Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers.*

Cette définition d'un système multi-agents est vérifiable dans la majorité des systèmes rencontrés dans la littérature.

L'environnement joue un rôle actif dans les SMA. Il possède une dynamique propre qui comprend l'apparition, le déplacement, et la disparition d'objets. La principale différence entre les agents et les objets présents dans l'environnement est le côté intentionnel des changements d'états.

La définition qui sied plus à notre conception d'un système multi-agents est celle énoncée par Demazeau [Demazeau95].

Définition : On appelle un système multi-agents, un système qui comporte les éléments suivants :

- *Un ensemble d'agents A.*
- *Un environnement E, c'est-à-dire un espace contenant les agents qui disposent ou non d'une métrique et qui possède une dynamique propre.*
- *Un ensemble I d'interactions. Ces interactions peuvent se dérouler entre agents ou entre les agents et l'environnement.*
- *Une organisation O qui peut être la résultante des comportements élémentaires des agents.*

3.2.3. Société d'agents intelligents ou société intelligente d'agents

Il existe deux écoles concernant la conception de systèmes multi-agents : l'école cognitive et l'école réactive [Ferber95]. La différence entre ces deux conceptions réside dans la définition même des agents utilisés. Doit-on considérer les agents comme des entités intelligentes qui cohabitent dans un environnement commun ou comme des éléments simples réagissant aux variations de l'environnement selon un schéma de type stimulus-réponse ? La question est donc de savoir si l'intelligence du système se trouve dans les agents eux-mêmes, on parlera alors de société d'agents intelligents, ou dans les interactions et l'organisation issues de la composition des comportements réactifs, on parlera dans ce cas de société intelligente d'agents.

Le choix de l'une ou l'autre des deux approches est déterminant pour la conception du système. En effet, celui-ci va déterminer si l'intelligence doit être injectée au niveau des agents eux-mêmes ou au niveau des interactions. Ces deux approches ne sont pas exclusives, aussi il est bien évidemment possible de combiner les deux types d'approches. En outre, ce n'est pas parce que le processus de décision est simple que les mécanismes de perception et d'action doivent l'être également.

3.2.3.1. Agents cognitifs

L'approche cognitive est la plus représentée dans la communauté de l'intelligence artificielle distribuée, et mise en œuvre dans plusieurs cadres applicatifs [Weis99], [Jennings98], [Mess00], [Sycara98] ou [Wooldridge95]. Elle trouve son origine dans la volonté de faire interagir des entités intelligentes par l'intermédiaire de communications explicites. Dans cette approche, chaque agent dispose de sa propre représentation du monde et de la tâche qu'il doit accomplir, d'un ensemble d'actions accessibles permettant de réaliser leur tâche et de facultés de communication avec les autres agents (cf. figure 8). On peut alors parler d'une société d'***agents intelligents.*** Cette interaction peut se faire soit de façon coopérative, comme par exemple dans [Minar99], soit de façon concurrentielle comme dans [Durand99]. Les agents d'un tel système sont également appelés agents ***intentionnels*** car ils possèdent une représentation de leurs buts et de la façon dont ils peuvent les atteindre.

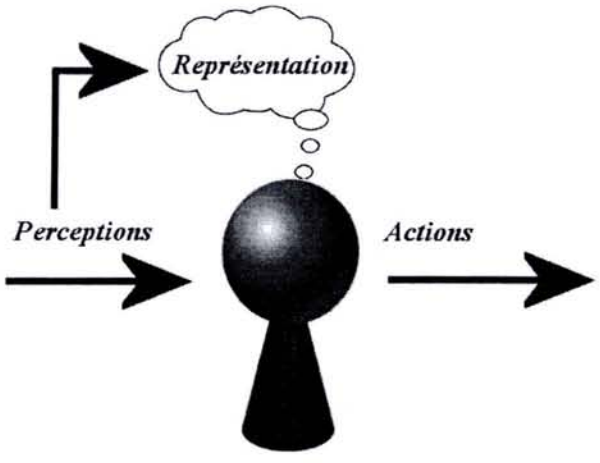

figure 8 : Représentation d'un agent cognitif

3.2.3.2. Agents réactifs

D'un autre point de vue, l'approche réactive considère que les agents ne doivent pas obligatoirement posséder une certaine forme d'intelligence pour que le système complet se comporte de façon intelligente, on parle alors d'***agents réactifs*** [Drogoul93], [Parunak97] (cf. figure 9). C'est souvent le cas lorsqu'on observe les populations d'insectes tels que les fourmis, les termites ou les araignées sociales dont les capacités

cognitives sont existantes mais limitées. Dans ce type d'approche, il n'existe ni de représentation du but du système ni de planification au préalable destinée à organiser l'enchaînement des comportements élémentaires des agents. Cependant, de tels systèmes peuvent résoudre des problèmes qualifiés de complexes tels que le problème du voyageur de commerce [Dorigo97] ou la détection de contours dans une image à niveau de gris [Ramos00]. Ceci montre que notre hypothèse d'utilisation d'agents réactifs pour effectuer une tâche de localisation et de suivi est plausible.

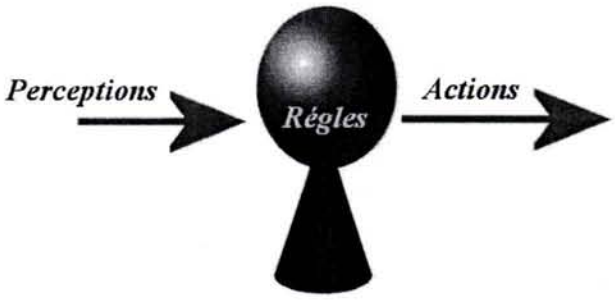

figure 9 : Représentation d'un agent réactif

3.3. L'environnement

L'environnement est un élément important dans le processus dynamique d'un système multi-agents. Cependant, il existe assez peu de documents concernant sa structure, sa construction et sa dynamique.

De façon usuelle, l'environnement est défini comme étant tout ce qui entoure les agents. Il possède une dynamique propre qui influe directement ou indirectement sur les actions effectuées par les agents. Pour Jean-Pierre Müller [Müller98], l'environnement est à la fois un lieu d'inscription permettant de partager le résultat des actions des agents constituant le système et un ensemble de contraintes sur la dynamique de celui-ci. Dans certains systèmes, l'environnement se réduit simplement à des canaux de communications entre les agents (Agents Web par exemple). Par contre, dans les systèmes d'inspiration biologique ou physique, l'environnement est essentiel à l'accomplissement des interactions entre les agents et donc au bon déroulement du processus de résolution du problème posé. Dans ce dernier cas, l'environnement possède en général une métrique.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

Lorsqu'il s'agit d'analyser des systèmes distribués réels, l'environnement du système doit s'attacher à représenter au mieux le lieu d'immersion réel des agents, tant du point de vue de sa dynamique que de celui de ses contraintes, de sa métrique ou de sa structure. Ainsi, si l'on veut simuler la dynamique de la recherche de nourriture chez la fourmi, l'environnement devra comporter des zones représentant le nid des fourmis, les sources de nourriture et les obstacles éventuels. En outre, il devra pouvoir gérer l'évaporation des substances chimiques déposées. Ces deux caractéristiques, topologique et dynamique, sont les deux principales caractéristiques de l'environnement.

Dans le cadre d'une approche constructiviste, l'environnement doit être conçu de façon à rendre compte des caractéristiques du problème à résoudre tout en mettant en place les contraintes nécessaires au bon déroulement du processus de résolution. Dans notre cas, l'environnement doit pouvoir rendre compte de la géométrie du monde réel et de la dynamique des cibles à suivre.

3.4. Les interactions.

Dans le cadre des SMA, une interaction peut être définie comme la mise en relation dynamique de deux ou plusieurs éléments du système par le biais d'un ensemble d'actions réciproques. Ces éléments peuvent être des agents ou des éléments de l'environnement ou encore l'environnement lui-même.

La notion d'interaction est au centre de la problématique dans les SMA. Elle peut être considérée à la fois comme la conséquence d'actions effectuées par plusieurs agents en même temps, mais aussi comme l'élément nécessaire à la constitution d'une organisation spatio-temporelle. Elle est la résultante des comportements individuels des agents et des contraintes de l'environnement.

Ainsi, les interactions ne sont donc pas forcément exprimées de façon explicite dans la description du système.

3.5. Organisation et auto-organisation

L'***organisation*** peut être définie comme une structure issue de l'arrangement des composants d'un système de façon à répondre à un objectif donné. Dans le cadre des systèmes multi-agents l'organisation peut être considérée comme un arrangement des agents et de leurs comportements conditionné par les contraintes imposées par l'environnement. Lorsque cette structure est la conséquence émergente de la

conjonction des interactions de "bas niveau" entre les éléments du système, on parle alors d'***auto-organisation***.

Dans les SMA, l'auto-organisation est un phénomène résultant de deux modes d'interaction élémentaires : une ***rétroaction positive*** et une ***rétroaction négative***. La rétroaction positive induit une amplification du phénomène d'organisation alors que la négative tend à stabiliser le phénomène [Müller98].

Lorsque l'on veut construire un SMA répondant à un problème posé, il s'agit donc de définir les interactions entre les agents et avec leur environnement afin de prendre en compte les contraintes du problème et de façon à obtenir une structure émergente pouvant être interprétée comme une solution..

Cependant, la construction d'un tel système est très délicate car il n'existe pas de méthode pour élaborer les interactions en fonction du problème à résoudre. En outre, le problème de la caractérisation de l'organisation, et donc de la solution obtenue, n'a pas encore été résolu de façon générale. Pourtant, l'intérêt de la caractérisation de l'organisation est indéniable car elle permettrait, entre autre, d'estimer la qualité du résultat obtenu et de réguler les paramètres des interactions en fonction de celle-ci.

3.6. Quelques exemples de systèmes réactifs

Comme nous l'avons expliqué dans l'introduction de ce document, nous avons choisis d'aborder le problème de la localisation et du suivi en utilisant des SMA constitués d'agents réactifs. Aussi, il est plus à propos de détailler ici des approches réactives résolvant des problèmes relativement complexes, plutôt que de mettre l'accent sur des approches cognitives traitant du même sujet. Ceci va permettre de mettre en évidence les grands principes que nous devons adopter pour construire notre proposition.

Les sujets abordés sont des problèmes récurrents en informatique tel que la détection du plus court chemin, la détection de transitions radiométriques dans des images réelles ou encore la résolution du problème du voyageur de commerce.

3.6.1. Fourragement et tri de couvain (brood sorting) chez les fourmis

L'exemple des fourmis est celui qui est le plus couramment utilisé dans la littérature ([Ferber95], [Bonabeau99] et [Parunak97]) lorsqu'il s'agit d'illustrer l'utilisation des SMA pour la résolution de problèmes simples.

Dans la nature, les fourmis construisent un ensemble de chemins qui connectent le nid aux différentes sources de nourriture. La construction de ces trajets a été expliquée par les biologistes sous la forme d'un enchaînement de comportements élémentaires de type stimulus/réponse. D'un point de vue mathématique, les réseaux formés correspondent à des arbres couvrants (spanning trees) minimaux [Goss89]. Les chemins suivis par les fourmis correspondent aux chemins les plus courts. En outre, la méthode de construction des réseaux ne peut pas être identifiée à un algorithme existant en théorie des graphes. Cette structure optimale du réseau émerge d'un ensemble d'actions individuelles simples exécutées par les fourmis (cf. figure 10).

Dans le cadre de l'utilisation des algorithmes fournis pour résoudre des problèmes informatiques, on peut considérer que chaque fourmi impliquée dans la tâche de fouragement possède 5 items comportementaux élémentaires déclenchés en fonction de son environnement local [Parunak97] :

- **Eviter** les obstacles
- **Explorer** l'environnement de manière aléatoire dans la direction générale du gradient de phéromones (substance chimique déposée par les fourmis). Lorsque aucune phéromone n'est détectée la fourmi suit un mouvement brownien (distribution uniforme de direction). Par contre, dans le cas où une phéromone est détectée la distribution de probabilité est modifiée de façon à favoriser la direction du déplacement selon la quantité de phéromone la plus importante.
- **Déposer** des phéromones à intervalle régulier tout en se dirigeant vers le nid si la fourmi porte de la nourriture. La position du nid est connue grâce à une diffusion d'une phéromone particulière au nid, c'est-à-dire que les fourmis connaissent la position du nid. Dans certaines simulations simplifiées, la position du nid n'est ni connue ni détectable et les fourmis continuent à se déplacer de façon aléatoire. Dans ce dernier cas, il y a toujours convergence du système vers un état stable mais cependant avec une vitesse plus faible.
- **Prendre** de la nourriture si la fourmi se trouve sur une source et si elle n'en transporte pas.
- **Poser** la nourriture si la fourmi en transporte et se trouve sur le nid.

Ces items comportementaux sont issus des hypothèses de travail posées par les biologistes à partir d'observations effectuées sur des populations de fourmis.

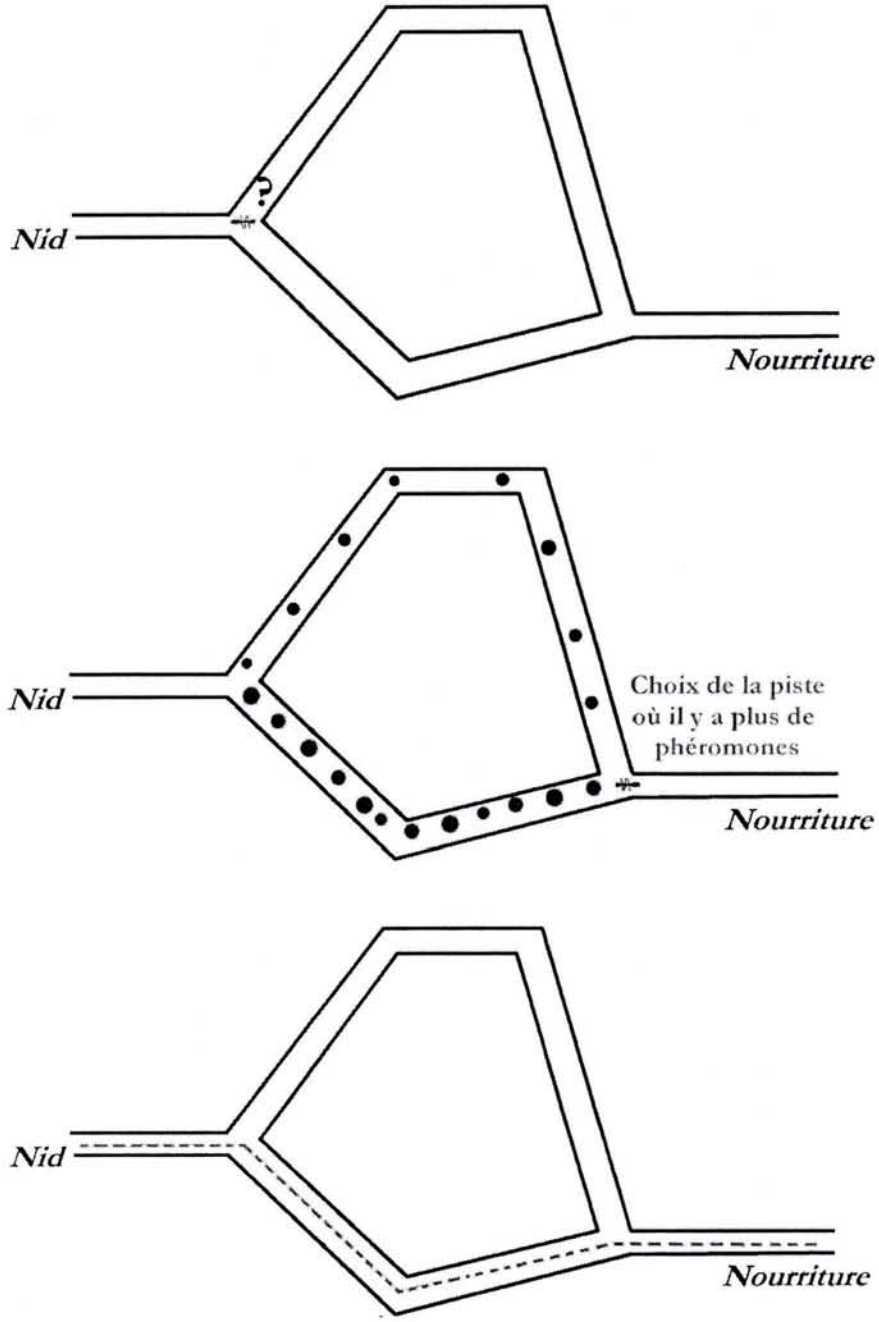

figure 10 : Recherche du plus court chemin à l'aide d'algorithme fourmis

De par la définition des items comportementaux, tous les chemins de phéromones conduisent vers une source de nourriture. Les premiers chemins du réseau ne sont pas forcement rectiligne mais, ils tendent à le devenir au fur et à mesure des passages des fourmis.

L'expérience de la figure 10 montre comment il est possible d'obtenir un comportement collectif émergent (obtention du plus cours chemin) à partir d'individus incapables de résoudre seuls le problème. Afin de déterminer quelle branche correspond au plus court chemin, il faut évaluer la probabilité qu'a une fourmi à prendre l'une des deux branches. La probabilité que la (m+1)ieme fourmi prenne la branche du haut est donnée par l'équation suivante :

$$P_U(m+1) = \frac{(U_m + k)^b}{(U_m + k)^b + (L_m + k)^b}$$

Équation 1

Dans cette équation, U_m correspond au nombre de fourmis étant passées par la branche supérieure et L_m au nombre de fourmis dans la branche inférieure.(k et b sont des paramètres destinés à adapter le système à la configuration de l'environnement [Dorigo99]).

Une des caractéristiques importantes de cette méthode de recherche du plus court chemin est sa faculté d'auto-adaptation aux changements de configurations de l'environnement. Lorsqu'un des chemins reliant le nid à la source de nourriture est obstrué, la quantité de phéromone présente dans celui-ci va diminuer jusqu'à disparaître. De façon simultanée cette quantité va augmenter dans les autres chemins alternatifs jusqu'à ce que l'un d'entre eux soit privilégié (cf. figure 11).

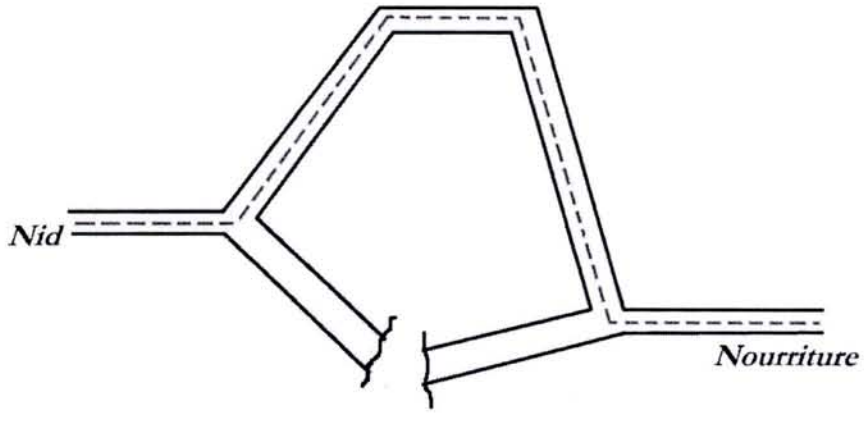

figure 11 : Adaptation de la recherche du plus court chemin

Dans cet exemple tiré de la biologie, nous retrouvons un certain nombre de caractéristiques importantes pour notre approche :

- Les fourmis possèdent des comportements simples et interagissent avec leur environnement de façon locale (détection de la nourriture et suivi du gradient de phéromones).
- Les fourmis possèdent une perception locale de leur environnement.
- L'environnement possède une dynamique propre sur laquelle les fourmis ne peuvent pas agir.
- Le résultat (construction du plus court chemin) est issu de la combinaison de la dynamique des agents et de celle de l'environnement.
- La simplicité et le caractère local des comportements induit un phénomène d'auto-adaptation qui est une caractéristique propre aux SMA réactifs.

Outre son utilisation pour la recherche du plus court chemin, le comportement de fouragement des fourmis est également utilisé dans d'autres applications. Il est notamment possible de les utiliser pour la résolution du problème du voyageur de commerce [Colorni96] ou encore pour la détection des contours des objets présents dans une image à niveau de gris [Ramos00].

Les approches de ce type sont utilisées dans beaucoup d'applications avec des résultats spectaculaires [Colorni96] et [Bonabeau99]. En ce qui concerne le problème du voyageur de commerce, les algorithmes « fourmis » sont comparables en performances aux approches de type algorithmes génétiques. Ces algorithmes sont également utilisés dans plusieurs problèmes d'optimisation tels que l'affectation quadratique (meilleure heuristique actuellement), le coloriage de graphe ou le routage de véhicules ou de paquets dans les réseaux informatiques [Dorigo96].Dans toutes ces applications, le résultat correspond à une configuration stable des agents découlant de l'accumulation de phénomènes locaux. Le principal problème rencontré dans tous les cas de figure est la difficulté de trouver une preuve mathématique de la convergence du système [Meuleau02] et [Birattari02]. Ainsi, il est délicat de caractériser de façon formelle l'obtention d'une organisation répondant au problème posé. Ce problème est récurrent dans toutes les méthodes de résolution de problème faisant appel aux SMA réactifs.

3.6.2. Construction de toile chez les araignées sociales

L'origine des travaux sur les araignées sociales en informatique, résulte d'une collaboration avec des biologistes désirant confirmer leurs hypothèses comportementales concernant la construction commune de toile ou la prédation, sans avoir à élaborer des expériences impliquant des animaux réels qui sont délicates à mettre en œuvre et très longues à converger.

Le modèle de construction de toiles par un ensemble d'araignées s'exprime par un nombre limité de comportements élémentaires qui dépendent de la présence de soie dans l'environnement :

- **Se déplacer**, soit en accédant à un piquet adjacent, choisi selon une probabilité uniforme, soit en suivant un fil de soie en fonction d'une probabilité d'attirance.
- **Poser** un fil de façon aléatoire.

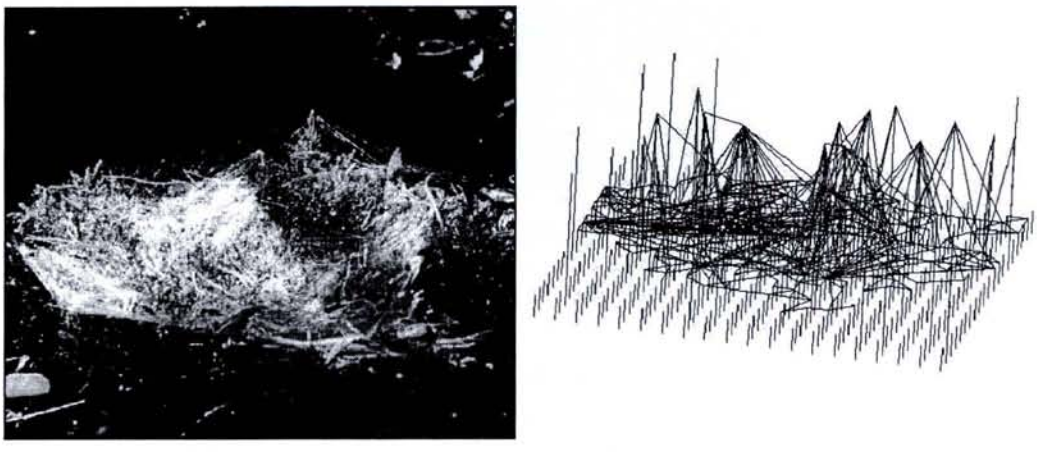

figure 12 : Exemple de construction de toile. L'image de gauche représente le modèle biologique tel qu'il peut être observé dans la nature. L'image de droite représente le modèle informatique reconstruit à partir des comportements individuels proposés par les biologistes.

A partir des résultats obtenus par les simulations informatiques, les biologistes ont pu confirmer leurs hypothèses. D'un point de vue informatique, comme cela est le cas pour d'autres sociétés d'arthropodes telles que les fourmis, les propriétés mises en évidence par les simulations ont permis d'envisager des méthodes de résolution alternatives pour des problèmes dont les enjeux dépassent le cadre de la biologie.

A l'instar des algorithmes fourmis, utilisés en traitement d'image pour la détection de transitions radiométriques dans les images à niveau de gris, le comportement de construction en commun de la toile a été adapté et a permis d'envisager un nouvel algorithme de détection de régions radiométriquement homogènes dans des images à niveaux de gris [Bourjot02], [Bourjot01] et [Thomas01] (cf. figure 13).

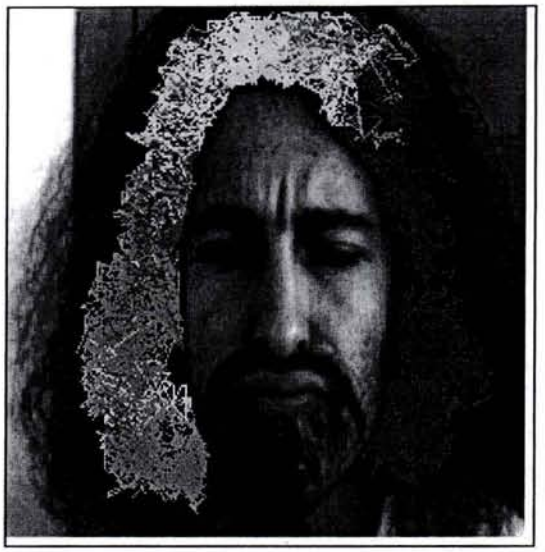

figure 13 : Détection de régions radiométriquement homogènes utilisant un algorithme inspiré des comportements des araignées sociales [Thomas01] (Chaque couleur correspond à une araignée, toutes les araignées se trouvant sur une toile unique)

De la même façon que pour les applications inspirées du comportement des fourmis, il n'existe pas de méthode pour calculer les paramètres optimaux fournissant le meilleur résultat possible si ce n'est un parcours empirique de l'espace d'état, soit de façon manuelle et intuitive, soit à l'aide de méthodes de type algorithmes génétiques. A l'instar des applications usuelles en traitement d'images, l'estimation de la pertinence du résultat se fait essentiellement de façon intuitive par observation directe.

3.6.3. Les boids et le déplacement en essaim (flocking)

L'origine des travaux sur le déplacement en essaim (flocking) provient de la volonté de modéliser, en image de synthèse, le mouvement des bancs de poissons ou des volées d'oiseaux sans avoir recours à une détermination rigide des trajectoires individuelles. Les premiers travaux ont été effectués par Craig Reynolds [Reynolds87] et sont présents dans de nombreux films. « *Batman returns* » de Tim Burton fut l'un des premiers à utiliser le modèle de déplacement en essaim en particulier pour la simulation de volées de chauves-souris et le déplacement des pingouins.

Les volées d'oiseaux, et les bancs de poissons, ont ceci de particulier qu'ils exhibent des comportements de coordination évolués tels que la synchronisation des changements de direction, l'évitement d'obstacles ou de collisions entre les membres du groupe (ou boid) uniquement en utilisant des comportements individuels simples. En outre, tous les déplacements de groupes se font sans leader défini.

Les comportements individuels sont destinés à maintenir une cohérence de l'ensemble des individus. Les items comportementaux sont au nombre de trois (cf figure 14) :

- **Comportement de séparation** : maintien d'une distance minimale entre un agent et les autres objets à proximité (y compris les autres agents)
- **Comportement de cohésion** : attirance des agents pour le centre de la volée.
- **Comportement d'alignement** : alignement local des vitesses des agents à la fois en orientation et en amplitude.

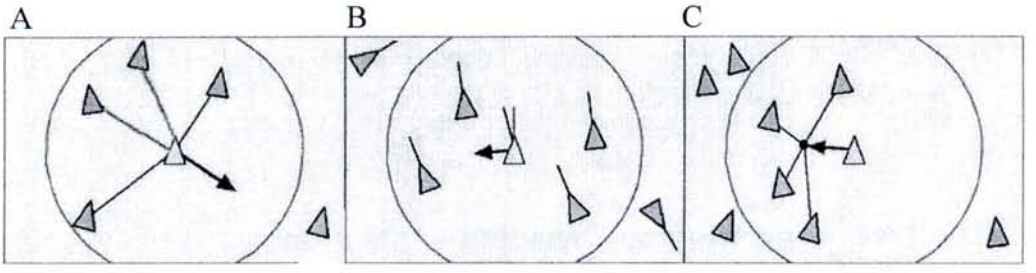

figure 14 : Représentation des comportements utilisés pour le flocking. Séparation (A), Alignement (B) et Cohésion (C). Le disque centré sur l'agent correspond à sa limite de perception. D'après [Reynolds87].

Dans une structure de flocking, les comportements individuels répondent de façon simultanée à une variation globale de la structure. Ces variations de configurations se propagent de proche en proche de façon locale. Tous les comportements sont ancrés dans un schéma local de perception et d'action : le mouvement d'un agent est conditionné par ses congénères les plus proches et influe uniquement sur ceux-ci. Ainsi, chaque agent possède une vision limitée paramétrée par une distance de perception et un angle de vision (cf. figure 15). Dans ce cadre applicatif, on peut donc parler d'agents situés.

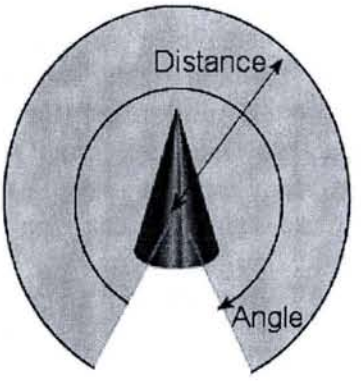

figure 15 : Les limitations de la perception d'un boid

Le mouvement généré par l'algorithme de vol en essaim peut être scripté, dans le sens où l'on donne une orientation générale au mouvement sous la forme d'une contrainte extérieure, ou impromptue. Dans ce dernier cas, le système converge rapidement vers une volée unique se déplaçant à une vitesse uniforme tant en direction qu'en amplitude. Les petites volées et les individus isolés sont absorbés par les volées de grande taille. La rencontre d'obstacle peut séparer les volées en plusieurs plus petites. L'initialisation d'un mouvement impromptu se fait de façon aléatoire au niveau des positions des boids, de leurs orientations et de leurs vitesses.

Un certain nombre d'éléments sont intéressants pour notre problématique :

- Les agents sont réactifs et situés
- La population est homogène
- Les comportements sont inspirés de la physique et de la biologie
- Les interactions permettent une adaptation automatique de la structure du groupe en cas d'obstacle.

3.6.4. Algorithmes réactifs d'évitement de collision appliqués à la gestion du trafic aérien

Outre les systèmes d'inspiration biologique que nous venons de voir, il existe des systèmes multi-agents réactifs s'appuyant sur des lois physiques usuelles sans pour autant s'inspirer d'un système complet existant. C'est notamment le cas des propositions faites par Karim Zeghal [Zeghal94] dans le cadre de l'évitement de collision entre des avions se trouvant à proximité les uns des autres (cf. figure 16).

Le principal intérêt de ce système est le fait qu'il a été élaboré en fonction d'un problème concret donné et non pas adapté d'un problème existant qui lui serait proche. Ainsi, la démarche est complètement différente des exemples que nous avons détaillés précédemment. En effet, il ne s'agit plus ici de trouver comment adapter un système existant à un problème donné, mais à construire un système en fonction d'un problème posé.

Les comportements d'évitement sont dictés par une application de forces physiques (champs de forces) calculées en fonction de la présence d'autres appareils dans le voisinage. En fonction du type de configuration spatiale rencontrée le comportement des agents peut suivre plusieurs modèles :

- *Coopératif* si l'avion rencontré possède le même type de comportement.
- *Non coopératif* si l'avion rencontré possède un autre mécanisme d'évitement
- *Passif* le comportement d'évitement est inhibé
- *Agressif* l'avion rencontré devient une cible.

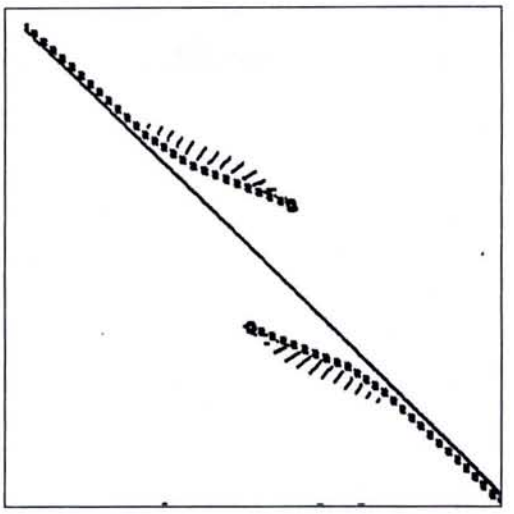

figure 16 : Exemple de comportement d'évitement. (d'après [Zeghal94])

Les principes comportementaux instaurés sont purement réactifs et ne prennent en compte que la configuration locale. De plus, il n'y a ni communication ni négociations entre les avions.

Enfin, il est intéressant de noter que la réponse fournie par le système n'est pas une configuration spatiale mais une dynamique des agents. Il n'y a donc pas d'état d'équilibre dans la configuration spatiale des agents.

3.6.5. The Particle Swarm Optimization

The Particle Swarm Optimization (PSO) ([Kennedy01]) est une méthode d'optimisation qui peut être comparée aux méthodes de programmation évolutionnistes telles que les algorithmes génétiques. En effet, celle-ci utilise également une "population" de solutions candidates, les particules, qui doivent évoluer afin d'obtenir une solution optimale ou quasi-optimale à un problème donné. L'évolution des particules s'effectue en prenant en compte trois principes fondamentaux dans la sélection des comportements :

- L'***évaluation*** des stimuli.
- La ***comparaison*** de la situation de la particule avec celles de ses voisins en fonction d'une valeur de référence proche de la notion de fitness des algorithmes génétiques.
- L'***imitation*** des comportements des particules du voisinage.

L'application de ces principes couplée à une évolution temporelle permet l'obtention d'une solution.

La principale différence avec les algorithmes génétiques est le principe d'évolution de la population dans l'hyper-plan des solutions, qui s'appuie sur des liens locaux entre les particules liens qui peuvent être comparés à des interactions entre agents. A l'initialisation, toutes les variables du système sont affectées à des valeurs aléatoires. En outre, à chacune des particules/agents est attribuée, de façon stochastique, une vitesse. A chaque itération, cette vitesse est modifiée en prenant en compte la meilleure position entre la position précédente de la particule et celles de ses voisines. Ainsi, la sélection, et donc la progression vers les minimums locaux, se fait en fonction des positions les meilleures du voisinage suivant une valeur de santé ("fitness") dépendante du problème.

Les principaux avantages de cette méthode sont :

- Sa simplicité notamment au niveau des mécanismes mis en œuvre qui sont facilement compréhensibles,
- Sa facilité à être implémentée
- Sa puissance
- Son faible coût.

Bien évidemment, cette méthode a été adaptée à plusieurs cadres applicatifs tels que le contrôle de l'état de charge d'une batterie d'un véhicule ou l'estimation de la performance d'êtres humains.

3.7. Discussion

Tous les systèmes étudiés dans ce chapitre correspondent à une approche de type SMA constitués d'agents réactifs. Cependant, chacun d'entre eux possède des particularités essentiellement liées aux choix de conceptions effectués au niveau des processus d'interaction et de la dynamique de l'environnement. Même s'il est possible de mettre en évidence des caractéristiques communes aux approches réactives, il est peu envisageable de concevoir un système réactif indépendant du problème que celui-ci doit résoudre.

L'utilisation de SMA pour la résolution de problème peut se substituer aux méthodes classiques. De par la distribution de ses éléments et la localité des unités de raisonnement (agents), ces approches sont plus adaptées à des problèmes physiquement ou conceptuellement distribués. En outre, elles possèdent des propriétés de flexibilité, de robustesse et d'adaptabilité.

Du point de vue réactif, la conception d'un SMA, dans le but de résoudre un problème donné revient à :

- Formaliser le problème sous la forme d'un environnement structuré.
- Concevoir les comportements et les interactions des agents de façon à élaborer le processus de résolution.

- Interpréter les résultats fournis (organisation) en fonction des caractéristiques de la tâche à résoudre.

Les éléments primordiaux dans la conception d'un tel système sont le choix de la représentation de l'environnement en adéquation avec les caractéristiques du problème à résoudre d'une part, et la définition des interactions et des comportements des agents d'autre part. En ce qui concerne les comportements, un des éléments clefs est le choix de ceux-ci en fonction de la situation de chacun des agents. Il existe plusieurs méthodes pour choisir et/ou combiner plusieurs items comportementaux dont :

- Les méthodes à base d'heuristiques qui consiste en l'utilisation de règles simples s'inspirant de considération logiques sur le déroulement du processus. (si *Agent[i] et Agent[j] trop proches et de même nature* alors *déplacement des agents*).
- Les méthodes probabilistes qui sont en général issues d'études statistiques sur un phénomène réel proche de celui que l'on veut résoudre. (A chaque pas de temps l'*Agent[i]* peut *se déplacer* avec une probabilité $P_1[i]$, *consommer* avec une probabilité $P_2[i]$ et *ne rien faire* avec une probabilité $P_3[i]$).
- Les méthodes inspirées de lois physique à base de combinaison de forces ou de champs de force dépendants de la situation de chacun des agents (*Accélération_Agent[i] = (Σforces / masse_Agent[i])*).

On peut bien évidemment combiner ces trois familles comportementales en fonction des connaissances que l'on a du processus à construire et des propriétés que l'on veut lui donner.

A la lumière des éléments que nous venons d'exposer, nous allons détailler, dans le chapitre suivant, nos choix conceptuels pour résoudre le problème de la localisation et du suivi de cibles multiples.

Chapitre 4 : Proposition d'un principe de résolution pour la localisation

Maintenant que les concepts théoriques sont posés, nous allons pouvoir préciser notre approche.

Dans ce chapitre, nous allons nous concentrer sur la description et le fonctionnement du processus de résolution. Dans un premier temps, nous allons exposer les principes généraux de notre architecture. Nous détaillerons ensuite son fonctionnement et les choix conceptuels que nous avons effectués.

4.1. Principes

4.1.1. Généralités

Nous avons choisi de structurer notre processus de résolution de problème à partir d'agents réactifs simples qui interagissent entre eux et avec leur environnement pour produire collectivement une réponse complexe qui correspond à une solution au problème posé. Comme nous l'avons exposé en introduction, les SMA constitués d'agents réactifs possèdent un certain nombre de propriétés qui sont adaptées au cadre applicatif que nous avons choisi. Dans la littérature, il y a peu de dispositif de localisation ou de fusion de données utilisant un processus de résolution à base d'agents réactifs. Pour ce type d'application, on a plutôt recourt à des agents cognitifs [Chang98]. En général, les agents réactifs sont plutôt utilisés pour effectuer du contrôle distribué de processus ou de robots mobiles [Simonin01] ou de la distribution des informations. Il est donc intéressant de montrer comment les principes que nous avons développés dans le chapitre précédent peuvent être appliqués dans le cadre de la localisation et du suivi de cibles.

En outre, nous avons choisi d'utiliser une population homogène d'agents. Ce choix est uniquement une hypothèse de développement de façon à simplifier l'ajout ou le retrait d'agents en cours de fonctionnement et de rendre le processus de résolution indépendant de la situation et des caractéristiques de chacun des agents. En outre, ce choix n'est pas limitatif. Il est tout à fait envisageable de modifier la population d'agents de façon à favoriser une spécification de chacun d'entre eux en fonction de leur situation et de la cible qu'ils suivent, sans pour autant devoir modifier les mécanismes régissant l'ensemble du système.

Nous avons vu, dans le chapitre précédent, qu'il y a trois grandes familles d'approches pour la mise en œuvre des mécanismes d'interactions : l'utilisation d'heuristiques, l'utilisation de formulations probabilistes, et l'utilisation de modèles physiques. Notre choix s'est porté sur une formulation physique, sous forme de forces de type attraction/répulsion. Dans notre cas, les avantages de cette formulation sont multiples :

- Elle permet de donner aux agents des comportements cohérents en adéquation avec les phénomènes physiques réels entrant en jeu dans le déplacement des cibles. Il est alors possible de se passer de certaines connaissances sur la dynamique du processus observé.
- L'expression analytique des forces peut s'exprimer simplement en s'inspirant des lois de la gravitation.
- Les fondements théoriques établis en physique classique peuvent nous servir pour construire des outils d'évaluation du processus et de l'organisation résultante.
- Il est aisé de conserver une cohérence de l'ensemble du système.

Nous avons vu en introduction que l'enjeu de notre approche est de savoir comment construire des comportements et des interactions entre les agents et leur environnement de façon à ce que le système fournisse une réponse au problème de la localisation et du suivi.

Un SMA constitué d'agents situés réactifs peut être décrit par les éléments suivants [Demazeau95] :

- L'environnement et ses lois d'évolution
- Les agents et leurs items comportementaux
- L'organisation résultante
- Les interactions entre les agents et entre les agents et leur environnement

La définition de l'environnement et celle de l'organisation correspondent à une description macroscopique du processus de résolution, tant du point de vue de la formalisation du problème à résoudre que de celui de la solution attendue. Par contre, la définition des agents et de leurs comportements fait référence à une description microscopique du processus.

Nous allons donc décrire comment nous allons construire ces éléments de façon adaptée à notre cadre applicatif dans le but de fournir une solution sous la forme d'une organisation spatio-temporelle simple.

4.1.2. Les différents éléments du système

Dans ce paragraphe, nous allons définir, de façon générale, les éléments constitutifs du dispositif selon la description des systèmes multi-agents donnée par Demazeau.

L'environnement

L'***environnement*** du système est une représentation abstraite du monde d'évolution des éléments à localiser. Il est représenté sous la forme d'un ensemble discret d'états correspondant chacun à une position possible d'une cible. Chaque cible potentielle est matérialisée par un ***plot*** localisé sur un état. Chaque plot possède donc une position et une altitude qui peut être considérée comme une valeur de confiance concernant l'information fournie. A chaque pas de temps, l'environnement est reconstruit en fonction de son état précédent, c'est-à-dire la position de tous les plots au temps t-1, et des nouvelles informations reçues au temps t. Il y a donc une accumulation des informations dans l'environnement. Afin d'éviter une saturation de l'environnement, la prise en compte des états au temps t-1 se fait modulo une pondération destinée à diminuer l'importance des anciens plots dans la reconstruction.

Les agents

Un ***agent*** est une entité se déplaçant dans son environnement sous l'influence de ses comportements et dont ***l'objectif implicite*** est de localiser et de suivre les plots présents dans son environnement en utilisant la perception partielle qu'il possède de celui-ci.

L'organisation

Nous supposons qu'il est possible d'obtenir des informations concernant les éléments que nous voulons localiser sous la forme de zones d'intérêt. Dans ce cas, la localisation peut donc être définie comme étant le résultat de la détection de la position de ces zones. D'un point de vue macroscopique, ce que nous voulons c'est donc une ***organisation*** spatio-temporelle des agents telle que l'agencement et la dynamique de ceux-ci rendent compte de l'état des cibles. Nous avons donc choisi d'effectuer cette localisation sous la forme d'une agglomération des agents sur les plots. La position d'une cible est alors donnée par l'estimation de la position du barycentre de l'agglomérat d'agents.

En outre, en l'absence de plot, les agents devront se répartir de façon uniforme dans l'environnement. Ceci permet, à la fois, d'éviter tout regroupement inopportun et de prévoir l'apparition de cibles éventuelles.

Nous pouvons considérer l'agrégation des agents sur les plots comme le résultat d'un phénomène d'amplification et l'équi-répartition dans l'environnement en l'absence de perception comme la conséquence d'un phénomène de régulation.

Les interactions

Etant donné que nous avons choisi d'utiliser un système de type ***auto-organisé***, les ***interactions*** des agents, entre eux et avec leur environnement, vont donc devoir rendre compte de deux tendances d'évolution, antagonistes dans leurs résultats.

De façon à obtenir ce double phénomène de focalisation et d'équi-répartition spatiale, il faut doter nos agents de deux comportements reflétant l'antagonisme des phénomènes globaux : un comportement d'***attraction*** responsable du processus de focalisation et donc de l'amplification du processus d'organisation (feed-back positif) et un comportement de ***répulsion*** qui contribue à la régulation (feed-back négatif) tout en induisant une répartition homogène en l'absence de plots. Le comportement de répulsion caractérise une interaction entre agents, alors que le comportement d'attraction traduit une interaction entre les agents et leur environnement.

Afin de réguler l'influence des deux comportements antagonistes en fonction de la situation des agents, nous avons introduit un état interne de satisfaction qui inhibe ou tout au moins diminue l'influence de la répulsion lorsqu'un agent se trouve sur un plot. Pour cela nous avons doté nos agents d'un item comportemental supplémentaire qui consiste en la ***consommation*** du plot sur lequel il se trouve. Cette consommation permet d'augmenter la satisfaction de l'agent. En outre, ce comportement a pour conséquence de faire disparaître les plots de façon à ce que l'environnement ne contienne que des informations récentes.

L'architecture que nous proposons peut donc être représentée de la façon suivante :

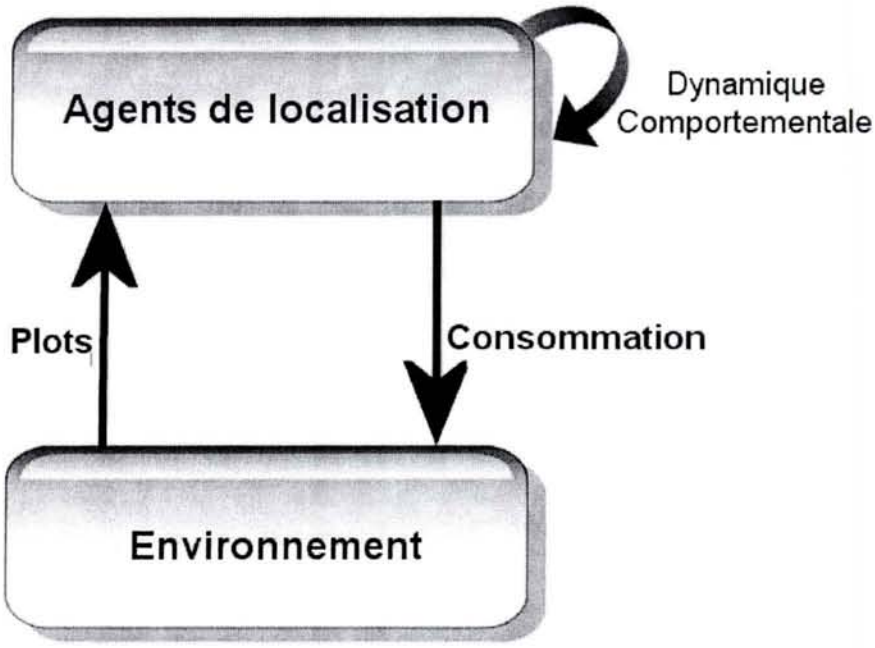

figure 17 : Architecture de notre système multi-agents

4.2. Description détaillée de l'architecture

4.2.1. Généralités

Si l'on se réfère à la méthodologie proposée par Müller [Müller98], construire un système multi-agents revient à définir une description formelle du phénomène global, une projection de ce phénomène en terme de structure d'interaction au niveau microscopique, et des comportements en adéquation avec le résultat attendu.

Nous venons de décrire le phénomène global sous la forme de deux tendances antagonistes qui sont la focalisation sur des zones d'intérêt et la répartition homogène en l'absence de cible potentielle. D'autre part, nous avons également traduit ces phénomènes globaux sous la forme de deux principes d'interaction entre les agents et avec leur environnement : attraction pour les plots et répulsion pour les autres agents. Il

ne nous reste donc plus qu'à définir de façon précise les items comportementaux dont on va doter nos agents.

Pour notre processus de résolution, nous avons choisi une approche d'inspiration physique. Les phénomènes physiques d'attraction et de répulsion sont des phénomènes étudiés depuis longtemps en mécanique classique tant au niveau des interactions entre les entités astronomiques (planètes, corps célestes, soleil,...) qu'au niveau particulaire (interactions entre les électrons et les noyaux atomiques par exemple). De ce fait, nous disposons d'exemples de formulation des forces dont on peut s'inspirer pour définir les équations régissant les interactions entre les agents et avec leur environnement.

Parmi les exemples de SMA que nous avons détaillés dans le chapitre précédent, les approches de Reynolds [Reynolds87] concernant les "Boids" et celle de Zeghal [Zeghal94] sur les trajectoires d'évitement de collision sont celles qui se rapprochent le plus de la physique. Cependant, ils ne traitent que du comportement de répulsion entre les agents.

Le comportement d'attraction a très peu été abordé à partir d'une formulation physique. Pourtant celui-ci est très intéressant à mettre en œuvre car, outre le phénomène gravitationnel, il faut prendre en compte, pour les distances faibles, des interactions entre la matière afin d'éviter l'effet de fronde gravitationnelle (lorsque la distance entre les deux éléments en attraction devient faible, la force d'attraction tend vers l'infini). Ce comportement a été traité, en particulier, par Olivier Simonin [Simonin01] sous la forme d'une force physique associée à une heuristique permettant de conserver les agents proches du pôle attractif. Dans notre cas, nous ne voulons pas avoir recours à des heuristiques car nous voulons conserver une cohérence physique à l'ensemble du système.

4.2.2. L'environnement

4.2.2.1. Représentation

Parmi les représentations possibles du monde, telles que les cartes de route, les cartes géométriques ou les représentations topographiques, nous avons choisi d'utiliser une grille d'occupation. La représentation que nous avons adoptée est très proche de ce qui est utilisé pour les algorithmes de planification à l'aide de processus de Markov [Laroche99]. Le choix de la représentation du monde, et donc du problème à résoudre, est primordial pour la construction du processus de raisonnement. Dans le cadre de la localisation et du suivi de cibles, la représentation sous la forme de grille d'occupation est celle qui est plus couramment utilisée. Son principal intérêt est de permettre une

visualisation directe des résultats ainsi qu'une formalisation aisée des données fournies par les différentes sources d'informations.

Le monde réel est supposé partiellement connu, c'est-à-dire qu'il est possible de connaître ses limites physiques, sa métrique et les obstacles qui le jalonnent. A partir de ces informations, nous construisons une carte représentative du monde réel dans laquelle nous plaçons les obstacles, murs, portes. En outre, nous avons choisi de nous placer dans le cas où la représentation des positions dans l'environnement s'effectue de façon discrète. L'environnement ainsi construit est donc composé de cases correspondant chacune à une position possible des cibles. Le choix de la taille de chacune de ces cases se fait essentiellement en prenant en compte des paramètres limitatifs pour les algorithmes utilisés.

Il peut paraître limitatif d'utiliser une représentation discrète du monde réel. Cependant, étant donné les imprécisions cumulées des sources d'informations tant au niveau des capteurs qu'au niveau des traitements utilisés, il semble judicieux d'utiliser une discrétisation des positions afin de lisser quelque peu les erreurs éventuelles. En outre, certaines sources d'information ne peuvent donner que des résultats discrets.

Cependant, bien que l'environnement soit discrétisé, nous avons choisi de caractériser le déplacement des agents de façon continue afin de gérer de façon plus fine l'agglomération. En effet, il serait inintéressant de posséder des agents ayant la même taille que les plots car, dans ce cas, nous perdrions tous les avantages de l'utilisation d'une approche de type SMA. La localisation reviendrait à former un empilement d'agents sur les plots, méthode qui peut être facilement approchée en détectant directement les percepts par une méthode simple de type « top-hat ». L'intérêt d'une telle approche est la possibilité d'estimer la position de la cible avec une granularité inférieure au pas de discrétisation de l'environnement. En effet, lorsqu'une cible se trouve dans une position réelle située entre deux ou plusieurs états de l'environnement, la perception va engendrer un ensemble de plots correspondant à tous les états proches induisant par la même une focalisation des agents sur une position intermédiaire résultante de l'influence de tous les plots.

De façon alternative, il est également possible de se limiter à un déplacement discret des agents en conservant cependant un pas de déplacement (distance entre deux positions successives d'un agent) beaucoup plus petit que la taille d'un état de l'environnement.

Ainsi, l'environnement est constitué d'un ensemble de cases, chacune étant caractérisée par une valeur $Z_P(t)$, comprise entre 0 et une valeur maximum, correspondant à la possibilité de présence d'une cible (altitude du plot). Lorsque cette valeur est égale à zéro, nous parlons de case de l'environnement. Par contre, lorsque cette valeur est non nulle, on parle de plot.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

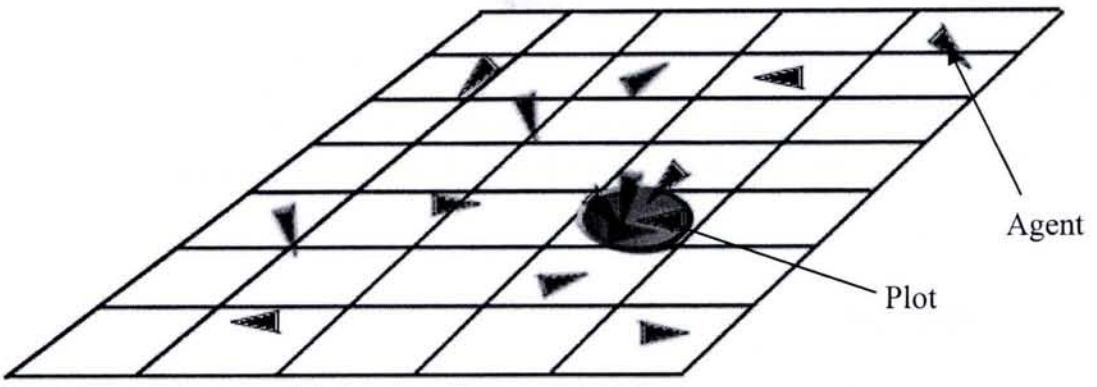

figure 18 : Agents dans leur environnement

4.2.2.2. Dynamique

Outre sa représentation, l'élément le plus important dans la définition de l'environnement des agents est la description de sa dynamique. Etant donné que l'on inscrit notre approche dans le cadre de la résolution d'un problème concret, il faut que la dynamique de l'environnement permette une représentation de celle du monde réel et des cibles que l'on doit localiser et suivre.

Les cibles réelles apparaissent, se déplacent et disparaissent, il faut donc que la dynamique de l'environnement permette une interprétation de ces trois phénomènes. Dans notre cas, ceux-ci sont traduits sous la forme de deux dynamiques d'accumulation (pour l'apparition des cibles) et d'évaporation (pour leur disparition). Le déplacement est, quant à lui, géré par la conjonction des deux dynamiques.

- Accumulation

Lorsqu'une nouvelle supposition de présence est fournie au système par une source d'information, il faut augmenter la valeur de la case correspondante selon la formule suivante.

$$Z_P(t) = Z_P(t-1) + P_C$$

Équation 2

$Z_P(t)$ représente l'altitude du plot P au temps t et P_C représente la valeur fournie par la source d'information. Ce mécanisme permet la prise en compte dans l'environnement des informations issues de plusieurs sources concernant la même cible ou des cibles différentes. La fusion de ces informations se fait par accumulation des valeurs dans les cases correspondantes de l'environnement. Pour des raisons pratiques, nous avons, en outre, choisi de plafonner les accumulations à une valeur maximum déterminée de façon empirique.

- Evaporation

Afin d'éviter l'accumulation d'informations périmées au sein de l'environnement et de gérer la disparition des cibles, nous avons décidé d'introduire un dispositif d'évaporation qui permet d'éliminer de façon progressive les plots qui sont apparus et qui ne sont pas renouvelés. Ainsi, ce mécanisme tend à faire disparaître les informations parasites, sans nécessiter un recours aux agents. L'équation régissant le phénomène d'évaporation est la suivante :

$$Z_P(t) = Z_P(t-1) * (1 - \varepsilon)$$

Équation 3

$Z_P(t)$ représente l'altitude du plot P au temps t et ε le coefficient d'évaporation compris entre 0 et 1.

- Déplacement

La gestion du déplacement des cibles par l'environnement est mise en place grâce à la conjonction des deux mécanismes d'accumulation et d'évaporation.

Par l'intermédiaire de ces phénomènes d'évaporation et d'accumulation, l'environnement possède une dynamique propre, dont le but est de maintenir une cohérence temporelle des sources d'informations. L'accumulation permet une prise en compte des nouvelles hypothèses de présence et un renforcement des anciennes alors que l'évaporation permet de contrôler la pertinence temporelle des informations tout en permettant la disparition des cibles. Ainsi, le déplacement d'une cible va générer une nouvelle hypothèse de présence correspondant à une nouvelle case de l'environnement. L'accumulation induit l'apparition de cette nouvelle position alors que l'évaporation va faire disparaître l'ancienne position connue (cf. figure suivante).

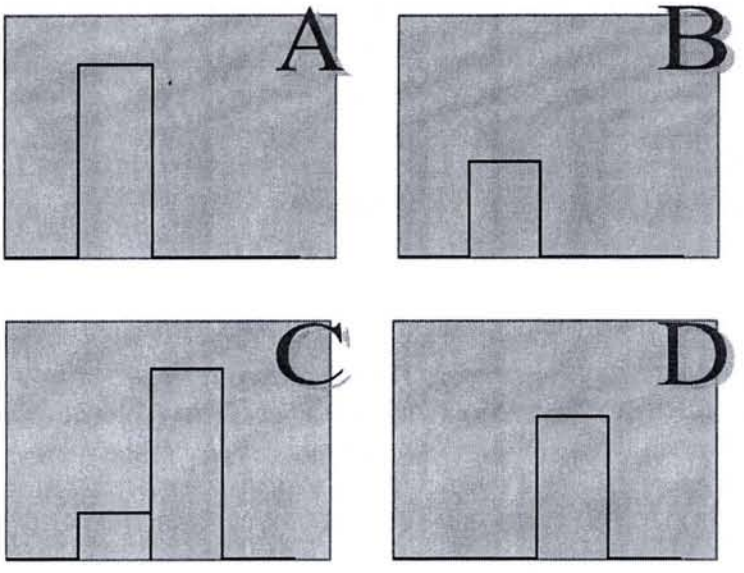

figure 19 : Prise en compte du déplacement des cibles. (A) Présence d'un plot dans l'environnement correspondant à une cible. (B) Sous le phénomène d'évaporation la hauteur du plot diminue. (C) La cible s'est déplacée formant ainsi un nouveau plot dans l'environnement (phénomène d'accumulation) L'altitude du premier plot continue à diminuer. (D) Le premier plot a complètement disparu, le nouveau commence à s'évaporer.

- Force de frottement et inertie

Pour le moment, ce que nous venons de définir concerne essentiellement le comportement propre de l'environnement. Outre cette dynamique propre, nous avons choisi d'introduire une interaction forte entre l'environnement et les agents qu'il contient, inspirée des phénomènes physiques que l'on rencontre dans le monde réel. En effet, l'environnement tel qu'il a été défini jusqu'à présent est conservatif, c'est-à-dire qu'il n'y a pas de perte d'énergie de quelque sorte que ce soit au niveau des agents. Ainsi, les effets de leurs comportements s'ajoutent à chaque pas de temps sans limitation, ce qui implique un mouvement amplifié de façon infinie. En outre, en l'absence de perte énergétique, un agent isolé, qui ne subit plus aucune interaction, continue son déplacement à vitesse constante induisant un désordre supplémentaire dans l'organisation du système, ce qui est contraire à nos objectifs.

Afin d'introduire des pertes énergétiques au cours des déplacements, nous avons choisi d'utiliser un principe inspiré des forces de frottement fluide. Ce frottement permet une inertie du mouvement des agents tout en favorisant la convergence vers un état stable.

L'expression analytique d'une telle force de frottement est la suivante :

$$\overrightarrow{F_{frottement}} = -\lambda \cdot \vec{V}$$

Équation 4

Dans cette formule, λ est le coefficient de frottement de l'environnement, il possède la même valeur quelque soit la zone de l'environnement, et $\vec{V}$ est la vitesse de l'agent considéré. Le coefficient λ sert à réguler l'influence de la force de frottement et donc l'inertie du système. Plus il sera grand, plus l'environnement s'opposera aux déplacements des agents.

4.2.3. Description des agents, de leurs comportements et des interactions

Les agents que nous allons utilisés ; nous les appellerons maintenant ***agents de localisation*** ; sont des entités virtuelles ***massiques*** qui possèdent une perception limitée de l'environnement dans lequel ils évoluent. Ainsi, ils ne perçoivent plus les plots et les autres agents au delà d'une distance faible par rapport à la taille du monde. Dans ce paragraphe, nous allons voir, dans un premier temps, comment est construit l'état interne de nos agents. Puis, nous détaillerons les différents comportements que nous avons définis.

4.2.3.1. Etat interne

Les agents de localisation possèdent un état interne qui rend compte de leur situation dans la société. Cet état interne est matérialisé sous la forme d'une mesure énergétique et d'un niveau de satisfaction.

- Energie

L'***énergie interne*** d'un agent est une grandeur qui prend en compte à la fois la proximité des autres agents sous la forme analogue à une ***énergie d'interaction*** et la présence d'un plot sous la forme d'une ***énergie potentielle***. Si l'on veut conserver une abstraction physique cohérente, il faut que l'état interne de l'agent reflète les influences des éléments présents dans l'environnement en adéquation avec les forces mises en place. Ainsi, plus les agents voisins sont proches, plus l'énergie d'interaction doit être forte. A contrario, lorsque l'agent se trouve sur un plot, son énergie potentielle doit être faible. A partir de ces considérations nous avons construit l'équation suivante pour caractériser l'état interne de l'agent.

$$E_A = -k.z_A + \sum_{A'} \frac{\lambda_{AA'}}{d^n_{AA'}}$$

Équation 5

Dans la première partie de l'équation représentant l'énergie potentielle de l'agent A, k représente la constante de gravitation pour l'agent et z_A la valeur de la case de l'environnement sur laquelle il se trouve, celle-ci pouvant être assimilée à une altitude. En ce qui concerne la constante de gravitation k, elle est en générale prise égale à ***m.g*** dans les systèmes réels avec ***m*** la masse de l'entité à considérer et ***g*** la constante de gravitation de l'environnement. Dans notre cas, cette constante sera définie de façon empirique en fonction de l'importance que l'on veut donner à l'énergie potentielle par rapport à l'énergie d'interaction. Si l'on conserve le paradigme de la mécanique classique, le fait de formuler de cette façon l'énergie potentielle en fonction de "l'altitude" de l'agent revient à considérer un plot comme étant un trou dans le continuum spatial de l'environnement ce qui nous rapproche de la représentation de la gravitation exposée par Einstein (cf. figure 20).

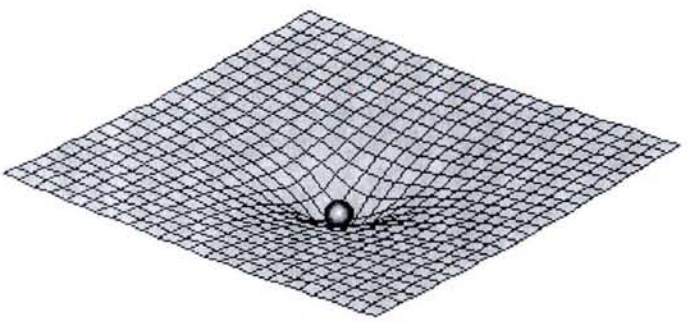

figure 20 : Espace-temps à deux dimensions incurvé par la présence d'une masse

La seconde partie de l'équation caractérise l'influence de l'énergie d'interaction de l'agent ***A***. $\lambda_{AA'}$ est le coefficient d'interaction entre les agents A et A', et $d_{AA'}$ est la distance qui les sépare. A l'instar du coefficient k de l'énergie potentielle, les coefficients $\lambda_{AA'}$ sont pris tous égaux et déterminés de façon empirique. D'autre part, *n* détermine l'ordre d'influence de la distance inter-agents dans le calcul de l'énergie. En général, les forces d'interaction sont des forces en $1/d^2$ ce qui implique que, du point de vue énergétique, l'influence de la distance est en 1/d. Il est donc légitime de prendre, dans un premier temps, une valeur unitaire pour *n*.

Si l'on veut effectuer, à l'instar de ce qui se fait habituellement en physique, le lien entre la dynamique due aux différentes forces et les variations de l'énergie interne des agents, on se rend compte que la tendance d'évolution d'un agent correspond à une minimisation de son état interne, soit en se plaçant sur un plot (diminution de l'énergie potentielle), soit en s'éloignant des autres agents (minimisation de l'énergie d'interaction)

Cette énergie interne est donc utile pour caractériser de manière énergétique les tendances d'évolution des agents. En outre, elle rend compte de la configuration spatiale locale autour de chaque agent. Cette grandeur peut éventuellement servir de point de départ à une mesure locale de la configuration spatio-temporelle du système.

Nous allons maintenant voir comment nous pouvons utiliser une partie de cette énergie pour effectuer un contrôle des différents comportements.

- <u>Satisfaction et contrôle comportemental</u>

Si nous suivons la tendance énergétique introduite dans le paragraphe précédent, nous pouvons supposer que l'agent est satisfait lorsque son énergie est minimale. A partir de cette constatation, il est donc naturel de définir un coefficient de satisfaction de l'agent qui sera déterminé comme étant égal à l'inverse de son énergie. Ainsi plus son énergie est faible, plus l'agent est satisfait.

A partir de cette notion de satisfaction, il est intéressant d'introduire une régulation entre les comportements antagonistes des agents de façon à accentuer le phénomène de focalisation lorsqu'il y a des informations dans l'environnement et à augmenter la répulsion dans les zones dépourvue de plot. Si l'on conserve la totalité de l'expression de l'énergie, il est impossible d'introduire une telle régulation car la formulation générale prend en compte à la fois la satisfaction due à la présence d'un plot et celle due à la répulsion. Afin de faciliter l'introduction de cette régulation, il est donc judicieux de ne conserver qu'un seul des deux éléments présents dans le calcul de l'énergie de l'agent. Nous avons donc choisi d'utiliser uniquement l'énergie potentielle pour caractériser la satisfaction de l'agent car celle-ci représente davantage l'objectif global du système qui est la détection de zones d'intérêt. La satisfaction d'un agent est donc définie de façon proportionnelle à l'altitude de ce dernier :

$$S_A = k\, z_A$$

Équation 6

Lorsque la satisfaction est grande, agent sur un plot par exemple, nous introduisons une diminution de la répulsion qu'il subit afin de conserver une cohésion du groupe sur la cible. Par contre, lorsque cette valeur est faible, le comportement de répulsion devient prépondérant, favorisant ainsi l'équi-répartition des agents et le retour à l'état stable fondamental.

La notion de satisfaction que nous venons de définir permet donc d'inhiber le comportements de répulsion des agents se trouvant sur un plot ce qui autorise plusieurs agents à rester sur le même plot et par conséquent à former un groupe.

4.2.3.2. Les comportements des agents

Les comportements des agents sont de deux types : les comportements dynamiques d'attraction et de répulsion consécutifs aux interactions et le comportement de consommation dont le but est de réguler la population sur les plots. Ces comportements se déroulent de façon simultanée et leurs influences respectives sont modulées en fonction de la situation de l'agent.

- <u>La consommation</u>

La consommation des plots est un comportement dynamique qui n'influe pas directement sur la focalisation des agents. Cependant, il est primordial pour le bon fonctionnement du système. Le comportement de consommation a pour but de faire disparaître les plots de l'environnement de façon à éliminer des informations désuètes.

Ainsi lorsqu'un agent se trouve sur un plot, celui-ci diminue, à chaque pas de temps, en intensité d'un certain coefficient multiplicateur α que l'on appelle appétit de l'agent. Par conséquent, lorsque beaucoup d'agents se trouvent sur un plot, celui-ci va disparaître très rapidement induisant un exode des agents de localisation. Le comportement de consommation permet donc de réguler la population présente sur les plots en empêchant une trop grande accumulation qui se traduirait par une absence d'agents dans une autre partie de l'environnement. L'équation suivante montre la formulation mathématique que nous avons utilisée :

$$Z_P(t) = Z_P(t-1) * (1 - \alpha)^n$$

Équation 7

Dans cette équation, $Z_P(t)$ représente l'altitude (ou l'amplitude) du plot P à l'instant t, α est l'appétit d'un agent et n le nombre d'agents présents sur le plot.

- Les interactions

Les interactions que nous utilisons pour décrire les comportements des agents de localisation sont inspirées de celles développées par Reynolds et adaptées à notre problématique [Reynolds87]. Ainsi, l'alignement, la séparation et la cohésion laissent place à deux comportements d'attraction et de répulsion. En outre, le comportement d'attraction que nous présentons peut être apparenté à la poursuite de la reine chez les abeilles.

Par rapport au « flocking », le comportement d'alignement a disparu. Cet item comportemental est nécessaire dans l'approche de Reynolds car il n'y a pas, dans ce cas, de direction de déplacement imposée *a priori*. En effet, dans son approche, la direction de déplacement de l'ensemble des agents est émergente en fonction des contributions de la direction de déplacement de chacun des agents. Ici, la direction est imposée par le mouvement des plots. Si ces derniers ne bougent pas, les agents se focalisent autour du point culminant tout en oscillant à cause de l'opposition des comportements.

- La répulsion entre les agents

La répulsion peut être comparée à un comportement de gravitation entre deux corps mais pris de façon négative. Elle est destinée à maintenir une répartition uniforme des agents dans les zones dépourvues de plots. A l'instar des forces d'interaction gravitationnelle, la répulsion est une force en $1/r^2$: l'amplitude de cette force est très importante lorsque les agents sont très proches et diminue rapidement lorsque ceux-ci s'éloignent les uns des autres.

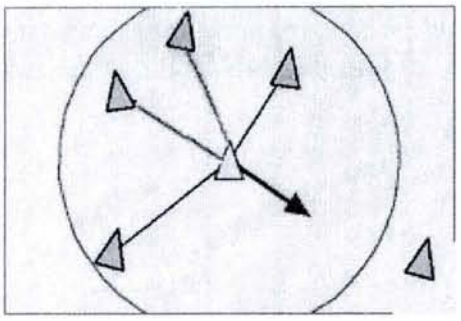

figure 21 : schéma du comportement de répulsion

L'équation suivante (Équation 8) montre l'expression analytique d'une telle force subie par l'agent A_j de part la présence de l'agent A_i dans la zone de perception. Le coefficient α est une constante multiplicative qui prend en compte la constante gravitationnelle de l'environnement et la satisfaction de l'agent A_j. D'un point de vue pratique, étant donné que l'environnement de nos agents est absolument virtuel, cette constante sert à moduler l'influence du comportement de répulsion par rapport aux autres comportements. Le vecteur reliant A_j et A_i est un vecteur de norme $\|A_jA_i\|$, en le divisant par $\|A_jA_i\|^3$, on obtient bien une force en $1/r^2$. m_i et m_j sont respectivement les masses de l'agent A_i et de l'agent A_j. Dans notre cas, la population étant homogène, les deux masses sont égales.

$$\overrightarrow{R_{ij}} = \alpha \cdot m_i m_j \frac{\overrightarrow{A_i A_j}}{\left\|\overrightarrow{A_i A_j}\right\|^3}$$

Équation 8

En appliquant l'expression analytique aux coordonnées des agents i et j, on obtient pour les coordonnées de la force de répulsion entre deux agents la formule suivante :

$$\begin{cases} Rx_{ij} = \alpha\, m_i m_j \dfrac{(x_j - x_i)}{\left((x_j - x_i)^2 + (y_j - y_i)^2\right)^{3/2}} \\ Ry_{ij} = \alpha\, m_i m_j \dfrac{(y_j - y_i)}{\left((x_j - x_i)^2 + (y_j - y_i)^2\right)^{3/2}} \end{cases}$$

Équation 9

Si on généralise la formule à tous les agents présents dans la zone de perception de l'agent j on obtient donc la formule suivante que nous avons utilisée dans notre application :

$$\begin{cases} Rx_j = \sum_{i \neq j} \left(\alpha \, m_i m_j \dfrac{(x_j - x_i)}{\left((x_j - x_i)^2 + (y_j - y_i)^2\right)^{3/2}} \right) \\ Ry_j = \sum_{i \neq j} \left(\alpha \, m_i m_j \dfrac{(y_j - y_i)}{\left((x_j - x_i)^2 + (y_j - y_i)^2\right)^{3/2}} \right) \end{cases}$$

Équation 10

Dans cette équation, x_i, y_i, m_i et x_j, y_j, m_j représentent respectivement les coordonnées et la masse des agents ***i*** et ***j***. Le coefficient α prend en compte la constante gravitationnelle de l'environnement des agents et la satisfaction de l'agent ***j***.

- L'attraction pour les plots

L'attraction s'inspire du comportement de cohésion de Reynolds. Contrairement à ce dernier, il ne s'agit pas d'une interaction entre agents mais d'une interaction entre les agents et leur environnement. Celle-ci se traduit par une attirance pour les plots et en particulier pour leur centre de gravité (centre des cases de l'environnement). La force résultante de ce comportement peut être comparée à une force de gravitation ou à une force d'interaction entre deux particules que l'on peut représenter de la façon suivante :

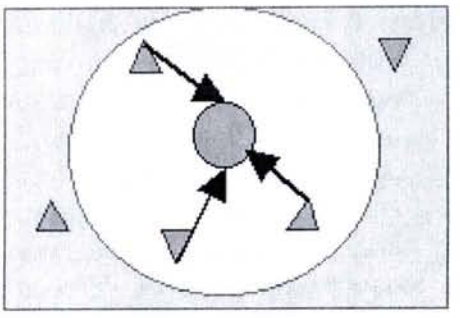

figure 22 : Schéma du comportement d'attraction

D'un point de vue analytique, il est possible de mettre ce comportement sous la forme suivante :

$$\overrightarrow{F_{\infty d}} = \beta \cdot m_A m_p \frac{\overrightarrow{A\,P}}{\left\| \overrightarrow{A\,P} \right\|^3}$$

Équation 11

Cette équation représente la force à exercer sur l'agent A due à la présence du plot P. Le vecteur reliant A à P est calculé en prenant en compte les positions respectives de l'agent et du plot dans l'environnement abstrait, la position du plot étant exprimée, dans l'espace continu des agents, comme étant la position de son centre. Les coefficients $\boldsymbol{m_A}$ et $\boldsymbol{m_P}$ sont les masses respectives de l'agent et du plot. En ce qui concerne le plot, sa masse est égal à son altitude.

La norme du vecteur étant égale à $\left\| \overrightarrow{AP} \right\|$, cette force possède bien une expression en $1/r^2$ qui impose une diminution importante de l'influence de ce comportement relativement à un accroissement de la distance entre l'agent concerné et le plot.

De la même façon que pour la force de répulsion, le coefficient β représente la constante de gravitation de l'environnement. En pratique, nous allons utiliser ce coefficient pour réguler l'influence du comportement d'attraction en fonction des autres paramètres du système.

Si on utilise la formule telle qu'elle est présentée dans l'équation précédente, le comportement d'attraction ne fournira pas le résultat escompté, c'est-à-dire la focalisation des agents sur les plots. En effet, si on regarde l'expression analytique, lorsque la distance entre les agents tend vers zéro l'amplitude de la force tend vers l'infini ce qui induit un phénomène de fronde lorsque les agents se rapprochent des plots. Ainsi, il faut améliorer le comportement d'attraction afin de réduire la force

d'attraction de façon progressive en fonction de la diminution de la distance entre l'agent et le plot. La solution proposée par Olivier Simonin [Simonin01] est l'utilisation d'une heuristique dont le but est d'inhiber la force d'attraction lorsque les agents se trouvent à proximité de la source d'attraction. Plutôt que d'introduire une heuristique, nous avons choisi de trouver une solution physiquement cohérente. Pour cela, nous nous sommes inspirés du modèle d'attraction d'une planète gazeuse en découpant le profil de la force d'attraction en trois zones en fonction de la distance par rapport au plot (zone d'attraction classique, zone atmosphérique et zone de contact avec le plot) (cf. figure 23).

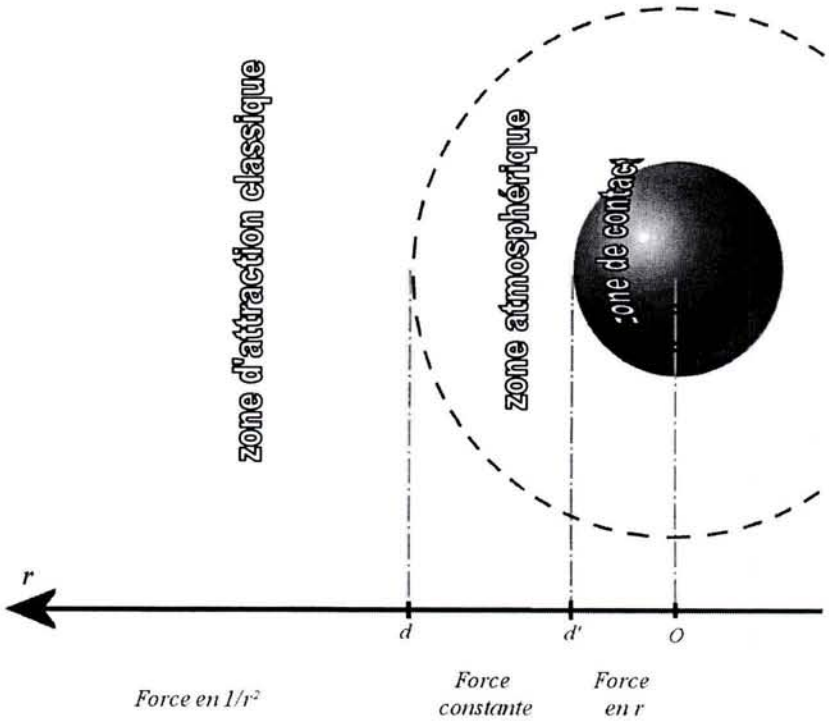

figure 23 : Les différentes zones du comportement d'attraction

La première étape, entre l'infini et une distance d du plot, se réfère à l'expression analytique que nous venons de développer (Équation 11). Ensuite, jusqu'à une distance d', le comportement d'attraction suit une loi constante dont le but est d'anihiler l'influence de la distance sur l'amplitude de la force (cf. Équation 12).

$$\overrightarrow{F_{dd'}} = \beta_1 \cdot m_A m_p \frac{\overrightarrow{A\ P}}{\left\| \overrightarrow{A\ P} \right\|}$$

Équation 12

Dans cette équation, le coefficient multiplicateur β1 sert à la fois à réguler l'influence de l'attraction et à prendre en compte la constante gravitationnelle de l'environnement. En outre, il est calculé afin d'assurer une continuité entre les différentes sections. L'équation suivante permet d'effectuer le lien entre β et β1 :

$$\beta_1 = \frac{\beta}{d^2}$$

Équation 13

Enfin, au delà de d' et jusqu'à l'approche du centre du plot, l'attraction suit une loi linéaire dont le but est de ralentir les agents et de les conserver dans la zone d'influence du plot. L'équation est alors la suivante :

$$\overrightarrow{F_{d'0}} = \beta_2 \cdot m_A m_p \cdot \overrightarrow{A\ P}$$

Équation 14

Le coefficient β2 est également calculé de façon à obtenir une continuité dans l'expression de la force :

$$\beta_2 = \frac{\beta}{d^2 \cdot d'}$$

Équation 15

On obtient donc le profil suivant pour le comportement d'attraction :

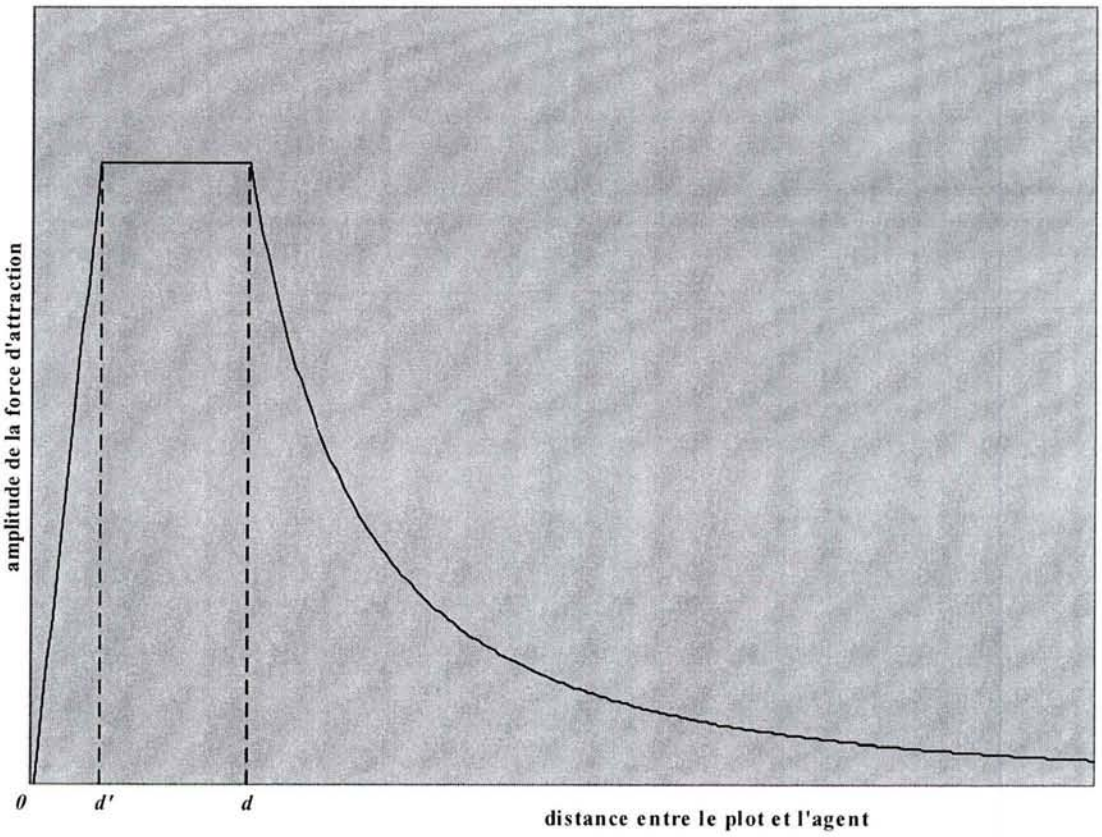

figure 24 : Profil de la force d'attraction.

En appliquant l'expression analytique aux coordonnées de l'agent A et du plot P, on obtient la formule suivante pour les coordonnées de la force d'attraction entre l'agent A et le plot P.

$$\begin{cases} Fx_{AP} = k\beta\, m_A m_P \dfrac{(x_P - x_A)}{\left((x_P - x_A)^2 + (y_P - y_A)^2\right)^{n/2}} \\ Fy_{AP} = k\beta\, m_A m_P \dfrac{(y_P - y_A)}{\left((x_P - x_A)^2 + (y_P - y_A)^2\right)^{n/2}} \end{cases}$$

Équation 16

$$\text{avec } (k,n) = \begin{cases} (\frac{1}{d'd^2},0) & si \quad \sqrt{(x_P - x_A)^2 + (y_P - y_A)^2} \in [0,d'] \\ (\frac{1}{d^2},1) & si \quad \sqrt{(x_P - x_A)^2 + (y_P - y_A)^2} \in [d';d] \\ (1,3) & si \quad \sqrt{(x_P - x_A)^2 + (y_P - y_A)^2} \in [d;\infty[\end{cases}$$

Si on généralise la formule à tous les plots présents dans la zone de perception de l'agent A, on obtient donc la formule suivante que nous avons utilisée dans notre application :

$$\begin{cases} Fx_A = \sum_i \left(k\beta \; m_A m_{Pi} \frac{(x_{Pi} - x_A)}{\left((x_{Pi} - x_A)^2 + (y_{Pi} - y_A)^2\right)^{n/2}} \right) \\ Fy_A = \sum_i \left(k\beta \; m_A m_{Pi} \frac{(y_{Pi} - y_A)}{\left((x_{Pi} - x_A)^2 + (y_{Pi} - y_A)^2\right)^{n/2}} \right) \end{cases}$$

Équation 17

$$\text{avec } (k,n) = \begin{cases} (\frac{1}{d'd^2},0) & si \quad \sqrt{(x_{Pi} - x_A)^2 + (y_{Pi} - y_A)^2} \in [0,d'] \\ (\frac{1}{d^2},1) & si \quad \sqrt{(x_{Pi} - x_A)^2 + (y_{Pi} - y_A)^2} \in [d';d] \\ (1,3) & si \quad \sqrt{(x_{Pi} - x_A)^2 + (y_{Pi} - y_A)^2} \in [d;\infty[\end{cases}$$

Dans cette équation, x_{Pi}, y_{Pi}, m_{Pi} et x_A, y_A, m_A représentent respectivement les coordonnées et la masse du plot ***i*** et de l'agent ***A***. Le coefficient β prend en compte la constante gravitationnelle de l'environnement des agents. Les paramètres ***k*** et ***n*** sont des variables contextuelles qui dépendent de la position de l'agent ***A*** par rapport à celle du plot ***Pi***.

4.2.4. Résolution des équations dynamiques

Nous avons choisi d'aborder le problème des déplacements des agents de façon continue en s'inspirant du modèle influence/réaction ([Ferber95], [Simonin01]) que l'on retrouve en physique dans le cadre de l'étude de la dynamique du point. La différence fondamentale avec le modèle influence/réaction est que les comportements ne sont plus engendrés par les agents en fonction de leur perception mais provoqués par l'influence de la présence des autres objets et agents de l'environnement. Cette approche permet de gérer plus facilement les accélérations et les changements de direction. Les déplacements des agents s'effectuent en utilisant les lois de la mécanique classique. Chaque comportement appliqué à un agent donne comme résultat un vecteur de force qui influence le mouvement. Bien évidemment, un agent possède des capacités de déplacement limitées tant du point de vue de la vitesse que de celui de l'accélération. Ceci permet d'éviter un régime d'oscillation pour la convergence des agents vers les plots.

L'application de la loi fondamentale de la dynamique permet de déterminer l'accélération de chacun des agents (Équation 18). ($\vec{\gamma}$ représente l'accélération et $\vec{F}_c$ Force engendrée par le comportement c)

$$\vec{\gamma} = \frac{1}{m} \cdot \sum_{Comportements} \vec{F}_c$$

Équation 18

L'intégration de cette dernière permet de déterminer la variation de vitesse de l'agent (Équation 19). Pour cette intégration, nous considérons que le temps est discret et que la variation entre deux pas de temps est de 1.

$$\vec{V}_t = \vec{V}_{t-1} + \frac{\Delta t}{m} \cdot \left(\vec{F}_{Répulsion} + \vec{F}_{Attraction} + \vec{F}_{frottement} \right)$$

Équation 19

En remplaçant la force de frottement par son expression et en réunissant les termes liés à la vitesse on obtient l'équation suivante :

$$\vec{V}_t(1+\frac{\Delta t}{m}\lambda)=\vec{V}_{t-1}+\frac{\Delta t}{m}\cdot\left(\vec{F}_{Répulsion}+\vec{F}_{Attraction}\right)$$

Équation 20

En intégrant une seconde fois, nous obtenons la variation de position de l'agent dans l'environnement.

$$\vec{X}_t=\vec{X}_{t-1}+\left(\vec{V}_{t-1}\cdot\Delta t+\frac{\Delta t^2}{2\cdot m}\cdot\left(\vec{F}_{Répulsion}+\vec{F}_{Attraction}\right)\right)\frac{1}{\left(1+\frac{\Delta t}{m}\lambda\right)}$$

Équation 21

En s'inspirant des lois de la physique, cette formulation du déplacement est très intéressante car elle permet une composition aisée des comportements. En particulier, il est facile de rajouter ou d'inhiber un comportement en jouant sur la présence de la force correspondante. En outre, il permet une inertie des agents qui est indispensable non seulement pour la prise en compte des positions passées des agents mais également pour l'anticipation des positions futures. Ainsi, le comportement d'un agent est une somme pondérée des comportements élémentaires, avec une pondération qui varie en fonction de la configuration locale. Le comportement résultant de cette pondération est appelé **vecteur d'intention ($\vec{I}$)** pouvant être exprimé de la façon suivante :

$$\vec{X}_t=\vec{X}_{t-1}+\left(\vec{V}_{t-1}\cdot\Delta t+\frac{\Delta t^2}{2\cdot m}\cdot\left(\vec{I}\right)\right)\frac{1}{\left(1+\frac{\Delta t}{m}\lambda\right)}$$

Équation 22

avec $\vec{I} = \vec{F}_{Répulsion}+\vec{F}_{Attraction}$

En plus de cette résolution dynamique, nous avons introduit une ***vitesse limite*** pour les agents afin d'éviter les problèmes d'amplification trop importante. Cette vitesse limite détermine directement la vitesse maximum des cibles que l'on sera en mesure de localiser. Cependant, les tests, que nous avons effectués et que nous présentons dans la

suite de ce travail, montrent que les agents, bien que bridés du point de vue de la vitesse, sont capables de suivre des cibles jusqu'à deux fois plus rapides.

4.3. Résultats et fonctionnement

- Résultats

Le résultat du processus de résolution est caractérisé par une formation de groupes autour des plots. Cette formation s'effectue en fonction de l'altitude des agents et en fonction de la proximité des voisins. Lorsqu'un agent, placé sur un plot, se trouve proche de ses voisins (voisinage de deux ou trois individus au plus) il y a formation d'un groupe. La détection des groupes se fait par la méthode des k-means. Ceci est un choix de notre part, on aurait très bien pu envisager une détection de groupe élémentaire effectuée en parcourant l'ensemble des positions des agents et en les regroupant deux par deux.

Etant donné la nature des comportements inter-agents, ceux-ci ne peuvent être proches les uns des autres que s'ils se trouvent sur un plot. Ainsi, dès que deux agents sont suffisamment proches, il y a formation d'un groupe. La position de ce groupe, et donc la position de la cible qu'il matérialise, correspond à la position du barycentre du groupe d'agents.

L'intérêt des groupes est de matérialiser la focalisation sur les plots et donc la localisation des entités mobiles.

- Fonctionnement

Lors de l'initialisation du système, les agents sont répartis de façon aléatoire dans l'environnement.

En l'absence de percept, seule la force de répulsion dicte le comportement de chacun des agents. Il en résulte une équi-répartition des agents dans tout l'environnement de façon à ce qu'ils soient le plus éloigné possible les uns des autres. Cette répartition spatiale constitue l'état stable fondamental du système. D'un point de vue énergétique, il correspond à une maximisation de l'énergie de chacun des agents ce qui peut être assimilé à un but local. En effet, en l'absence de percepts, l'énergie potentielle de tous les agents est nulle. La seule façon de maximiser leur état interne est donc de minimiser la partie correspondant à la contribution des interactions. Ceci est réalisé lorsque la distance entre les agents est la plus grande possible.

L'arrivée d'un stimulus perceptif perturbe l'état d'équilibre du système. Les agents proches du percept introduit vont tendre à être attirés par ce dernier. Ceci va se traduire par une agglomération d'un certain nombre d'agents sur la position de la perturbation. Les autres agents, n'étant pas attirés par le percept, vont occuper l'espace laissé libre. Si on ne prend pas en compte le comportement de consommation, le système tend vers un autre état stable qui correspond à un état de maximum énergétique pour chacun des agents.

Une fois que les agents se trouvent sur un plot, ceux-ci vont le consommer : c'est-à-dire qu'ils vont diminuer l'altitude du plot correspondant. Une fois que celui-ci est totalement consommé, il y a diminution de la valeur de satisfaction et donc une séparation des agents. Deux cas de figure sont alors possibles : soit l'objet à localiser est toujours détectable, soit il n'est plus détecté du tout.

Dans le premier cas, si l'objet n'a pas bougé, le groupe n'ayant pas changé de place, l'état de stabilité est déjà atteint. Si l'objet s'est déplacé, le système va évoluer vers un autre état de stabilité (focalisation des agents proches). Si la dynamique de rafraîchissement est suffisamment rapide le déplacement va être très faible. Ainsi, les agents se trouvant sur le plot, une fois l'état de stabilité atteint, seront essentiellement ceux qui faisaient partie d'un premier groupe créé. Dans ce cas, on considère que le groupe est le même que celui présent au pas de temps précédent ($g(t)=g(t+1)$). L'enchaînement de ces positions successives permet de construire les trajectoires des cibles suivies.

Dans le cas où l'objet n'est plus détecté du tout le système revient dans son état stable fondamental et le groupe disparaît.

4.4. Discussion

Dans ce chapitre, nous avons décrit le processus de résolution à base d'agents réactifs que proposons. Pour cela, nous avons défini :

- les agents et leurs variables internes,
- les mécanismes d'interaction entre les agents et avec les plots présents dans l'environnement
- la dynamique de l'environnement et son influence sur les interactions
- le résultat collectif émergent désiré (focalisation et équi-répartition)

Conformément à la piste que nous avons décidé de suivre, les agents que nous avons construits sont bien de type réactif. Ils ne possèdent pas de représentation ni de la tâche individuelle qu'ils doivent accomplir (attraction pour les zones d'intérêt) ni de la tâche collective (construction de groupes sur les plots). Leurs perceptions se limitent à la

détection des plots et des autres agents, faite de façon indirecte lorsque les agents se trouvent dans l'une ou l'autre des régions d'influence des forces d'attraction et de répulsion. Leurs actions sont conditionnées à la fois par les forces qu'ils subissent et par leur situation (consommation des plots liée à l'altitude de ceux-ci)

En ce qui concerne les interactions, nous avons choisi une formulation physique du même type que celle que l'on peut rencontrer en physique de la matière ou en astrophysique avec quelques adaptations aux contraintes des SMA. Nous nous sommes attachés à maintenir une cohérence physique à l'ensemble des interactions. En particulier, nous avons choisi d'utiliser une formulation de la force d'attraction analogue à celle générée par une planète gazeuse, certes simplifiée, plutôt que d'avoir recours à des heuristiques d'inhibition de comportement pouvant paraître artificielles.

Pour représenter l'environnement, nous avons utilisé une grille d'occupation comme c'est souvent le cas en localisation et en suivi. Ce choix est adapté au problème que nous voulons résoudre car il permet, d'une part, une interprétation naturelle des résultats fournis par les processus et, d'autre part, une gestion simplifiée de l'intégration des sources d'information. En outre, la comparaison avec les autres algorithmes de localisation et de suivi est plus aisée.

Le processus que nous venons de décrire respecte bien les contraintes que nous nous étions fixées en introduction :

- L'environnement est partiellement connu (on ne connaît que sa métrique).
- Il n'y a pas de connaissance *a priori* sur les sources d'informations.
- Le processus de résolution ne nécessite pas de construction ou de manipulation de représentations complexes du problème à résoudre.
- Les cibles ne sont pas connues à l'avance.
- Le raisonnement et la prise de décision sont simples et de type stimulus/réponse.

Par rapport à l'architecture que nous avons proposé en introduction, il manque encore deux éléments à déterminer. Dans un premier temps, il faut vérifier que les mécanismes mis en place au niveau microscopique (agents et interactions) sont capables de fournir un résultat collectif, interprétable de façon macroscopique, conforme à ce qui est désiré. Dans un second temps, il faut définir un mécanisme permettant de construire, à partir d'observations du monde réel, un environnement adapté au processus de résolution que nous venons de définir et à ses contraintes. Ce mécanisme a pour but d'alimenter en percepts l'environnement des agents de localisation.

Chapitre 5 : Tests simulés : mise en œuvre et résultats

Si le chapitre précédent avait pour but de décrire de façon détaillée l'architecture de notre système multi-agents, il s'agit maintenant de tester le bon fonctionnement de celui-ci tout en dégageant un certain nombre de caractéristiques et de qualités qui illustrent le bien fondé de nos choix de conception. Enfin, il faut également évaluer les performances de ce système pour la tâche qui lui est destinée même si ce dernier point n'est pas le plus important puisque notre but n'est pas de construire une méthode de localisation et de suivi ultime qui rivalise en performance avec les autres méthodes couramment utilisées [Thrun99] mais de définir un système réactif, simple et disposant de capacités décisionnelles limitées pouvant néanmoins résoudre une tâche complexe à partir de sources d'informations diverses et distribuées.

L'objectif de ces tests est double. Dans un premier temps, nous voulons vérifier que les mécanismes microscopiques définis dans le chapitre précédent fournissent la réponse collective désirée tout en mettant en évidence les propriétés liées au modèle. Dans un second temps, il s'agit de quantifier les performances du processus par rapport à la tâche de localisation.

5.1. Mise en œuvre

5.1.1. Premiers essais avec Starlogo®

Avant d'effectuer des essais de l'ensemble du système en situation réelle avec de vrais robots. Nous avons effectué des simulations sur différents outils destinés à la construction de systèmes multi-agents.

Dans un premier temps nous avons testé la viabilité des comportements des agents à travers l'outil Starlogo® [Starlogo]. Le but de ces tests était de vérifier que les agents se focalisaient correctement sur les plots tout en conservant une répartition homogène sur les zones non stimulées. Le calcul des déplacements s'effectuait alors de façon discrète pour chacun des agents en fonction de ses perceptions. Les plots étaient générés de façon aléatoire et leur influence sur les agents était amplifiée par un phénomène de diffusion analogue à celui des phéromones. Ces premiers tests ont permis de mettre en évidence la pertinence des comportements choisis tout en mettant en avant des limitations dues notamment au caractère discret de la matérialisation des déplacements et au modèle utilisé par l'outil de conception. Avec cet outil de développement, il est notamment impossible de gérer de manière acceptable les variations de vitesse et d'accélération. Par conséquent nous avons choisi d'utiliser une autre plate-forme de développement de SMA permettant de gérer les interactions et la représentation de l'environnement à un plus bas niveau d'abstraction.

5.1.2. Utilisation de Madkit®

Ensuite, nous avons implanté cette méthode de gestion des comportements à l'aide de la plate-forme agent Madkit® [Madkit] développée par J. Ferber et O. Guknecht. En particulier, nous avons utilisé le moteur synchrone proposé par cet outil pour gérer les différentes dynamiques de notre système.

L'architecture de test est composée de trois parties :

- Un étage de construction de l'environnement constitué de générateurs de plots
- Le processus de résolution
- Un étage de détection des groupes formés

Les générateurs de plots sont des éléments logiciels qui fournissent une succession de plots situés soit sur une position fixe de l'environnement soit sur une trajectoire choisie. Il nous est donc possible de générer :

- des plots statiques à une position donnée.
- des plots répartis de façon aléatoire dans l'environnement.
- des trajectoires rectilignes.
- des sinusoïdes.
- des trajectoires rectilignes doubles permettant d'engendrer des collisions et des séparations de cibles.

En outre, le générateur de plots est capable de fournir du bruit sur les trajectoires émises : soit sous la forme d'un bruit blanc uniformément réparti sur la totalité de l'environnement (plots parasites), soit sous la forme d'un bruit de décalage au niveau de la trajectoire ce qui équivaut à introduire une imprécision sur les informations émises par le générateur de plots.

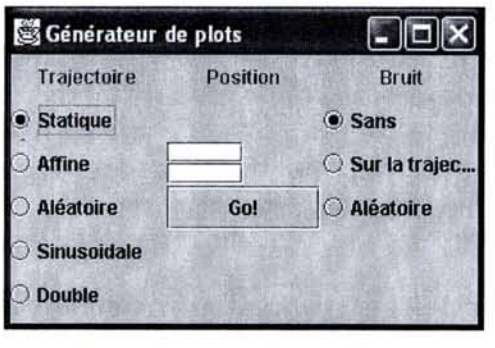

figure 25: Le générateur de plots

La détection des groupes s'effectue à partir des positions des agents fournies par le processus de résolution. Cette étape s'effectue de façon séparée car elle n'est pas conceptuellement liée au processus mais correspond davantage à une interprétation du résultat collectif fourni par ce dernier.

Afin de faciliter la mise en œuvre de l'architecture sur plusieurs machines, chacune de ces parties communique avec les autres par l'intermédiaire de « socket » La structure de l'application de test est donc la suivante :

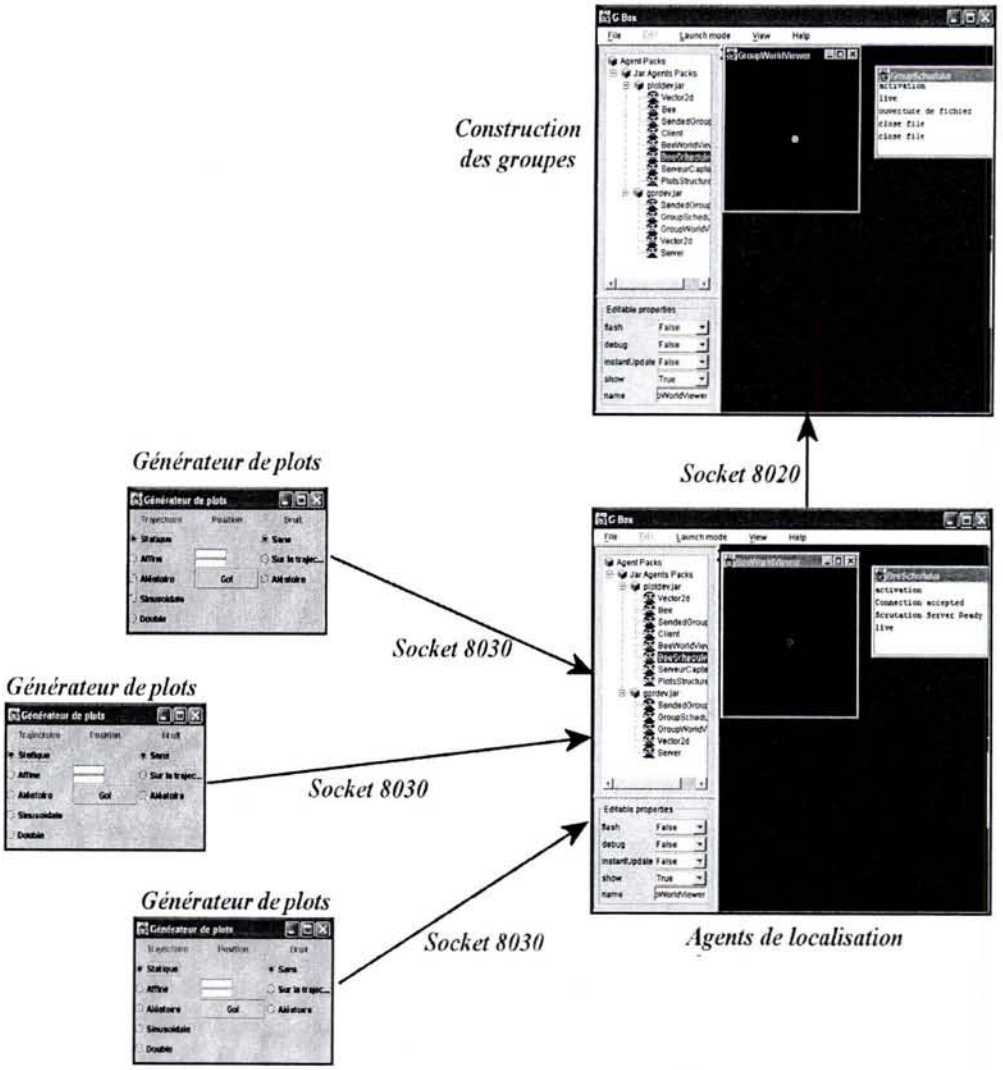

figure 26 : Structure de l'architecture de test

5.2. *Tests et résultats*

Comme nous l'avons déjà dit précédemment, lors de ces tests, nous voulons à la fois montrer les propriétés liées au modèle utilisé et quantifier les performances du processus relativement à la tâche de localisation.

Les tests que nous avons effectués sont donc les suivants :

- Tests préliminaires destinés à vérifier le bon fonctionnement du système en adéquation avec la tâche à réaliser.

- Tests avec plusieurs générateurs de plots dont le but est de montrer l'indépendance du processus par rapport au nombre de sources d'informations

- Tests de localisation avec un nombre variable de cibles afin de mettre en évidence la possibilité de gérer plusieurs cibles de façon simultanée sans connaissance *a priori* de leur nombre.

- Tests de localisation statique et dynamique (vitesse constante, et trajectoire sinusoïdale) dont le but est de mesurer les erreur de localisation en fonction d'une trajectoire connue à l'avance pour les plots.

- Tests de localisation avec un nombre variable d'agents.

- Tests de séparation à partir de deux trajectoires proches ou se rapprochant destinés à mesurer la distance minimum décelable entre deux plots proches.

- Tests de robustesse aux bruits avec un bruit blanc sur l'ensemble de l'environnement ou un bruit blanc uniquement sur la trajectoire. Le but de ces tests est de montrer comment le système se comporte en cas de déviation par rapport au fonctionnement idéal.

- Tests de suivi, à partir de trajectoires connues, destinés à mesurer l'erreur de suivi (prise en compte du décalage temporel entre la position du plot et la position détectée).

En ce qui concerne les tests qualitatifs, pour lesquels il y a une mesure de l'erreur de localisation, nous avons effectués entre une vingtaine et une cinquantaine d'expérience avec les mêmes types de trajectoire et les mêmes paramètres de simulation. Les résultats exposés sont donc des résultats moyens sur l'ensemble des expériences effectuées.

5.2.1.1. Tests préliminaires

5.2.1.1.1. Tests de fonctionnement

Dans un premier temps, nous avons effectué une série de tests permettant de mettre en évidence la viabilité des comportements dont sont dotés nos agents. Nous voulons à la fois tester si le processus fourni bien une réponse collective sous la forme d'une focalisation sur les plots et évaluer l'influence du nombre d'agents sur l'état stable en absence de plot.

Pour cela, nous avons généré des plots statiques à des positions connues de l'environnement et nous avons observé l'évolution de la structure spatio-temporelle des agents sous l'influence des comportements et des interactions.

Le premier de ces tests a consisté en la vérification de l'obtention de l'état stable fondamental en l'absence de percepts (cf. figure 27)

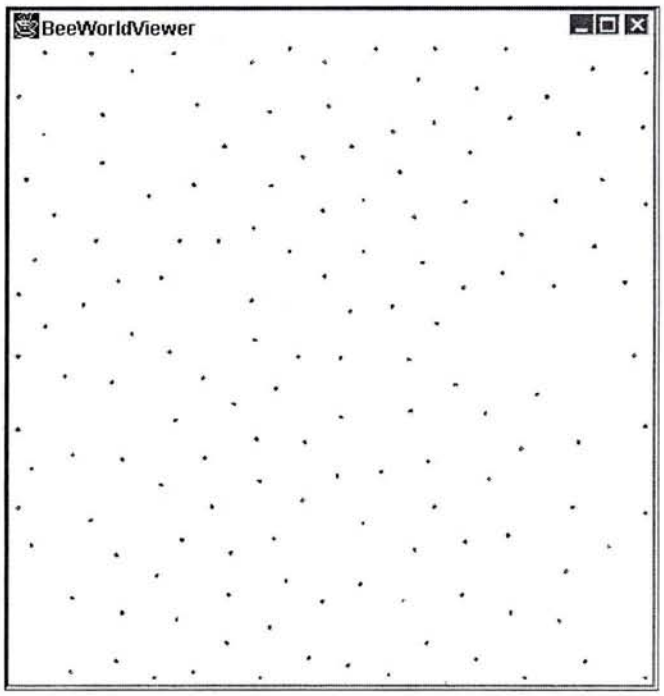

figure 27 : l'état stable fondamental

Ensuite, nous avons vérifié que l'arrivée d'un percept modifiait bien l'état d'équilibre du système de façon à obtenir une focalisation d'agents sur le plot créé formant ainsi un groupe (cf. figure 28). Dans la deuxième partie de cette figure, nous pouvons remarquer que les agents, qui ne sont pas attirés par le plot, n'ont pas comblé l'espace laissé libre autour de ce dernier. Ceci est dû aux paramètres utilisés pour réaliser cette illustration qui ont été définis de telle sorte que les agents non attirés restent dans un état d'équilibre autour de la zone d'attraction du plot. Pour cela il a suffit de limiter la portée du comportement de répulsion.

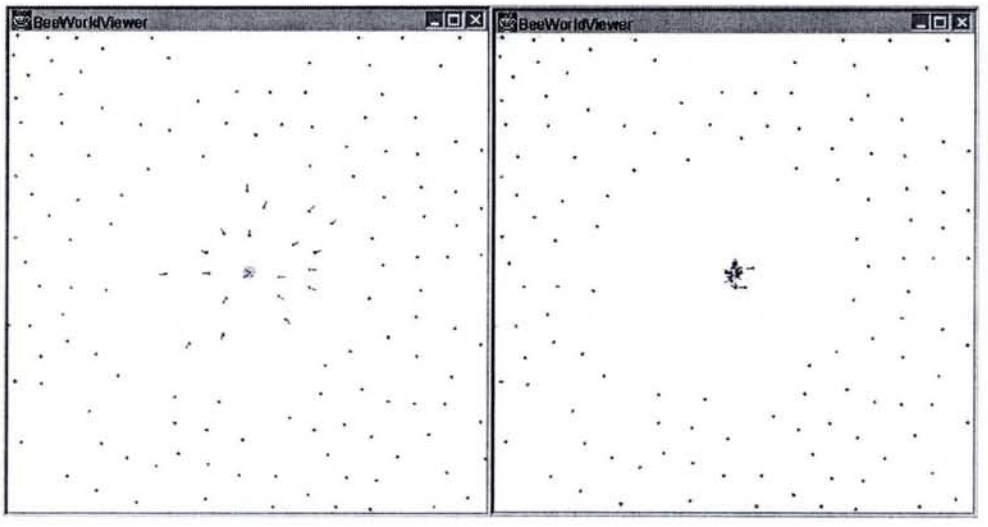

figure 28 : Convergence du système vers un état stable après l'apparition d'un percept

Enfin, nous avons testé que le fait que la disparition du plot induisait un retour du système dans l'état stable fondamental (cf. figure 29).

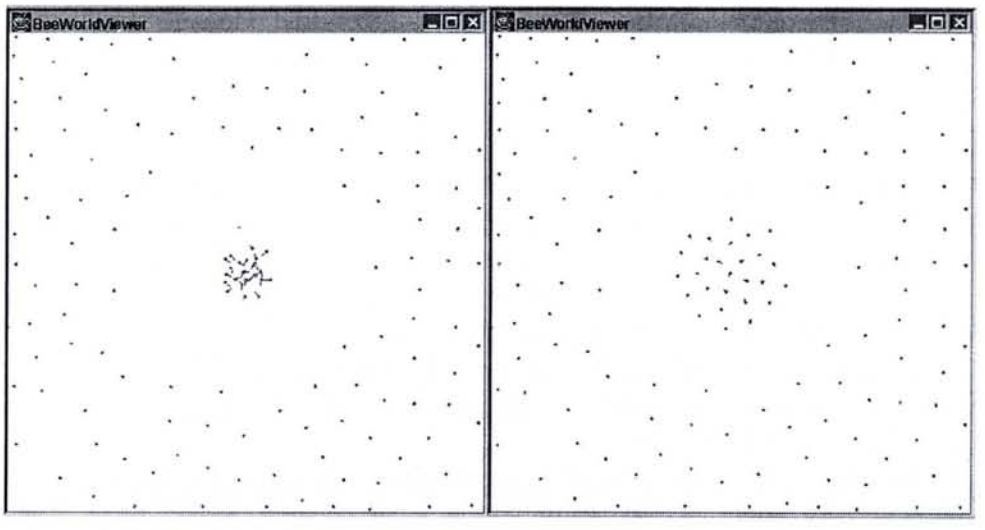

figure 29 : Retour du système dans l'état stable fondamental

Tous ces tests préliminaires ont été effectués avec des nombres d'agents différents. Le fonctionnement du système n'est pas perturbé par la variation du nombre d'agents. Cependant lorsque le nombre est très faible devant la taille de l'environnement la focalisation ne se fait pas de façon aussi naturelle car la distance entre les agents et le plot peut être très importante par rapport aux ordre de grandeurs mis en jeu dans les comportements. En outre, dans ce cas, l'état d'équilibre fondamental est rapidement obtenu. Aussi, dans certains cas où le nombre d'agents est vraiment très faible, il est tout à fait possible que le plot ne soit pas détecté.

Dans le cas contraire, lorsque les agents sont très nombreux l'état stable est très difficile à obtenir (cf. figure 30). Il y a en fait une oscillation permanente des agents. Par contre, lorsqu'un percept fait son apparition, celui-ci est immédiatement détecté. Enfin, il est également possible d'ajouter ou de retirer des agents en cours de fonctionnement sans perturber le fonctionnement de l'ensemble. Le système tend alors vers un nouvel état d'équilibre.

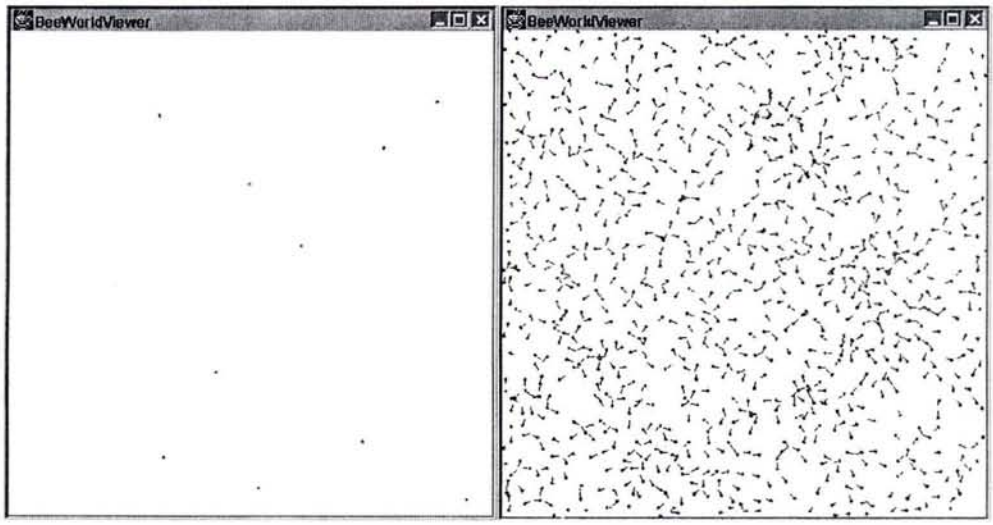

figure 30 : Influence du nombre d'agents sur l'équilibre du système

Ces tests montrent que les mécanismes d'attraction et de répulsion permettent de fournir la réponse collective attendue. D'autre part, ils montrent également que le nombre d'agent n'influe pas sur le résultat. Cependant, il est plus judicieux d'utiliser un nombre d'agents suffisamment important, du même ordre que la taille de l'environnement, afin de faciliter la mise en œuvre des interactions.

De manière empirique, nous nous sommes rendu compte qu'il était impossible de maintenir un état stable non oscillant lorsque le nombre d'agent était plus grand que la racine du nombre total d'états.

Pour la suite des tests, nous avons choisi d'utiliser un environnement d'une taille raisonnable pour une tâche de localisation (200 états sur 200 états) avec un nombre d'agents suffisamment important au regard de la taille de l'environnement (150 agents).

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

Ce nombre d'agent permet d'obtenir un état stable fondamental dépourvu d'agitation tout en maintenant une zone de couverture suffisamment importante.

5.2.1.1.2. Tests avec plusieurs générateurs de plots et avec plusieurs cibles

Le but de ces tests est de mettre en évidence l'indépendance du processus de résolution vis-à-vis du nombre de sources d'information et donc du nombre de générateurs disponibles. Nous voulons, en outre, vérifier le bon fonctionnement des dynamiques d'accumulation et d'évaporation.

Pour effectuer ces tests, nous avons donc utilisé plusieurs générateurs de plots sur des machines différentes, fournissant chacun des plots dans l'environnement des agents. En ce qui concerne le choix de la nature des plots, nous nous sommes limités à l'utilisation de plusieurs trajectoires rectilignes uniformes et de plots fixes fournis par plusieurs générateurs de façon simultanée. En outre, nous avons fait en sorte que le nombre de générateurs simultanément connectés varie en cours de fonctionnement de façon à vérifier que la connexion et la déconnexion des générateurs n'influent pas sur le processus.

L'évaluation des résultats a été faite à partir d'observations sur le résultat fourni et sur sa dynamique.

D'un point de vue local, nous avons observé que les agents se comportent de la même façon que lorsqu'il n'y a qu'un seul plot dans l'environnement. Par contre, lorsque le nombre de cibles est un peu trop important (environ la moitié du nombre d'agents), il n'y a plus localisation de toutes les cibles.

D'autre part, il n'y a pas de ralentissement au niveau de l'exécution ni de perturbation de fonctionnement lorsqu'on modifie le nombre de générateurs.

Enfin, il n'y a aucune différence entre l'utilisation du processus avec plusieurs générateurs fournissant chacun un plot et celle avec un seul générateur fournissant le même nombre de plots.

5.2.1.2. Tests de localisation

Ces tests de localisation ont plusieurs buts. Par leur intermédiaire, il s'agit de :

- vérifier que l'organisation émergente correspond bien à une focalisation sur les régions contenant une cible potentielle et matérialisée par un ou plusieurs plots.

- mesurer l'erreur de localisation c'est-à-dire la distance qui sépare la position des groupes formés de la trajectoire effective du ou des plots.
- mesurer la variation de performance en fonction de la vitesse relative des plots par rapport à la vitesse maximale des agents. En particulier, nous voulons voir si les agents se comportent conformément aux prévisions pour des vitesses de plots proches de la vitesse maximale définie pour les agents. Nous voulons également mettre en évidence les déviations éventuelles de la réponse du système lorsque la vitesse des plots est plus grande que celle des agents.

Ces tests ne prennent pas en compte l'aspect dynamique, c'est-à-dire que les mesures d'erreur sont faites à partir de la distance la plus courte par rapport à la trajectoire idéale. Il est donc possible d'avoir un retard temporel du groupe d'agents focalisés par rapport au plot sans que cela soit mis en évidence par ces premiers résultats. Le but est ici d'estimer la capacité à rester localisé sur une trajectoire définie.

- <u>Tests de localisation statique</u>

Pour ce premier test, nous avons choisi d'utiliser un cas élémentaire de localisation, puisqu'il va s'agir de trouver la position d'un plot statique dans l'environnement des agents. Par cette expérience, nous voulons à la fois vérifier l'efficacité du système dans un cas très simple tout en évaluant la stabilité de la réponse obtenue si l'on réitère un grand nombre de fois l'expérience.

Pour estimer la performance du processus pour la tâche de localisation, nous avons calculé, dans chacune des expériences, la distance moyenne séparant le centre du plot généré (centre de la case) du centre de gravité du groupe d'agents focalisés.

Dans un premier temps, en ce qui concerne la stabilité de la réponse, les tests montrent à chaque fois des résultats similaires quant à la mesure de l'erreur moyenne de localisation et de son écart type.

Le tableau suivant montre les résultats moyens que nous avons obtenus au cours d'une cinquantaine d'essais :

	en X		en Y	
	en nombre d'états de l'environnement	en pourcentage par rapport à la taille du monde	en nombre d'états de l'environnement	en pourcentage par rapport à la taille du monde
Erreur moyenne	0,1892	0,09%	0,2947	0,15%
Ecart Type	0,3884	0,19%	1,0316	0,52%

figure 31 : Tableau montrant l'erreur et l'écart type de la localisation statique pour un monde de taille 200 x 200 et pour 150 agents.

Ces résultats sont plutôt intéressants car ils montrent une bonne précision de la localisation statique. Cependant, il est possible de remarquer que, contrairement à des algorithmes classiques de détection de plot, il y a une certaine imprécision résiduelle. Celle-ci est essentiellement due au caractère dynamique du processus de résolution. En effet, sous l'action du comportement de consommation et de la variation de la satisfaction des agents, une oscillation apparaît car, si le groupe reste globalement à la même place, les agents qui le compose ne sont pas forcément toujours les mêmes.

- Tests avec un nombre variable d'agents

Afin de vérifier qu'il est possible de faire varier le nombre d'agent en cours de fonctionnement, nous avons effectué des tests dans les mêmes conditions que précédemment. Nous avons donc généré un plot à une position fixe donnée de l'environnement tout en faisant varier le nombre d'agent de façon aléatoire entre +10% et -10% du nombre initial. Nous avons ensuite mesuré la distance moyenne entre le centre du plot et la position du groupe (centre de gravité).

Les résultats obtenus sont tout à fait similaires à ceux obtenus précédemment, erreur moyenne de l'ordre de 0,1% et écart type de l'ordre de 0,2%. Ceux-ci montrent donc que le nombre d'agents peut varier en cours de fonctionnement, dans des limites raisonnables, sans pour autant affecter la qualité du résultat collectif.

- Tests de localisation dynamique

Maintenant que nous avons mesuré l'erreur statique de localisation, nous allons procéder à la mesure de l'erreur dynamique. Afin de mesurer cette erreur, nous avons généré des trajectoires rectilignes et sinusoïdales en faisant varier les paramètres de vitesse ou de fréquence d'oscillation. Pour chaque trajectoire, nous avons mesuré la distance séparant la position des plots de la trajectoire idéale qui est bien évidemment connue. Dans ces tests, nous n'avons pas pris en compte le décalage temporel éventuel

entre le plot et le groupe d'agents car nous sommes restés dans des zones de fonctionnement assez proches du cas statique (vitesse des cibles très lente) ne permettant pas de mesurer de façon satisfaisante un éventuel décalage temporel. En outre, nous voulons mesurer ici la capacité des groupes à rester sur une trajectoire définie et non pas la capacité de ceux-ci à suivre une cible.

Pour réaliser ces tests, nous avons généré des plots parcourant une trajectoire rectiligne dans un premier temps, et une trajectoire sinusoïdale ensuite. De la même façon que pour le cas statique, nous avons mesuré la distance moyenne entre la position du groupe et son équivalent sur la trajectoire idéale suivi par la cible.

La figure 32 et la figure 33 montrent les premiers tests effectués avec une trajectoire rectiligne à vitesse constante, égale à la moitié de la vitesse maximum des agents. Le tableau qui suit présente les résultats moyens obtenus pour un grand nombre de passages.

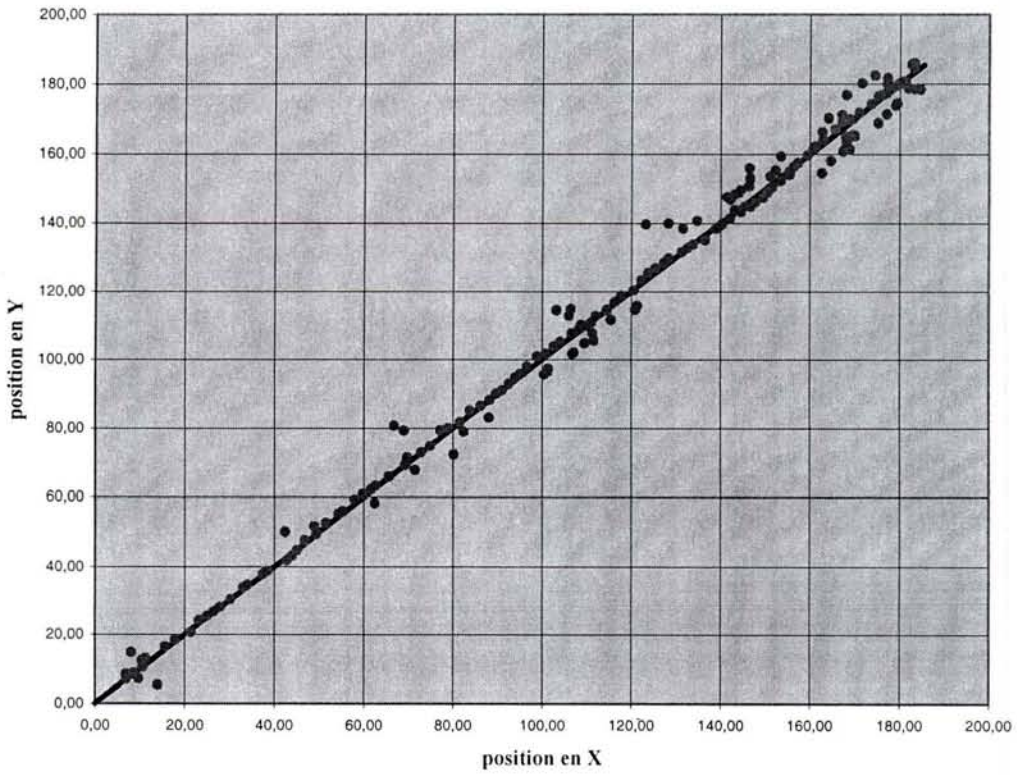

figure 32 : Suivi de trajectoire rectiligne uniforme avec une vitesse relative égale à 0,5. La droite représente la trajectoire du plot et les points représentent les positions détectées des groupes.

	en X		en Y	
	en nombre d'états de l'environnement	en pourcentage par rapport à la taille du monde	en nombre d'états de l'environnement	en pourcentage par rapport à la taille du monde
Erreur moyenne	1,6620	0,83%	0,3258	0,16%
Ecart Type	4,7961	2,40%	4,6158	2,31%

figure 33 : Tableau montrant l'erreur et l'écart type de la localisation dynamique sur une trajectoire rectiligne pour un monde de taille 200 x 200 et pour 150 agents.

L'erreur et l'écart type obtenus lors de ces tests sont nettement plus élevés que dans le cas de la localisation statique. Néanmoins, ces valeurs restent tout à fait acceptables aux vues de la nature des mécanismes utilisés.

Pour les seconds tests, nous avons utilisé une trajectoire sinusoïdale. Ce choix n'est pas anodin, car ce type de courbe permet de mettre en évidence le comportement du système lors de changements de trajectoire importants, au niveau des extrema, et faibles, au niveau des points d'inflexion. Ces tests ont, en particulier, mis en évidence les effets indésirables de l'inertie des groupes d'agents.

La

figure 34 montre un exemple de suivi de trajectoire sinusoïdale. Quant au tableau suivant, il montre les calculs d'erreur moyenne obtenue au cours de ces tests.

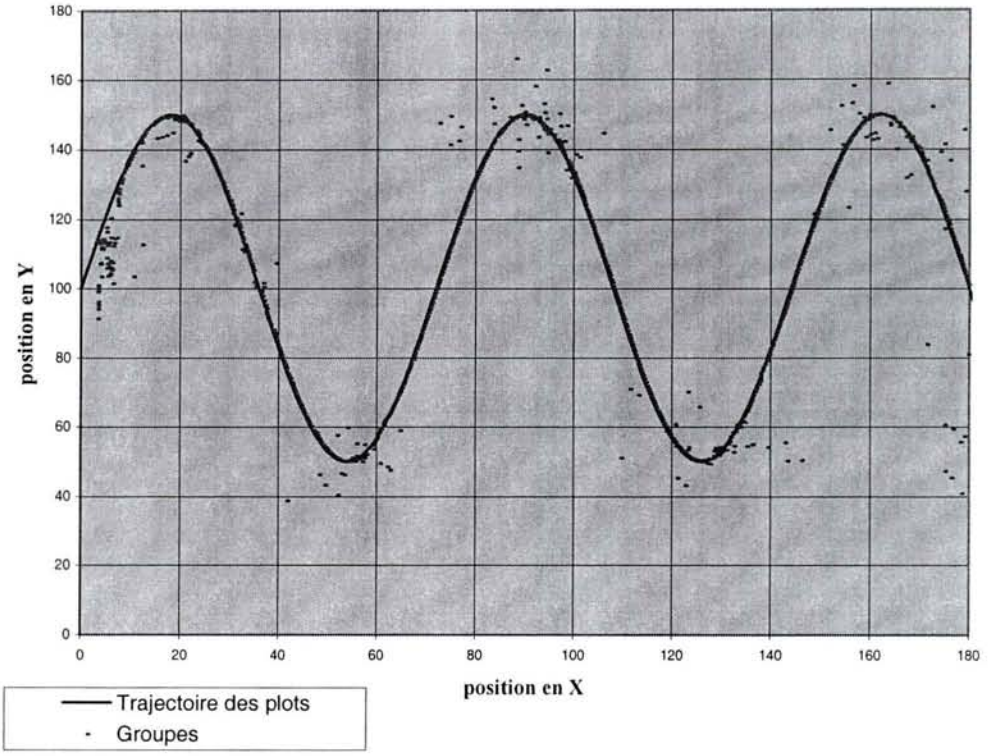

figure 34 : Suivi de trajectoire sinusoïdale avec une vitesse relative inférieure ou égale à 0,5

	Erreur moyenne		Ecart Type	
Global	7,14517826	3,57%	7,55280799	3,78%
Régions linéaires	4,95141931	2,48%	4,995238	2,50%
Régions de changement d'orientation	8,4877012	4,24%	8,93946402	4,47%

figure 35 : Mesures de l'erreur moyenne et de l'écart type pour une trajectoire sinusoïdale pour différentes portions du parcourt

Dans le cas de la trajectoire rectiligne à vitesse constante, le système se comporte bien avec une erreur relative proche de celle obtenue en localisation statique. Ceci était prévisible : étant donné que la vitesse ne change pas en orientation, l'inertie implique que les agents et donc que le groupe conserve la même trajectoire, le sens du déplacement étant induit par le déplacement des plots.

Dans le cas sinusoïdal, le système se comporte de façon adaptée dans les portions où la vitesse est quasi constante, dans les zones de changement de concavité de la sinusoïde par exemple, avec néanmoins un résultat un peu inférieur à celui obtenu pour des trajectoires rectilignes simples essentiellement dû au fait que la vitesse de la cible n'est pas constante dans les portions quasi rectilignes.

Par contre, l'imprécision de la localisation est plus grande dans les zones où l'accélération est minimale mais où le vecteur tangent à la trajectoire change fortement d'orientation. Cette augmentation de l'erreur a pour cause l'inertie du système qui tend à faire conserver au groupe sa trajectoire rectiligne. Le mouvement du groupe s'incurve donc avec un certain retard sous l'action des nouveaux plots.

Cependant la variation de l'erreur n'est pas très importante : environ 2% de différence. En outre, ce type de variation est également présent dans toutes les méthodes prédictives telles que le filtrage de Kalman [Koller94] ou les approches Markoviennes [Burgard97].

5.2.1.3. Tests de séparation

Les tests de fusion de plots ont pour but de déterminer la distance de discrimination du système, c'est-à-dire la distance minimale qu'il est possible d'avoir entre deux plots avant que ceux-ci ne soient confondus en un seul groupe. Par ces tests, nous voulons également déterminer à partir de quel moment il est possible de différencier et de détecter deux groupes qui se séparent.

Pour effectuer les tests de fusion de plots, nous avons utilisé deux générateurs différents synchronisés dans le temps fournissant chacun une trajectoire rectiligne uniforme. Ces trajectoires ont été choisies pour que les plots soient suffisamment éloignés au départ (x proche de 0) et confondus en un point proche de la limite du monde (x proche de 200).

Nous avons ensuite déterminé graphiquement la distance de discrimination en considérant que la distinction entre les plots n'est plus possible lorsque les impacts de groupes formés sont éloignés de la trajectoire idéale de plus de la moitié de la distance séparant les plots.

La figure suivante montre une réalisation de ces tests. D'autres part, nous avons matérialisé de manière graphique la ***distance de discrimination*** que nous avons définie comme étant ***la distance minimale jusqu'à laquelle il est possible de différencier les deux plots***.

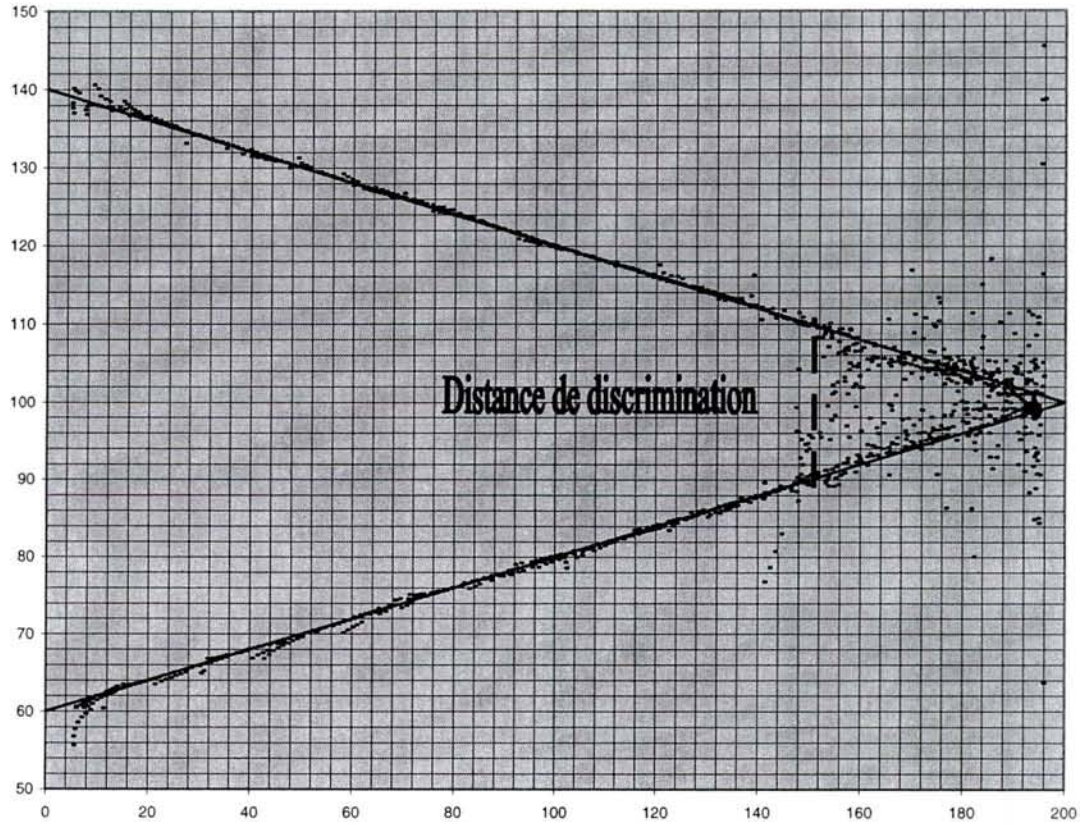

figure 36 : Fusion de deux trajectoires et mesure de la distance de discrimination pour une distance de perception des agents de 25 pixels. (distance de discrimination mesurée : 19,41 pixels)

Ces tests montre que la distance de discrimination est du même ordre que la distance d'attraction des plots (rayon de la zone de diminution de l'attraction proche du centre du plot). La courbe suivante (cf. **figure 37**), construite à partir d'une dizaine de réalisations pour chaque distance d'attraction, montre l'évolution de la distance de discrimination en fonction de celle-ci. L'évolution est en fait quasi-linéaire, et les distances sont à peu près du même ordre même si la distance de discrimination est légèrement plus faible. En outre, l'écart type est très faible, de l'ordre de 1%.

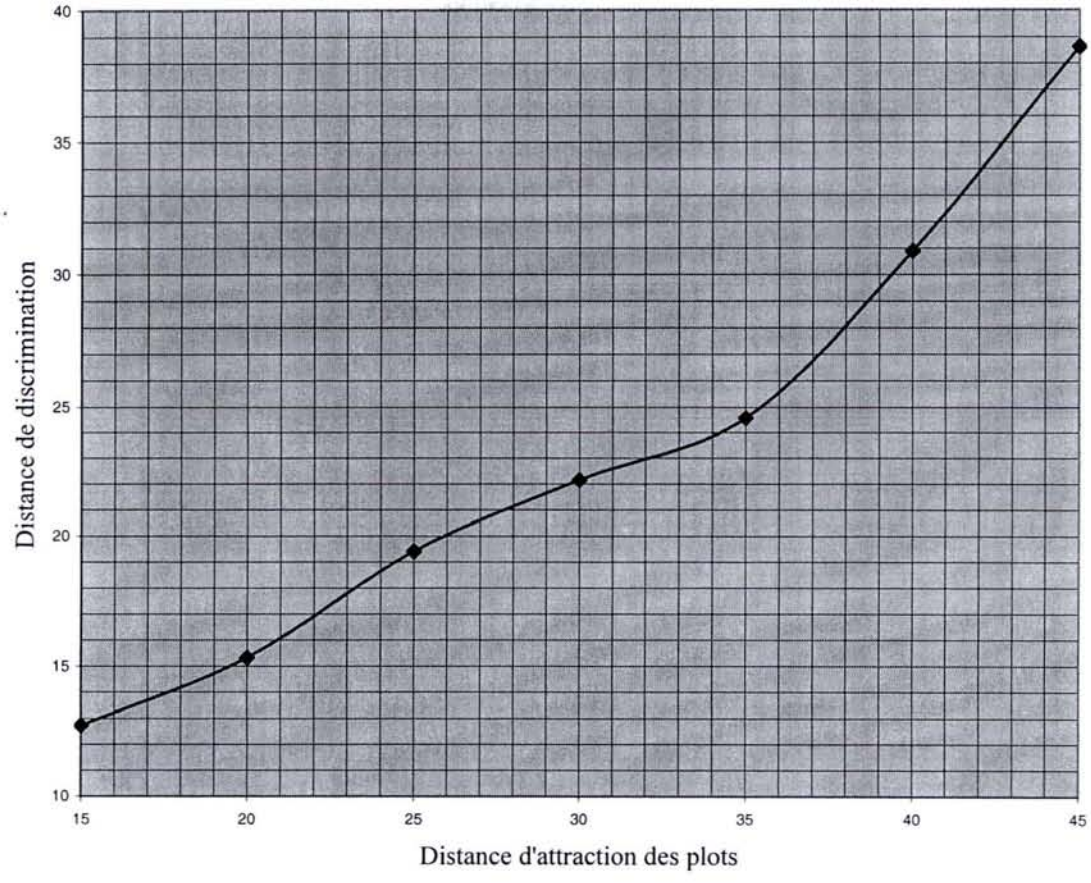

figure 37 : Evolution de la distance de discrimination en fonction de la distance d'attraction des plots

En ce qui concerne les tests de séparations, nous avons utilisés, à l'instar des tests précédents, deux générateurs de plots pour effectuer la mesure de la ***distance de séparation*** du système, que nous avons définie comme étant ***la distance à partir de laquelle il est possible de distinguer deux plots qui se séparent***.

Les trajectoires que nous avons utilisées sont confondues sur une première partie du parcours, puis elles se séparent formant un angle de 45° entre elles. La figure suivante montre une réalisation de ce test et représente graphiquement la distance de séparation.

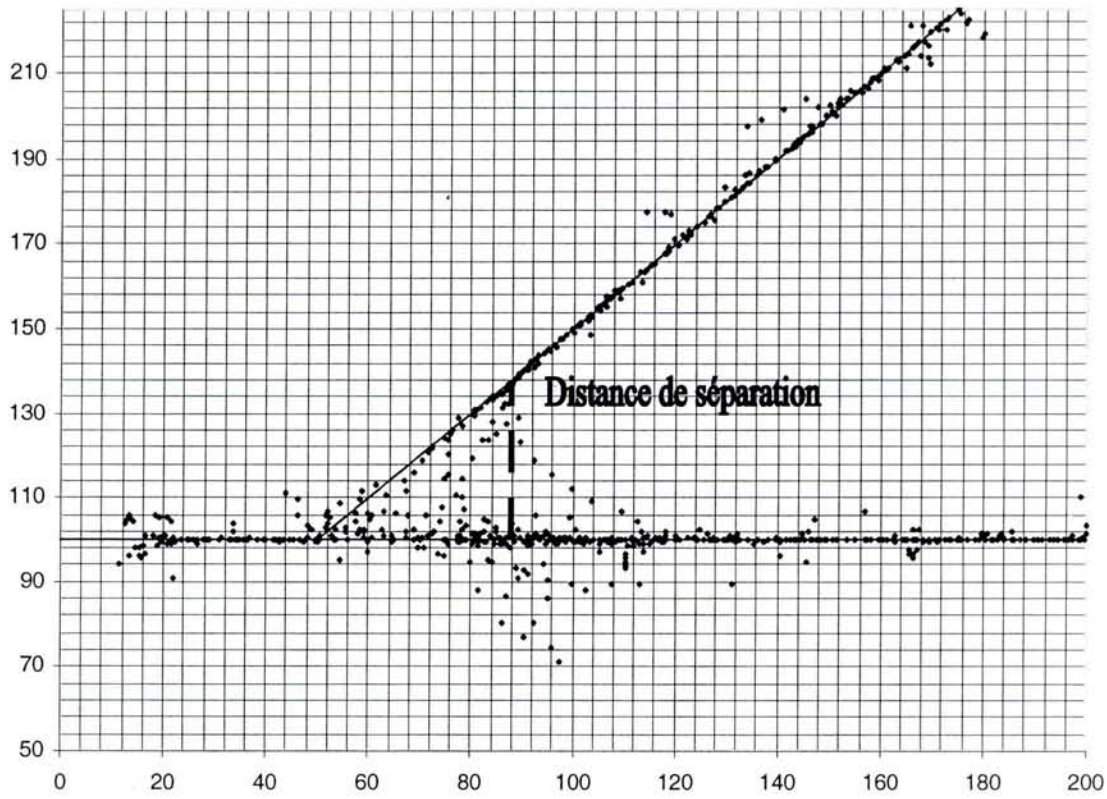

figure 38 : Séparation de deux trajectoires et distance de séparation pour une distance de perception de 45 (distance de séparation mesurée : 38,12)

La fusion des trajectoires se déroule de façon idéale car les groupes d'agents arrivent sur le même plot. Ainsi, le seul phénomène important décelable est une légère confusion due aux agents qui quittent le groupe fusionné en fin de parcours. Cet exode a pour cause le phénomène de régulation de la population des agents sur les plots consécutif à l'équilibre instauré par le comportement de consommation.

Pour ce qui est de la séparation, les résultats sont du même ordre que ceux obtenus pour la fusion : la distance de séparation quasi égale à la distance de discrimination. Cependant, la situation est plus difficile à résoudre pour le système. En effet, de part l'inertie des agents, la trajectoire qui prolonge la partie commune est favorisée par rapport à celle qui se sépare ce qui explique une zone de floue lors de la séparation. Par contre, au bout d'une distance relativement faible, on obtient de nouveau deux trajectoires sous l'action de deux phénomènes :

- la séparation du groupe initial qui est le phénomène minoritaire.
- le recrutement de nouveaux agents présents sur la trajectoire du second plot.

5.2.1.4. Tests de robustesse

Le but de ces tests est d'évaluer la robustesse du processus par rapport aux bruits générés dans l'environnement d'une part (plots parasites) et à ceux générés sur la trajectoire (imprécision des capteurs) d'autre part.

- Bruit blanc dans l'environnement

Lors de ces tests, nous avons décidé de générer, en plus des trajectoires matérialisant les cibles à localiser, des plots parasites dont la position varie de façon aléatoire. Le but de ces manipulations est de mesurer la réactivité du système aux plots parasites et donc aux fausses informations. Cette réactivité dépend essentiellement de la vitesse d'évaporation des plots. Ainsi, si le plot reste trop longtemps dans l'environnement le processus va considérer, à juste titre, que celui-ci correspond à une cible potentielle.

Pour ce test, nous avons utilisé deux générateurs différents : l'un d'entre eux fournissant au système des trajectoires non bruitées, et l'autre induisant des cibles parasites de façon aléatoire et avec une fréquence variable. Nous avons mesuré le nombre de plots détectés pendant une période de temps fixe en faisant varier le coefficient d'évaporation. Nous ramenons ensuite la mesure à une évaluation du nombre de plots détectés par minute de façon à avoir des valeurs comprises entre 0 et 20.

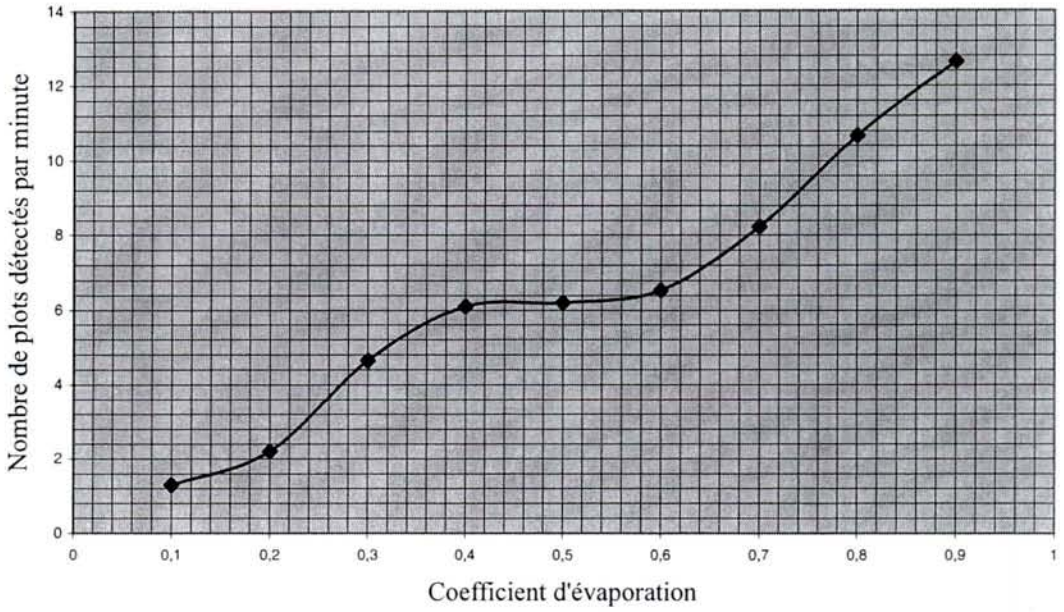

figure 39 : Courbe évolution du nombre de plots détectés en fonction du coefficient d'évaporation. Le générateur utilisé produit un plot toutes les secondes.

Ces tests mettent en évidence l'importance de la dynamique de l'environnement dans le processus de résolution. En effet, à la lumière des résultats obtenus nous pouvons remarquer que l'environnement, et en particulier ses lois d'évolution, sont tout aussi importantes dans le processus de résolution et donc dans le résultat obtenu que les comportements des agents.

- <u>Bruit blanc sur la trajectoire</u>

Afin de mettre davantage à l'épreuve notre système, nous avons fait des mesures d'erreur de localisation avec un bruit blanc sur la trajectoire. L'objectif est ici de voir si l'inertie du système est suffisamment importante pour conserver une trajectoire malgré la présence d'imprécisions sur les informations fournies par les générateurs de plots.

Contrairement aux tests précédents, les résultats obtenus ne sont pas très probants. En effet, le système est trop réactif aux oscillations dues au bruit sur la trajectoire : le groupe d'agents suit la trajectoire et ses oscillations. Il est bien évidemment possible de compenser cette réactivité importante en augmentant l'inertie des agents (augmentation de la masse par exemple) mais le processus ne va alors être capable de suivre uniquement des cibles se déplaçant de façon rectiligne. En effet, on retrouve ici le

problème de suivi de changement de trajectoire que nous avons déjà rencontré lors des tests avec des plots suivant une trajectoire sinusoïdale.

Ce manque de robustesse au bruit sur la trajectoire est-il réellement rédhibitoire pour les objectifs que nous sommes fixés ainsi que pour notre cadre applicatif ? En fait, même si l'inertie des agents permet de faciliter le suivi en compensant, de façon locale, une partie du bruit présent sur la position des plots, la construction d'une trajectoire fait appel à des considérations globales sur l'enchaînement des positions successives des groupes. Il est donc tout à fait normal de ne pas pouvoir compenser de façon locale l'intégralité des déviations. En effet, il est difficile de demander à un SMA qui réagit de façon locale grâce à des agents, qui ont de plus une perception limitée, de gérer des phénomènes qui ne peuvent être essentiellement appréhendés que de façon globale.

5.2.1.5. Tests de suivi

Une fois les tests de localisation et de robustesse réalisés, nous avons effectué une série de mesures de suivi permettant d'évaluer l'efficacité temporelle de la méthode pour une seule cible mobile. Pour cela, nous avons utilisé un générateur (puis deux) fournissant une trajectoire rectiligne. Le processus permet de fournir en cours de fonctionnement et à espace de temps régulier la position des plots présents dans l'environnement et celles des groupes formés. Nous avons donc utilisé cette fonctionnalité pour mesurer le décalage en position entre le plot et le groupe formé à chaque pas de temps.

- <u>Tests avec une seule cible</u>

Les premiers tests ont permis de déterminer l'erreur et l'écart type obtenus lors du suivi de plots sur une trajectoire rectiligne. La figure suivante montre le résultat du suivi d'une telle trajectoire. Ces tests ont été effectués pour des vitesses relatives différentes. Cependant, nous n'avons pas utilisé de vitesse trop lente, rapport de vitesse inférieur à 1, car, dans ce cas, le système se comporte quasiment comme en localisation statique avec des erreurs du même ordre de grandeur. Nous avons tout d'abord effectué des simulations avec des plots ayant la même vitesse maximale que celle des agents de localisation. Nous avons ensuite augmenté la vitesse des plots afin de vérifier la robustesse de l'architecture pour des cibles se déplaçant rapidement. Pour la représentation graphique, nous considérons que la trajectoire des plots est idéale, c'est-à-dire qu'elle passe par le centre de chacun des plots placés dans l'environnement. Ce que nous voulons mesurer ici ce n'est pas l'erreur de localisation (ceci à déjà été fait dans le début de ce paragraphe) mais l'erreur dynamique, c'est-à-dire la différence temporelle de localisation entre la position du plot et celle du groupe d'agent.

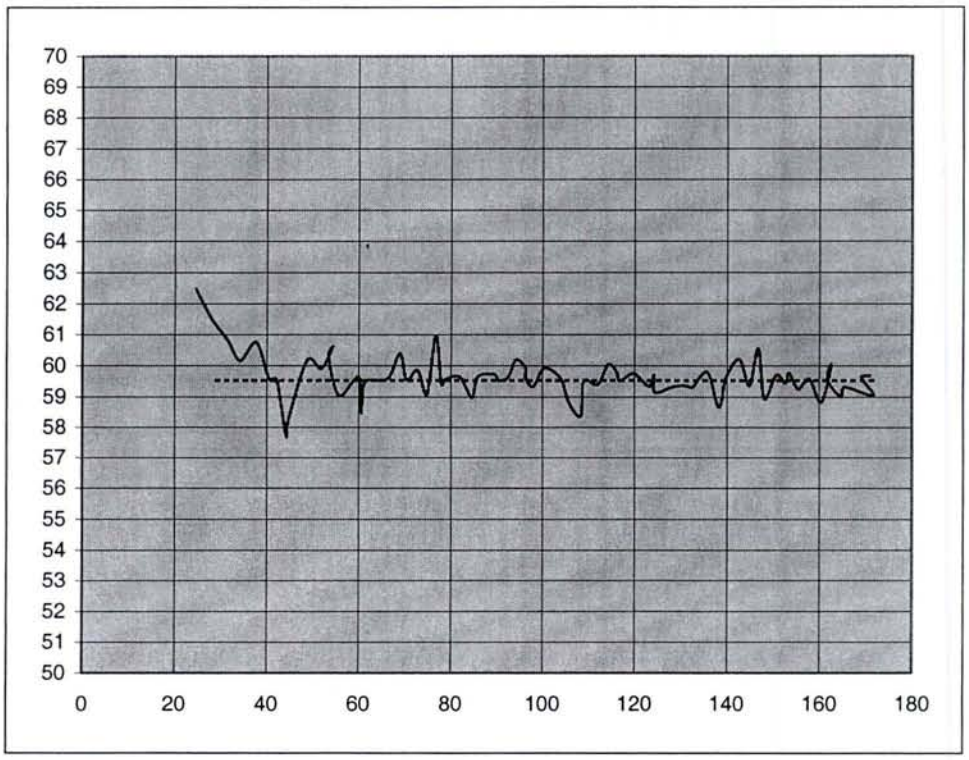

figure 40 : Un exemple de suivi de trajectoire. En pointillés la trajectoire idéale suivie par les plots et en continu la trajectoire détectée par les agents de localisation. L'axe des abscisses et celui des ordonnées représentent les coordonnées des positions successives.

La forte oscillation de la trajectoire en début de suivi correspond à un problème d'initialisation. En effet, les agents étant positionnés de façon aléatoire lors de la mise en œuvre de l'application, il est possible que certains d'entre eux soient suffisamment proches pour former des groupes créant ainsi une fausse trajectoire. Par contre, une fois que les comportements influent sur les déplacements des agents, les groupes se forment correctement sur la trajectoire. Ceci nous montre la nécessité de l'attente de l'état stable fondamental avant toute utilisation du système.

La trajectoire s'effectuant à y constant, les résultats obtenus pour l'erreur de suivi en y sont du même ordre que ceux obtenus pour l'erreur de localisation. Par contre, en ce qui concerne le décalage temporel en x, il varie de façon relativement importante en fonction de la vitesse relative des plots. La figure 41 montre l'erreur temporelle moyenne et l'écart type moyen calculés à partir d'une vingtaine de réalisations pour chacune des vitesses.

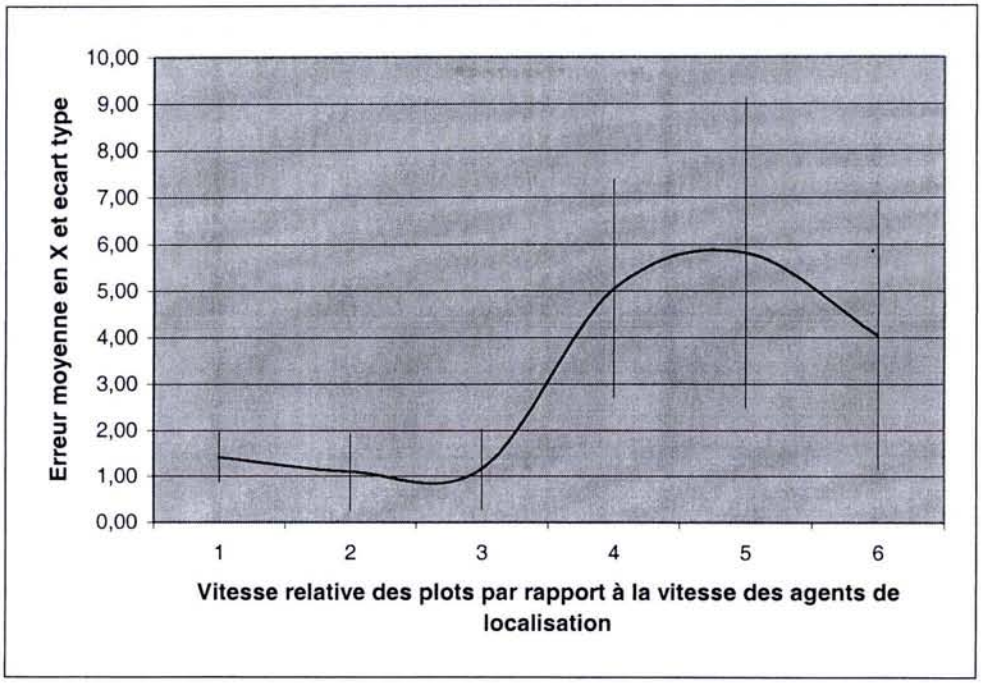

figure 41 : Erreur temporelle moyenne et écart type pour différentes vitesses relatives

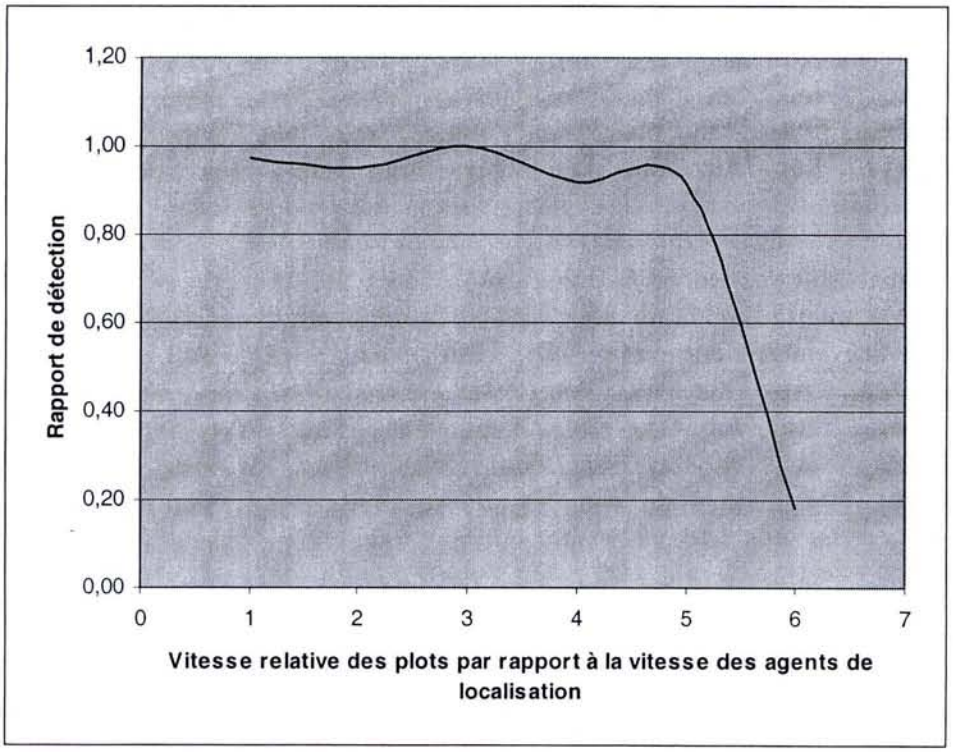

figure 42 : Rapport de détection (nombre de groupes formés/ nombre plots générés) en fonction de la vitesse relative

On remarque que lorsque la vitesse relative croît (plot plus rapide que la vitesse maximum des agents) l'erreur moyenne et surtout l'écart-type augmente. Néanmoins, cette augmentation n'est flagrante qu'à partir d'un rapport de vitesse de l'ordre de 3. En outre, on peut remarquer une amélioration des résultats lorsque la vitesse relative est très grande. Ceci est à mettre en relation avec la chute du rapport de détection, rapport entre le nombre de groupes formés et le nombre de plots générés. En effet, étant donné que ce dernier est très faible (la cible n'est pas souvent détectée), à chaque création de groupe, on est quasiment assuré que la cible est correctement localisée.

- Plusieurs cibles se rencontrant

Enfin la dernière série de tests a été réalisée afin de mettre en évidence la possibilité qu'a le système de suivre plusieurs cibles simultanément et ceci même dans le cas ou les trajectoires des cibles se croisent. Pour effectuer ces tests, nous avons utilisé deux générateurs de plots synchronisés dispensant des trajectoires rectilignes sécantes en un point situé au centre de l'environnement. Afin d'estimer la réponse du système, nous avons relevé les positions successives des groupes d'agents au cours d'une dizaine de simulations.

La figure 43 montre les trajectoires obtenues lors de plusieurs simulations de suivi de plots se déplaçant de la même façon ayant une vitesse relative 3 fois supérieure à la vitesse maximum des agents.

Ces résultats montrent que, même dans un cas limite, le système est capable de suivre plusieurs plots de façon simultanée y compris lorsque les trajectoires de ceux-ci se croisent. En fait, le nombre de plots présents dans l'environnement a très peu d'importance étant donné que le système fourni une réponse locale aux perturbations perceptives.

Les résultats de ce suivi de deux cibles dont les trajectoires se coupent sont intéressants à interpréter. D'un premier abord, on pourrait conclure que les groupes d'agents ne suivent pas correctement les cibles puisque les trajectoires de groupes ne correspondent pas aux trajectoires de celles-ci. Cependant, si l'on raisonne d'un point de vue physique, les trajectoires des cibles ne peuvent pas avoir d'existence réelle car elles correspondraient à un choc entre les deux cibles suivies. La seule solution physique plausible est un évitement du choc et une poursuite des trajectoires dans un sens perpendiculaire au sens de déplacement initial. C'est cette réponse qui est fournie par le système.

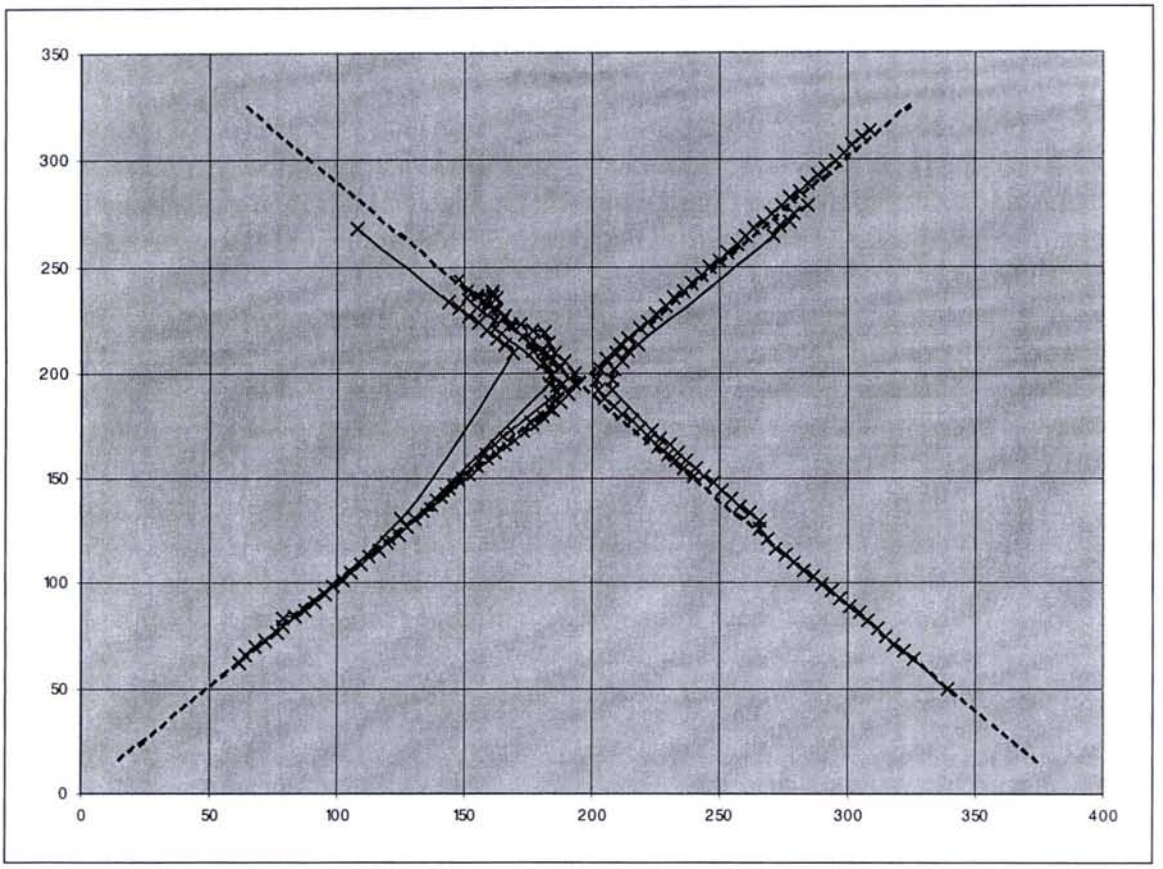

figure 43 : Suivis de deux plots

Si la vitesse relative des plots est faible, on obtient effectivement une fusion des groupes puis une séparation des trajectoires de façon analogue à ce que l'on a pu observer dans le cas des tests de localisation pour la séparation et la fusion de trajectoire.

Il existe donc plusieurs cas en fonction de la vitesse relative des plots :

- Pour une vitesse relative faible, il y a fusion puis séparation de groupes.
- Pour une vitesse relative forte, il y a apparition de trajectoires d'évitement (cf. figure 43)

5.3. Discussion et Commentaires

Il est légitime de s'interroger sur la pertinence de l'utilisation d'agents pour la localisation des plots fournis par l'ensemble sources d'informations, alors qu'il semble plus naturel d'utiliser des algorithmes classiques de traitement d'images tels que des seuillages ou des détections de régions.

A la lumière des premiers résultats obtenus, nous pouvons remarquer que l'utilisation des agents permet d'obtenir des résultats de bonne qualité mais avec une mise en œuvre beaucoup plus simple qu'avec des méthodes classiques. En outre, cette approche réactive possède des intérêts importants que nous avons pu observer au cours de ces différents tests :

- L'inertie des agents permet une anticipation des déplacements analogue à celle observée lors de l'utilisation d'un filtre de Kalman.
- Les comportements permettent une persistance de l'information en l'absence de rafraîchissements fréquents, ce qui rend possible l'utilisation de plusieurs capteurs aux capacités limitées.
- La gestion de l'apparition et de la disparition de nouvelles cibles se fait de façon automatique.

Cependant, le principal inconvénient de ce type de système est la mise au point des paramètres des comportements pour obtenir le résultat escompté. Ainsi, même si le système est relativement simple à concevoir, il est très difficile à régler. En effet, il est assez délicat de mesurer l'influence des paramètres comportementaux sur le résultat au niveau macroscopique. En outre, le système dispose d'un grand nombre de paramètres, parfois interdépendants, qui agissent directement ou indirectement sur la qualité du résultat final.

Néanmoins, ces tests nous ont permis d'isoler un certains nombres de paramètres critiques qui conditionnent directement la qualité du résultat obtenu.

Pour ce qui est des comportements antagonistes d'attraction et de répulsion, nous nous sommes rendus compte, grâce aux différents tests de performances réalisés en faisant varier les paramètres du modèle, que le comportement d'attraction est l'élément qui influe le plus sur la qualité du résultat.

Il est donc primordial de bien évaluer ses paramètres. Le comportement de répulsion ne conditionne que l'état du système dans les zones dépourvues de plots. Dans ce cas, ce qui est important, c'est la proportion de répulsion dans le comportement de chacun des agents. Si le comportement d'attraction est prépondérant, les agents vont massivement se concentrer sur les régions d'intérêt en délaissant les régions vides de plots. Le système ne sera alors plus capable de prendre en compte l'arrivée de nouvelles cibles.

L'inertie des groupes d'agents est surtout importante pour le suivi des cibles. Elle permet, de façon indirecte, de prendre en compte les paramètres dynamiques du groupe

d'agents pour prévoir son évolution future. Le réglage de l'inertie se fait selon deux paramètres dépendant à la fois des paramètres intrinsèques des agents (masse) et de la physique de l'environnement (frottement). Par expérience, nous nous sommes rendu compte qu'il était plus aisé de contrôler l'inertie en jouant sur le coefficient de frottement de l'ensemble du système plutôt que de modifier la masse des agents. En effet, la plage de modification du coefficient de frottement étant bornée (entre 0 et 1) il est plus facile de régler la proportion de frottement dans l'expression générale des forces. En ce qui concerne la masse, nous n'avons pas de valeur de référence. Ainsi, il est délicat de modifier ce paramètre pour influer sur l'inertie. Nous avons donc choisi de le conserver égal à 1 dans l'ensemble des expérimentations.

Les paramètres de régulation conditionnent également la qualité du résultat obtenu puisqu'ils influent directement sur la taille de la population d'agents présente sur un plot et donc sur la cohérence du groupe formé. Le coefficient d'évaporation permet de jouer sur la qualité de la prise en compte des plots parasites : plus il est important (proche de 1) moins les plots parasites sont pris en compte. Celui-ci est un paramètre de régulation isotrope puisque son influence ne varie pas quelque soit la zone de l'environnement concernée. L'appétit des agents permet également de modifier le nombre d'agents présents sur chacun des plots. Son but est d'obtenir un nombre d'agent en adéquation avec l'importance (altitude) du plot. Plus ce coefficient est élevé (proche de 1), plus la population sera faible sur le plot. A l'opposé du coefficient d'évaporation, celui-ci est un paramètre de régulation anisotrope puisque son influence dépend du nombre d'agents présents sur le plot.

Enfin, le dernier paramètre critique est le nombre d'agents à partir duquel est formé un groupe. Si ce nombre est faible (proche de 2), de nombreux groupes parasites vont apparaître y compris dans les zones dépourvues de plots. Par contre, si celui-ci est trop important, certaines cibles réelles ne seront pas détectées.

En ce qui concerne le réglage initial des paramètres, nous nous sommes inspirés des formulations usuelles des forces physiques dans le cadre de l'attraction/répulsion entre deux particules. Plus que la valeur absolue de chacun des paramètres, ce qui est important, ce sont les proportions relatives de chacun d'entre eux. Une autre solution que nous aurions pu envisager pour résoudre ce problème de réglage initial serait d'effectuer un apprentissage des paramètres sur un cas de figure où l'on connaît la réponse attendue. Cependant, dans ce dernier cas, il est tout de même nécessaire de posséder une valeur initiale des paramètres pas trop éloignée des réglages optimaux de façon à éviter les extrema locaux. Il est donc, dans tous les cas, nécessaire de s'appuyer sur la connaissance physique des phénomènes utilisés.

Bien que le système permette d'effectuer un suivi et une localisation précise de deux ou plusieurs cibles dont les trajectoires se croisent, il est impossible de dire si celles-ci se sont effectivement croisées ou si elles ont rebroussé chemin après leur rencontre.

5.4. Bilan

Ces tests ont permis de mettre en évidence l'intérêt des comportements proposés pour la tâche de localisation par focalisation. Nous avons, en particulier, testé le bon fonctionnement des deux comportements antagonistes d'attraction et de répulsion.

D'autre part, nous avons effectué des expériences destinées à montrer que l'organisation obtenue à partir de ces comportements permet de caractériser la position des cibles et de réaliser leur suivi. En outre, nous avons essayé de voir les limitations du système ainsi défini en utilisant des situations critiques et en faisant varier un certain nombre de paramètres.

Par rapport aux objectifs fixés initialement, nous possédons maintenant, grâce à ces tests, des éléments de réponse permettant de mettre en évidence certaines caractéristiques. En particulier :

- Le dispositif est capable de gérer un nombre de cibles quelconque et variable en cours de fonctionnement. *(tests avec plusieurs générateurs)*
- Le nombre de source d'informations peut également être variable en cours de fonctionnement. *(tests avec plusieurs générateurs)*
- Le système permet de gérer la panne de sources d'information de part le principe d'accumulation de l'environnement. *(tests avec plusieurs générateurs fournissant la même information)*
- Le système étant dynamique, il est possible d'ajouter ou de retirer des agents en cours de fonctionnement rendant ainsi possible une adaptation des moyens de résolution en fonction de la complexité du problème à traiter. *(tests de localisation statique)*
- La population étant homogène, le processus de résolution est identique quelle que soit la configuration initiale du problème et surtout quelles que soient ses variations en cours de fonctionnement. *(tests multi-cibles)*

Chapitre 6 : Passage au monde réel : Comment introduire la perception ?

Dans le chapitre précédent, nous avons montré que notre proposition d'architecture donnait des résultats encourageants (du point de vue de leur pertinence vis-à-vis des objectifs et des contraintes fixés) sur des cas de simulation proches de notre cadre applicatif. Afin de vérifier que le comportement du système, tel que nous l'avons défini, permet bien de localiser et de suivre plusieurs cibles de façon simultanée, il faut vérifier que ce fonctionnement donne des résultats tout aussi acceptables en situation réelle. En outre, il faut que le processus conserve les propriétés mises en évidence lors des tests simulés telles que la robustesse vis-à-vis de la qualité des sources d'informations, l'utilisation d'un nombre variable capteurs ou la possibilité de suivre plusieurs cibles différentes de façon simultanée.

En s'inspirant de l'architecture que nous avions pour le test (cf. figure 26), le lien avec le monde réel doit s'effectuer au niveau de la construction du problème dans l'environnement des agents. Il s'agit donc de remplacer les générateurs de plots par des sources d'informations issues du monde réel.

Le problème de la perception dans un tel système n'est pas anodin. En effet, il ne s'agit pas uniquement de fournir des informations au système, mais également d'effectuer une réflexion à la fois sur ce que peut apporter la perception tant au niveau des caractéristiques générales du système qu'au niveau de la qualité de l'information fournie et surtout jusqu'où doit-on aller dans la sophistication des traitements pour que le système soit efficace. Ainsi, dans ce chapitre nous voulons répondre à un certain nombre de questions concernant la perception du monde réel :

- Jusqu'où peut-on et doit-on complexifier les traitements destinés à fournir des informations à notre système ?
- Est-ce que celles-ci peuvent toutes être obtenues ?
- Quelles sont les caractéristiques et les propriétés que nous nous sommes fixées qui peuvent être obtenues par une perception adaptée ?

Enfin, il ne faut pas confondre la perception du monde réel destinée à construire une représentation abstraite sous la forme d'un environnement pour nos agents de localisation, et la perception de ceux-ci même si ces deux notions sont liées. En effet, la perception des agents est directement conditionnée par la configuration de l'environnement et donc par la perception qu'a le système du monde réel.

Dans ce chapitre, nous allons donc nous attacher à effectuer ce lien avec le problème physique concret en mettant en place quelques principes de construction d'une structure générique, l'***unité perceptive***, regroupant un ou des capteurs avec des unités de traitements de l'information, et fournissant des indications numériques structurées sur les variations du problème à résoudre.

Dans un premier temps, nous allons définir le concept d'unité perceptive. Puis, nous détaillerons la construction de deux unités perceptives que nous avons utilisées pour effectuer les tests sur des cibles réels.

6.1. Concept d'unité perceptive

Pour envisager une activité de localisation dans un environnement donné, il faut pouvoir construire une représentation de la position d'un objet par rapport au monde dans lequel il se trouve. Cette opération, et par extension, la représentation du monde requièrent une perception des caractéristiques de l'environnement que l'on veut modéliser. Cette perception s'effectue par l'intermédiaire de dispositifs appelés capteurs. Ces dispositifs d'acquisition ont pour principe de transformer la configuration observable du monde en une succession de données numériques ou analogiques. Ces données doivent ensuite être interprétées pour fournir une représentation abstraite et partielle de l'environnement. En effet, le principe de la localisation est de trouver, à partir de la représentation du monde, la position qui correspond le plus à la perception courante. Il s'agit donc d'effectuer une comparaison entre une représentation du monde immédiate fournie par les différents dispositifs d'acquisition avec une représentation de référence préétablie.

6.1.1. Les capteurs physiques.

Il existe de nombreux capteurs disponibles sur le marché possédant chacun des caractéristiques spécifiques et des limitations dues essentiellement à leur conception et/ou à leur principe de fonctionnement. Ces capteurs peuvent être généraux, c'est-à-dire capables de fournir des informations riches et diversifiées sur l'environnement, c'est le cas des caméras, ou spécifiques attachés à la détection d'un évènement particulier, comme un détecteur de CO_2 par exemple. La majeure partie d'entre eux est utilisable dans le cadre d'une application de localisation. Ainsi, l'utilisation des capteurs peut être envisagée de plusieurs façons différentes. En effet, ceux-ci peuvent être fixes dans l'environnement comme une caméra de surveillance par exemple, ou être embarqués sur l'objet à localiser, on parlera alors d'auto-localisation. Dans ce cas, nous pouvons noter deux types de capteurs bien différenciés : les capteurs proprioceptifs et les capteurs extéroceptifs. Les capteurs proprioceptifs fournissent des informations sur l'état interne du robot (nombre de tours de roue, niveau de charge de la batterie,...) alors que les capteurs extéroceptifs fournissent des informations sur l'environnement entourant le robot.

Parmi les capteurs proprioceptifs, l'odomètre est celui qui est le plus facilement exploitable dans notre cadre applicatif. Son principe est d'estimer les déplacements du robot à partir des rotations élémentaires estimées au niveau des roues ou des moteurs. (cf. annexes)

Les capteurs extéroceptifs peuvent être classés principalement en deux catégories : les capteurs de proximité ou de contact et les capteurs de distance. Les premiers servent

essentiellement de capteurs de bas niveau pour implanter les réflexes élémentaires des robots. Les seconds sont ceux qui sont les plus appropriés à une application de localisation. Parmi ces capteurs nous pouvons citer les sonars, les télémètres ou les capteurs infrarouges. Les caméras ne peuvent pas être classées parmi les capteurs de distance même si elles peuvent très bien réaliser cette tâche. En fait, les dispositifs imageurs ont ceci de particulier que la tâche à laquelle ils sont destinés ne dépend pas de la conception du capteur mais essentiellement du traitement qu'on lui adjoint. Cette particularité en fait donc des capteurs privilégiés pour toutes les applications robotiques.

Quelque soit sa nature, un capteur physique peut être considéré comme un dispositif permettant d'acquérir et de traiter les informations du monde réel pour donner un résultat sous la forme de signaux analogiques ou numériques. (cf. figure 44).

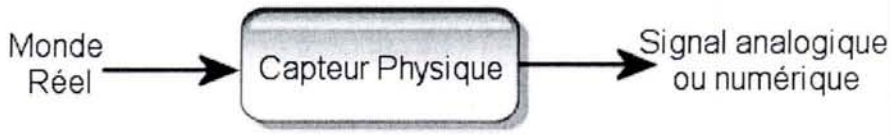

figure 44 : Représentation schématique d'un capteur physique.

Ces signaux ne sont pas directement exploitables dans le cadre d'une application concrète, et ils ne permettent pas de construire facilement une représentation abstraite du monde réel. C'est pourquoi ils sont dans tous les cas associés à des unités de traitement permettant de transformer les signaux numériques ou analogiques obtenus en une information structurée.

6.1.2. L'unité perceptive

6.1.2.1. Principe général

Une fois le problème de la perception résolu et le choix des capteurs effectué, il faut donc choisir une méthode pour réaliser la construction de la représentation abstraite du monde. Parmi les méthodes les plus utilisées, nous pouvons citer les cartes des routes, les cartes géométriques, les grilles d'occupation ou les histogrammes d'occupation [Olson99], [Duckett01]. Ces dernières sont assez faciles à mettre en œuvre mais possèdent des limitations assez contraignantes concernant le stockage des informations qui devient très vite problématique avec les machines courantes. En outre, la plupart de

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

ces méthodes requièrent l'utilisation d'algorithmes de fusion afin de pouvoir bénéficier de l'apport de plusieurs types de capteurs distribués dans le temps et dans l'espace.

En général, les traitements utilisés sont directement liés à la nature du capteur, aussi le choix du traitement est toujours délicat puisqu'il doit prendre en compte à la fois les limitations du capteur et la nature des informations qu'il doit fournir.

L'***unité perceptive***, ou capteur logique, peut être définie comme la conjonction entre un ***capteur physique*** et une ***unité de traitement*** de l'information dont le but est de fournir des données numériques en adéquation avec la représentation du monde choisie (cf. figure 45).

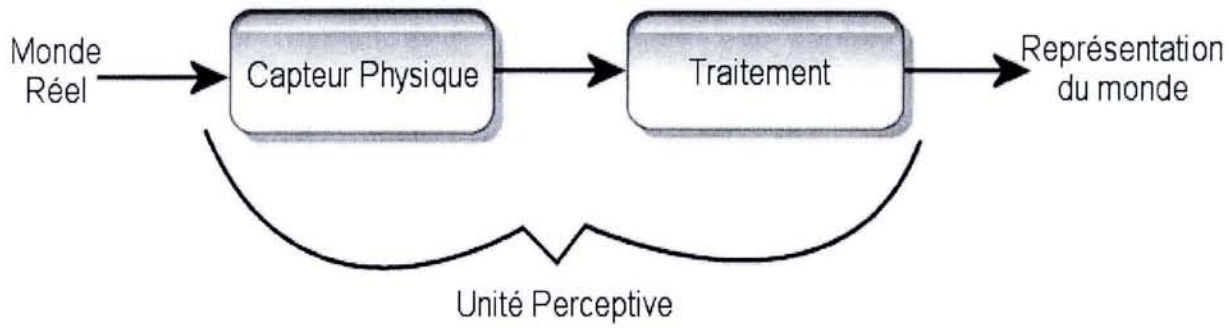

figure 45 : Représentation schématique d'une unité perceptive (ou capteur logique)

Comme nous pouvons le voir dans la définition que nous venons de proposer, le choix des traitements de l'unité perceptive est dicté par la représentation du monde choisie.

6.1.2.2. Intérêts pour notre architecture

Nous avons vu que la localisation avait pour principe de comparer la représentation immédiate du monde fournie par les capteurs avec une représentation de référence. En considérant le modèle de représentation choisi, localiser une cible dans le monde réel reviendra à trouver le couple position/orientation qui correspond le mieux à l'objet parmi ceux proposés par la représentation abstraite.

Dans ce contexte, nous considérons donc une unité perceptive comme étant un élément fournissant un ensemble de couples position/orientation probables pour une ou plusieurs cibles données à partir des observations fournies par un capteur physique (cf. figure 46) .

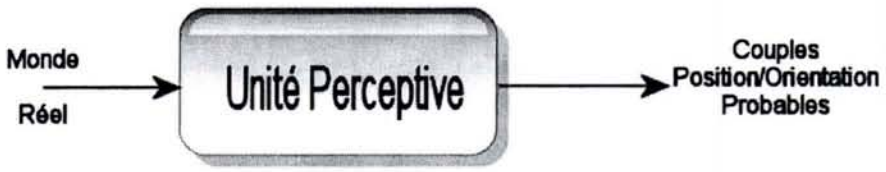

figure 46 : Le modèle d'unité perceptive que nous utilisons.

L'intérêt, pour notre approche, est de pouvoir utiliser ces unités perceptives en remplacement des capteurs virtuels utilisés dans le chapitre précédent. Dans le cadre de notre architecture, nous ne manipulons donc pas directement les capteurs et les unités de traitement mais leurs abstractions sous la forme d'UP.

6.2. Construction de deux unités perceptives simples

Nous allons maintenant appliquer le concept d'unité perceptive à deux cas concrets qui se différencient uniquement par la nature du traitement utilisé.

Pour le capteur physique, nous avons choisi d'utiliser une caméra CCD classique, l'unité de traitement étant assurée soit par une Analyse en Composantes Principales soit par un élément inspiré du flot optique et du filtre de Kalman.[Roumeliotis99]

6.2.1. L'analyse en composantes principales

La méthode présentée dans la suite de ce chapitre a été développée initialement pour effectuer de la reconnaissance de visages. Elle a été ensuite adaptée à la localisation d'un robot mobile dans un environnement réel constitué de couloirs [Gechter99]. Cette application est le premier dispositif de localisation réalisé au cours de cette thèse. Elle est essentiellement destinée à mettre en évidence la possibilité d'interpréter de manière globale les images fournies par une caméra noir et blanc classique sans utiliser des algorithmes d'extraction de primitives qui sont à la fois coûteux et difficiles à calibrer sous des conditions de prise de vue variables (cf. annexes). En outre, cette application permet de disposer d'une première source d'information, éminemment simple, pour notre architecture de localisation capable de fournir des résultats sous la forme désirée à partir d'observation sur le monde réel.

Le principe qui prévaut ici est la comparaison la plus simple possible entre une image acquise par la caméra et celles contenues dans une base représentative de l'environnement. Le modèle de représentation du monde est celui que nous avons

détaillé dans le chapitre précédent. Afin de construire une base qui représente l'environnement réel, et qui s'adapte à cette représentation abstraite, l'idée la plus simple est de prendre une image, sous des conditions standards d'éclairage, pour chacun des états possibles dans la représentation du monde. Ainsi, nous avons choisi d'effectuer une prise de vue pour chacun des couples Position/Orientation possibles. La comparaison s'effectue ensuite à partir de caractéristiques globales calculées de façon systématique.

Ainsi, la méthode choisie est inspirée d'une méthode de classification extrêmement utilisée pour trier des ensembles de données multidimensionnelles homogènes [Abdi98] : l'Analyse en Composantes Principales (ACP). Elle est également connue sous le nom de transformation de Karhunen et Loeve particulièrement en Automatique. L'ACP permet de rechercher les axes de discrimination maximum d'un nuage de points dans un espace de dimension quelconque. La classification des données s'effectue à partir des projections des points du nuage sur ces axes. La détermination de ceux-ci est réalisée en calculant les vecteurs propres de la matrice de covariance du système de points, le pouvoir discriminant de chaque axe étant déterminé par sa valeur propre. Plus celle-ci est élevée plus l'axe engendré par le vecteur propre est discriminant. L'ACP peut s'adapter à différents types de capteurs tels que les caméras ([Pourraz99], [Sim99a]), les télémètres laser [Crowley98] ou tout autre capteur pouvant fournir des données ***homogènes***, c'est-à-dire, dans notre cas, des données pouvant être exprimées dans une ***base vectorielle fixe*** et dont les composantes dans cette base sont ***bornées***. Les images en niveaux de gris, par exemple, peuvent être mises sous la forme d'un vecteur de dimension le nombre de pixels dans une image et dont chacune des composantes est comprise entre 0 et 255. Ainsi, il faut que tous les vecteurs fournis par le ou les capteurs soient de même taille et de dynamique finie. Dans la suite, nous allons développer le principe de l'Analyse en Composantes Principales en s'appuyant sur une utilisation avec un capteur de type caméra CCD noir et blanc.

Afin de pouvoir utiliser l'ACP et effectuer de la reconnaissance de position à partir d'une caméra embarquée, il faut construire au préalable une base d'images représentatives de l'environnement dans lequel l'entité mobile va évoluer. Ceci peut paraître très limitatif mais il est malheureusement nécessaire de procéder de la sorte étant donné que ce type d'algorithmes fonctionne par comparaison vision immédiate / vision mémorisée. Par contre, il peut être judicieux d'effectuer les premières reconnaissances avec une base minimale qui sera enrichie en fonction des déplacements de l'entité mobile. Comme dans la plupart des algorithmes utilisés en localisation, il s'agit ensuite de trouver l'élément de la base qui correspond le mieux à une acquisition donnée. Dans le cadre de l'utilisation de l'ACP avec des images, cette correspondance est évaluée en fonction des propriétés radiométriques des différentes acquisitions. Dans ce contexte, l'utilisation de l'algorithme de traitement permet d'orienter cette recherche uniquement sur des propriétés globales des images contrairement aux algorithmes classiques ([Zingaretti98]) qui s'attachent davantage à la recherche d'indices significatifs dans les scènes observées. Néanmoins, il n'est pas nécessaire ou utile d'utiliser l'ACP sur la globalité des informations fournies. Il est parfois plus judicieux d'employer cette méthode uniquement sur des portions significatives des images

([Pourraz99], [Sim99a], [Sim99b], [Sim99c], [Sim99d], [Sim98a], [Sim98b], [Abdi98], [Moghaddam96]). Cette dernière approche constitue en fait la majeure partie des publications. Cependant, nous avons choisi d'appliquer l'analyse en composantes principales à la totalité des images car ceci permet d'éviter une étape supplémentaire d'interprétation et d'améliorer le temps de réponse de l'algorithme en se passant d'une étape interprétative supplémentaire.

Afin de pouvoir utiliser l'ACP en reconnaissance visuelle, il faut tout d'abord considérer qu'une image est en fait un point dans un espace de dimension égale au nombre de pixels qu'elle contient, les images de la base étant supposées être toutes de la même taille afin de vérifier le critère d'homogénéité. L'ensemble des images de la base constitue alors un nuage de points dans un espace multidimensionnel. Il suffit d'utiliser l'ACP sur ce nuage de points pour calculer les axes les plus discriminants pour l'ensemble des images.

Dans le cadre de cette thèse, nous avons développé plusieurs unités perceptives plus ou moins complexes, à partir de l'ACP. Outre l'utilisation de l'ACP simple, nous lui avons également adjoint des méthodes de prise en compte du déplacement de la cible soit sous la forme de voisinages, statiques ou dynamiques, soit sous la forme de processus de Markov partiellement observables (POMDP) [Gechter00], [Gechter01a], et [Gechter01b]. En outre, nous avons également étudié la possibilité de fusionner plusieurs capteurs soit à partir des POMDP (fusion probabiliste), soit directement à l'aide de l'ACP (fusion de percepts) [Gechter01c] et [Gechter02].

Les différents algorithmes que nous avons mis en œuvre possèdent des caractéristiques intéressantes vis-à-vis des objectifs et des contraintes que nous nous sommes fixés :

- Il n'y a pas de représentation complexe du monde ou de la tâche à accomplir.
- Il y a une continuité dans les perceptions permettant d'envisager la construction d'une trajectoire.
- Il est possible d'utiliser plusieurs sources d'informations (fusion de percept et fusion probabiliste)

Cependant, ils ne permettent pas d'obtenir toutes les propriétés désirées :

- La localisation multi-cibles est délicate à mettre en œuvre (obligation d'ajouter une étape de fusion)
- Il est impossible de faire varier le nombre de sources d'information (fusion de percept)
- Le processus de résolution est directement lié à la nature des capteurs et des traitements utilisés.

6.2.2. Utilisation d'un filtre de Kalman

L'autre unité perceptive que nous avons utilisée a été développée au sein de l'équipe MAIA par Loïc Pelissier dans le cadre du suivi de personnes âgées à domicile. Ce dispositif est inspiré des méthodes de traitements d'images de type filtre de Kalman ou Flot optique [DeCarlo00], [Nagel98] (cf. annexes)

Le fonctionnement de cette unité perceptive a pour principe l'évaluation de la différence pixel à pixel entre une image de référence représentant la zone d'évolution des cibles et l'image acquise. Il est possible d'assimiler ce type d'approche à un filtre de Kalman d'ordre 0. L'image différence représente les éléments de la scène qui sont en mouvement ou qui ne se trouvent plus à la place qu'ils occupaient dans l'image de référence. Ce dispositif permet de détecter toute sorte de cible sans pré-requis sur la nature de la scène observée.

Après l'obtention de l'image des cibles en mouvement il s'agit d'en déterminer la position. Pour cela, nous avons employé une méthode de détection de régions par balayage avec fusion pour éviter les problèmes de détection de régions non convexes.

Cette étape est ensuite suivie d'une triangulation permettant de retrouver la position réelle à partir de connaissances sur les conditions de prise de vue. (cf. annexes)

L'unité perceptive fourni comme résultat la position dans le monde réel de toutes les régions détectées. Il suffit ensuite d'adapter ces positions à la configuration de l'environnement des agents.

Cette unité perceptive permet de résoudre certaines des caractéristiques désirées :

- Elle permet la localisation multi-cibles en nombre quelconque et variable et de façon simultanée.
- Elle ne nécessite pas de connaissance sur les cibles.

Par contre, elle possède quelques inconvénients :

- Elle est directement liée à un capteur physique et ne peut donc pas pallier une éventuelle panne de celui-ci.
- La fusion de capteurs n'est pas envisageable hormis en ajoutant une étape supplémentaire permettant de gérer plusieurs unités perceptives du même type.
- Sa portée est limitée par la configuration du dispositif physique.

6.3. Conclusion

6.3.1. Bilan

Dans le cadre d'une application réelle, la perception possède un rôle très important car elle conditionne la formalisation du problème à résoudre et donc la qualité des résultats. Le concept d'***unité perceptive*** permet de regrouper derrière un formalisme commun toutes les sources d'informations utilisables par notre système. Les contraintes alors définies sont d'ordre général et concernent essentiellement la nature des données fournies. En effet, nous n'avons imposé aucunes contraintes concernant la nature des capteurs ou celle des traitements.

A partir d'algorithmes élémentaires nous avons montré qu'il était possible de construire plusieurs unités perceptives différentes et de complexité variable tout en conservant une adéquation avec le formalisme.

6.3.2. Discussion

Le problème de l'élaboration d'unités perceptives complexes est l'obligation d'injecter dans le traitement une certaine connaissance sur le problème à résoudre ce qui alourdi le processus de construction de l'environnement. En ce qui nous concerne, nous ne nous sommes jamais fixés de condition sur une éventuelle simplicité des perceptions, nous voulons juste que le processus de décision soit le plus simple possible. En outre, si l'on se réfère aux SMA biologiques, tels que celui des fourmis par exemple, celles-ci possèdent bien des primitives décisionnelles simples mais ce n'est pas pour autant que le mécanisme de perception des phéromones n'est pas compliqué.

Si l'on se réfère au travail que nous avons réalisé sur la perception et sur l'exploitation des capteurs physiques, nous pouvons remarquer qu'il est possible d'introduire peu à peu des éléments intelligents dans les différents traitements afin d'obtenir les caractéristiques désirées. Cependant, dans certains cas, les modifications sont plus d'ordre architectural que d'ordre technique. Ainsi, il est éventuellement possible d'envisager la localisation multi-cibles à partir d'algorithmes analogues à ceux que nous venons de décrire, mais au prix d'une refonte complète de la chaîne de traitement de l'information, en ajoutant des couples capteurs/traitements élémentaires et en introduisant une surcouche globale de gestion et de fusion des perceptions.

Mis à part ces modifications importantes, il est néanmoins possible de résoudre un certain nombre de problèmes tout en restant raisonnable du point de vue de la

complexité de l'interprétation. Par exemple, nous avons montré qu'il était possible de résoudre en partie le problème de l'utilisation d'un nombre variable de capteurs, tout en s'assurant d'une continuité dans les localisations successives. Pour le moment, notre algorithme, s'appuyant sur l'ACP, ne permet que l'utilisation de capteurs embarqués. Nous avons fait volontairement ce choix car, d'une part, il est plus difficile de construire un dispositif de localisation avec une focalisation interne et d'autre part, les algorithmes de localisation externe sont assez courants et en général performants.

Enfin, certaines caractéristiques obtenues en complexifiant les traitements sont déjà présentes dans notre architecture à base d'agents réactifs. La continuité des trajectoires, par exemple, est déjà obtenue grâce au phénomène d'inertie des groupes. L'utilisation de plusieurs générateurs de plots nous a permis de mettre en évidence la possibilité de gérer plusieurs sources d'informations indépendantes, le système multi-agents effectuant une fusion des informations par l'intermédiaire de la dynamique de l'environnement.

Chapitre 7 : Application à la localisation en robotique mobile

Le but de ce chapitre est de montrer la viabilité de notre architecture dans une situation réelle. Le principal objectif ici est de montrer que les résultats précédemment obtenus peuvent être vérifiés en situation réelle. En outre, nous voulons également voir quels sont les cas où le dispositif multi-agents est plus avantageux qu'un dispositif de localisation classique ainsi que ceux où les résultats obtenus sont moins probants. Pour ce faire, nous allons effectuer une série de tests comparatifs entre notre dispositif et un algorithme utilisant un filtre de Kalman (cf. description dans le paragraphe 2.4.2.1).

Pour ces tests réels, nous avons été obligés de changer de type de plate-forme par rapport aux essais préliminaires de localisation introduits dans le chapitre 6 et présentés en détail dans [Gechter01c]. En effet, le développement de ces dispositifs de localisation classiques a été effectué en parallèle avec celui de l'architecture multi-agents et celui du robot « Simplet » de l'équipe MAIA. Au début de ce développement, nous espérions construire plusieurs robots à partir de la même architecture ce qui nous aurait fourni un ensemble de plates-formes de test adapté à notre application. Malheureusement, le développement de l'architecture matérielle ayant pris quelque peu de retard, nous avons dû nous reporter sur une autre architecture pouvant convenir à la définition de notre cadre applicatif. Nous avons donc choisi d'utiliser les robots « footballeurs » disponibles au sein de notre équipe de recherche.

Après une brève présentation du cadre expérimental, nous allons donc exposer les résultats des différents tests que nous avons effectués. Nous commencerons par exposer les résultas des tests préliminaires destinés à confirmer les résultats obtenus dans le chapitre 5. Ensuite, nous comparerons notre dispositif avec un filtre de Kalman sur une trajectoire ovoïde parcourue par un robot. Nous étudierons, en outre, l'apport de la fusion de percept sur les performances de notre dispositif. Enfin, nous conclurons en comparant les propriétés obtenues en situation réelle avec celles déjà montrées lors des simulations.

7.1. Le cadre expérimental

Comme nous l'avons expliqué dans l'introduction, nous avons utilisé les minis robots footballeurs de l'équipe pour effectuer les tests de localisation de cibles multiples.

Le dispositif expérimental est composé d'une dizaine de robots de type char disposant chacun de deux roues motrices et de supports métalliques simulant une liaison de type roue libre avec le sol (cf. figure 47).

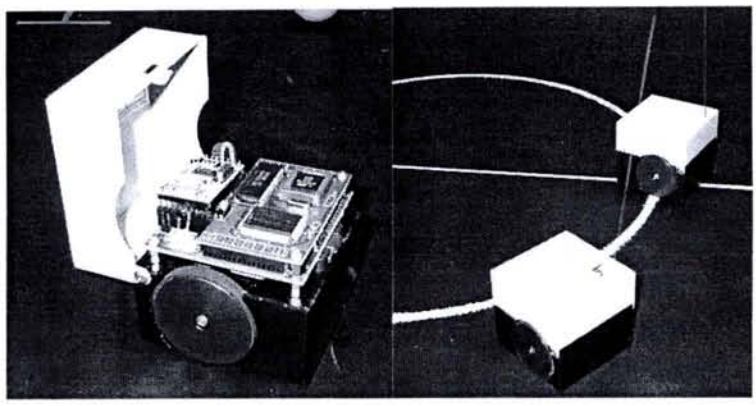

figure 47 : Les minis robots de tests

Ces robots sont commandés par un PC envoyant des ordres à chacun d'entre eux sous la forme de couple de vitesse pour chacune des deux roues motrices. La perception de l'environnement et de la position des robots s'effectue par une caméra CCD unique disposée sur un mat télescopique situé au dessus du plateau d'évolution. Bien que ne disposant réellement que d'un seul capteur, il est possible de construire plusieurs unités perceptives (UP) différentes en modifiant les algorithmes de traitement utilisés

En ce qui concerne l'architecture du dispositif de localisation, nous avons conservé le même type d'organisation que celle utilisée pour les tests simulés (cf. figure 26). La seule différence se situe dans le remplacement des générateurs de plots par le ou les dispositifs de perceptions. La transmission entre les unités perceptives et le système réactif s'effectue toujours via un socket sur un port défini permettant ainsi une délocalisation du processus de résolution sur une machine différente de celle utilisée pour le contrôle et la perception de l'environnement.

7.2. Tests préliminaires

7.2.1. Tests de continuité de la trajectoire

L'objectif de ces tests est de montrer que l'inertie des groupes d'agents, obtenue grâce à l'utilisation de la satisfaction (retard de la disparition du groupe) et aux forces physiques mises en jeu (ralentissement progressif du groupe), permet de compenser, en partie, l'absence momentanée d'information concernant une cible.

Pour ces tests, nous avons choisi de coupler la caméra CCD de notre installation à un algorithme de traitement à base d'analyse en composantes principales analogue à celui

utilisé pour les tests de construction d'unités perceptives (paragraphe 6.2). Etant donné que l'ACP est réputée pour fournir une information discrète sur la position des robots (cf. figure 48), elle peut être considérée comme une source d'information de mauvaise qualité fournissant de façon discontinue les informations.

Nous avons ensuite, à partir des informations fournies par l'ACP, détecté les positions successives prises par le même groupe d'agents au cours du déplacement du robot. La figure 49 représente un des tests que nous avons effectué.

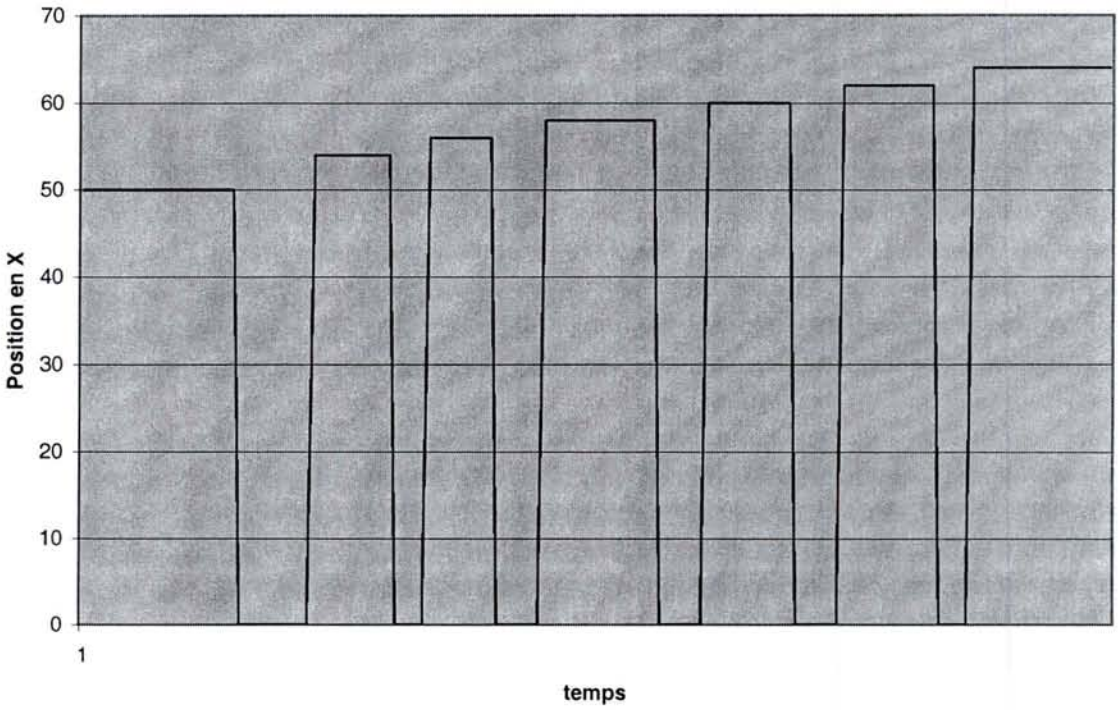

figure 48 : Percepts discontinus fournis par l'ACP (position en X en ordonnée) au cours du temps lors d'une phase de tests réels

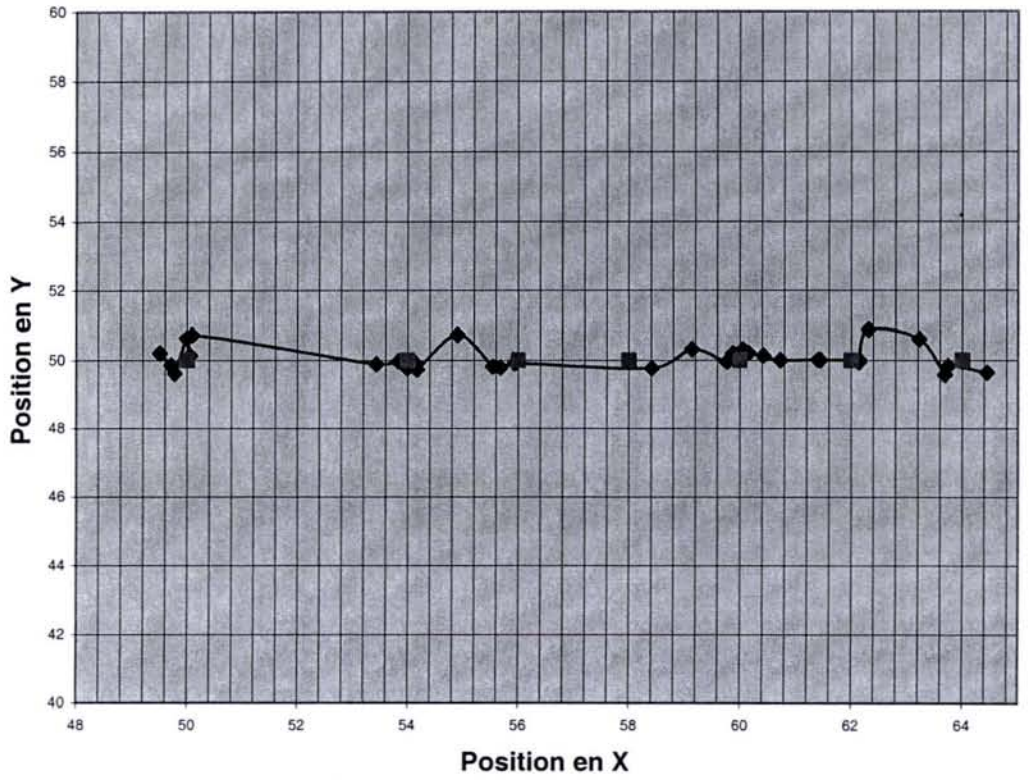

figure 49 : Ensemble des positions fournies par les dispositif réactif en fonction des percepts issus de l'ACP

Nous pouvons voir, à partir des résultats présentés sur la figure 49, que le processus de localisation est tout de même tributaire de la qualité de la source d'information. En effet, il n'y a pas de lissage complet de la trajectoire de la cible détectée. Néanmoins, nous pouvons remarquer que, bien que les percepts ne soient présents que ponctuellement dans l'environnement, le système conserve la trace de la cible. Ce phénomène est essentiellement dû à la conservation de la cohésion des groupes sous l'influence de la valeur de satisfaction des agents qui inhibe la répulsion. Ainsi, même s'il n'est pas possible de deviner totalement la trajectoire des cibles entre deux percepts, le système permet tout de même de conserver la trace d'une cible dans l'environnement en attendant l'apparition d'un nouveau percept. Il y a donc bien une continuité dans la trajectoire, c'est le même groupe du début à la fin du suivi, mais le parcours de celle-ci se fait en fonction des percepts fournis, le déplacement est lent autour des percepts et rapide entre ceux-ci.

7.2.2. Test de suivi avec un seul robot

Pour les tests qui suivent, nous avons choisi de changer d'unité perceptive devant les grandes limitations reconnues de l'analyse en composantes principales (discontinuité des perceptions, forte sensibilité aux conditions de prise de vue,...). Nous avons donc utilisé l'unité perceptive construite à partir d'algorithmes de type filtre de Kalman.

Au delà du but de montrer que le système fonctionne en situation réelle, l'objectif de ces tests est d'estimer l'erreur de localisation connaissant les positions successives du robot dans le monde réel et de comparer l'influence de la partie perceptive à celle du processus de résolution. Nous nous sommes limités à un seul robot afin de faciliter les différentes mesures d'erreur.

Au cours de ceux-ci, nous avons donc mesuré la position réelle du robot de façon directe sur la table et comparé celle-ci avec les informations issues de l'unité perceptive d'une part et du dispositif de localisation d'autre part.

Les résultats obtenus à partir d'une vingtaine de manipulations sont présentés dans le tableau suivant :

	Erreur (pourcentage / Taille du monde)	***Ecart Type (pourcentage / Taille du monde)***
Dispositif de perception par rapport à la position réelle	0,92%	0,96%
Localisation réactive / Perception	0,98%	1,11%

figure 50 : Erreur et écart type sur la localisation en situation réelle

Ces tests ont été effectués avec une vitesse de robot très faible afin de s'affranchir du phénomène de retard induit par le dispositif réactif lorsque la vitesse des agents devient faible par rapport à la vitesse des cibles d'une part, et des ralentissements dans le traitement des informations à cause de la complexité de l'algorithme d'autre part. En effet, notre objectif n'est pas de mesurer les limitations dues à la façon dont a été

programmé l'algorithme de localisation mais celles induites par le mécanisme de fonctionnement.

Les résultats du tableau précédent montrent que le dispositif de localisation réactif génère à peu près autant d'erreur que le dispositif de perception. Les résultats obtenus sont assez encourageants. Cependant, ils se dégradent fortement lorsque le robot va de plus en plus vite pour les deux raisons précitées.

7.2.3. Tests avec plusieurs robots

- Tests simples

De la même façon que pour les tests précédents, nous voulons vérifier que le système fonctionne correctement avec plusieurs cibles. Pour cela, nous utilisons deux robots ayant des trajectoires différentes pouvant éventuellement se croiser. Nous évaluons ensuite l'erreur générée par le processus, calculée à partir de la différence entre la position supposée par l'unité perceptive et celle déterminée par le centre de gravité des groupes d'agents respectifs.

Nous avons également étendu ces tests à un nombre de robots supérieur à deux. Dans tous les cas, le système a fonctionné de façon adaptée en détectant toutes les cibles présentes sur la table et sans ralentissement par rapport au cas mono-cible.

Nous avons, bien sûr, retrouvé dans ce cas de fonctionnement les limitations présentes lors de la phase de tests simulés lorsque deux robots sont proches les uns des autres.

- Ajout et suppression de cible en cours de fonctionnement

En plus des tests précédents, nous sommes également intéressés à l'apparition et à la disparition des cibles. Nous avons donc observé la réaction du processus en faisant varier le nombre de cibles présentes sur la table. Conformément à ce que nous avions mis en évidence lors des tests simulés, le fait d'enlever ou d'ajouter des cibles en cours de fonctionnement n'affecte pas le fonctionnement du processus.

7.2.4. Tests avec plusieurs unités perceptives

Afin de vérifier la possibilité d'utiliser de façon simultanée ou de façon alternative plusieurs unités perceptives, nous avons placé une seconde caméra dans l'environnement mais avec un angle différent de la première. Nous avons ensuite essayé de faire varier en cours de fonctionnement le nombre de sources d'information en supprimant en alternance l'un ou l'autre des deux dispositifs. Dans tous les cas, le système n'a pas ressenti de variation de fonctionnement ce qui montre bien que le processus de localisation réactif est robuste aux variations du nombre d'unités perceptives utilisées.

En outre, nous avons également ajouté aux dispositifs de perception des générateurs de plots virtuels fournissant des informations supplémentaires dans l'environnement des agents. Dans le cas de générateur de bruit aléatoire, le système montre la même robustesse qu'en tests simulés. En outre, si les générateurs fournissent des trajectoires classiques, le système considère que celles-ci sont de nouvelles cibles à localiser. Il n'y a donc pas de différence entre l'information fournie par les capteurs réels et celle fournie par les générateurs de plot.

Enfin, le dernier test que nous avons effectué a pour but pour montrer la capacité de notre dispositif à gérer des environnements de grande taille qui ne sont pas intégralement couvert par les unités perceptives. Pour cela, nous considérons que la zone couverte par les capteurs réels ne couvre qu'une partie de l'environnement des agents, la seconde partie de celui-ci étant couverte par des générateurs de plots analogues à ceux que nous avons utilisés lors des tests simulés. (cf. figure 51)

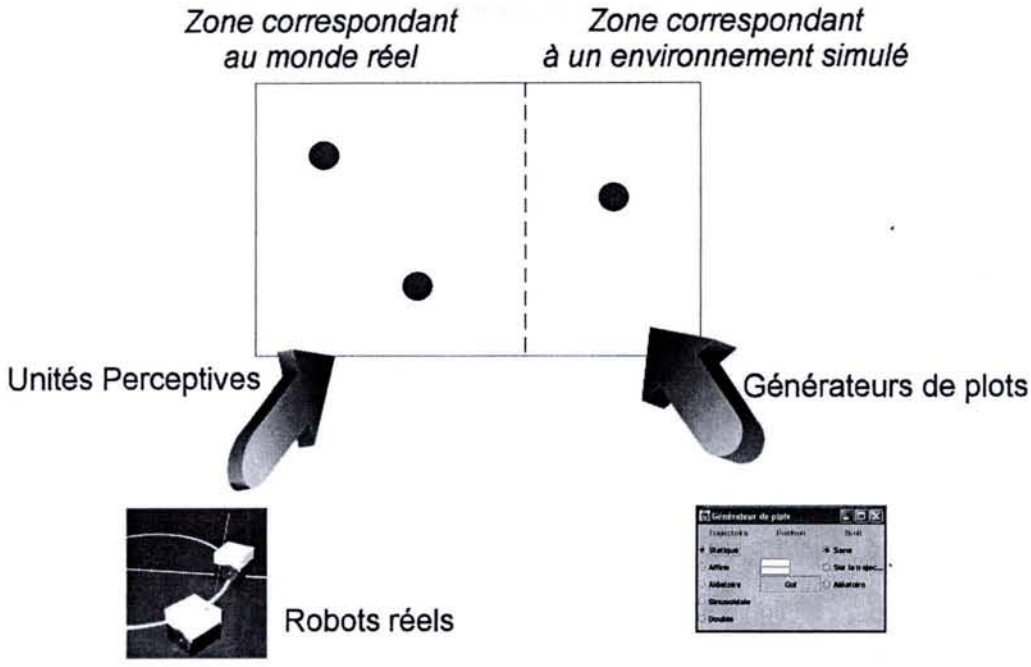

figure 51 : Schéma de l'architecture de tests pour des environnements de grande taille

Les tests effectués avec cette architecture montrent que le processus de localisation réactif fonctionne de manière transparente par rapport à la complexité du dispositif d'observation. Ainsi, tout se passe comme si les cibles se trouvaient dans la même zone du monde réel. C'est-à-dire qu'il n'y a pas de différence de traitement entre les informations fournies par les unités perceptives et celles issues des générateurs de plots.

Cette expérience montre qu'il est possible d'utiliser notre proposition d'architecture pour effectuer de la localisation de cibles multiples dans des cas où l'environnement ne peut pas être réduit à une surface entièrement perceptible. Il est donc tout à fait envisageable d'effectuer de la localisation dans des locaux de bureau par exemple où l'environnement global est décomposé en petites parcelles localement perceptibles.

7.3. Tests comparatifs avec un filtre de Kalman

7.3.1. Objectifs et mise en œuvre

L'objectif de ces tests est d'évaluer les performances de notre dispositif de localisation et de suivi vis-à-vis d'un algorithme de référence en suivi de cible : le filtre de Kalman. Notre but n'est pas de concurrencer la méthode de filtrage de Kalman, mais d'effectuer une comparaison objective entre ces deux approches pour évaluer leurs performances relatives.

Pour ces tests, nous avons comparé l'erreur moyenne et l'écart type de cette erreur pour chacun de ces dispositifs. Ne pouvant pas connaître la trajectoire réellement suivie par le robot, nous avons été obligé d'effectuer ce comparatif relativement à une ***trajectoire de référence*** fournie par une unité perceptive. Peut-être aurait-il été plus intéressant, de mesurer les performances à partir d'une trajectoire réelle. Cependant, la trajectoire utilisée permet tout de même de mettre en évidence les performances comparées des deux algorithmes. Dans chacune des figures présentées par la suite, la trajectoire débute en début de ligne droite supérieur en partant vers la gauche, la trajectoire étant parcourue dans le sens anti-trigonométrique.

Pour mesurer l'erreur et l'écart type, nous avons comparé à intervalle de temps régulier la position de la cible estimée par le dispositif de perception avec celles estimées à la fois par le filtre de Kalman d'une part et par le dispositif multi-agents d'autre part. Etant donné la taille de l'environnement des agents utilisé et la taille de l'espace réel de test, nous avons mesuré qu'une erreur de 1 correspond à un décalage de 1,5 cm sur l'espace de test, celui-ci étant un carré de 2 m de coté. Cet ordre de grandeur est valable pour tous les tests comparatifs.

7.3.2. Le filtre de Kalman utilisé

Le filtre que nous avons utilisé est un filtre de Kalman classique. Dans un premier temps, on évalue la position prédite par le filtre en utilisant des informations sur la dynamique de cible ainsi qu'une estimation de l'erreur commise lors de cette prédiction sous la forme d'une matrice de covariance. Le résultat ainsi obtenu est affiné, dans un second temps, par l'observation effectuée sur le processus et l'estimation *a priori* de l'erreur d'observation (matrice de covariance).

En ce qui concerne le modèle, nous avons choisi d'adopter un modèle de trajectoire linéaire pour la cible à suivre. C'est-à-dire que la position prédite correspond à un

déplacement de la cible depuis la position courante en suivant un vecteur de déplacement correspondant à la vitesse courante. Ce modèle simple est plutôt bien adapté à la trajectoire utilisée puisque celle-ci est essentiellement composée de lignes droites de grande taille reliées par des portions courbes possédant un rayon de courbure assez grand. (cf. figure 52)

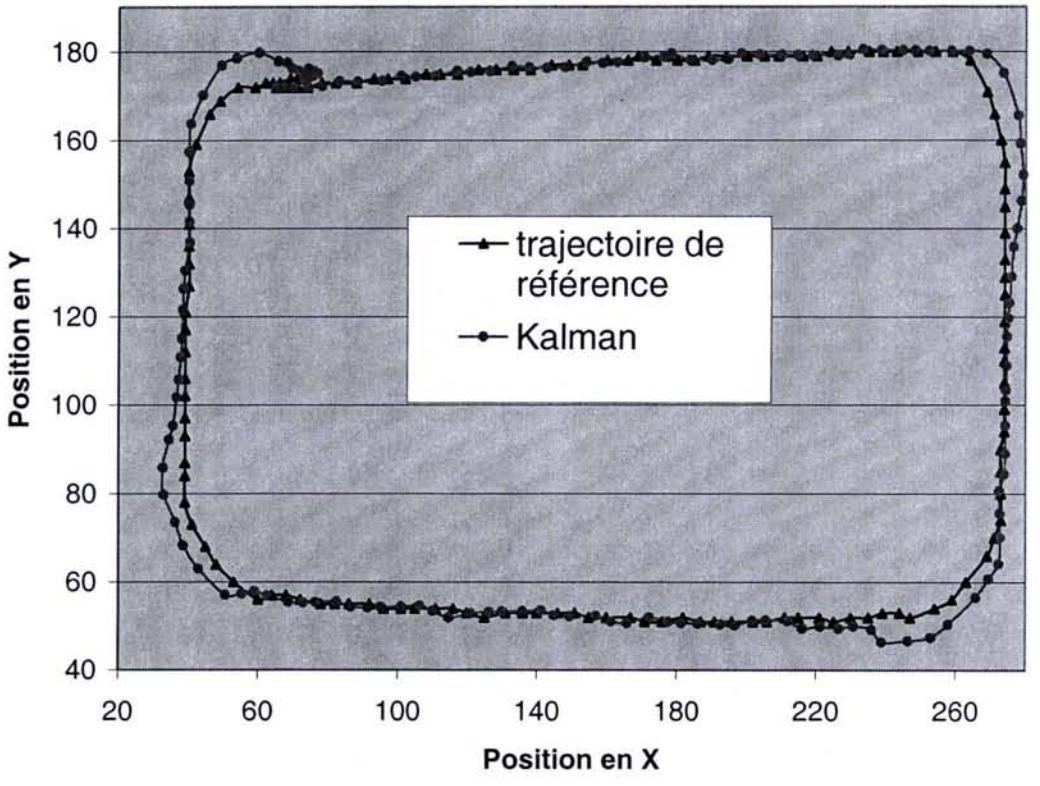

figure 52 : Exemple de trajectoire de référence et de filtre de Kalman

Pour les réglages du filtre, nous nous sommes contentés de modifier les matrices de covariance concernant l'erreur d'observation et l'erreur de prédiction. Le reste des paramètres du filtre étant déterminé de façon analytique à partir du modèle linéaire de déplacement, le vecteur d'état de la cible étant composé de sa position et de sa vitesse.

7.3.3. Résultats à partir d'une source d'informations de référence

La courbe présentée dans la figure 53, montre le résultat d'un suivi de cible effectué avec notre algorithme multi-agent. La trajectoire de référence est une trajectoire ovoïde estimée à partir d'observations sur les déplacements d'un robot réel de type robot footballeur.

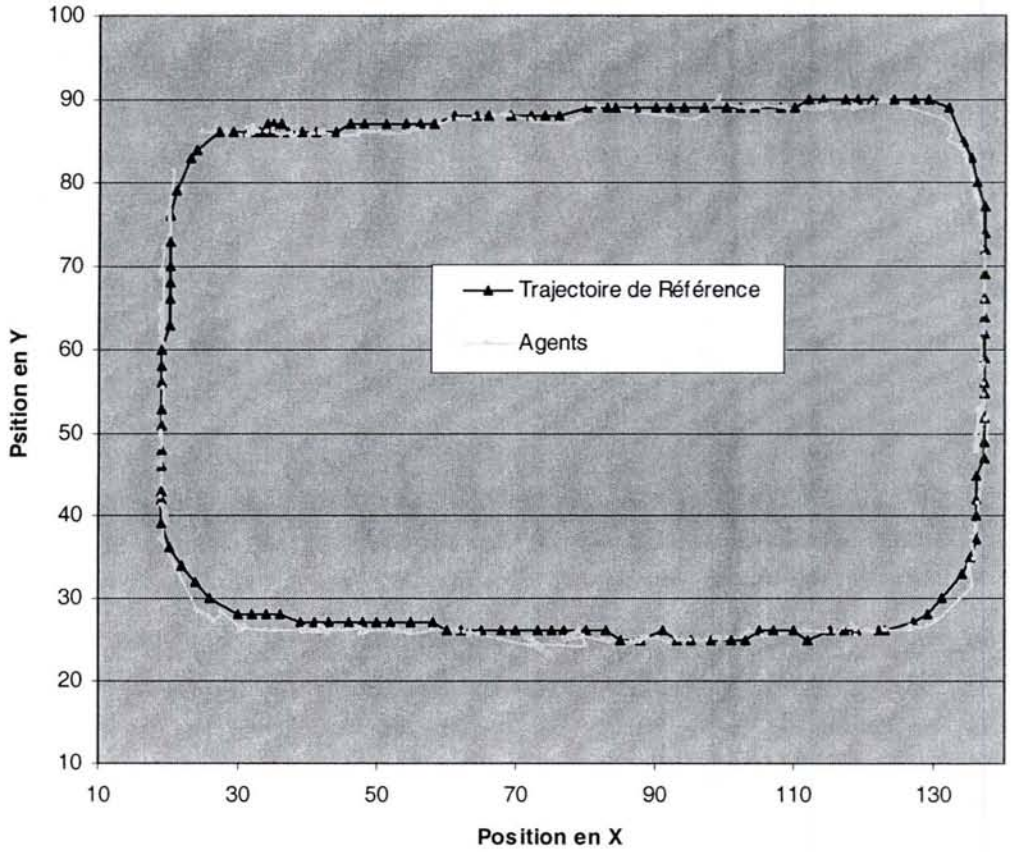

figure 53 : Exemple de trajectoire de référence et de suivi avec les agents

Par rapport au suivi effectué avec un filtre de Kalman (cf. figure 52), le suivi obtenu est moins lissé : il y a plus d'oscillations autour de la trajectoire de référence. Par contre, les décalages lors des changements de direction sont plus rapidement rattrapés que pour un filtre de Kalman. Le tableau suivant montre les résultats comparatifs des deux algorithmes sur un ensemble de trajectoires ovoïdes parcourues par le robot.

	Filtre de Kalman	Système Multi-Agent
Erreur Moyenne	0,421	0,715
Ecart Type	0,510	1,292

figure 54 : Résultats comparatifs entre Kalman et le système multi-agent

Les résultats exposés dans la figure 54 montrent un avantage au filtre de Kalman surtout en ce qui concerne l'écart type. Néanmoins, les résultats obtenus pour l'erreur moyenne sont du même ordre ce qui est encourageant concernant la qualité de la réponse fournie par notre dispositif. La principale différence entre les algorithmes se situe dans la connaissance du processus observé. En particulier, notre dispositif ne nécessite pas de connaissance concernant la position initiale de la cible à suivre ce qui n'est pas le cas pour le filtre de Kalman.

7.3.4. Résultats avec une source d'informations bruitée

Pour ces tests, nous avons choisi de bruiter la trajectoire de référence avec un bruit blanc gaussien d'écart type correspondant au double de l'erreur commise par le dispositif multi-agent dans le cas où la trajectoire est non bruitée (écart type de 2)

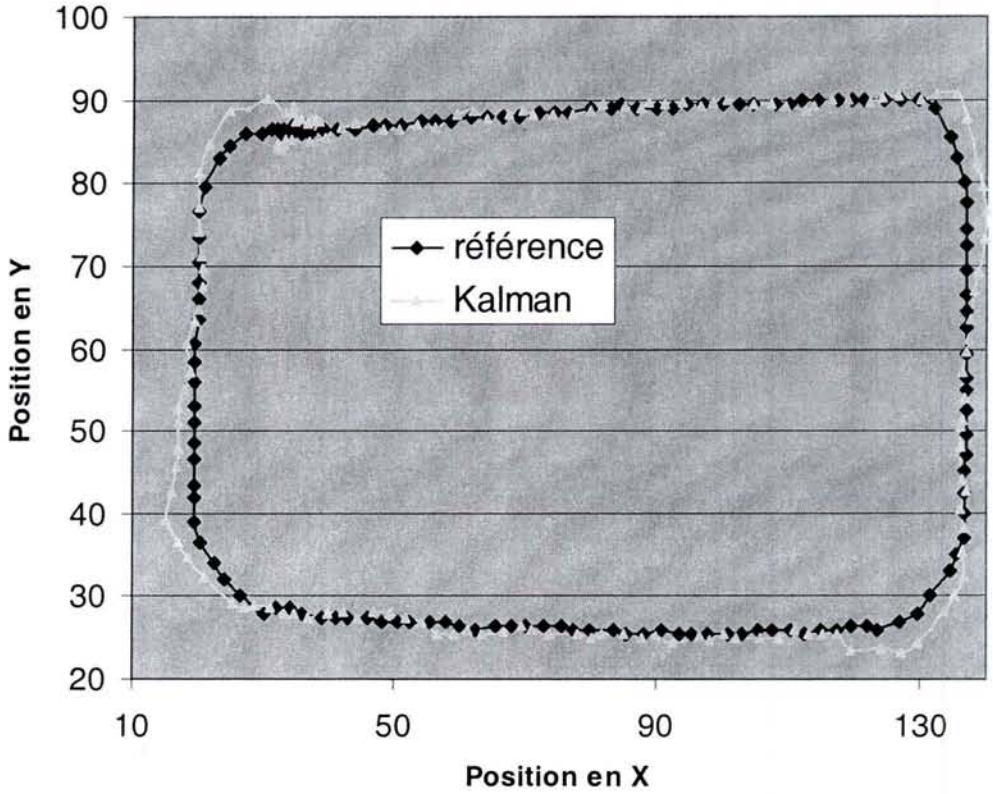

figure 55 : Résultats obtenus avec un filtre de Kalman avec une trajectoire de référence bruitée

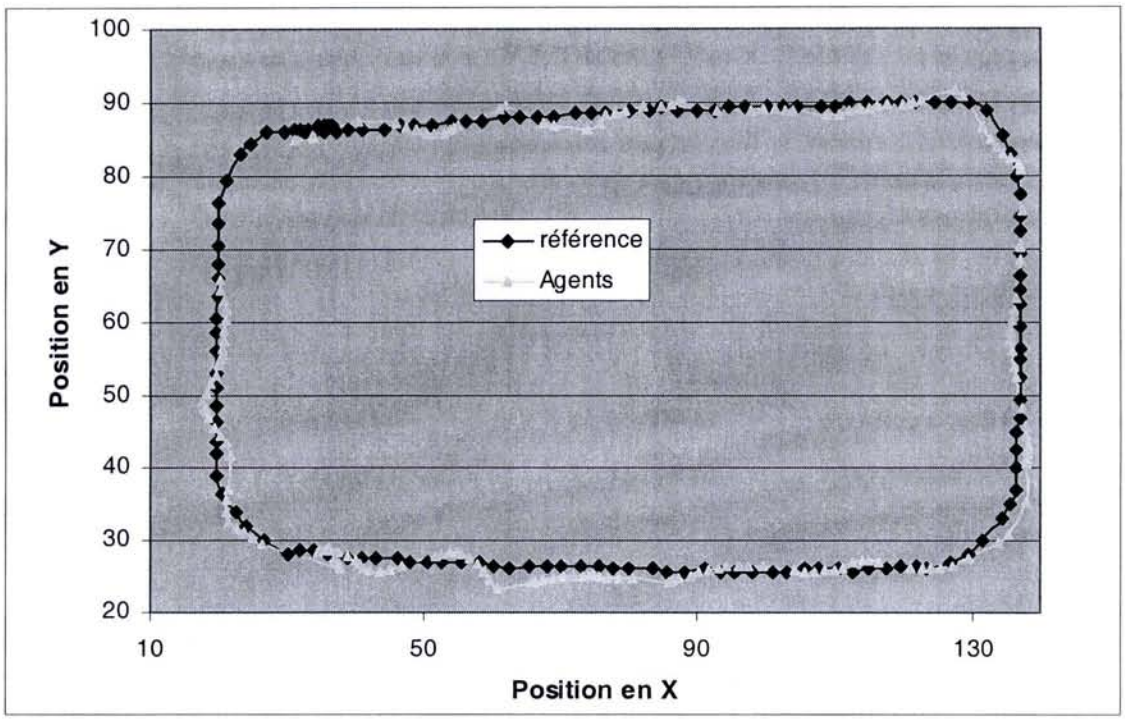

figure 56 : Résultats obtenus avec le dispositif multi-agent avec une trajectoire de référence bruitée

	Filtre de Kalman	Système Multi-Agent
Erreur Moyenne	1,064	0,953
Ecart Type	1,004	1,350

figure 57 : tableau comparatif entre le filtre de Kalman et le dispositf multi-agent pour une référence bruitée

Les trois figures précédentes confirment à la fois les résultats comparatifs obtenus avec une trajectoire de référence non bruitée et les résultats obtenus lors des tests simulés avec le dispositif multi-agent.

Les deux algorithmes testés montrent ici quasiment les mêmes performances vis-à-vis du bruit. Ni l'un ni l'autre ne s'avère être très robuste à ce type d'information.

A l'instar de ce nous avions mis en évidence lors des tests simulés avec une trajectoire bruitée, le dispositif multi-agent est très réactif au bruit. Aussi, le groupe d'agents suit le bruit de façon assez précise ce qui se caractérise par une oscillation un peu plus prononcée autour de la trajectoire idéale.

7.3.5. Bilan

Les résultats obtenus, lors de ces tests comparatifs (cf. figure 58), montrent que le filtre de Kalman est légèrement meilleur que notre dispositif de suivi pour des trajectoires non bruitée, cette différence s'estompant avec l'introduction d'un bruit blanc sur la trajectoire. Cependant, la mise au point du filtre de Kalman nécessite beaucoup plus de connaissance sur le processus observé. En effet, lorsqu'il s'agit de construire un filtre de Kalman, il est impératif de connaître :

- La position initiale de la cible
- La matrice de transition entre deux états successifs de la cible (modèle de déplacement)
- L'erreur commise sur la prédiction et donc sur le modèle dynamique
- L'erreur commise sur les observations

		Filtre de Kalman	Système Multi-Agent
Source = trajectoire de référence	*Erreur Moyenne*	0,421	0,715
	Ecart Type	0,510	1,292
Source = trajectoire bruitée	*Erreur Moyenne*	1,0640	0,9530
	Ecart Type	1,0040	1,1830

figure 58 : tableau récapitulatif des résultats comparatifs entre notre approche et un filtre de Kalman

Le dispositif multi-agent permet de s'affranchir d'une partie de cette connaissance.

D'une part, dans notre système, la position initiale de la cible est inconnue. La détection de nouvelles cibles se fait de façon automatique. Ce problème de l'initialisation des pistes est un problème crucial lorsqu'on utilise un filtre de Kalman, notamment dans le cas d'applications militaires de suivi de pistes radar pour lesquelles le problème n'est pas encore totalement résolu.

D'autre part, il n'y a pas de connaissance sur l'erreur d'observation. Ceci peut effectivement poser un problème pour notre système puisque une information fausse conduit directement à une évaluation fausse de la position de la cible. Cependant, il est possible d'introduire cette connaissance de l'erreur en modulant l'altitude des plots fournis par les Unité Perceptives (UP) en fonction de leurs qualités. Ainsi, un percept de moins bonne qualité se traduira par un plot ayant une altitude plus faible. En outre, cette lacune peut en partie être compensée par la possibilité de fusionner les informations issues de différentes UP ce qui n'est pas le cas du filtre de Kalman classique.

Enfin, les deux dispositifs nécessitent une connaissance du modèle dynamique des cibles. Dans le cas du dispositif multi-agent, ce modèle est implicite et présent dans les mécanismes d'interactions mis en place. En effet, le fait d'utiliser des forces physiques avec prise en compte de l'inertie constitue en soi une prise en compte d'une forme de

modèle dynamique. Pour le filtre de Kalman, le modèle est plus rigide dans le sens où il est difficilement modifiable en cours de fonctionnement, si ce n'est pas la prise en compte d'une erreur de prédiction. Dans le cas du dispositif multi-agent, les paramètres qui régissent les interactions sont tout à fait modifiables en cours de fonctionnement. En outre, on peut envisager une modification non globale des paramètres en favorisant une spécialisation des groupes d'agents en fonction de la cible suivie.

7.4. Tests de Fusion

7.4.1. Objectifs et mise en œuvre

L'objectif de ces tests est de mettre en évidence l'importance de la fusion dans l'amélioration des résultats de localisation et de suivi dans un cadre réel avec des sources d'informations qui peuvent parfois être bruitées.

Dans ce cadre, nous avons mené deux séries de tests.

La première consiste en l'utilisation de trois trajectoires bruitées avec des bruits blancs gaussiens différents possédant des paramètres analogues à celui que nous avons utilisé pour les tests précédents (écart type de 2%). La figure 59 montre les trois sources d'informations bruitées utilisées.

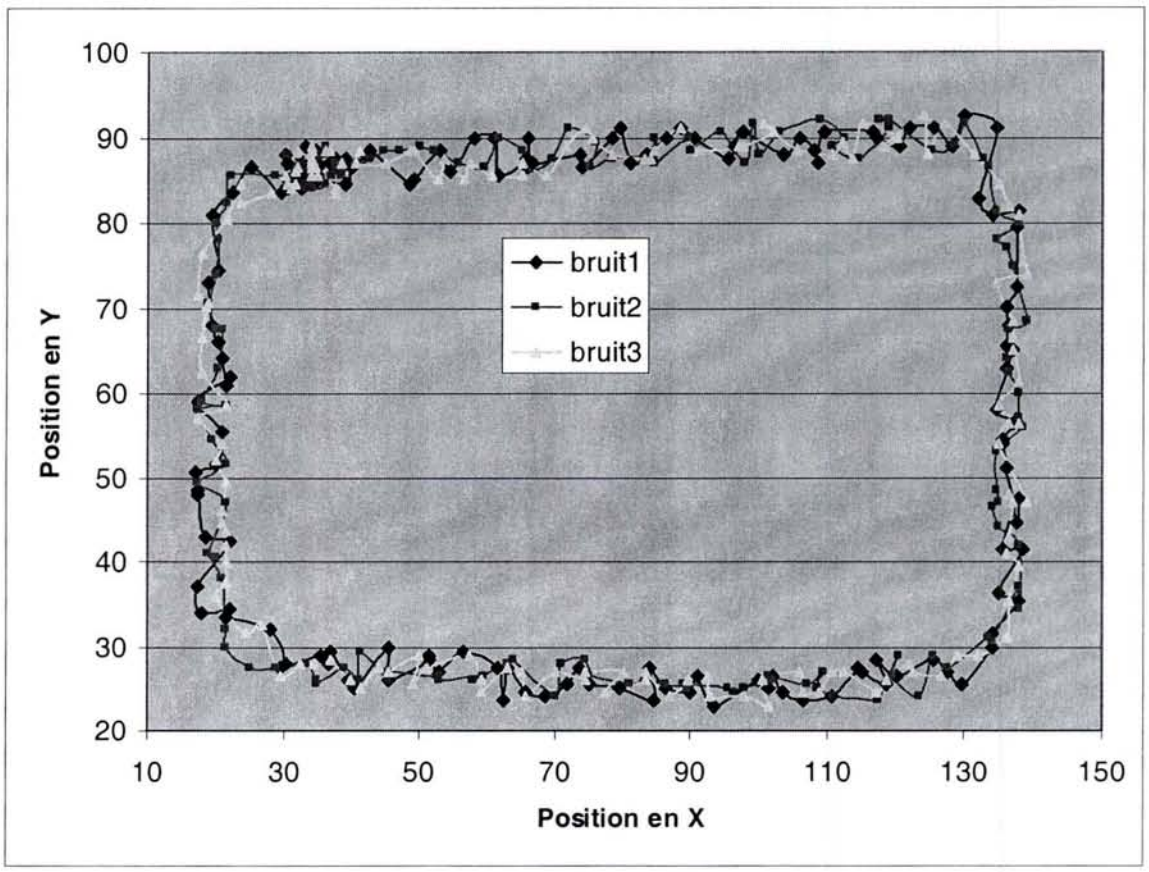

figure 59 : Les trois sources bruitées utilisées pour les tests de fusion

Pour la seconde série, nous avons utilisé, en entrée de notre dispositif multi-agent, les résultats de trois filtres de Kalman différents. L'association filtre de Kalman / Dispositif d'acquisition peut être considérée comme une unité perceptive à part entière conforme à la définition que l'on en a donné dans le chapitre précédent. Ceci permet donc d'associer de façon conjointe une méthode de traitement du signal performante avec notre dispositif dans le but d'améliorer les résultats obtenus. Dans le cadre, de l'utilisation des trois filtres, nous avons également effectué des tests en modifiant la pondération de chacun des filtres par l'intermédiaire de variations de l'altitude des plots fournis par chacun des dispositifs. De la même façon que pour les tests comparatifs, une erreur de 1 correspond à un décalage de 1,5 cm sur l'espace d'évolution des robots.

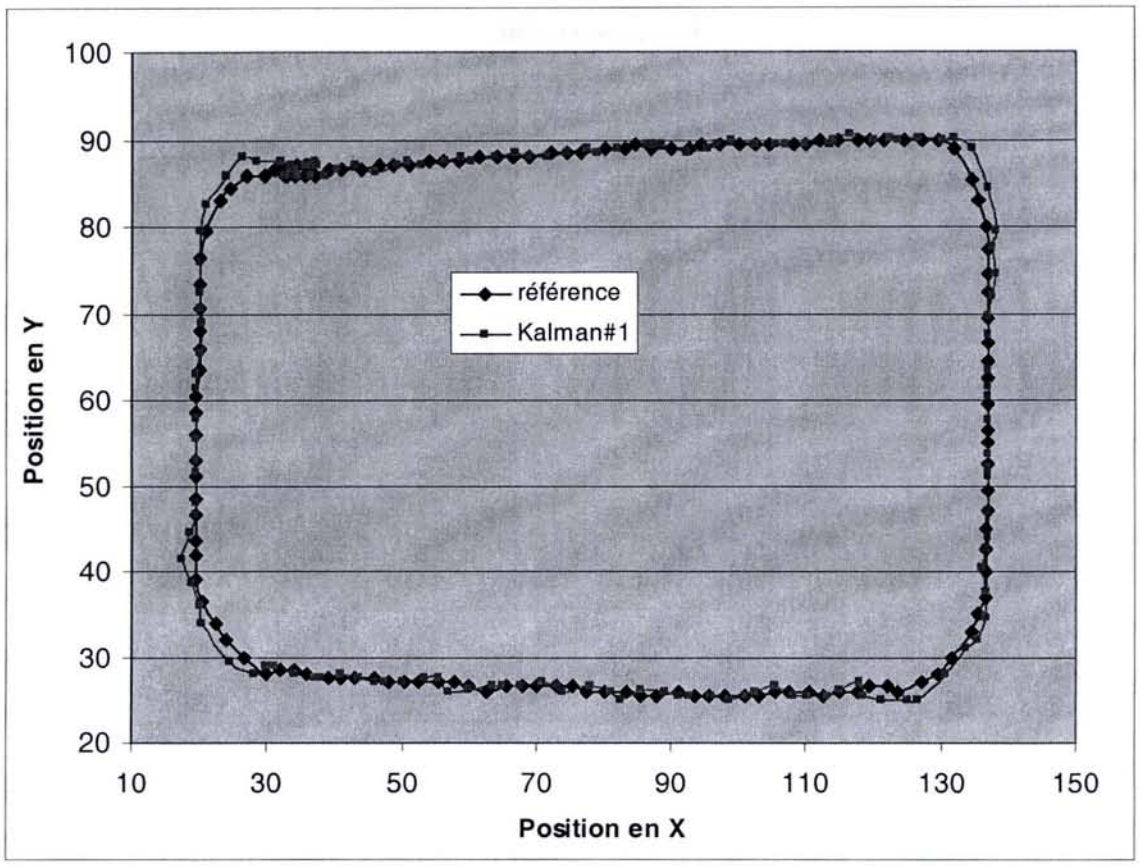

figure 60 : le premier filtre de Kalman utilisé

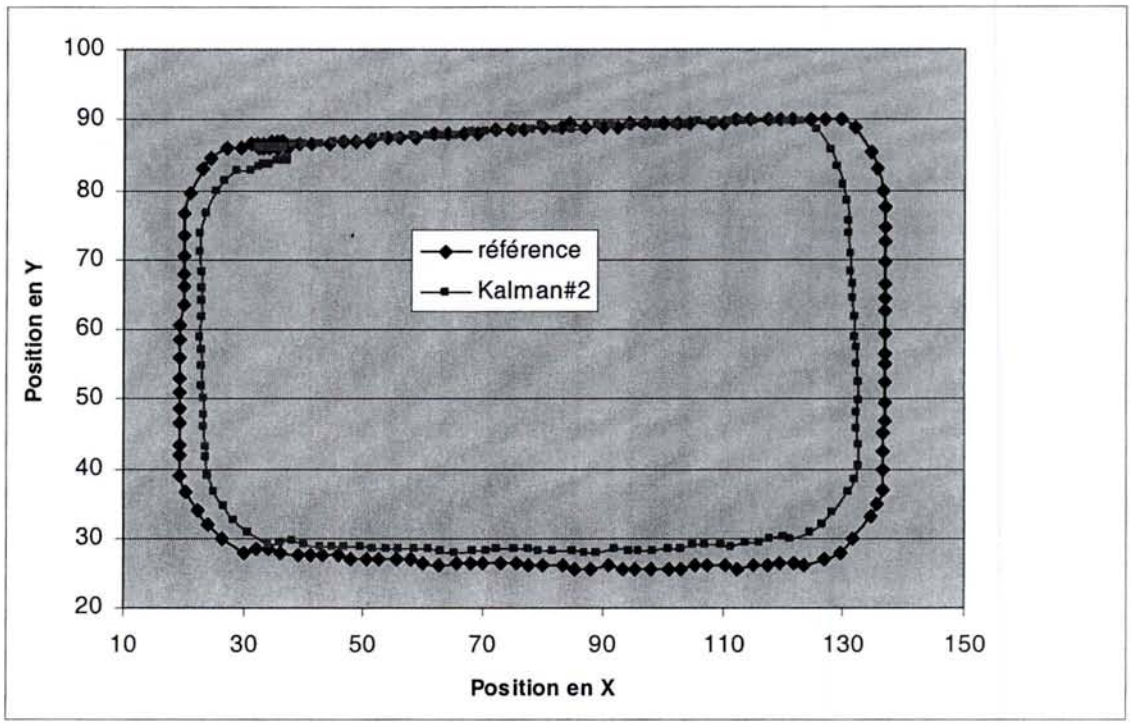

figure 61 : le second filtre de Kalman utilisé

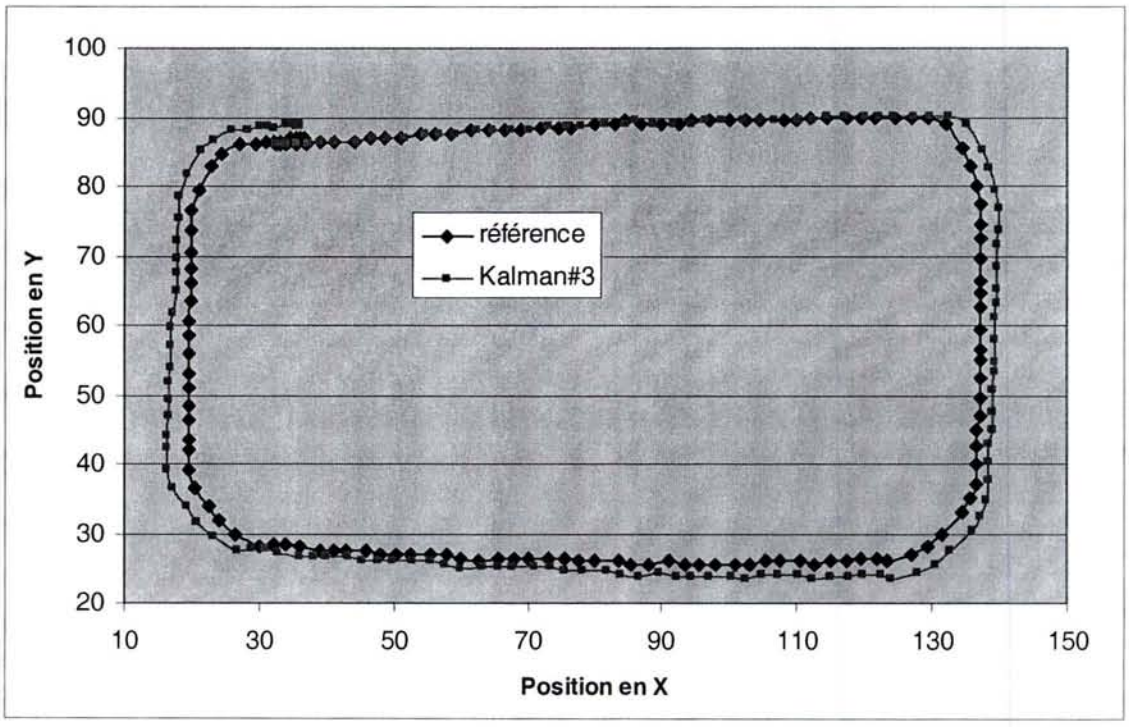

figure 62 : le troisième filtre de Kalman utilisé

Les trois figures précédentes montrent les trois filtres de Kalman utilisés. Le premier constitue un filtre de bonne qualité avec une proportion équilibrée entre la prédiction due au modèle dynamique et la correction engendrée par les observations. Le second représente un filtre où la phase de prédiction est prépondérante et où le paramètre temporel du modèle (temps de passage d'un état au suivant) est sous-estimé ce qui se traduit par une anticipation de l'évolution. Le dernier filtre est en fait identique au second sauf pour le facteur temporel qui est, cette fois-ci, sur-estimé.

Une fois les sources d'informations mises en place, nous avons mesuré l'erreur moyenne et l'écart type engendrés par notre dispositif relativement à la trajectoire idéale fournie par l'unité perceptive de référence.

7.4.2. A partir de plusieurs sources bruitées

La courbe suivante (cf. figure 63) nous montre la trajectoire fournie par notre dispositif après fusion des informations issues des trois sources bruitées pondérées de façon identique.

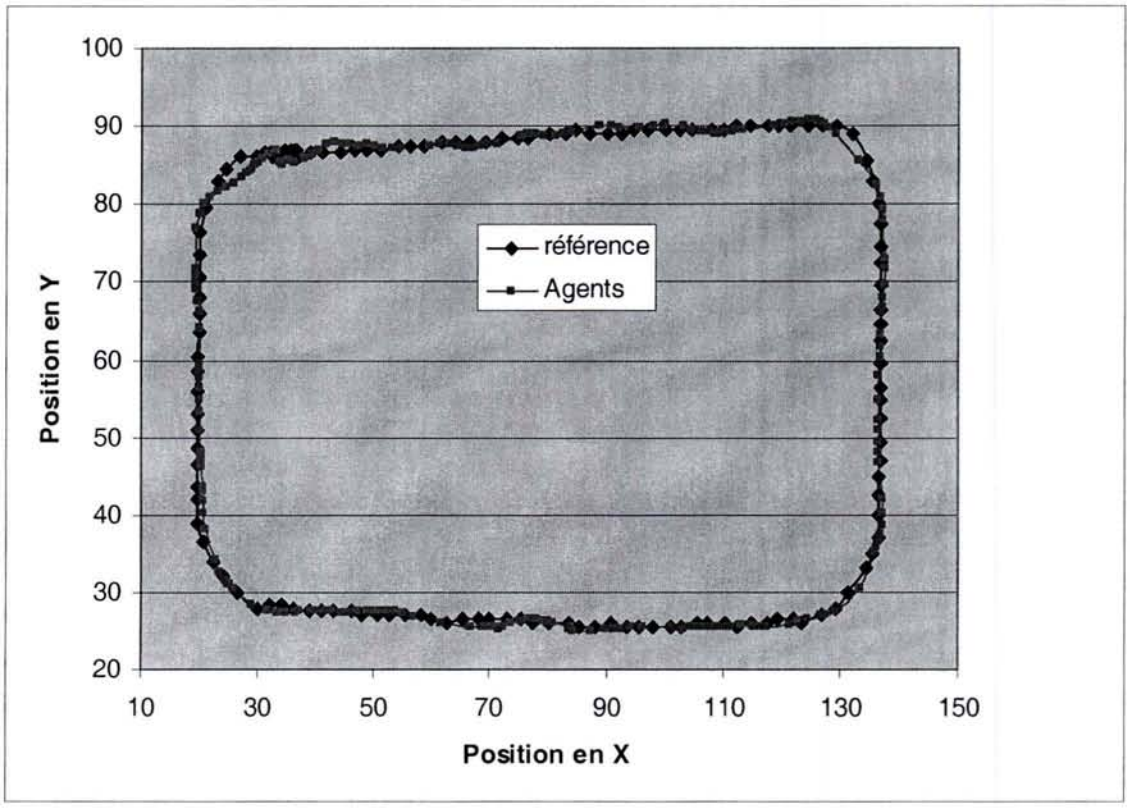

figure 63 : Résultats obtenus par le dispositif multi-agent à partir de trois sources d'informations bruitées

	Système Multi-Agent
Erreur Moyenne	0,533
Ecart Type	0,392

figure 64 : erreur moyenne et écart type obtenus à partir de trois sources bruitées

Les résultats présentés dans le tableau précédent montrent une nette amélioration des performances de notre algorithme lorsque la fusion est utilisée. Par rapport, aux résultats avec une trajectoire bruitée seule, l'amélioration est de 44%. Dans le cas présent, les résultats sont meilleurs que ceux obtenus avec pour seule entrée la trajectoire de référence. En effet, la fusion d'informations permet de compenser les oscillations autour de la trajectoire idéale ce qui se traduit par un meilleur résultat global et surtout par un écart type nettement moins important (environ 25% de l'écart type obtenu avec la trajectoire de référence). La fusion des informations présente donc un

grand intérêt pour l'amélioration de la qualité des résultats obtenus. Elle permet notamment de compenser les effets des différents bruits sur le résultat final.

7.4.3. A partir de trois filtres de Kalman différents

La figure 65 montre la trajectoire fournie par notre dispositif multi-agent en utilisant comme source trois filtres de Kalman différents pondérés de façon uniforme.

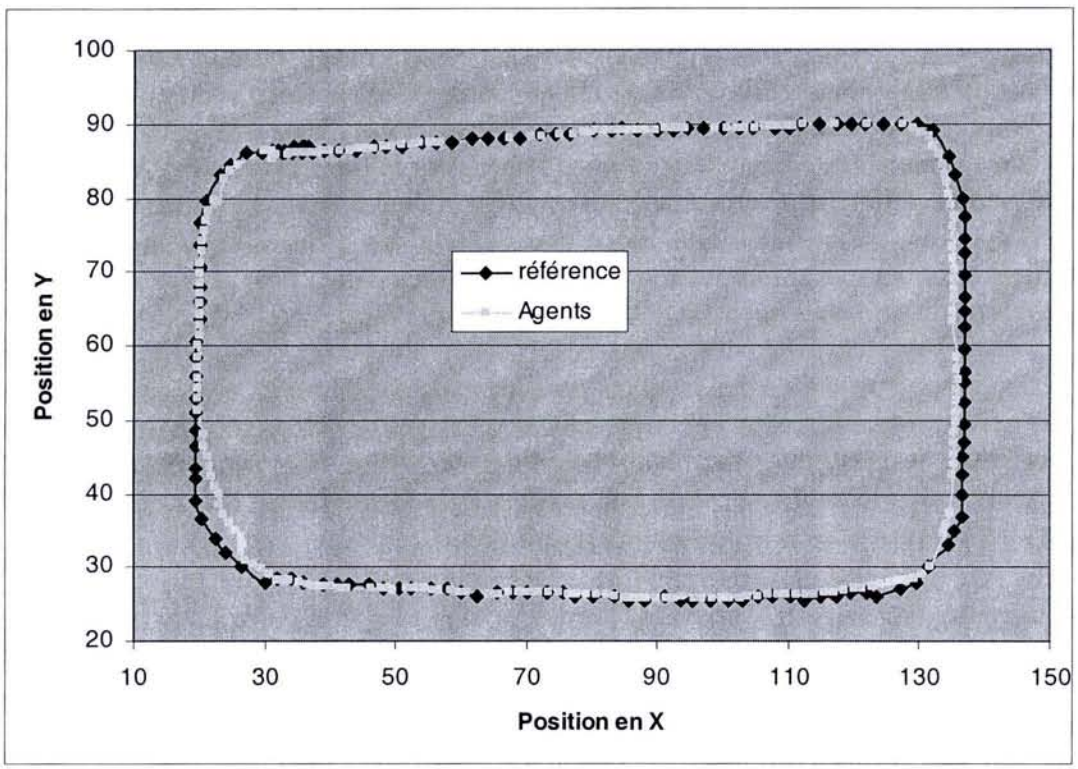

figure 65 : Résultats obtenus avec plusieurs filtres de Kalman comme sources d'informations (pondération uniforme)

	Système Multi-Agent
Erreur Moyenne	0,613
Ecart Type	0,769

figure 66 : Erreur moyenne et écart type avec trois filtres de Kalman pondérés de façon uniforme

Les résultats exposés dans le tableau précédent sont un peu meilleurs que ceux obtenus avec une trajectoire de référence non bruitée utilisée comme source d'information unique. Néanmoins, le gain n'est pas très important (14,2%). On observe notamment que la trajectoire a tendance à se trouver à l'intérieur de la trajectoire idéale. Ceci est dû au fait que le filtre de Kalman qui sous-estime le facteur temporel fourni une trajectoire qui possède plus de décalage avec la trajectoire idéale que celle fournie par le filtre sur estimateur. Une pondération plus faible pour la trajectoire intérieure aurait sans doute permis d'obtenir un résultat bien meilleur. D'autre part, on peut remarquer que l'écart type est nettement moins élévé que dans le cas d'une source unique. Le fait d'utiliser trois sources différentes, ayant une cohérence entre elles, permet de réduire les oscillations du groupe d'agents autour de la trajectoire idéale.

Dans le but d'améliorer les résultats obtenus avec des filtres de Kalman pondérés de façon uniforme, nous avons choisi de modifier les pondérations en favorisant le filtre qui effectue le meilleur suivi. Nous avons donc utilisé une pondération deux fois supérieure pour le meilleur filtre par rapport à la pondération des deux autres. La trajectoire alors obtenue est présentée sur la figure 67.

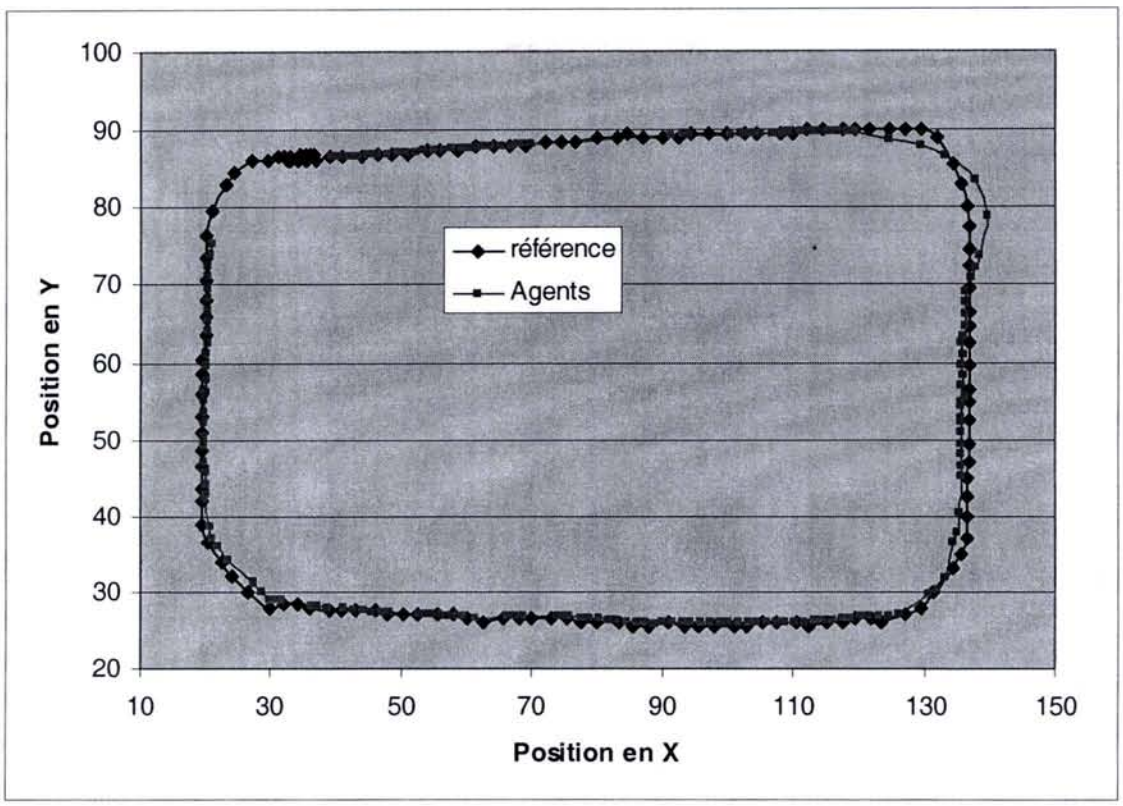

figure 67 : Résultats obtenus avec plusieurs filtres de Kalman comme sources d'informations (pondération non uniforme)

	Système Multi-Agent
Erreur Moyenne	0,471
Ecart Type	0,478

figure 68 : Erreur moyenne et écart type avec trois filtres de Kalman pondérés de façon non uniforme

Cette fois-ci, les résultats sont nettement meilleurs que pour le test précédent. Nous avons alors un gain de 34% par rapport à une utilisation avec la trajectoire idéale seule. Cependant, le résultat est un peu moins bon que celui obtenu avec le filtre de Kalman seul. Ceci est parfaitement normal puisque le fait de fusionner les trois sources d'information corrompt les informations obtenues par le meilleur des filtres. De la même façon que pour le cas précédent, l'écart type est assez faible à cause du processus de fusion des informations.

Afin de mettre davantage en évidence l'importance de la fusion, nous avons décidé d'ajouter aux trois filtres de Kalman, pondérés de manière uniforme, la trajectoire de référence. Le tableau suivant montre les résultats obtenus

	Système Multi-Agent
Erreur Moyenne	0,313
Ecart Type	0,380

figure 69 : Résultats obtenus avec une fusion de trois filtres de Kalman et de la trajectoire de référence.

Les résultats, présentés dans le tableau précédent, sont les meilleurs que nous ayons obtenus (gain de 56% sur l'erreur par rapport à l'utilisation de la trajectoire de référence comme seule source d'informations cf. figure 70). Ceci est très encourageant pour notre dispositif puisqu'ils montrent que les choix architecturaux (connection et déconnection à loisir de plusieurs sources d'informations) et conceptuels (mécanismes de fusion, concept d'unité perceptive) que nous avons effectués sont justifiés. En outre, l'ajout et la suppression de sources d'information sont suffisamment simples pour pouvoir envisager la construction de dispositifs de localisation très élaborés, utilisant un nombre important d'unités perceptives différentes, et utilisant le même dispositif multi-agent.

Entrée	Trajectoire de référence	3 sources bruitées	3 filtres de Kalman pondérés de façon uniforme	3 filtres de Kalman avec prépondérance pour le meilleur	3 filtres de Kalman pondérés de façon uniforme + trajectoire de référence
Erreur Moyenne	0,715	0,533	0,613	0,471	0,313
Ecart Type	1,292	0,392	0,769	0,478	0,380

figure 70 : tableau récapitulatif des résultats obtenus en fusionnant différent percepts.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

7.5. Conclusion

Comme nous l'avons dit en introduction de ce chapitre, ces tests ont pour principal objectif de montrer que notre processus de localisation et de suivi fonctionne également dans un cadre réel. En outre, ils ont permis de confirmer certaines propriétés déjà mises en évidence lors des tests simulés :

- Le processus permet de compenser en partie l'absence momentanée d'information concernant une cible *(tests avec l'ACP).*
- Il n'engendre pas d'imprécision supplémentaire dans le cadre réel par rapport aux tests simulés, l'erreur de localisation et son écart type sont du même ordre *(tests de suivi avec un seul robot).*
- Il est capable de suivre plusieurs cibles de façon simultanée et en nombre variable *(tests avec plusieurs cibles).*
- La variation du nombre d'unités perceptives n'influe pas sur le processus de localisation et de suivi. *(tests avec plusieurs unités perceptives).*

D'autre part, ces tests ont permis d'approfondir l'évaluation de notre dispositif notamment en effectuant un comparatif avec le filtrage de Kalman utilisé dans de nombreuses applications de suivi. Les résultats bruts, obtenus avec une trajectoire de référence comme unique source d'informations, de notre algorithme sont sensiblement moins bons que ceux fournis par Kalman. Par contre, les résultats s'équilibrent lorsque la source devient bruitée. Outre ces tests de performance, nous avons mis en évidence les points communs et les différences entre ces deux algorithmes. Bien que fournissant des résultats de meilleure qualité, le filtre de Kalman nécessite une plus grande connaissance du processus observé. Si les deux algorithmes intègrent une connaissance du modèle dynamique de la cible, notre dispositif ne nécessite pas de connaissance particulière concernant la position initiale de la cible, l'initialisation des pistes se faisant de façon automatique.

Enfin, nous avons également testé le mécanisme de fusion de notre dispositif. Ces tests ont permis de mettre en évidence l'intérêt de la fusion d'informations dans le cadre du suivi de cible. Nous avons notamment vu que le mécanisme mis en place permettait :

- de compenser le bruit présent sur les différentes sources.
- de diminuer les oscillations du groupe d'agents autour de la trajectoire idéale.
- d'utiliser des algorithmes évolué, tels que le filtre de Kalman, comme source unité perceptive.
- De moduler l'importance des résultats fournis par les unités perceptives en fonction de leur qualité.

Dans sa version actuelle, notre dispositif de localisation ne peut pas être utilisé de façon confortable avec des robots fonctionnant au maximum de leur potentiel. En effet, il reste de nombreuses limitations principalement dues à l'implémentation de

l'algorithme et à la puissance limitée des machines de tests. Néanmoins, les problèmes rencontrés ne sont pas liés ni à notre démarche conceptuelle ni au modèle que nous proposons, mais à sa réalisation matérielle. La levée de ces problèmes ne devrait donc pas poser trop de difficulté pourvu que l'on se penche sur l'optimisation du programme et que l'algorithme utilisé soit optimisable.

Chapitre 8 : Conclusion

8.1. Rappel du sujet

L'objectif de cette thèse est de montrer qu'il est possible de construire un système fournissant une réponse intelligente à un problème complexe réel à partir d'entités simples, disposant de mécanismes de prise de décision limités.

La démarche que nous avons choisie d'adopter s'inspire des approches de type systèmes auto-organisés. Une telle approche implique que l'obtention d'une réponse du système provient d'un état d'équilibre entre une dynamique propre à l'environnement et une dynamique issue de la conjonction des interactions entre les agents. Les points clefs d'une telle démarche sont :

- La construction d'un environnement dynamique caractérisant le problème à résoudre à partir d'observation du monde réel.
- L'élaboration d'un processus de résolution permettant de fournir une réponse au problème posé par l'environnement.
- L'interprétation des résultats fournis par le processus en fonction des contraintes liées à la tâche à résoudre.

Afin de mettre en œuvre cette démarche, nous avons choisi d'appliquer ce modèle à un cadre applicatif complexe et souvent abordé en robotique : la localisation et le suivi. Cet objectif concret nous a permis d'adapter notre proposition à une application type non triviale, tout en nous laissant l'opportunité de démontrer l'utilité et le bien fondé de notre approche dans un cadre réel.

8.2. Bilan et Apports

Au cours de cette thèse, nous avons abordé la résolution du problème de localisation et de suivi selon la démarche suivante. Dans un premier temps, nous avons construit une architecture réactive s'appuyant sur des lois physiques d'interaction. Ensuite, nous avons effectué son évaluation en simulation. Enfin, nous avons expérimenté cette architecture en situation réelle.

Nous présentons ci-dessous le bilan selon deux optiques : la première, liée au modèle proposé, expose ses avantages et ses limitations et la seconde qui présente l'emploi de l'architecture dans le cadre applicatif.

8.2.1. Modèle

Nous avons, dans cette thèse, défini un processus de résolution utilisant des agents réactifs simples interagissant entre eux et avec leur environnement en fonction de lois physiques.

L'utilisation que nous avons faite des SMA constitués d'agents réactifs permet de respecter les propriétés et les contraintes que nous avions énoncées en introduction. En particulier, le processus de résolution que nous avons construit ne nécessite pas de construction ou de manipulation de représentations complexes du problème, l'environnement étant représenté sous la forme d'un ensemble d'états et de plots.

Un des principaux intérêts de notre approche est de montrer comment la modélisation du contrôle via l'interaction avec l'environnement peut se faire par l'intermédiaire de modèles tirés de la physique classique. L'utilisation que nous faisons de ces modèles peut être assimilée à l'élaboration d'une physique de l'environnement. Ainsi, ce qui est modifié lors de la mise au point du système, ce n'est pas la façon dont les agents vont réagir aux variations de l'environnement mais comment ce dernier va influer sur les éléments qu'il contient.

D'autre part, la dynamique de gestion des nouvelles sources d'information est également une caractéristique intéressante de notre approche. En effet, le fait de gérer l'apport de nouvelles sources, dynamiquement, à travers un principe d'accumulation de percepts dans l'environnement des agents permet d'aborder le problème de la fusion d'informations d'un point de vue réactif. Cette fusion réactive n'est rendue possible que parce que nous nous sommes affranchis du coté symbolique des sources d'informations tout en généralisant, sous la forme d'unités perceptives, les mécanismes de construction dynamique de l'environnement. Ainsi, là où il était possible de voir uniquement le désir de normaliser les protocoles d'acheminement de l'information vers le système, il faut également déceler, dans le concept d'unité perceptive, l'ambition d'une formalisation des principes de construction de percepts. Le but final de ces derniers est d'aborder la fusion sous une forme réactive plus simple à mettre en place que les méthodes classiques.

8.2.2. Application

Les différents tests effectués, tant en simulation qu'avec des robots réels, nous ont permis de montrer que l'utilisation de SMA constitués d'agents réactifs permet de résoudre la problématique de la localisation et du suivi de cibles. En outre, cette approche réactive est caractérisée par la mise en évidence d'un certain nombre de propriétés remarquables :

- Le système étant dynamique, il est possible d'ajouter ou de retirer des agents en cours de fonctionnement rendant ainsi possible une adaptation des moyens de résolution en fonction de la complexité du problème à traiter.
- La population étant homogène, le processus de résolution est identique quelle que soit la configuration initiale du problème et surtout quelles que soient les variations de celui-ci en cours de fonctionnement.
- Le dispositif est capable de gérer un nombre de cibles quelconque et variable en cours de fonctionnement.
- Le nombre d'unités perceptives peut également être variable en cours de fonctionnement, ce qui est assez délicat à envisager dans le cadre d'autres algorithmes de localisation centrés sur l'exploitation des capteurs.
- Le système permet de gérer la panne des unités perceptives de par le principe d'accumulation de l'environnement.
- De par son indépendance à la nature des sources d'informations, et contrairement aux approches classiques dans lesquelles le domaine de localisation dépend directement de la portée du capteur, il est possible d'envisager l'utilisation de notre approche avec des environnement réels de taille importante et/ou divisés en plusieurs sous-environnements. En effet, il est tout à fait envisageable d'utiliser notre processus, sans aucune modification, pour des environnements complexes tels qu'un ensemble de bureaux reliés les uns aux autres et disposant de capteurs dont la perception est limitée par la présence d'obstacles tels que des murs ou des portes.

Les tests comparatifs nous ont permis de mettre en évidence les qualités de notre approche vis-à-vis d'une méthode de suivi éprouvée. Les résultats obtenus montrent que notre dispositif est, dans l'absolu, moins performant qu'un filtre de Kalman. Cependant, lorsque cette source devient bruitée, les résultats de notre approche se dégradent un peu moins et se traduisent essentiellement par une accentuation des oscillations autour de la trajectoire idéale (augmentation de l'écart type de l'erreur).

Dés lors que l'on utilise les mécanismes de fusion d'informations de notre dispositif, les résultats deviennent meilleurs que ceux obtenus avec un filtre de Kalman. En outre, la souplesse de notre approche vis-à-vis de la nature intrinsèque des unités perceptives permet d'envisager une utilisation conjointe avec le filtrage de Kalman afin de bénéficier des avantages des deux méthodes.

La principale limitation de notre dispositif est essentiellement d'ordre technique. En effet, le fait d'avoir recours à un grand nombre d'agents et d'utiliser d'importantes ressources réseau pour la transmission provenant des unités perceptives augmente, de façon importante, le temps de réponse du système. Pour le moment, il est donc délicat d'envisager une utilisation avec des environnements de grande taille qui nécessite un nombre important d'agents.

Ainsi, notre objectif n'était pas de proposer une méthode supplémentaire de localisation mais d'aborder le problème sous un regard neuf. Il s'agit d'envisager le processus de résolution de problème non pas comme le choix d'un ensemble de traitements dépendants de la nature des sources d'informations mais comme une unité de raisonnement à laquelle il faut adapter la connaissance que l'on a du problème.

Non seulement nous avons réussi à proposer une architecture simple, indépendante des capteurs et des traitements utilisés, mais en plus celle-ci permet d'effectuer une tâche de localisation et de suivi avec des performances tout à fait acceptables.

8.3. Problèmes non encore résolus

Les tests de performances effectués sur le système permettent d'avoir une idée sur le bon fonctionnement du processus de résolution. Cependant, les moyens employés pour caractériser les qualités de notre approche sont essentiellement des déductions faites à partir de situations typiques. Pour le moment, il n'est pas possible de déterminer un critère de qualité directement à partir des mécanismes mis en place, il faut avoir recours à des mesures multiples du résultat (position des groupes, nombre des groupes, ...) tout en connaissant les paramètres des situations de tests auxquelles le dispositif est confronté. Nous n'avons pas encore réussi à fournir une grandeur mesurable permettant la caractérisation de la performance du système indépendamment de toute connaissance sur les caractéristiques du problème à résoudre.

De cette absence de mesure découle le second problème que nous n'avons pas encore résolu : celui de la mise au point des paramètres comportementaux et de la physique de l'environnement. En effet, toute la phase de mise au point a été réalisée de façon empirique à l'aide des mesures d'erreur réalisées, même si nous avons été aidés par le fait que le modèle physique sous-jacent permette une prédiction assez aisée de l'influence des paramètres. Cette difficulté dans la mise au point est un inconvénient majeur de ce type d'approche mais qui est commun à toutes les approches à base d'agents. Néanmoins, le fait que notre système repose sur des lois physiques existantes permet d'envisager la possibilité de construire une mesure s'inspirant de celles utilisées dans les systèmes physiques classiques.

Enfin, le dernier problème à résoudre est la détermination des trajectoires. En effet, pour le moment, nous considérons une trajectoire comme étant une succession de localisations cohérentes dans l'espace et dans le temps. Cette détermination n'est pas satisfaisante car elle ne permet pas encore la construction de trajectoire continue pour les différentes cibles et empêche donc d'approfondir la possibilité qu'a notre dispositif à identifier les cibles à partir de leur dynamique. Cependant, je pense qu'il n'est pas possible de traiter directement le problème des trajectoires d'un point de vue local au sein des agents de localisation. En effet, la construction d'une trajectoire ne peut être perçue que comme un phénomène macroscopique. Celles-ci ne peuvent donc pas *a*

priori être élaborées à partir de considérations locales. Ceci n'est donc pas un problème de notre processus de résolution, mais davantage un problème de l'architecture globale. Pour le résoudre, il faudrait donc ajouter à notre dispositif, une partie interprétative de haut niveau s'appuyant sur les caractéristiques des groupes formés.

8.4. Améliorations et perspectives

Dans le paragraphe précédent, nous avons vu que certains problèmes rencontrés au cours de l'élaboration et de la validation de notre dispositif n'avaient pas encore été résolus. Cependant, nous disposons de plusieurs pistes pour améliorer les limitations rencontrées et étendre les perspectives d'utilisation de notre approche.

8.4.1. Du point de vue applicatif

Du point de vue applicatif, le principal objectif à court terme, est l'optimisation du programme afin de le rendre exploitable avec des robots fonctionnant à leur vitesse optimale.

Dans un premier temps, il s'agit de s'affranchir des cotés limitatifs de la plate-forme de développement madkit® qui n'est pas complètement adaptée à notre application car elle possède de nombreuses fonctionnalités inutiles pour notre application, qui alourdissent le traitement. Il nous faudra donc remplacer le moteur synchrone utilisé par un autre mécanisme de gestion des interactions propre à notre application.

En outre, il est également nécessaire de se pencher davantage sur l'algorithme utilisé afin d'optimiser certains de ses éléments qui sont trop coûteux en ressources vis-à-vis de la tâche requise. Enfin, il est également possible d'envisager la partie affichage sous la forme d'une architecture client/serveur, afin de libérer un maximum de ressources pour le cœur du système.

Pour le moment, les tests avec des robots réels n'ont été effectués que dans le cadre de la localisation et du suivi avec capteurs externes. *A priori*, l'utilisation de notre modèle réactif pour l'auto-localisation ne devrait pas poser de problème car celui-ci est indépendant de la nature des sources d'informations. Cependant, il est indispensable de réaliser des tests plus approfondis concernant l'utilisation de capteurs embarqués.

Le dispositif que nous avons créé s'occupe uniquement de la partie perceptive et permet uniquement de fournir des informations sur les positions et les mouvements des robots. Aussi, il serait intéressant d'essayer d'intégrer une structure de contrôle des robots utilisant le même formalisme. Une solution possible consisterait en l'intégration de cette structure de contrôle en tant que perturbation du système sous la forme de plots fictifs générés artificiellement. L'objectif de ces plots fictifs serait d'influer sur le

comportement des groupes d'agents qui pourrait alors être liés de façon unique aux robots réels. Ce lien entre les groupes et les robots exige au préalable une identification des cibles suivies.

8.4.2. Du point de vue du modèle

Pour le moment, le réglage des paramètres a été effectué de façon empirique à partir des mesures d'erreur réalisées pour des situations typiques. Il serait intéressant d'exploiter ces mesures afin, par exemple, d'effectuer un apprentissage total ou partiel des paramètres du système. Parmi les algorithmes d'apprentissage existants, on peut très bien envisager l'utilisation de méthodes de descente de gradient ou d'apprentissage par renforcement pour améliorer la réponse du système à une configuration spatio-temporelle du monde réel donnée. Cependant, dans le cadre de l'apprentissage, il faudra également prendre en compte davantage les phénomène locaux que les interprétations de l'organisation globale.

Nous avons déjà effectué quelques tests d'apprentissage sur des trajectoires ovoïdes bruitées du même type que celles présentées dans le chapitre 7. Les premiers résultats que nous avons obtenus sont très encourageants (diminution de l'erreur de l'ordre de 50%) mais ils demandent à être approfondis. En effet, nous pourrions effectuer un apprentissage non pas sur une trajectoire mixte comme un ovale mais sur des trajectoires typiques différentes (trajectoires rectilignes, cercles, portion de courbes, ...). Ces différents apprentissages permettraient de disposer d'un ensemble de paramètres permettant de caractériser des trajectoires différentes. Dès lors, on peut envisager d'adapter, en cours de fonctionnement, les paramètres de certains groupes d'agents en fonction de la cible suivie. Cette spécialisation des comportements des agents devrait être en mesure d'améliorer les résultats obtenus tout en nous offrant l'opportunité d'effectuer une identification des type cibles, chaque type étant caractérisé par un ensemble de paramètres du modèle.

Les mesures d'erreur effectuées ne permettent d'envisager qu'un apprentissage sur des cas typiques (trajectoires rectilignes, trajectoires sinusoïdale, cibles statiques,...). Par contre, il n'est pas possible avec de telle mesure de doter notre processus de capacité d'auto-adaptation. Pour cela, il faut disposer d'une grandeur mesurable permettant de caractériser la performance du système sans connaissance sur la configuration du problème à résoudre (mesure de vitesse d'obtention d'une configuration stable, estimation de la qualité de la stabilité,...). Parmi les pistes que nous pensons être dignes d'intérêt, la notion d'entropie semble être une solution de choix puisqu'elle est directement inspirée du monde de la physique et semble être en mesure de fournir une estimation du désordre spatio-temporel des agents de localisation. Cependant, les méthodes de calcul de l'entropie présentes dans la littérature [Balch00], [Parunak01], [Kanada94] ne semblent pas forcément pouvoir apporter une solution efficace à notre problème. En effet, celles-ci s'attachent davantage à estimer

l'organisation spatiale d'un point de vue global (niveau macroscopique) sans forcément prendre en compte les phénomènes (niveau microscopique) qui ont conduit à une telle organisation. Le mieux que l'on puisse espérer est alors une simple caractérisation mathématique des différents états macroscopiques stables (état de stabilité fondamental ou focalisation sur un nombre de plots défini). Ainsi, si l'on veut que cette mesure de désordre rende compte des phénomènes microscopiques, il faut élaborer une formulation s'appuyant sur les bases comportementales des agents et s'inspirant du mode de calcul de l'entropie dans les différentes branches de la physique. Il m'est d'avis que la solution réside dans l'ambivalence de la notion d'entropie et de son interprétation énergétique puisque l'obtention d'un extrema entropique correspond toujours à un extrema énergétique. Dans notre cas, la solution consisterait éventuellement à calculer l'entropie à partir de l'énergie calculée pour chacun des agents et donc à partir de leur état interne.

De plus, lorsque l'on regarde l'ensemble de l'architecture proposée, il est possible de la considérer comme une unité perceptive particulière, étant donné que c'est un dispositif fournissant des données numériques représentant un certain état du monde réel. Il serait donc intéressant de mettre en cascade plusieurs architectures du même type de façon à répondre à un problème plus complexe. Ainsi, d'un point de vue conceptuel, il serait judicieux d'approfondir cette idée en proposant, par exemple, un principe de résolution qui consisterait en la séparation du problème à résoudre en plusieurs sous problèmes plus simples, et pouvant être résolus par une approche similaire à celle que nous venons de proposer. La résolution du problème initial se ferait donc par assemblage de structures réactives.

Enfin, et à plus long terme, étant donné que notre système repose sur des lois physiques éprouvées, il serait intéressant de formaliser d'un point de vue mathématique tout le processus de construction d'un système multi-agents réactif devant résoudre une tâche donnée. Cette formalisation mathématique devra sûrement s'inspirer des concepts mis en place dans les phénomènes de physique non linéaire ou encore dans la théorie du chaos. L'objectif serait alors de montrer qu'il est possible de construire une théorie mathématique permettant d'expliquer l'obtention des propriétés observées.

Chapitre 9 : Bibliographie

[Abdi97] H. Abdi, D. Valentin and B.G. Edelman, *Eigenfeatures as intermediate level representations: The case for PCA models*, Brain and Behavioural Sciences December 1997.

[Abdi98] Abdi, H., Valentin, D., Edelman, B., *Eigenfeatures as intermediate level representations: the case for PCA models*. Brain and Behavioral Sciences, 21, 17-18. (1998)

[Abutaleb89] A.S. Abutaleb, *Automatic Thresholding of gray-level pictures using two-dimensional entropy,* Computer Vision Graphics Image Process 47 (1989) 22-32

[Ayache89] Nicholas Ayache, *Vision Stéréoscopique et perception multisensorielle: Application à la robotique mobile,* Inter Edition Science Informatique 1989.

[Ayari96] I. Ayari, *Fusion multi-capteurs dans un cadre multi-agents: application à un robot mobile,* Thèse de doctorat de L'Université Henri Poincaré (NANCY I) janvier 1996.

[Baerveldt01] A-J. Baerveldt, *A vision system for object verification and localization based on local features*, Robotics and Autonomous Systems 34 (2001) 83-92.

[Balch00] T. Balch, *Hierarchic Social Entropy: An Information Theoretic Measure of Robot Group Diversity*, Autonomous Robots, vol. 8, no. 3, July 2000

[Balch97a] T. Balch, *Learning roles: Behavioral diversity in robot teams*, College of Computing Technical Report GIT-CC-97-12, Georgia Institute of Technology, Atlanta, Georgia, March 1997

[Balch97b] T. Balch, *Social entropy: a new metric for learning multi-robot teams*, In Proc. 10th International FLAIRS Conference (FLAIRS-97).

[Baujard92] O. Baujard, *Conception d'un Environnement de Développement pour la Résolution de Problèmes : Apport de L'Intelligence Artificielle Distribuée et Application à la Vision*, Thèse de doctorat de l'Université Joseph Fourier (GRENOBLE I) octobre 1992.

[Bellot02] Bellot, David. *Fusion de données avec des réseaux bayésiens pour la modélisation des systèmes dynamiques et son application en télémédecine*. Thèse de l'université H. Poincaré NANCY I 2002. 170 p

[Bererton00] C. Bererton, L.E. Navarro-Serment, R. Grabowski, C. J.J. Paredis & P. K. Khosla, *Millibots: Small Distributed Robots for Surveillance and Mapping.* Accepted to be presented at the Government Microcircuit Applications Conference, to be held 20-23 March 2000.

[Birattari02] Birattari M., Di Caro G., Dorigo M., *Toward the formal foundation of Ant Programming*, Ant Algorithms - Proceedings of ANTS 2002, Third International Workshop on Ant Algorithms, Brussels, Belgium, September 12-14, 2002, Lecture Notes in Computer Science Vol. 2463, Springer-Verlag, 2002.

[Bonabeau94] Bonabeau E., Theraulaz G., *'Intelligence collective*, Hermès Paris 1994.

[Bonabeau99] Bonabeau E., Dorigo M., Theraulaz G., *Swarm Intelligence : From Natural to Artificial Systems*, Oxford University Press 1999.

[Bourjot02] Christine Bourjot, Vincent Chevrier, Vincent Thomas: How social spiders inspired an approach To region detection. AAMAS 2002: 426-433

[Bourjot03] Bourjot, Christine and Chevrier, Vincent and Thomas, Vincent. *A new swarm mechanism based on social spiders colonies: from web weaving to region detection*. Web Intelligence and Agent Systems: An International Journal - WIAS. 2003. vol 1. n° 1. pp.47-64

[Bouzid00a] Bouzid Makram, Charpillet François, Chevrier Vincent et Vialle Stéphane. *Parallélisation d'un simulateur d'agents situés.* Journées scientifiques du CCH, janvier 2000.Nancy, France.

[Bouzid00b] Bouzid Makram, Chevrier Vincent, Vialle Stéphane et Charpillet François, *A stochastic model of interaction for situated agents and its parallel implementation*. ACIDCA, 2000. Monastir, Tunisie.

[Bouzid01] Makram Bouzid, *Contribution à la modélisation de l'interaction agent/environnement : Modélisation stochastique et simulation parallèle*, doctorat de l'Université Henri Poincaré NANCY I 2001

[Braitenberg84] V. Braitenberg, *Vehicles, Experiments in synthetic Psychology*, The MIT Press 1984.

[Burgard97] W. Burgard, D. Fox, S. Thrun, *Active Markov Localization*, 15th international Joint Conference on Artificiel Intelligence (IJCAI 97)

[Carré99] J.Carré, A.Machonin, and P.Glize, *Un système multi-agent auto-organisateur pour l'apprentissage d'un profil utilisateur*. In Septièmes Journées Francophones pour l'Intelligence Artificielle Distribuée et les Systèmes Multi-agents, La Réunion, November 10 1999.

[Cheng98] H D Cheng and Y H Chen, *Thresholding Based on Fuzzy Partition of Two-Dimensional Histogram*, 14th International Conference on Pattern Recognition 16 - 20 August, 1998 Brisbane, Australia.

[Cheng99] H D Cheng and Y H Chen, *Fuzzy partition of two-dimensional histogram and its application to thresholding*, Pattern Recognition 825-843 1999

[Chong98] Chee-Yee Chong, *Distributed Architectures for Data Fusion*, Fusion '98 International Conference p 84-91.

[Colorni92] Alberto Colorni, Marco Dorigo, Vittorio Maniezzo: *An Investigation of some Properties of an ``Ant Algorithm''*. PPSN 1992:

[Cootes95] Tim F. Cootes, Christopher J. Taylor, David H. Cooper, and Jim Graham. *Active shape models - their training and application.* Computer Vision and Image Understanding, 61(1):38--59, January 1995.

[Crowley98] J.L. Crowley, F. Wallner, and B. Schiele, *Position estimation using principal components of range data*, Proceedings of the IEEE International Conference on Robotics and Automation (Leuven, Belgium), May 1998, pp. 3121--3128.

[DeCarlo00] DeCarlo D., Metaxas D., *Optical flow Constarints on Deformable Models with Applications to Face Tracking*, International Journal of Computer Vision, 38(2), pp 99-127, July 2000

[Dellaert99] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, *Monte Carlo Localization for Mobile Robots*, IEEE International Conference on Robotics and Automation (ICRA99), May, 1999.

[Demazeau95] Demazeau, Y., *From Interactions to Collective Behaviour in Agent-Based Systems,* European Conference on Cognitive Science, Saint-Malo, April 1995

[Demirli00] K. Demirli and I.B. Türksen, *Sonar based mobile robot localization by using fuzzy triangulation*, Robotics and Autonomous Systems, 33 (2-3) (2000) pp. 109-123

[Dev97] A. Dev, B.J.A Kröse, F.C.A. Groen, *Where are you driving to?" Heading Direction for a Mobile Robot from Optical Flow*, Proceedings of the IEEE International Conference on Robotics and Automation 1997

[Dorigo96] Marco Dorigo, Vittorio Maniezzo, Alberto Colorni, *The Ant System: Optimization by a colony of cooperating agents,* IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics (1996)

[Dorigo97] Dorigo M. & L.M. Gambardella (1997). *Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem.* IEEE Transactions on Evolutionary Computation, 1, 1, 53-66. (Also Tecnical Report TR/IRIDIA/1996-5, IRIDIA, Université Libre de Bruxelles.)

[Dorigo99] Marco Dorigo, Gianni Di Caro, Luca M. Gambardella, *Ant Algorithms for Discrete Optimization*, Proceedings of the Congress on Evolutionary Computation (1999)

[Drogoul93] Drogoul A. *De la simulation multi-agents à la résolution collective de problèmes* PhD Thesis, University Paris 6, November 1993.

[Duckett01] T. Duckett, U. Nehmzow, *Mobile robot self-organization using occupancy histograms and a mixture of Gaussian location hypotheses*, Robotics and Autonomous Systems, 34 (2001) pp. 117-129.

[Duckett98] T. Duckett, U. Nehmzow, *Mobile robot self-localisation and measurement of performance in middle scale environments,* Robotics and Autonomous Systems 24 (1998) 57-69.

[Durand99] S. Durand, *Représentation de points de vue multiples dans une situation d'urgence : une modélisation par des organisations d'agents*, Thèse de doctorat de l'Université du Havre 1999

[Faugeras92] Faugeras O. et Zhang Z., *3D Dynamic Scene Analysis*, Springer Verlag 1992.

[Ferber95] J. Ferber, *Les systèmes multi-agents*, , InterEditions, 1995

[Ferber99] Jacques Ferber *Multi-Agent System: An Introduction to Distributed Artificial Intelligence Harlow*: Addison Wesley Longman 1999

[Fox98] D. Fox, W. Burgard, S. Thrun, *Active Markov localization for mobile robots*, Robotics and Autonomous Systems 25 (2001) pp 195-207

[Fox99a] Dieter Fox, Wolfram Burgard, Hannes Kruppa, Sebastian Thrun *A Monte Carlo Algorithm for Multi-Robot Localization* AAAI/IAAI

[Fox99b] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. *Monte carlo localization: Efficient position estimation for mobile robots*. In Proc. of AAAI-99, 1999

[Gazendam98] Gazendam H.W.M., Jorna R.J., *Theories about architecture and performance of multi-agent systems* University of Groningen, Research Institute SOM (Systems, Organisations and Management) Research Report 1998.

[Gechter99] Gechter Franck, *Construction d'un module de stéréovision utilisant des arêtes verticales pour un robot mobile*, Rapport de DEA Université Louis Pasteur Strasbourg.

[Gechter00] Gechter, Franck and Charpillet, François. *Vision Based Localisation for a Mobile Robot*. In International Conference on Tools with Artificial Intelligence. 2000. 8 p.

[Gechter01a] Gechter, Franck and Thomas, Vincent and Charpillet, François. *Robot Localization by Stochastic Vision Based Device*. In The 5th World Multi-Conference on Systemics, Cybernetics and Informatics - SCI 2001 – The 7th International Conference on Information Systems Analysis and Synthesis - ISAS 2001. (Orlando, FL, USA). 2001. 3 p.

[Gechter01b] Gechter, Franck and Charpillet, François. *Markov Based Localization Device for a Mobile Robot*. In The 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space: A New Space Odyssey - i-SAIRAS 2001 . (Montréal, Canada). 2001. 7 p.

[Gechter01c] Gechter, Franck et Thomas, Vincent et Charpillet, François. *Utilisation d'un module de vision stochastique pour localiser un robot mobile*. In Troisièmes Journées Nationales de la Recherche en Robotique - JNRR'2001. (Hyères, France). 2001. 7 p.

[Gechter02] GECHTER Franck – *Une méthode de localisation d'un robot mobile utilisant une approche multi-agents pour la fusion de données* – 15eme Journées des Jeunes Chercheurs en Robotique JJCR2002 – Strasbourg, 31 janvier,1er février 2002

[Giai95] Giai-Checa Bernard, *Analyse du mouvement et de la structure au sein de sequences d'images monoculaires*, Thèse de l'Université de Nice-Sophia Antipolis, Ecole doctorale SPI mars 1995.

[Goss90] Goss S. BEckers R. Deneubourg J.L. Aron S. Pasteels J.M., *Trail Laying and Trail Following Can Solve Foraging Problems for Ant Colonies*, in Hughes R.N. (ed.) NATO ASI Series, Vol. G 20, Behavioural Mechanisms of Food Selection, Springer Verlag Berlin 1990

[Grabowski99] Grabowski, R., Navarro-Serment, L. E., Paredis, C.J.J., and Khosla, P. *Heterogeneous Teams of Modular Robots for Mapping and Exploration*, in Autonomous Robots (special issue on heterogeneous multirobot systems)1999.

[Hartley97] R.I. Hartley, P.Sturm, *Triangulation, Computer Vision and image understanding* vol.68, N°2, November 1997, pp. 146-157.

[Henkel00] R.D. Henkel, *"Synchronization, Coherence-Detection and Three-Dimensional Vision"*, Technical Report, Institute of Theoretical Neurophysics, University of Bremen 2000.

[Horaud93a] Radu Horaud et Olivier Monga, *Vision par Ordinateur : Outils Fondamentaux,* Traité Des Nouvelles Technologies. Editions Hermès 1993

[Horaud93b] Radu Horaud et Olivier Monga, *Vision Stéréoscopique*, chapitre 6, pages 187-228 dans [Horaud93a]

[Iyengar98] G. Iyengar and A. Lippman, *Clustering images using relative entropy for efficient retrieval*, In Workshop on Very Low bitrate Video Coding. University of Illinois-Urbana Champaign, Oct 1998.

[Jennings98] N. R. Jennings, T. J. Norman, and P. Faratin *"ADEPT: An Agent-based Approach to Business Process Management"* ACM SIGMOD Record 27 (4) 32-39. (1998)

[Jennings98a] N.R.Jennings, S. Parsons, P. Noriega and C. Sierra *"On Argumentation-Based Negotiation"* Proc. Int. Workshop on Multi-Agent Systems, Boston, USA. (1998)

[Jennings98b] N. R. Jennings, K. Sycara and M. Wooldridge *"A Roadmap of Agent Research and Development"* Int Journal of Autonomous Agents and Multi-Agent Systems 1 (1) 7-38. (1998)

[Kanada94] Y. Kanada, and M. Hirokawa: *Stochastic Problem Solving by Local Computation based on Self-organization Paradigm*, IEEE 27th Hawaii International Conference on System Sciences, pp. 82-91, 1994.

[Kennedy01] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Publisher 2001 ISBN 1-55860-595-9

[Koller93] D. Koller, J. Weber, and J. Malik, *Robust multiple car tracking with occlusion reasonning*, Technical Report UCB, 1993.

[Koller94] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, S. Russell, *Towards Robust Automatic Traffic Scene Analysis in Real-Time,* Proceedings of the International Conference on Pattern Recognition 1994.

[Laroche00] Laroche, Pierre. *Processus Décisionnels de Markov appliqués à la planification sous incertitude*. Thèse d'université. Nancy, 2000.

[Laroche99] Laroche, Pierre and Charpillet, François and Schott, René. *Mobile Robotics Planning using Abstract Markov Decision Processe*, International Conference on Tools with Artificial Intelligence , ICTAI. (Chicago, Illinois). 1999. pp.299-306

[Laubach99] Sharon L. Laubach, Clark F. Olson, Joel W. Burdick, and Samad Hayati *Long Range Navigation for Mars Rovers Using Sensor-Based Path Planning and Visual Localisation,* In Proceedings of the 5th International Symposium on Artificial Intelligence, Robotics and Automation in Space, 1999.

[Léon] http:// perso.club-internet.fr / jac_leon/ PhyContemporaine/ relagen/ relagen10.htm

[LeBer98] Le Ber, Florence and Chevrier, Vincent and Dury, Arnaud. *A Multi-Agent System for the Simulation of Land Use Organization.* In 3rd IFAC/CIGR Workshop on Artificial Intelligence in Agriculture. (Makuhari, Chiba, Japon). 1998. 6 p

[Lopez98] J.A. Lopez-Orozco, J.M. de la Cruz, J. Sanz, J. Flores, *Multisensor fusion of environment measures using Bayesian Networks*, Fusion'98 International Conference p 487-493.

[M.R.Jean97] M.R Jean , S Pesty, *Emergence et SMA*, V Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents, AFCET, AFIA, La Colle-sur-Loup, Quinqueton, Thomas, Trousse (eds), p. 323-342. 1997

[Madkit] http://www.madkit.org/

[Marchand01] E. Marchand, P. Bouthemy, F. Chaumette, *A 2D-3D model-based approach to real-time visual tracking,* Image and Vision Computing, 19(13):941-955, Novembre 2001.

[Marengoni99] Mauricio MARENGONI, Allen HANSON, Schlomo ZILBERSTEIN, and Edward RIS EMAN, *Control in a 3D Reconstruction System using Selective Perception*, Proceedings of the 7th IEEE International Conference on Computer Vision, Kerkyra, Greece, 1999

[Mees00] W. Mees, *Contribution à l'analyse distribuée de scènes : Application aux images satellitaires multi-spectrales haute résolution*, Thèse de doctorat de L'Université Henri Poincaré (NANCY I) janvier 2000.

[Meuleau02] N. Meuleau and M. Dorigo. *Ant colony optimization and stochastic gradient descent*. Artificial Life, 8(2):103-121, 2002. Also available as technical report TR/IRIDIA/2000-36, IRIDIA, Université Libre de Bruxelles.

[Minar99] Nelson Minar, Kwindla Hultman Kramer, Pattie Maes, *Cooperating Mobile Agents for Mapping Networks* Proceedings of the First Hungarian National Conference on Agent Based Computation (1999)

[Moghaddam96] Baback Moghaddam, Alex Pentland, *Probabilistic Visual Learning for Object Representation*, IEEE Transactions on Pattern Analysis and Machine Intelligence 1996.

[Morris99] Morris, D T; Donnison, C, *Identifying the neuroretinal rim boundary using dynamic contours*, Image and Vision Computing, vol 17 (3-4), pp 169-174, ISSN 0262-8856 1999.

[Mück00] Klaus Mück, Hans-Hellmut Nagel, Markus Middendorf, *Data-Driven Extraction of Curved Intersection Lanemarks from Road Traffic Image Sequences.* ECCV (2) 2000: 411-427

[Müller98] J.-P.Müller, *Méthodologie de conception de systèmes multi-agents de résolution de problèmes par émergence*, dans JFIADSMA'98, Hermès, novembre 1998

[Murray96a] D. Murray, J. Little, *Stereo vision based mapping and navigation for mobile robots*, Computer Science Dept University of British Columbia Vancouver 1996.

[Murray96b] D. Murray, J. Little, *Using real-time stereo vision for mobile robot navigation*, Computer Science Dept University of British Columbia Vancouver 1996.

[Nagel98] Hans-Hellmut Nagel, A. Gehrke, *Bildbereichsbasierte Verfolgung von Straßenfahrzeugen durch adaptive Schätzung und Segmentierung von Optischen-Fluß-Feldern.* DAGM-Symposium 1998: 314-321

[Olson97a] C. F. Olson, *Mobile Robot Self-Localization by Iconic Matching of Range Maps*, In Proceedings of the 8th International Conference on Advanced Robotics, pp 447-452, 1997.

[Olson97b] C. F. Olson, *Visual Localization Methods for Mars Rovers using Lander, Rover and Descent Imagery*, In Proceedings of the 4th International Symposium on Artificial Intelligence, Robotics, and Automation in Space, pages 413-418, 1997.

[Olson99] C. F. Olson, *Subpixel Localization and Uncertainty Estimation Using Occupancy Grids,* Proceedings of the IEEE International Conference on Robotics and Automation Detroit May 1999.

[Parunak00] Van Dyke Parunak H. , Brueckner S. , *Ant-Like Missionaries and Cannibals: Synthetic Pheromones for Distributed Motion Control*, Under review for Agents 2000.

[Parunak01] Van Dyke Parunak H., Brueckner S., *Entropy and Self-Organization in Multi-Agent Systems*, International Conference on Autonomous Agents 2001.

[Parunak97] H. V. D. Parunak. *'Go to the Ant': Engineering Principles from Natural Agent Systems*. Annals of Operations Research, 75:69-101, 1997.

[Peixoto00] P. Peixoto, J. Batista, H. J. Araujo, *Integration of information from several vision systems for a common task of surveillance*, Robotics and Autonomous Systems 31 (2000) 99-108

[Peixoto01] P. Peixoto, J. Batista, H. J. Araujo, *Real-time human activity monitoring exploring multiple vision sensors*, Robotics and Autonomous Systems 35 (2001) 221-228

[Picard98] R.W. Picard, *Toward Agents that Recognize Emotion*, Actes Proceedings IMAGINA, March 1998, pp 153-165, Monaco.

[Pourraz99] F. Pourraz and J. L. Crowley, *Continuity Properties of the Appearance Manifold for Mobile Robot Position Estimation*, Workshop on Perception for Mobile Agents, CVPRí99 June 1999.

[Pratt78] William K. Pratt, *Digital Image Processing*, Wiley Introscience publication 1978

[Press92] Press W.H., Teukolski S. A., Vetterling W.T and Flannery B.D., *Numerical Recipies in C* : the art of scientific Computing, Cambridge University Press, second Edition 1992.

[Ramos00] V. Ramos, F. Almeida, *Artificial Ant Colonies in Digital Image Habitats* - A Mass Behaviour Effect Study on Pattern Recognition, ANTS'2000 Brussels Belgique septembre 2000

[Reynolds87] Reynolds, C. W. (1987) *Flocks, Herds, and Schools: A Distributed Behavioral Model, in Computer Graphics*, 21(4) (SIGGRAPH '87 Conference Proceedings) pages 25-34.

[Rizzi01] A. Rizzi and R. Cassinis, *A robot self-localization system based on omnidirectional color images*, Robotics and Autonomous Systems, 34 (1) (2001) pp. 23-38

[Rogova98] G. L. Rogova, R. Menon, *Decision Fusion for Learning in Pattern Recognition*, Fusion '98 International Conference p 191-198.

[Roumeliotis99] Roumeliotis S.I., Sukhatme G.S., and Bekey G.A, *Circumventing Dynamic Modeling: Evaluation of the Error-State Kalman Filter applied to Mobile Robot Localization,* IEEE International Conference on Robotics and Automation , Detroit, May 10-15 1999

[Sauter01] John Sauter, H.D.V. Parunak, S. Brueckner,R. Matthews, *Tuning Synthetic Pheromones With Evolutionary Computing*, Evolutionary COmputation and Multi-Agent Systems (ECOMAS) (2001)

[Schulz01] D. Schulz, W. Burgard, *Probalistic state estimation of dynamic objects with a moving mobile robot*, Robotics and Autonomous Systems 34 (2001) 107-115

[Sellem00] Sellem P., Luzeaux D., *Répartition de la perception dans un système distribué de robots autonomes,* JFIADSMA 2000

[Shannon49] C.E. Shannon, *the Mathematical Theory of Communication*, University of Illinois Press 1949.

[Sim98a] R. Sim, *Mobile robot localization from learned landmarks,* Master's Thesis, McGill University. July, 1998.

[Sim98b] R. Sim and G. Dudek, *Mobile robot localization from learned landmarks,* Proceedings of IEEE/RSJ Conference on Intelligent Robots and Systems (IROS), Victoria, BC, Oct 1998

[Sim99a] R. Sim and G. Dudek, *Learning Environmental Features for Pose Estimation,* CVPR Workshop on Perception for Mobile Agents, Ft. Collins, CO, June 1999.

[Sim99b] R. Sim and G. Dudek, *Learning and Evaluating Visual Features for Pose Estimation*, Proceedings of International Conference on Computer Vision, Kerkyra, Greece, September 1999.

[Sim99c] R. Sim and G. Dudek, *Learning Visual Landmarks for Pose Estimation,* Proceedings of International Conference on Robotics and Automation (ICRA), Detroit, MI, May 1999.

[Sim99d] R. Sim and G. Dudek, *Visual Landmarks for Pose Estimation*, Canadian Artificial Intelligence, No. 43, Spring 1999 pp13-17

[Simonin01] O. Simonin, *Le modèle satisfaction-altruisme : coopération et résoltuion de conflits entre agents situés réactifs, application à la robotique*, Thèse (Montpellier) 2001.

[Simmons95] R. SIMMONS and S. KOENIG, *Probabilistic Robot Navigation in Partially Observable Environments*, International Joint Conference on Artificial Intelligence, Montreal Canada, August 1995.

[Srinivasan99] M.V. Srinivasan and al., *Robot navigation inspired by principles of insect vision*, Robotics and Autonomous Systems 26 (1999) 203-216.

[Starck00] J. Starck and F. Murtagh and F. Bonnarel, *Multiscale entropy for semantic description of images and signals,* J.L. Starck, F. Murtagh and F. Bonnarel, Multiscale entropy for semantic description of images and signals, 2000, submitted.

[Starlogo] http://education.mit.edu/starlogo/

[Sycara98a] K. Sycara *Levels of Adaptivity in Systems of Coordinating Information Agents Cooperative Information Agents* II, M. Klusch, G. Weiss, ed., Springer Verlag, 1998.

[Sycara98b] K. Sycara *Multiagent Systems* AI Magazine, Vol. 10, No. 2, 1998, pp. 79-93.

[Tarel95] Jean-Philippe Tarel et André Gagalowicz, *Calibration de camera à base d'ellipses*, INRIA Domaine de Voluceau, Rocquencourt 1995.

[Terzopoulos94] D. Terzopoulos, X. Tu, R. Grzeszczuk, *Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world,"*, Artificial Life, 1, 4, December, 1994, 327-351.

[Terzopoulos99] D. Terzopoulos, *Artificial life for computer graphics*, Communications of the ACM, 42(8), August, 1999, 32-42.

[Theraulaz97] G. Theraulaz, F. Spitz, *Auto-organisation et comportement*, Hermès Paris 1997.

[Thomas01] V. Thomas. *Evaluation de systèmes multi-agents d'inspiration biologique. Application à l'extraction de régions dans une image*. DEA UHP Nancy I. 2001.

[Thrun99] Sebastian Thrun *Monte Carlo Hidden Markov Models: Learning Non-Parametric Models of Partially Observable Stochastic Processes* (99) , Proc. 16th International Conf. on Machine Learning

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

[Touretzky96] D. Touretzky, M. Mozer and M. Hasselmo, *Advances in Neural Information Processing Systems 8*, MIT Press, Cambridge, MA, 1996.

[Tsai87] Tsai R. Y., *An efficient and accurate camera calibration technique for 3D Machine vision*, IEEE J. of Robotics and Automation, RA-3 :364-374 1987

[Verlinde98] P. Verlinde, G. Maître, E. Mayoraz, *Decision fusion using a multi-linear classifier*, Fusion '98 International Conference p 47-53.

[Wallner97] Frank Wallner, Bernt Schiele and James L. Crowley, *Position Estimation for a Mobile Robot From Principal Components of Laser Range Data*, In 5th International Symposium on Intelligent Robotic Systems'97, Stockholm, Sweden, July 1997. Also to appear in Robotics and Autonomous Systems, 1998.

[Waltz90] E. Waltz, J. Llinas, *Multisensor data fusion,* Artech House 1990.

[Weis99] G. Weis, *Multiagent Systems : Amodern approach to Distributed Artificial Intelligence*, The MIT Press 1999.

[Wooldridge95] Wooldridge, M., and Jennings, N. R. *Intelligent Agents: Theory and Practice*. The Knowledge Engineering Review 10(2): 115-152. 1995.

[Worrall92] Worrall A. D., Sullivan G.D. and Baker K.D., *A simple, intuitive camera calibration tool for natural images*, the University of Reading Bershire, RG62AY, UK

[Yang98] Ming-Hsuan Yang, Narendra Ahuja, *Detecting Human Faces in Color Images*, In Proceedings of the 1998 IEEE International Conference on Image Processing (ICIP 98), pp. 127-130, vol. 1, Chicago, October, 1998.

[Zeghal94] Zeghal K., Ferber J., *A reactive approach for distributed air traffic control*, In proc. Of Avignon'94 pp381-390

[Zeigler93] Zeigler, Bernard P.. *A Systems Methodology for Structuring Families of Models at Multiple Levels of Resolution*, Variable Resolution Modeling Conference, Tucson, Arizona, CF-103-DARPA, RAND, Santa Monica, Californie 1993.

[Zilberstein00] S. Zilberstein and A.I. Mouaddib, *Optimizing Resource Utilization in Planetary Rovers*, Proceedings of the 2nd NASA International Workshop on Planning and Scheduling for Space, pp. 163-168, March 2000.

[Zilberstein01] S. Zilberstein, F. Charpillet, and P. Chassaing, *Optimal Sequencing of Contract Algorithms*, To Appear in Annals of Mathematics and Artificial Intelligence, 2001.

[Zilberstein99] Shlomo Zilberstein et Abdel-Illah Mouadib, *Reactive control for dynamic progressive processing*, IJCAI 99.

[Zingaretti98] P. Zingaretti, A. Carbonaro, *Route following based on adaptative visual landmark matching*, Robotics and Autonomous Systems 25 (1998) 177-184.

Chapitre 10 : Annexes

10.1. Construire une représentation du monde : Utilisation des capteurs

10.1.1. But du chapitre

Avant de réaliser une application robotique, et en particulier une application de localisation, il faut impérativement savoir exploiter correctement les percepts fournis. Cette exploitation passe par la connaissance des principes de fonctionnement, et surtout des limitations, des capteurs utilisables sur une plate-forme mobile.

Il existe dans la littérature de nombreux ouvrages expliquant le principe de fonctionnement de l'un ou l'autre des capteurs les plus courants. Cependant, je n'ai jamais pu trouver d'ouvrage effectuant une synthèse des éléments caractéristiques des capteurs les plus couramment utilisés en robotique mobile. C'est pourquoi, j'ai décidé de consacrer un chapitre de mon manuscrit à une telle synthèse.

Ce chapitre a donc pour but de passer en revue les capteurs les plus couramment employés dans une application robotique. Mon but n'est pas d'être exhaustif, aussi j'ai choisi de ne pas traiter le cas des capteurs spécifiques à une application tel que les capteurs de CO_2. La description des capteurs effectuée dans ce chapitre est réalisée en se plaçant du point de vue de l'utilisateur potentiel sans rentrer dans les détails techniques liés aux choix technologiques des constructeurs.

10.1.2. Les différents types de capteurs

Il existe de nombreux types de capteurs utilisables dans le cadre d'une application de localisation. Ces capteurs peuvent être généraux, c'est à dire capable de fournir des informations riches et diversifiées sur l'environnement, c'est le cas des caméras notamment, ou spécifiques attachés à la détection d'un évènement particulier, comme un détecteur de CO_2 par exemple. Dans le cas de l'auto-localisation, nous pouvons noter deux types de capteurs bien différenciés : les capteurs proprioceptifs et les capteurs extéroceptifs. Les capteurs proprioceptifs fournissent des informations sur l'état interne du robot (nombre de tours de roue, niveau de charge de la batterie,...) alors que les capteurs extéroceptifs fournissent des informations sur l'environnement entourant le robot.

Nous allons détailler le principe de fonctionnement des capteurs les plus utiles dans le cadre de la localisation et de l'auto-localisation. Ainsi, nous ne détaillerons pas certains capteurs élémentaires tel que les capteurs de charge des batteries car ils n'apportent pas de données pertinentes pour notre application. Le but de la suite du document est d'effectuer un bilan de l'influence sur la localisation de chaque capteur pertinent.

10.1.2.1. L'odomètre

L'odomètre est l'un des capteurs proprioceptifs les plus courants sur les robots mobiles propulsés par des ensembles roue/moteur. Le principe de ce capteur est de prendre en compte les rotations élémentaires au niveau des roues pour estimer le déplacement du robot. Ce type de capteurs requiert un étalonnage préalable afin de prendre en compte les caractéristiques mécaniques de l'ensemble roue/moteur. En outre, ce capteur est très imprécis car il ne prend pas en compte les phénomènes de glissement des roues sur le sol et les éventuels patinages lors de la rotation des moteurs. Etant donné que la majorité de ces dispositifs procède de façon incrémentale, l'odomètre devient de moins en moins précis au cours des déplacements du robot.

L'odomètre est surtout indispensable lorsque les dispositifs de propulsion utilisés pour déplacer le robot sont des moteurs à courant continu. En effet, contrairement aux moteurs pas à pas, ceux-ci ne sont pas commandés en nombre de tours de roue mais sous la forme d'une tension variable appliquée à ses bornes. Cette commande s'effectue la plupart du temps en modulation de pulsation PWM (Pulse Width Modulation). Le principe de cette méthode est de modifier le rapport cyclique entre la taille du front haut et la taille du front bas pour contrôler la vitesse de rotation du moteur (cf. figure 71). Etant donné l'imprécision de cette loi de commande, il est donc nécessaire de disposer d'un capteur permettant de contrôler la rotation des moteurs.

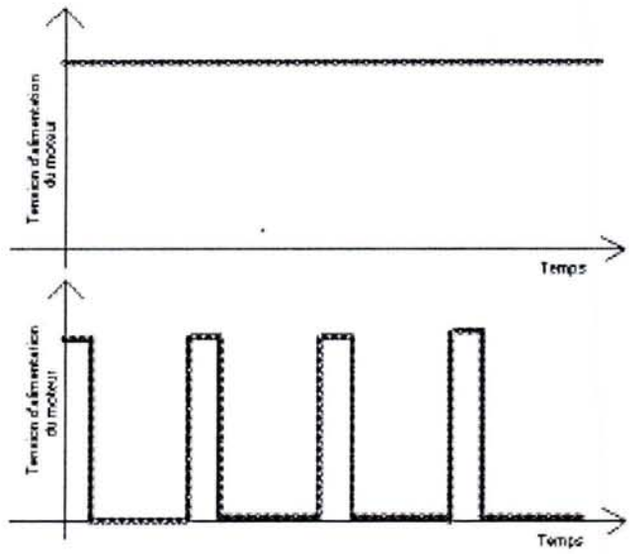

figure 71 : Exemples de commande de moteur par PWM, moteur à 100% (en haut) et moteur à 30% (en bas)

Il existe plusieurs méthodes pour construire un odomètre. Dans un premier temps, il est possible de réaliser un odomètre par méthode potentiométrique. Dans ce cas, une résistance variable est associée à la rotation de la roue. La mesure de la résistance fournit une estimation de la position de la roue, et le nombre d'alternances entre résistance faible et résistance forte permet de compter le nombre de tours de roue.

Une autre méthode couramment utilisée est l'adjonction d'un disque ajouré à chacune des roues. Ce disque est alors placé entre un émetteur lumineux de type diode infra-rouge et un récepteur photo-sensible de type photodiode ou phototransistor. Connaissant le nombre d'orifices dans le disque, il est assez simple de déduire le nombre de tours de roue en fonction du nombre d'alternances. Le principal problème de cette méthode est la difficulté de mettre en place le dispositif et surtout de la calibrer de façon précise.

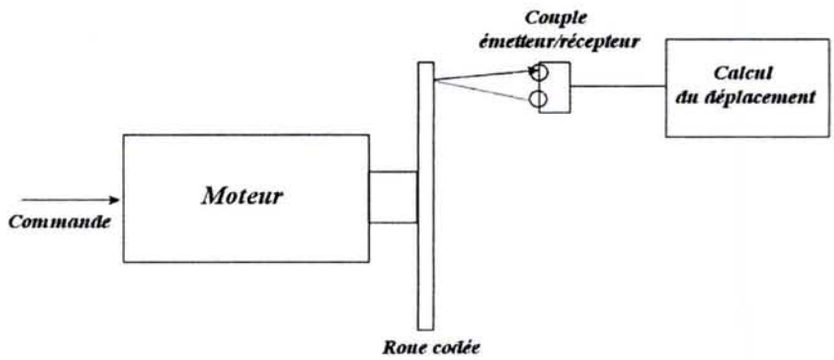

figure 72 : Principe de fonctionnement de l'odomètre à disque

Enfin, la dernière méthode, dérivée de la précédente, consiste à rassembler le couple émetteur/récepteur du même coté de la roue. L'alternance de luminosité s'effectue en liant à la roue un disque composé de secteurs blancs et de secteurs noirs. (cf. figure 72 et figure 73) Cette méthode se montre efficace et sa précision dépend du nombre de secteurs angulaires.

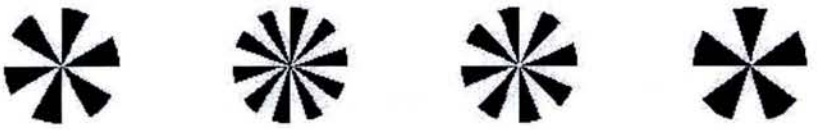

figure 73 : Quelques exemples de roues d'odomètre

Les autres capteurs proprioceptifs ne concernent pas directement la localisation puisque leur finalité est essentiellement destinée à fournir des renseignements sur l'état interne du robot (charge des batteries, états des capteurs disponibles, température de fonctionnement, ...). Par contre, les capteurs extéroceptifs sont nombreux à pouvoir intervenir dans une tâche de localisation.

10.1.2.2. Les capteurs de proximité ou de contact

Les capteurs de proximité ou de contact sont les plus courants sur les robots mobiles car ils sont très simples à mettre en œuvre et permettent d'élaborer des réflexes de bas niveau. Parmi ces capteurs de proximité les plus courants sont les capteurs de contact construits à partir de micro-interrupteurs. Etant donné que ces interrupteurs ne sont pas suffisamment robustes pour absorber eux-mêmes les chocs, ils sont en général inscrits dans des dispositifs mécaniques adaptés

Une autre possibilité est de simuler les capteurs de contact en utilisant des barrières lumineuses construites à partir de couples émetteur/récepteur infra-rouges. Ces dispositifs peuvent être considérés comme étant des capteurs logiques similaires à des interrupteurs mais réagissant à partir de distances faibles mais non nulles.

Ces capteurs sont très utiles en robotiques mobiles car ils permettent de construire une barrière proche du robot évitant ainsi les chocs en cas de mauvais fonctionnement ou de mauvaises manipulations. Ces dispositifs d'acquisition ne sont cependant pas très efficaces pour évaluer la position du robot dans un environnement car les informations

fournies ne sont pas assez riches. Néanmoins, ils peuvent fournir un complément d'information indispensable pour résoudre certaines localisations ambiguës.

Les capteurs de contact sont surtout utilisés dans la mise en œuvre de comportements réactifs, tels que des évitements de collision. Par contre, ceux-ci se révèlent nettement moins utiles pour des comportements de haut niveau. Dans ce cas, l'utilisation de capteurs de mesure de distance peut être digne d'intérêt.

10.1.2.3. Les capteurs de distance

Les capteurs dédiés à la mesure d'une distance entre l'objet sur lequel ils se trouvent et les obstacles à proximité sont en général construits à partir d'un couple émetteur/récepteur qui peut soit utiliser une onde lumineuse soit une onde sonore. Le principe général d'un tel dispositif d'acquisition est de mesurer la ou les différences entre les caractéristiques de l'onde émise et celle reçue après réflexion sur les obstacles (cf. figure 74). Ces mesures différentielles peuvent être des mesures de décalage temporel ou des mesures d'amplitude du signal reçu en fonction de celle du signal émis.

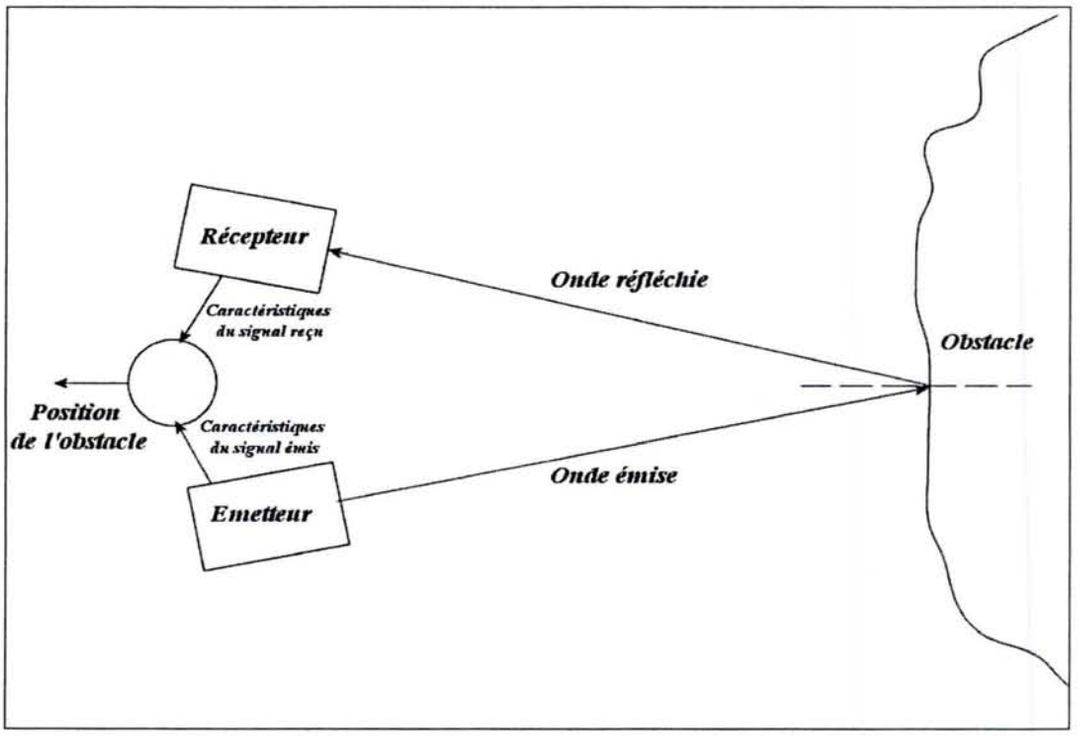

figure 74 : Principe de fonctionnement d'un capteur de distance

Les capteurs de distance sont très utiles en robotique car ils permettent de maintenir le robot à l'écart des obstacles les plus dangereux pour son déplacement. Pour la localisation, ils sont très précieux car ils permettent une estimation pertinente de la position relative des robots par rapport aux obstacles. Néanmoins, ils possèdent des inconvénients essentiellement dus à leur principe de fonctionnement : en effet, ce type de capteur fonctionne parfaitement si les obstacles sont perpendiculaires (ou avec un angle d'incidence proche de 90 degrés) au vecteur de propagation de l'onde émise. Dans ce cas, la réflexion s'effectue de façon idéale et le récepteur reçoit la quasi totalité de l'onde réfléchie modulo les pertes dues à la propagation dans l'air.

Par contre, lorsque l'angle d'incidence est trop faible l'onde est déviée et ne peut pas être récupérée par le récepteur. Ce phénomène est appelé la spécularité (cf. figure 75).

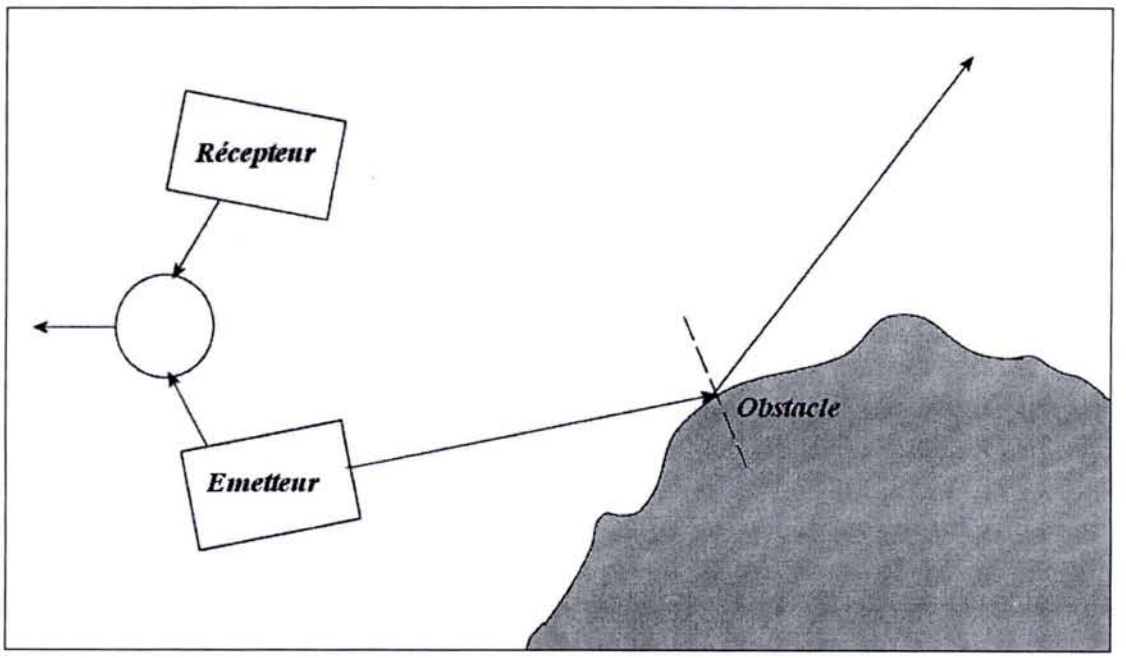

figure 75 : La spécularité

Dans le cas de la figure 75, il n'y a pas de mesure possible car le récepteur ne reçoit aucun signal provenant de l'émetteur. Néanmoins, il existe également des cas où la mesure est possible mais erronée. C'est ce qui arrive lorsqu'il y a un phénomène de réflexions multiples dues à une configuration spatiale spécifique entre les obstacles et le capteur. (cf. figure 76).

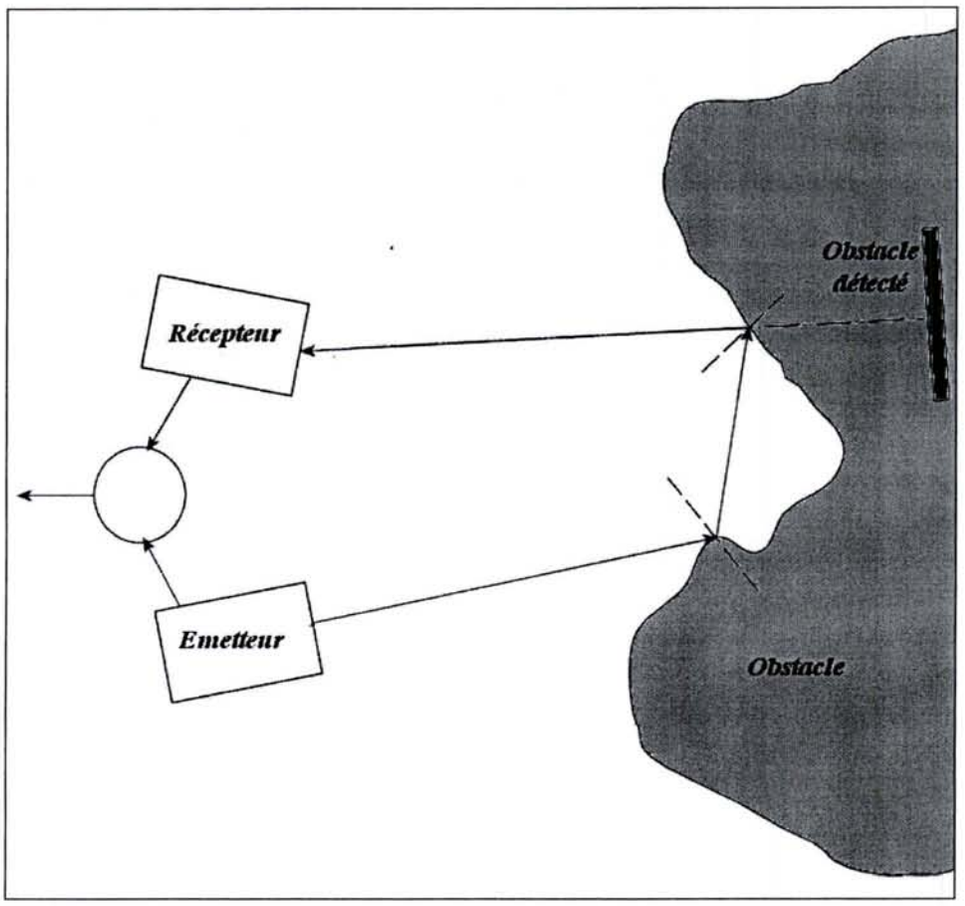

figure 76 : Phénomène de réflexions multiples

Enfin, étant donné que l'onde émise est spatialement répartie, dans le sens où le front de l'onde ne peut pas être assimilé à un point, il est fort possible que l'onde rencontre deux obstacles différents. Dans ce cas de figure, la mesure effectuée est une moyenne des distances aux obstacles rencontrés. (cf. figure 77)

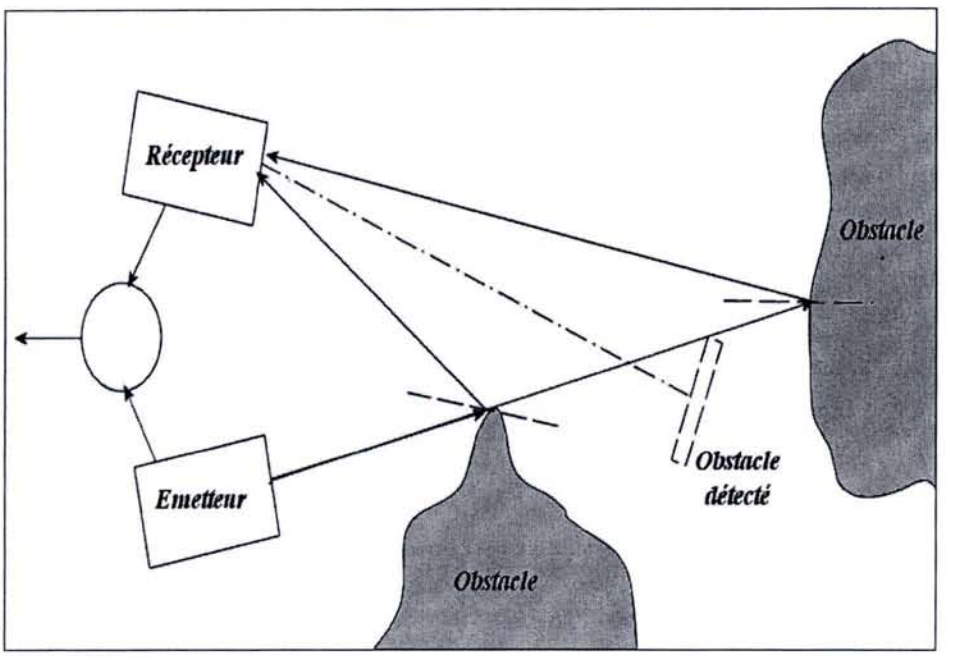

figure 77 : Phénomène de moyennage

10.1.2.3.1. Le télémètre laser

Le principe de fonctionnement d'un télémètre laser est celui énoncé dans le paragraphe précédent. L'onde émise est fournie par un laser, en général un laser à diode, qui émet une onde quasi monochromatique dont la longueur d'onde se situe dans les rouges ou dans les proches infra-rouges. Le récepteur est en général une caméra CCD adaptée dont la position est calibrée en fonction de la position de la source lumineuse. Ce dispositif permet d'effectuer une mesure dans un région bien spécifique. Pour obtenir un plus grand nombre de mesures, il existe plusieurs méthodes qui permettent d'utiliser la même source mais avec une configuration d'émission différente. Il est par exemple possible de transformer le faisceau laser en un plan de faisceaux dont les intersections avec les obstacles de l'environnement vont être reçues par la caméra. (cf. figure 78) Une autre possibilité consiste en une utilisation d'un jeu de miroir rotatif pour balayer la scène. (cf. figure 79)

figure 78 : Un modèle de télémètre laser utilisant une association caméra CCD/laser à diode (Nomadic Technologies)

Les télémètres fournissent en général des informations de bonne qualité, cependant leur coût et leur masse ne les prédisposent pas à toutes les plates-formes robotiques mais plus particulièrement aux structures les plus imposantes. Les informations qu'ils fournissent sont beaucoup plus précises que celles issues des capteurs de type sonars ou infrarouges.

figure 79 : Un modèle de télémètre laser à miroir (SICK)

10.1.2.3.2. Les sonars

Les sonars fonctionnent selon le même principe que le télémètre laser. L'onde émise ici est une onde ultra-sonique. Le couple émetteur/récepteur est remplacé par un dispositif unique qui fonctionne tantôt comme un émetteur et tantôt comme un récepteur. Ce dispositif, appelé transducteur, fournit une onde sonore dans un cône d'angle au sommet de valeur assez élevée. La mesure de la distance entre le capteur et l'obstacle s'effectue en évaluant le temps de vol d'un train d'ondes entre l'émission et la réception.

Les sonars sont en général répartis dans une ceinture permettant la mesure de la distance tout autour du robot (cf. figure 80). Les capteurs disposés de cette façon doivent être synchronisés en émission pour ne pas apporter d'imprécisions supplémentaires dues à la réception des trains d'onde émis par les capteurs adjacents. En outre, les capteurs fonctionnent de façon périodique pour augmenter le nombre des mesures et ainsi limiter les imprécisions, qui sont plus présentes que pour le télémètre notamment à cause de l'étendue du cône d'émission et de la perte d'énergie lors de la propagation de l'onde dans l'air.

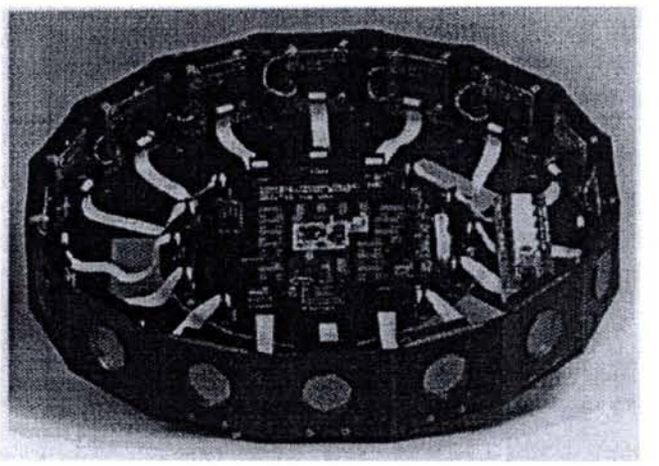

figure 80 : La ceinture de sonars du robot "Gaston" (Nomad200 Nomadic Techologies)

10.1.2.3.3. Les capteurs infrarouges

Nous avons vu précédemment que les capteurs infrarouges sont essentiellement connus pour être utilisés en tant que capteurs de proximité. En effet, étant donné que l'onde émise est une onde lumineuse, il semble impossible d'utiliser la méthode de mesure du temps de vol de l'onde. En outre, comme la lumière émise n'est pas du type laser, son étendue ne permet pas de localiser facilement le rayon réfléchi. Cependant, il est possible de mesurer des distances à l'aide de dispositifs infrarouges. La société Wany®, par exemple, a déposé un brevet sur l'utilisation de capteurs infrarouges pour effectuer de la mesure de distance. Le principe de fonctionnement de ces capteurs est

basé sur la mesure de la quantité d'énergie reçue par le récepteur en fonction de la quantité d'énergie émise. Ce rapport d'énergie est lié au temps de déplacement de l'onde dans l'air, notamment par l'intermédiaire de la dissipation énergétique induite par le milieu de propagation.

Outre les limitations classiques inhérentes aux capteurs de distance, la précision de la mesure est également limitée par les différentes propriétés radiométriques des obstacles rencontrés. Ainsi, la position d'un mur noir, de part ses propriétés d'absorption de la lumière, va être mesurée à une distance plus grande qu'un mur blanc placé au même endroit.

10.1.2.4. Les dispositifs imageurs

Les capteurs que nous avons présentés jusqu'à présent sont spécifiques à des tâches de positionnement relatif du robot par rapport à des obstacles présents dans l'environnement. Hormis le télémètre laser qui fournit des informations d'une grande qualité, les autres dispositifs d'acquisition sont très bruités et ne permettent essentiellement que l'élaboration de comportements réactifs de type stimulus/réponse. Néanmoins, il n'est pas rare de trouver dans la littérature des applications de localisation ou de cartographie réalisées à l'aide de capteurs de faible performance tels que les sonars [Demirli00]. Cependant, pour des applications où la détection et l'identification des objets sont primordiales, il est nécessaire d'avoir recours à des capteurs plus riches en terme de quantité d'informations. Le capteur qui est alors le plus utilisé est la caméra.

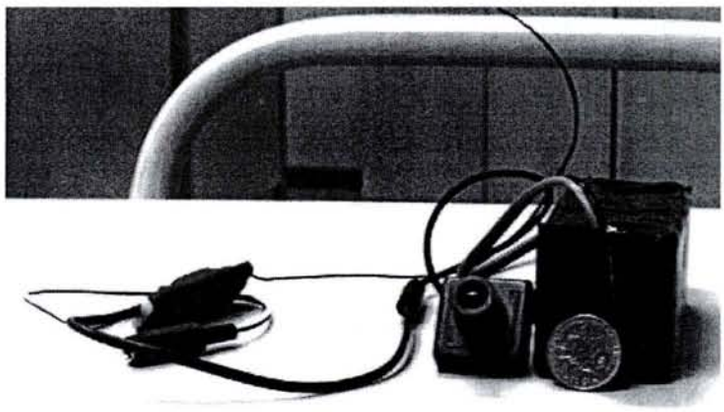

figure 81 : Un exemple de caméra miniature (avec transmetteur vidéo HF) utilisée sur les robots de l'équipe MAIA

Depuis quelques années, la taille (cf. figure 81) et surtout les prix des dispositifs imageurs ont fortement diminués, ce qui tend à imposer les caméras comme capteur essentiel à toute application robotique ou domotique. Bien qu'il soit fort intéressant de disposer d'une caméra, l'apport d'informations important qu'elle fournit va de pair avec une augmentation indispensable des ressources de traitement. En outre, bien qu'il soit naturel pour l'être humain d'interpréter les informations visuelles, il n'est pas aussi facile de formaliser cette interprétation sous forme algorithmique. Néanmoins, la richesse des informations fournies par les dispositifs imageurs en font un capteur indispensable dès lors que l'on veut accomplir des tâches complexes et précises. C'est donc naturellement que nous nous sommes concentrés sur l'utilisation des caméras dans le cadre de notre application.

Maintenant que nous avons exploré l'ensemble des capteurs les plus importants pour la localisation, nous allons détailler quelques algorithmes couramment utilisés en robotique mobile afin d'exploiter les caméras. Nous séparons ces algorithmes en deux familles en fonction du nombre de dispositifs d'acquisition auxquels ils sont appliqués.

10.2. L'exploitation des capteurs

10.2.1. But de ce chapitre

Nous avons vu, dans le chapitre précédent, que chaque type de capteurs avait ses caractéristiques propres et surtout ses domaines d'application privilégiés. Ainsi, il est indispensable de choisir correctement le type de capteur à utiliser lorsque l'on désire aborder une application telle que la localisation d'un robot mobile. Outre le choix du capteur, il faut également prendre en compte le choix des algorithmes à utiliser.

Le but de ce chapitre est d'effectuer un état de l'art de l'existant en matière de traitement d'image adapté à une tâche de localisation (externe ou auto-localisation). Nous avons choisis de nous focaliser sur l'utilisation d'une caméra comme capteur principal car c'est, de notre point de vue, le capteur qui permet de fournir la plus grande richesse d'information sur une scène observée, certes au prix d'un traitement des information parfois coûteux. Notre objectif est de pouvoir utiliser un capteur imageur tout en limitant le coût des traitements nécessaires à l'extraction de l'information. En d'autres termes, notre but est de développer une méthode de localisation où la partie cognitive et interprétative est la plus limité possible. Ainsi, nous voulons montre qu'il est possible de réaliser des tâches complexes en utilisant des modules de traitements séparément élémentaires. En outre, nous voulons également étudier la possibilité d'utilisation de plusieurs caméras de façon complémentaire.

Dans ce chapitre nous allons voir, dans un premier temps, les méthodes les plus couramment utilisées dans le cadre d'une application mono-capteur. Ensuite nous verrons comment il est possible d'aborder le cadre d'utilisation de plusieurs capteurs.

10.2.2. Cas mono-capteur

Etant donné que le moteur de cette thèse est la localisation et le suivi d'entités mobiles, je vais essentiellement m'attacher, dans cette partie, à expliciter les algorithmes les plus couramment utilisés dans ce cadre applicatif, bien qu'il existe d'autres types d'applications ayant recours à une perception mono-capteur.

Afin d'obtenir des résultats pertinents et précis lors de l'utilisation d'un capteur unique, il faut que ce dernier soit suffisamment riche pour en extraire le maximum d'informations. Ce principe fondamental écarte de ce type d'applications les capteurs très bruités et perceptivement limités tels que les sonars ou encore les capteurs infrarouges. Les seuls capteurs envisageables dans le cadre d'une localisation à capteurs unique sont le télémètre laser et la caméra. Nous remarquons au passage, que ces dispositifs sont en fait des capteurs multiples puisque le télémètre laser effectue plusieurs mesures d'une même scène en utilisant des dispositifs mécaniques alors que la caméra est en fait une structure composée de plusieurs photosites pouvant chacun être considéré comme un capteur à part entière.

Les applications utilisant un télémètre laser sont souvent basées sur une cartographie de l'environnement effectuée par accumulation des acquisitions lors des déplacements du capteur [Wallner97], [Crowley98]. D'autre part, un télémètre laser fixe peut également être utilisé pour la localisation en comparant une acquisition dépourvue de présence (acquisition de référence) avec les acquisitions courantes. Dans tout les cas, une localisation précise ne peut d'effectuer qu'en fournissant une métrique aux acquisitions qui est étroitement liée à la métrique du monde réel. Ce lien entre les différentes métriques s'appelle l'étalonnage (ou calibration) et se retrouve également lors de l'utilisation de caméras.

L'utilisation d'une caméra pour la localisation est beaucoup plus courante. Ceci s'explique non seulement par le faible coût d'achat d'une caméra par rapport à un télémètre, mais également par la plus grande facilité d'interprétation des informations fournies.

10.2.2.1. Méthodes couramment utilisées

Nous allons exposer quelques une des méthodes couramment utilisées en traitement d'image pour effectuer de la localisation et du suivi d'objet. Bien évidemment, nous ne pouvons pas parcourir ici toutes les méthodes utilisées en traitement d'image pour effectuer de la localisation, de l'identification ou du suivi. Il est possible de retrouver la majorité des méthodes élémentaires de traitement d'image dans des ouvrages spécifiques [Horaud93a], [Pratt78].

10.2.2.1.1. Localisation d'objets dans une scène

Les méthodes utilisables pour localiser un objet dans un environnement différent sensiblement en fonction de la position du capteur. En effet, la détection de la position d'un objet mobile ne se fera pas de la même façon si le capteur est placé sur l'objet ou si celui-ci est placé dans l'environnement. Nous allons voir, dans un premier temps, le cas où le capteur est embarqué. Nous verrons ensuite comment localiser un objet en observant la scène depuis l'extérieur.

Capteurs embarqués

Dans le cas où le capteur est placé sur l'objet mobile, la localisation consiste essentiellement en deux étapes. Dans la première, il s'agit de reconnaître la scène ou les objets qu'elle contient. Ensuite, il faut retrouver, à partir des caractéristiques extraites, la position absolue de l'environnement qui correspond à cette observation. En général cette seconde étape est une triangulation purement géométrique qui permet de déduire la position du capteur, et donc de l'entité mobile, en fonction de sa position relative par rapport aux éléments caractéristiques observés. Nous allons principalement détailler la partie concernant la détection des objets dans la scène car elle possède de nombreux points communs avec les algorithmes utilisés pour la localisation avec capteurs externes.

Il existe plusieurs manières d'aborder la détection d'objets dans une scène. La méthode la plus intuitive consiste à balayer de façon systématique l'image pour rechercher des éléments caractéristiques similaires à ceux des objets contenus dans une base construite à l'avance. Le principal inconvénient de cette méthode est qu'il faut disposer d'une base d'objets suffisamment représentative pour permettre une détection précise. En outre, chaque objet de la base devra être stocké soit sous plusieurs tailles et orientations, soit sous forme vectorielle afin de permettre une détection quelque soit sa position par rapport au capteur. En fonction de la taille et de la position des objets dans l'image, il est possible de déduire la position du capteur par rapport aux objets par triangulation simple. Ce dernier point nécessite une cartographie complète de l'environnement contenant la position de tous les objets. Cette méthode est de moins en moins utilisée car elle est très coûteuse bien que permettant l'obtention d'estimations de positions très précises.

Une autre méthode, un peu moins coûteuse, consiste à extraire et à interpréter des primitives, intersections d'arrêtes par exemple, extraites des images pour détecter la configuration caractéristique d'une certaine position dans l'environnement. Cette technique donne d'excellents résultats lorsque les conditions de prise de vue sont contrôlables, c'est pourquoi elle est surtout utilisée pour des applications de contrôle de processus industriels tel que le contrôle de la configuration d'essieux sur une chaîne de montage (cf. figure 82). En général, les primitives à rechercher sont des formes géométriques simples, pouvant être mises en évidence avec des algorithmes de traitement d'image élémentaires (filtre de Sobel, Gradient, Laplacien,...).

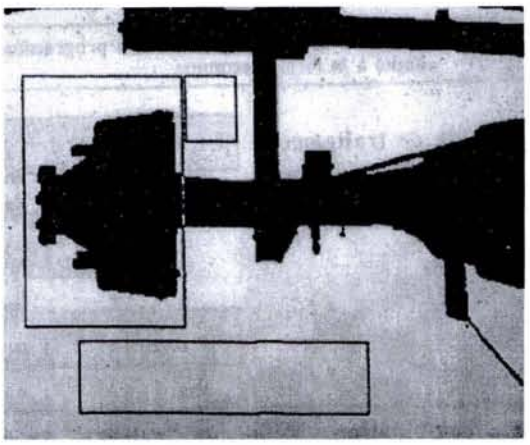

figure 82 : Localisation et identification d'essieux sur une chaîne de montage (Ford)

En ce qui concerne la robotique mobile, il est plus délicat d'utiliser cette méthode mise à part dans des locaux où les conditions de luminosité ne sont pas dépendantes de la lumière du jour. Elle est donc adaptée uniquement à un environnement structuré d'intérieur tel que des couloirs par exemple [Gechter99] (cf. figure 83). Néanmoins, ce principe de localisation est couramment utilisé en robotique mobile notamment au sein d'algorithme de stéréovision [Ayache89], [Baerveldt01], [Zingaretti98].

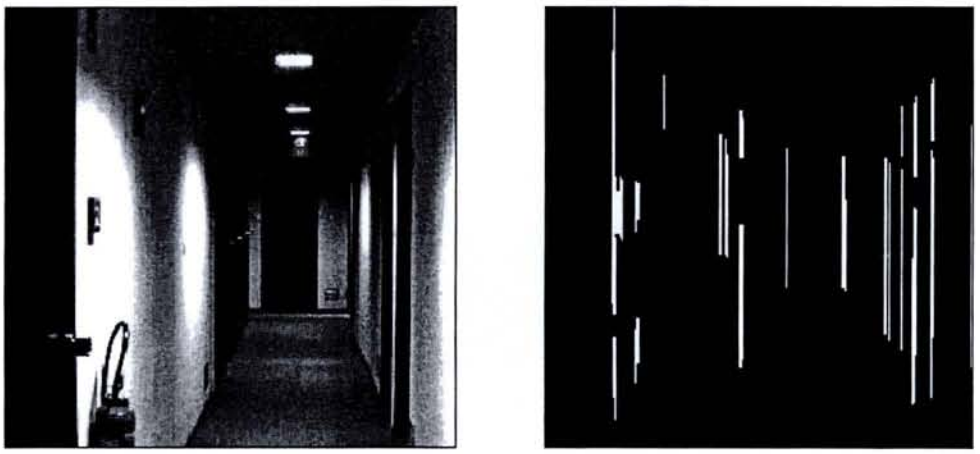

figure 83 : Extraction de primitives (arrêtes verticales)

Enfin, une des méthodes d'extraction de primitives de plus en plus développées actuellement consiste en l'utilisation de contours actifs pour extraire des images des objets dont on connaît des caractéristiques géométriques immuables, mais dont la forme peut être sujette à des variations, tels que des visages (cf. figure 84 et figure 85), des éléments anatomiques ou des routes. Cette méthode est une alternative efficace aux

algorithmes usuels de filtrage d'autant qu'elle est très robuste aux variations de qualité des prises de vue [Cootes95], [Morris99].

figure 84 : Détection d'un visage dans une image (d'après [Cootes95])

figure 85 : Détection de la frontière de la rétine (d'après [Morris99])

Capteurs fixes dans l'environnement

Dans le cas où les capteurs utilisés sont fixes dans l'environnement, toutes les méthodes décrites précédemment sont toujours valables. Cette fois-ci, par contre, la localisation ne s'effectue pas selon le même processus. En effet, il ne s'agit plus maintenant de localiser le capteur relativement à des objets observés dont la position est

connue mais au contraire de localiser les objets détectés relativement au capteur dont la position est fixe et connue. Dans ce cas, la localisation des objets est beaucoup plus aisée puisque l'étape de triangulation finale n'est plus requise. Néanmoins, la reconstruction de la position absolue des objets nécessite toujours un recours à la géométrie comme l'explique le paragraphe suivant.

10.2.2.1.2. Reconstruction d'une position dans l'espace

L'utilisation d'une caméra peut être largement suffisante pour certaines tâches de localisation dans l'espace. L'étalonnage permet de retrouver, pour un pixel donné, la droite sur laquelle se trouve son équivalent réel. Pour faire une reconstruction fiable, il reste donc une indétermination à lever. On peut s'affranchir de ce problème en prenant en compte la géométrie de la scène par rapport à la position du capteur et en émettant des hypothèses sur les objets à localiser. Dans le cadre de la localisation d'individus dans une pièce, par exemple, nous pouvons supposer, sans que l'hypothèse ne soit trop restrictive, que les entités mobiles à localiser vont se trouver sur le sol de l'environnement. Dans ce cas, il suffit de connaître l'équation du plan constitué par le sol, soit dans le repère lié à l'environnement soit dans le repère lié à la caméra. La position dans l'espace de l'entité peut donc être calculée comme étant l'intersection entre le plan du sol et la droite de reconstruction. (cf. figure 86)

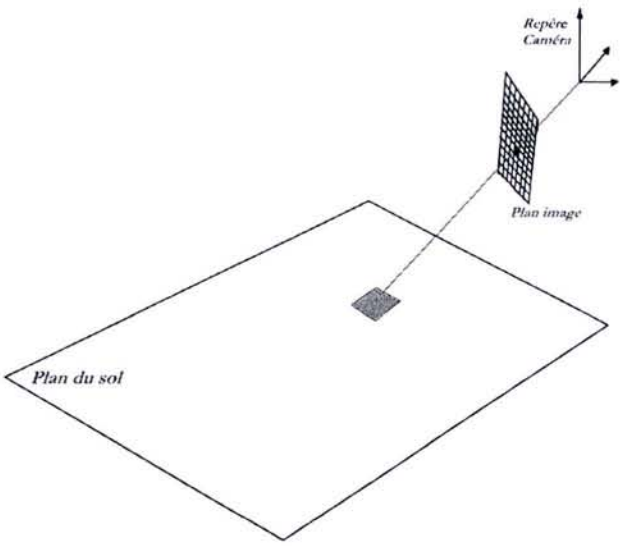

figure 86 : Reconstruction à l'aide d'une seule caméra

A partir de ces observations, nous pouvons remarquer que le placement dans l'espace des capteurs est une étape primordiale dans tout algorithme de localisation. Dans notre

cas particulier, si le plan image de la caméra est perpendiculaire au plan du sol (caméra à l'horizontal), il est impossible d'utiliser la totalité du capteur pour extraire des informations pertinentes. Les zones supérieures de l'image (image du plafond) et les zones centrales (droite de reconstruction parallèles au plan du sol) ne pourront par être utilisées. Uniquement les pixels situés en bas de l'image pourront être exploités moyennant une certaine réserve. En effet, étant donné que l'angle d'incidence entre la droite de reconstruction et le plan du sol est très faible, la précision ne sera pas optimale. Pour obtenir la meilleure précision possible, il faut que le plan image de la caméra soit parallèle au plan du sol ce qui correspond à une caméra placée verticalement. En plaçant la caméra de cette manière, on s'assure que les pixels de l'image vont correspondre à des surfaces au sol de petite taille et surtout que ces tailles seront homogènes de part et d'autre du pixel central (cf. figure 87).

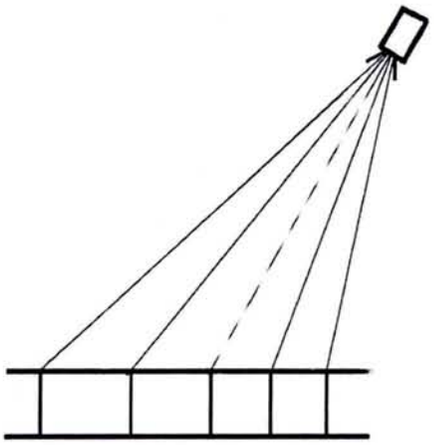

figure 87 : Variation de le surface au sol correspondant à chaque pixel en fonction de l'éloignement par rapport à l'axe optique (en pointillé)

10.2.3. Cas multi-capteurs

10.2.3.1. Fusion de données

Dès lors que l'utilisation de plusieurs capteurs s'avère indispensable, il est nécessaire d'avoir recours à une fusion de l'information fournie. Le choix de la méthode de fusion à employer est directement lié à la nature des sources d'informations (information numérique, information symbolique,...) et à la forme des résultats que l'on veut obtenir.

La fusion de données peut être définie, de façon générale, comme étant l'intégration de données provenant de sources multiples (capteurs, modules de décision, modules d'interprétation,...) dans le but de construire une interprétation pertinente et spécifique sur une entité, une activité ou un événement. Cette interprétation doit en particulier prendre en compte les incohérences entre les différentes sources et fournir un résultat plus précis et plus riche par rapport à ce que l'on pourrait obtenir en exploitant séparément les informations initiales.

D'un point de vue formel, le but de la localisation est de construire, à partir de plusieurs sources d'information {S1,S2,...,Sn}, une source de connaissance Sc. Cette source doit rassembler l'information contenue dans les sources initiales de façon à être "meilleure" du point de vue quantitatif, qualitatif, confiance ou sémantique (information d'un plus haut niveau d'abstraction) [Dutech99]. Les sources d'informations peuvent être concurrentielles ou complémentaires et fournir des informations qui, une fois combinées, vont permettre de construire une connaissance meilleure.

D'après [Ayari96], un système de fusion de données peut se décomposer en deux niveaux :

Premier niveau : ce niveau concerne la fusion de données et les opérations formant des hypothèses partielles.

Second niveau : ce niveau constitue l'évaluation de la situation, l'inférence et l'interprétation.

Dans la plupart des cas, la fusion est un processus dirigé à la fois par les données et par les buts. En effet, Les données conditionnent la précision et la véracité de résultats alors que les buts contribuent au choix de la conduite du processus d'interprétation.

Il existe bien évidemment de nombreuses méthodes pour réaliser de la fusion de données. Néanmoins, dans la suite de ce chapitre, j'ai choisi de ne citer que celles qui sont le plus adaptées à mon cadre applicatif.

10.2.3.2. Algorithmes classiques en fusion de données

10.2.3.2.1. Fusion statistique

Le principe de la fusion statistique est de construire une hypothèse de décision à partir de données, fournies par des sources d'information différentes, en prenant en compte les connaissances *a priori* sur le processus d'observation.

Cette fusion à pour principe l'utilisation d'un modèle des observables. Celle-ci possède en ce sens des propriétés similaires aux méthodes de fusion utilisant les principes de la théorie de l'information. Ainsi, elle nécessite des connaissances *a priori* sur le processus d'observation. Nous pouvons classer dans cette catégorie des méthodes telles que l'inférence classique, la méthode de Dempster-Shafer et l'approche Bayesienne [Lopez-Orozco98].

Nous détaillerons plus en détail une méthode de fusion probabiliste dans le chapitre IV de la Partie III.

10.2.3.2.2. Fusion physique

Principe général

Le principe de cette fusion est d'assembler un ensemble de sources d'informations ou d'observation dont les caractéristiques sont définies à partir d'un modèle précis. En d'autres termes, il s'agit d'adapter l'information issue des sources initiales pour qu'elles correspondent à un modèle prédéfini.

La stéréovision rentre dans le cadre de ce type de fusion. En effet, il s'agit alors de fusionner la position d'amers (points dans les images représentant la même source initiale) dans des vues différentes de la même scène dans le but de reconstruire une position dans l'espace. Nous allons donc voir plus en détail le principe de fonctionnement de la stéréovision dans le cadre d'une application robotique typique. Nous avons choisi la stéréovision pour illustrer le concept de fusion physique car cette méthode est certainement celle la plus couramment utilisée en robotique dès qu'il s'agit d'effectuer une reconstruction 3D dans l'environnement réel.

Un exemple de fusion physique:la stéréovision

Stéréovision : définition

La stéréovision est un procédé qui permet d'extrapoler la position d'un objet dans l'espace à partir d'un ou de plusieurs points de vue différents. Ce procédé est abondamment utilisé en robotique mobile notamment en détection et évitement d'obstacles. La reconstruction de la position d'un objet dans l'espace peut se faire à partir d'une seule image notamment par des techniques de détection de cohérence [Henkel00] ou en connaissance des informations supplémentaires sur les conditions de prise de vue. Cependant, il est plus courant d'effectuer de la reconstruction de position dans l'espace à l'aide de plusieurs points de vue sur une scène. Dans la majorité des cas, la stéréovision est réalisée à l'aide de deux, voire trois, caméras [Laubach99], [Olson97a], [Olson97b], [Murray96a] et [Murray96b]. Le principe général est alors de lier des éléments caractéristiques de la scène (arrêtes, angles, objets,...) présents dans toutes les images et d'extrapoler leurs positions en prenant en compte la géométrie du dispositif d'acquisition et le modèle des caméras. La géométrie du dispositif d'acquisition et le modèle des caméras sont déterminés lors d'une phase préalable appelée étalonnage (ou calibration).

Etalonnage : principe général

L'étalonnage dépend essentiellement de la modélisation préalable du dispositif d'acquisition. Celle-ci doit tenir compte de tous les paramètres susceptibles d'entrer en jeu lors de la capture d'une image. Le modèle doit représenter toute la chaîne d'acquisition, c'est-à-dire tous les éléments constituant l'optique, la position du capteur dans l'espace, l'échantillonnage spatial s'il s'agit d'un capteur CCD et l'électronique. Pour cela, il existe plusieurs modèles de caractérisation des paramètres définissant la caméra. En général, la modélisation comporte essentiellement les paramètres liés à l'optique et à l'échantillonnage avec plus ou moins de précision.

La modélisation la plus élémentaire est dite à sténopé [Faugeras92]. Ce modèle consiste à caractériser la transformation entre l'espace à trois dimensions et le plan à deux dimensions par une projection perspective en utilisant comme centre de celle-ci le centre optique du capteur. L'échantillonnage est, quant à lui, modélisé par quatre paramètres donnant la position du centre de projection dans le plan image ainsi que les pas d'échantillonnage de la matrice CCD. La modélisation peut donc être décomposée en deux étapes. La première consiste en une transformation géométrique permettant de passer du repère de la scène au repère associé au capteur. Ce passage de l'espace de la scène au plan image est régit par six paramètres : trois pour la rotation et trois pour la translation. La seconde étape est l'échantillonnage spatial, régit quant à lui par quatre paramètres : la position de l'image du centre de projection (deux paramètres) et les pas

d'échantillonnage vertical et horizontal. L'étalonnage du capteur consiste donc à déterminer tous ces paramètres.

D'autres méthodes utilisent un modèle à sténopé amélioré pour tenir compte des distorsions engendrées par l'optique [Tsai87]. Ces distorsions peuvent être de trois types [Tarel95]. Elles peuvent par exemple être radiales, c'est le type de distorsion le plus souvent modélisé. Elles sont directement liées à la qualité de l'optique utilisée. Elles peuvent également être dues au décentrage des lentilles constituant l'optique par rapport à l'axe. Enfin, elles peuvent résulter d'erreurs de parallélisme entre les lentilles dans le cas où l'optique est composée de plusieurs éléments. La prise en compte de ces distorsions apporte une non-linéarité au système d'équations à résoudre pour déterminer les paramètres du capteur. Ce qui implique d'autres méthodes de résolution.

En outre, l'étalonnage nécessite la connaissance précise de la position d'objets de référence utilisés pour effectuer une prise de vue étalon. Ceci peut être réalisé à partir de scène réelle dans le cadre notamment d'un étalonnage automatique du capteur, mais s'effectue en général à l'aide d'une grille d'étalonnage judicieusement disposée devant le dispositif d'acquisition. Cette grille peut prendre différentes formes telles que celle d'un quadrillage ou d'une mire constituée de disques de dimensions différentes. La méthode la plus utilisée consiste à associer les points significatifs de la grille, dont la position dans le repère scénique est parfaitement connue, avec leur correspondant dans l'image acquise. En utilisant un nombre suffisant de points, on peut trouver, plus ou moins aisément, les paramètres liés au capteur, la difficulté de la résolution dépendant de la modélisation préalable. Lors de cette résolution, on pourrait se contenter d'un nombre réduit de points. Cependant, l'expérience montre qu'il faut utiliser un grand nombre de points pour accroître la précision du modèle. Il s'agit d'obtenir un système surdéterminé que l'on résoudra par exemple par la méthode des moindres carrés [Press92].

Il est également possible d'effectuer des associations entre des indices autres que la position de certains points significatifs de la scène et de l'image. On peut, par exemple, utiliser les moments géométriques de figures se trouvant sur la grille d'étalonnage [Tarel95]. Cette méthode semble donner des résultats bien meilleurs que les méthodes classiques de déterminations des paramètres du dispositif d'acquisition. En effet, avec cette méthode, les précisions de mesure sont inférieures au pixel. Il faut prendre cependant quelques précautions lors de l'acquisition des images. Cette méthode peut être très intéressante lorsque l'on fait de la métrologie optique, mais dans notre cas, une méthode classique semble suffisante étant donné les imprécisions au niveau des déplacements du robot. Il reste ensuite à déterminer si la modélisation des distorsions est nécessaire pour l'application ou si un modèle à sténopé simple peut convenir. Cependant, l'étalonnage des caméras ne se faisant pas pendant l'exécution des tâches du robot, il n'est pas pénalisant d'utiliser un modèle plus approfondi pour caractériser les caméras.

Dans le cadre de la vision stéréoscopique, il ne suffit pas d'étalonner indépendamment les deux caméras. Il faut en outre étalonner les transformations

nécessaires au passage du repère d'une caméra à celui de l'autre. Pour cela, on modélise la transformation entre les deux caméras comme étant composée d'une rotation et d'une translation. Il faut donc déterminer les six paramètres caractérisant cette transformation. Du point de vue du protocole opératoire, on peut utiliser les images acquises pour l'étalonnage individuel des capteurs à condition que celles-ci aient été prises au même moment et sans mouvement d'une caméra par rapport à l'autre. Le lien entre les deux capteurs est déterminé grâce à la géométrie épipolaire.

Il faut également signaler l'existence de méthodes d'étalonnage automatique qui utilisent directement des éléments de la scène pour déterminer tous les paramètres du système d'acquisition. On peut également utiliser des méthodes d'étalonnage supervisées qui requièrent l'intervention d'un opérateur pour déterminer les points utiles pour calculer les correspondances entre la scène en trois dimensions et l'image 2D. [Worrall92]

Pour résumer, le principe général de l'étalonnage peut être décomposé en plusieurs étapes :

- La détermination des paramètres significatifs du processus d'acquisition (type de capteur, filtres,...)
- L'acquisition d'une scène connue, mesurable avec précision. (utilisation d'une mire par exemple)
- La construction de liens entre les éléments connus de la scène observée et leurs représentants perceptifs.
- La détermination des paramètres.

L'annexe précédente explique en détail les étapes de la détermination des paramètres du dispositif d'acquisition dans le cadre d'une modélisation simple du dispositif d'acquisition.

<u>Acquisition d'une scène mesurable et construction des liens</u>.

Dans le cadre de l'utilisation d'une caméra, la scène la plus facilement exploitable est une mire judicieusement choisie en fonction de la précision et des caractéristiques désirées. Une mire est un objet construit avec des éléments significatifs (lignes, points, ...) mesurables. Les paramètres typiques d'une mire sont : la distance entre les points ou les lignes, les tailles relatives des figures constituant la mire ou encore leur nombre.

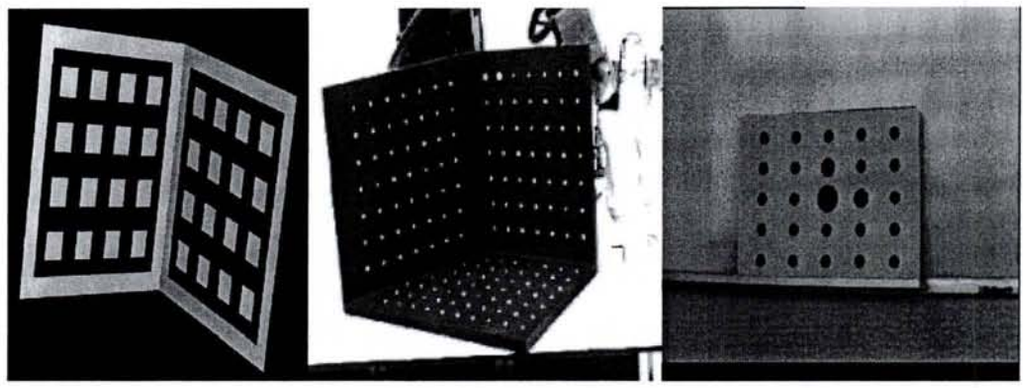

figure 88 : Exemples de mires utilisées [Baerveldt01] à gauche et [Horaud93b] au centre et [Gechter99] à droite.

Les positions dans l'espace des éléments caractéristiques de la mire sont facilement calculables puisqu'on connaît tous les paramètres de l'objet et sa position par rapport au capteur. Il faut ensuite repérer la position des objets caractéristiques dans l'image acquise en utilisant les méthodes classiques utilisées en traitement d'image. Les objets qui doivent être détectés sont en générale facilement isolables des autres éléments de l'image. Ce sont souvent des points, des formes géométriques simples (carrés ou disques) ou des éléments périodiques. Le nombre d'éléments présents sur la mire doit être suffisamment important pour obtenir un système surdimensionné que l'on peut résoudre par des méthodes classiques. Néanmoins, la méthode la plus utilisée pour l'étalonnage des caméras est la méthode Tsai [Tsai87] en deux étapes qui est détaillée en annexe.

Une fois tous les paramètres du modèle connus, il est relativement aisé de calculer, pour un point donné de l'image numérique, le rayon de vue sur lequel se trouve le point de la scène correspondant.

Nombre de caméras nécessaires

Dans le cas où il n'est pas possible de posséder des informations supplémentaires sur les positions éventuelles des cibles à détecter, il faut pouvoir disposer d'une autre source d'informations. En général, on a recours à une seconde caméra étalonnée, voire parfois une troisième caméra. Un objet réel se retrouvant dans les deux (ou trois) images peut alors être localisé en calculant l'intersection des droites reconstruites à partir des informations sur chacun des capteurs utilisés. Ceci est bien entendu le cas idéal, mais il se pose deux problèmes délicats à résoudre :

Même en faisant un étalonnage très précis de chacun des capteurs mis en œuvre, il n'est pas certain que les droites soient sécantes en un seul point. En effet, étant donné que les caméras effectuent un échantillonnage de la scène, il se peut qu'il y ait un léger décalage dans la localisation des objets dans les images notamment à cause de la

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

présence de pixels mixtes. En outre, la précision obtenue dépend fortement de la méthode de triangulation choisie [Hartley97].

En outre, la tête stéréoscopique doit avoir une certaine géométrie pour permettre aux capteurs d'avoir une zone d'observation commune tout en conservant un éloignement suffisant entre les dispositifs d'acquisition pour permettre une plus grande précision.

10.2.4. Conclusion

Nous avons vu, dans ce chapitre, que les méthodes utilisées pour effectuer de la localisation en robotique mobile, que ce soit de la localisation externe ou de l'auto-localisation, sont en générale composée de deux étapes. La première est constituée d'un ou de plusieurs algorithmes de traitement d'images destinés à interpréter la scène observée. La seconde est une étape de recalage des éléments reconnus dans l'environnement réel en utilisant des méthodes de géométrie projective élémentaire.

Le principal inconvénient de toutes les méthodes présentées dans ce chapitre est leur complexité et leur besoin en ressources matérielles pour obtenir des temps de réponses raisonnables dans le cadre d'une application sur un robot réel. En effet, toutes ces méthodes passent par une interprétation des données visuelles dont les calculs ne sont pas toujours triviaux.

Même si l'on peut s'inspirer de certains éléments des méthodes exposées dans ce chapitre, il est nécessaire de se tourner vers des principes de raisonnement qui ne sont pas directement liées au traitement d'images mais plutôt à d'autres domaines de l'informatique tel que la classification.

10.3. La détermination des paramètres d'acquisition d'un dispositif imageur dans le cas d'un modèle simple.

Nous allons voir maintenant un exemple de paramétrage d'une caméra CCD standard. Cet exemple est couramment utilisé en stéréovision ou en métrologie.

Lorsque des caméras sont utilisées dans le but de mesurer des objets dans une scène en trois dimensions, il est nécessaire de connaître, avec le plus de précision possible, les transformations qui permettent le passage entre un objet concret de la scène 3D et son correspondant dans l'image numérique. Ainsi, il faut modéliser le processus d'acquisition de l'image en fonction de la position dans l'espace des objets et des capteurs. En outre, pour que la modélisation soit complète, il faut également tenir compte des éléments qui constituent le capteur. S'il s'agit d'une caméra CCD par exemple, il faut considérer la qualité de l'optique et les caractéristiques de la matrice tels que la taille des éléments photosensibles ou la distance inter-éléments. Le principe de l'étalonnage est de déterminer, à l'aide d'un objet-étalon, les différentes caractéristiques du modèle. Aussi, cette procédure est fortement liée à la modélisation de l'acquisition.

10.3.1. Le modèle classique

Il existe plusieurs méthodes pour paramètrer un dispositif d'acquisition. Le modèle le plus utilisé est dit à « sténopé ». La caméra est alors considérée comme un trou d'épingle qui constitue le centre optique du capteur (cf. figure 89). L'image acquise est l'intersection entre les rayons issus des points de la scène 3D passant par le centre optique et un plan situé derrière le trou d'épingle et distant de la longueur f .

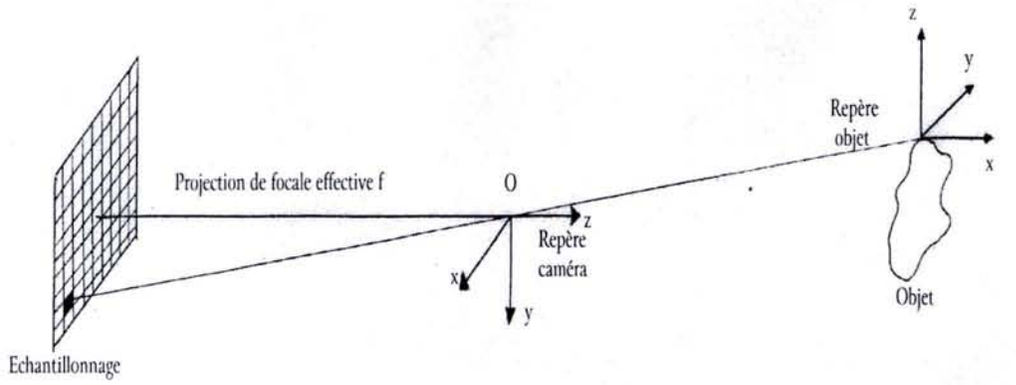

figure 89 : Le modèle à trou d'épingle

La formation de l'image revient donc à une transformation rigide, rotation et translation, et à une projection. Si l'on considère ensuite que le plan image supporte une matrice CCD, il faut ajouter au modèle les paramètres de discrétisation de la matrice. On obtient donc avec les paramètres suivants :

6 paramètres pour la transformation rigide qui correspondent aux 3 degrés de liberté de la rotation et aux 3 degrés de liberté de la translation. Ces six éléments constituent les paramètres extrinsèques de la caméra.

1 paramètre pour la projection. Il correspond à la focale effective f du capteur. On peut remarquer, dans la figure 90, que la focale effective f est positive alors que cette dernière est négative dans la figure 1. Dans ce cas, tout se déroule comme si les rayons étaient interceptés par le plan avant de passer par le trou d'épingle.

4 paramètres pour la discrétisation (2 paramètres pour la position de la projection du centre optique et 2 pour les pas d'échantillonnage suivant les deux directions). Ces cinq éléments sont les paramètres intrinsèques de la caméra.

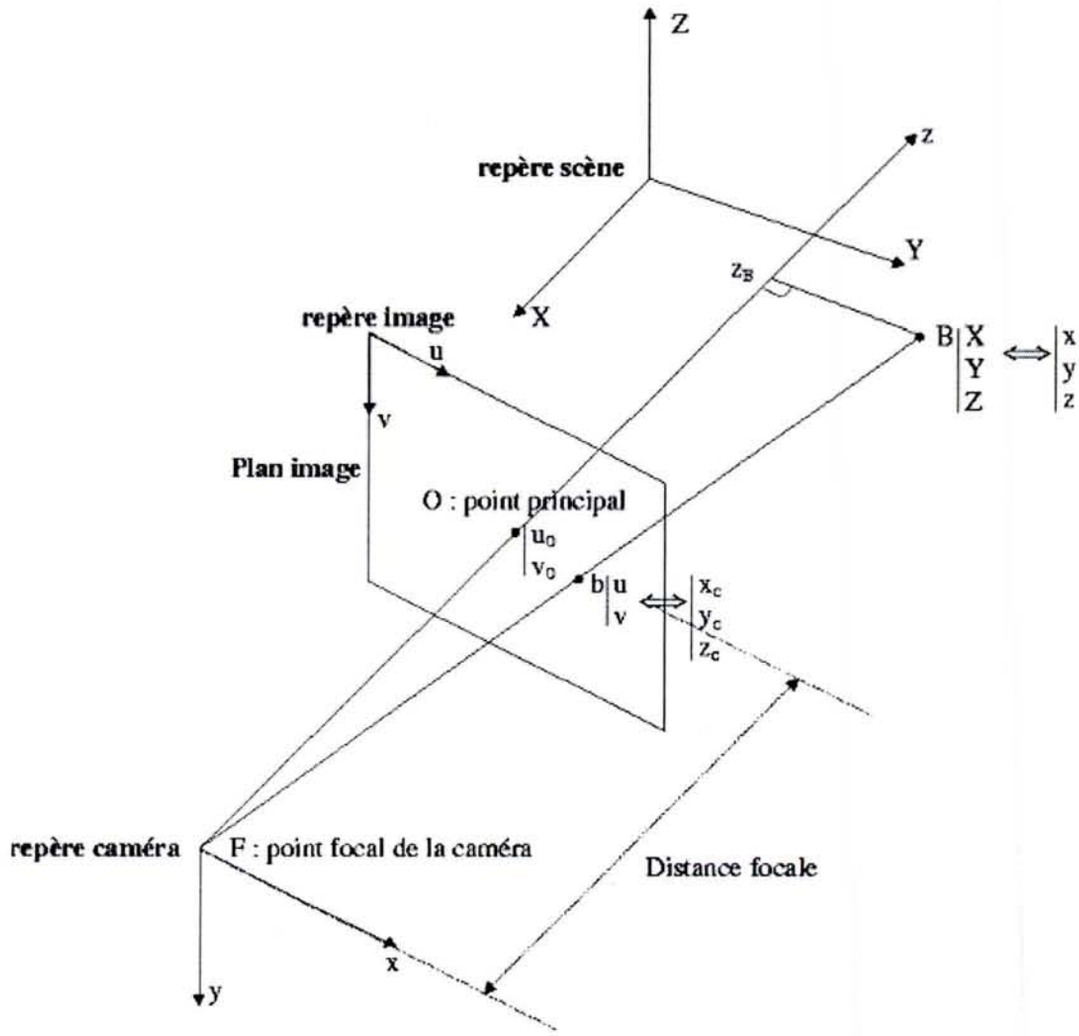

figure 90 : Le modèle à sténopé (d'après [Horaud93b])

Ce modèle a pour avantage d'être très simple, mais il ne tient pas compte de l'optique en amont de la caméra. Il donne de très bons résultats pourvu que l'optique soit de suffisamment bonne qualité. Pour une utilisation demandant davantage de précision, il faut également tenir compte des distorsions générées par l'optique de la caméra. Dans ce cas le modèle utilisé est un modèle à sténopé amélioré. Celui-ci diffère essentiellement de part l'ajout de plusieurs paramètres rendant compte de la distorsion radiale de l'objectif.

Transformation / Repère	Paramètres / Coordonnées	Equations
Repère de la Scène	(x_w, y_w, z_w)	
Transformation Rigide: Repère de la **Scène** ➔ Repère **Caméra**	<u>Rotation</u> : $(r_1, ..., r_9)$ <u>Translation</u>: (T_x, T_y, T_z)	$\begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 & T_x \\ r_4 & r_5 & r_6 & T_y \\ r_7 & r_8 & r_9 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix}$
Repère de la Caméra	(x, y, z)	
Projection Perspective : (Modèle sténopé)	Focale effective: f	$X_u = f.\frac{x}{z}$ $Y_u = f.\frac{y}{z}$
Repère Image (sans distorsion)	(X_u, Y_u)	
Distorsion Radiale de l'objectif:	Coefficients de distorsion: κ_1 et κ_2	$\begin{cases} X_u = X_d.(1+(\kappa_1.r^2 + \kappa_2.r^4)) \\ Y_u = Y_d.(1+(\kappa_1.r^2 + \kappa_2.r^4)) \end{cases}$ avec $r^2=(X_d^2 + Y_d^2)$
Repère Image (avec distorsion)	(X_d, Y_d)	
Acquisition par le capteur: (matrice CCD)	Centre de projection : C_x, C_y Coefficient d'incertitude: s_x	$\begin{cases} X_f = \frac{s_x}{d'_x}. X_d + C_x \\ Y_f = \frac{Y_d}{d_y} + C_y \end{cases}$ avec $d'_x = d_x . \frac{N_{cx}}{N_{fx}}$
Coordonnées dans l'image numérique	(X_f, Y_f)	

Tableau 1: Les étapes de la transformation: Monde Réel ➔ Image Numérique

10.3.2. Un modèle amélioré

Le modèle présenté ici est dérivé du modèle à sténopé classique. Il fut utilisé en particulier par Tsai [Tsai87] dans le cadre de sa méthode d'étalonnage. Il s'agit d'un modèle à sténopé amélioré qui tient compte de la distorsion radiale engendrée par l'objectif sous la forme de deux coefficients κ_1 et κ_2. Il faut donc ajouter aux trois étapes du modèle à sténopé classique (Transformation rigide, Projection et Echantillonnage), une quatrième étape liée à la distorsion. Cette étape supplémentaire se situe entre la projection sur le plan image et l'échantillonnage. Les étapes de la formation de l'image sont présentées dans le tableau page 227.

Détermination des paramètres:

La détermination des paramètres peut être réalisée selon différentes méthodes (détermination des paramètres). La méthode présentée ici est étroitement liée au modèle décrit dans le paragraphe précédent. Elle a été développée par Tsai [Tsai87] et se décompose en deux étapes successives. Chacune d'entre elles permet de déterminer les paramètres utilisés pour décrire le modèle de la caméra. Ces deux étapes sont dictées par plusieurs observations concernant le parallélisme entre $P_{oz}P$ et O_iP_d. (cf. figure 91)

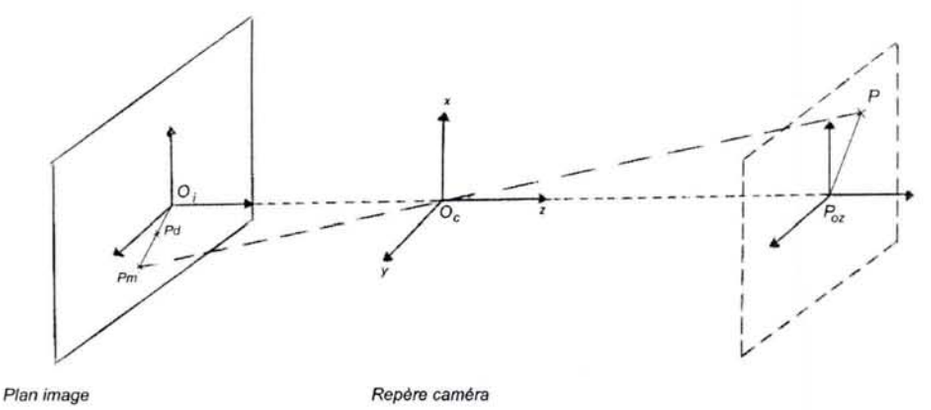

figure 91 : La condition de parallélisme de la méthode Tsai.

P_{oz} est le point de l'espace qui possède la même coordonnée en z, dans le repère lié à la caméra, que le point P. Les trois principales observations sont les suivantes :

La distorsion étant considérée comme étant radiale, elle ne modifie aucunement le parallélisme entre $P_{oz}P$ et O_iP_d puisqu'elle affecte de la même manière les coordonnées en X et Y.

La projection de focale effective f ne modifie pas non plus le parallélisme entre les deux segments car, de la même façon que la distorsion, elle affecte les coordonnées X et Y dans les mêmes conditions.

La translation en T_z ne modifie pas non plus le parallélisme entre les deux segments.

Première étape

La première étape a pour but de calculer les paramètres de la rotation ainsi que les paramètres Tx et Ty de la translation. Avant tout traitement, il faut chercher dans les données constructeur les paramètres N_{cx}, N_{fx}, d_x et d_y. N_{cx} représente le nombres d'éléments de la matrice CCD dans la direction x, N_{fx} correspond au nombre de pixels dans une ligne de l'image numérique, d_x et d_y sont respectivement la distance en millimètres entre les éléments dans la direction x et la direction y. On débute ensuite l'étalonnage par l'acquisition d'une image de la mire étalon.

Après avoir effectué cette acquisition, il faut déterminer l'ensemble des points que l'on va utiliser pour effectuer la mise en correspondance. Ces points, appelés points d'étalonnage, sont déterminés à partir des éléments présents sur la mire. Il est envisageable, par exemple, d'utiliser les centres de gravité ou encore les extrémités des motifs disposés sur le plateau étalon. Les coordonnées de ces points sont notées (X_{fi}, Y_{fi}) avec i compris entre 1 et N, N étant le nombre de points d'étalonnage utilisés. Le pixel central est pris comme origine du plan image dans l'image numérique : sa position est notée (C_x, C_y). Il s'agit ensuite de calculer les coordonnées des points dans l'image réelle distordue à partir de celles des points d'étalonnage dans l'image numérique. Ce calcul s'effectue à l'aide des relations suivantes :

$$\begin{cases} X_{di} = d'_x \dfrac{(X_{fi} - C_x)}{s_x} \\ Y_{di} = d_y\,(Y_{fi} - C_y) \end{cases}$$

Équation 23

avec $d'_x = d_x . N_{cx}/N_{fx}$ et (X_{di}, Y_{di}) les coordonnées dans l'image réelle distordue, (X_{fi}, Y_{fi}) les coordonnées dans l'image numérique et s_x un paramètre d'incertitude dans la détermination de X dû aux imprécisions du balayage au niveau du capteur (en général ce facteur est très proche de 1).

Soit les cinq inconnues suivantes, $\frac{r_1}{T_y}, \frac{r_2}{T_y}, \frac{T_x}{T_y}, \frac{r_4}{T_y}$ et $\frac{r_5}{T_y}$ que l'on notera respectivement a_1, a_2, a_3, a_4 et a_5. Cette seconde partie de l'étalonnage consiste à déterminer ces cinq inconnues. Pour cela, il faut utiliser la matrice L définie par :

$$L = (a_1, a_2, a_3, a_4, a_5)$$

Équation 24

D'après l'observation sur le parallélisme, on obtient la relation suivante :

$$(Y_{di}.x_{wi}, Y_{di}.y_{wi}, Y_{di}, -X_{di}.x_{wi}, -X_{di}.y_{wi}).L^T = X_{di}$$

Équation 25

avec (xwi, ywi, zwi) les coordonnées du point étalon i dans le repère lié à l'objet. La variation de i se fait entre 1 et N. Si N est supérieur à 5, le système est alors surdéterminé et sa résolution peut, par exemple, s'effectuer par le calcul de la matrice pseudo-inverse.

Après cette résolution, on obtient donc la valeur des cinq inconnues précédentes. Il s'agit donc ensuite de calculer les paramètres de la rotation ($r_1,\ldots,r_9$) ainsi que les paramètres T_x et T_y de la translation. Ce calcul s'effectue en trois étapes successives. Dans un premier temps, il faut calculer $|T_y|$ à partir des cinq inconnues précédentes. Pour cela, nous définissons la matrice C par la relation suivante :

$$C = \begin{pmatrix} r_1/T_y & r_2/T_y \\ r_4/T_y & r_5/T_y \end{pmatrix} = \begin{pmatrix} a1 & a2 \\ a3 & a4 \end{pmatrix}$$

Équation 26

La matrice C est orthogonale : en effet, il s'agit d'une sous-matrice de la matrice de rotation multipliée par le scalaire $\frac{1}{T_y}$. Si les deux colonnes (lignes) de C sont non nulles, on obtient alors :

$$T_y^2 = \frac{\left(S_r - \sqrt{(S_r^2 - 4.(a_1\, a_5 - a_4\, a_2)^2)}\right)}{(2.(a_1.a_5 - a_4.a_2)^2)}$$

Équation 27

avec $S_r=(a_1^2+a_2^2+a_4^2+a_5^2)$ sinon l'expression de T_y est donnée par :

$$T_y^2 = \frac{1}{(a_i^2+a_j^2)}$$

Équation 28

avec a_i et a_j les éléments de la matrice C non nuls.

Ce calcul nous permet donc d'obtenir *Ty* au signe près. Il ne reste donc plus qu'à chercher ce signe. Pour cela, nous prenons un point d'étalonnage éloigné du centre de coordonnées (X,Y) dans le plan image et de coordonnées (x_w,y_w,z_w) dans le repère lié à l'objet. Nous supposons, en outre, que le signe de Ty est +1. Calculons ensuite r_1, r_2, T_x, r_4 et r_5 en multipliant respectivement a_1, a_2, a_3, a_4 et a_5 par T_y. Il faut calculer ensuite les deux expressions suivantes :

$$\begin{cases} x = r_1.x_w+r_2.y_w+T_x \\ y = r_4.x_w+r_5.y_w+T_y, \end{cases}$$

Équation 29

Si x et X, et y et Y sont de signes identiques, alors T_y est positif, sinon T_y est bien évidemment négatif.

Il ne reste plus maintenant qu'à calculer les coefficients de la matrice de rotation. Les coefficients r_1 , r_2 , r_4 et r_5 sont déjà connus grâce à a1, a2, a4 et a5. La matrice de rotation étant une matrice orthonormale, le calcul des coefficients des deux premières lignes s'effectue de façon très rapide en tenant compte de la norme des lignes et de leur orthogonalité. La dernière d'entre elles est calculée en effectuant le produit scalaire avec les deux premières et en utilisant le fait que la matrice de rotation possède un déterminant valant ± 1. On obtient donc la matrice de rotation suivante pour une focale effective f positive (la détermination préalable du signe de la focale effective se fait par résolution du système surdéterminé décrit dans la seconde étape):

$$R = \begin{pmatrix} r_1 & r_2 & \sqrt{1-r_1^2-r_2^2} \\ r_4 & r_5 & s\sqrt{1-r_4^2-r_5^2} \\ r_7 & r_8 & r_9 \end{pmatrix} = \begin{pmatrix} L_1 \\ L_2 \\ L_3 \end{pmatrix}$$

Équation 30

avec $s=-sgn(r_1r_4+r_2r_5)$, $L_i.L_j^T=\delta_{i,j}$ pour $i,j\in\{1,2,3\}$ et $\det(R)=\pm1$

Si la focale effective est négative, la matrice de rotation s'obtient en fonction de la matrice précédente par l'expression :

$$R=\begin{pmatrix} r_1 & r_2 & -\sqrt{1-r_1^2-r_2^2} \\ r_4 & r_5 & -s\sqrt{1-r_4^2-r_5^2} \\ -r_7 & -r_8 & r_9 \end{pmatrix}$$

Équation 31

Seconde étape

Cette seconde étape sert à déterminer les coefficients *Tz* de la translation et la focale effective *f*. En considérant que les coefficients de distorsion sont nuls, on obtient le système surdéterminé suivant dans lequel *Tz* et *f* sont les inconnues et (X_i, Y_i) sont les coordonnées des points d'étalonnage:

$$(y_i \quad -d_y.Y_i).\begin{pmatrix} f \\ T_z \end{pmatrix} = (r_7.x_{wi}+r_8.y_{wi}).d_y.Y_i$$

Équation 32

avec $y_i=r_4.x_{wi}+r_5.y_{wi}+T_y$

Cette résolution nous permet d'obtenir une estimation des paramètres *Tx* et *f*. Cette estimation permet de connaître, dans un premier temps, le signe de *f* et donc de choisir la forme de la matrice de rotation. Il faut ensuite bien évidemment calculer les solutions exactes de *Tx* et de *f* ainsi que celles de κ_1 et κ_2, les deux paramètres de distorsion. Pour effectuer ce calcul, on résout l'équation suivante en utilisant un algorithme standard d'optimisation avec *Tx*, *f*, κ_1 et κ_2 comme inconnues. On prend, de plus, comme hypothèses de départ, les approximations calculées précédemment pour *Tx* et *f*, et 0 pour les coefficients de distorsion.

$$d_y.Y+d_y.Y.(\kappa_1.r^2+\kappa_2.r^4) = f.\frac{(r_4.x_w+r_5.y_w+r_6.z_w+T_y)}{(r_7.x_w+r_8.y_w+r_9.z_w+T_z)}$$

Équation 33

$$r = \sqrt{\left(\frac{d'_x}{s_x}.X\right)^2 + (d_y.Y)^2}$$

Équation 34

avec X= Xf-Cx , Y=Yf-Cy .

10.4. L'intégration temporelle : exemples de la littérature

10.4.1. Introduction

L'intégration du temps dans les paramètres de perception est indispensable à la fois pour prédire les déplacements des cibles à localiser et suivre, et pour assurer la continuité dans les perceptions.

L'intérêt de connaître les déplacements du robot entre deux positions à localiser est double. Il s'agit, d'une part, de limiter les recherches dans les positions possibles du robot en prenant en compte le déplacement et la ou les positions précédemment évaluées. D'autre part, ceci permet de conserver un historique des déplacements et donc de construire une trajectoire dans l'environnement.

L'ajout de la prise en compte des informations sur le déplacement du robot peut être réalisé au travers de plusieurs méthodes.

Dans un premier temps, il est envisageable d'utiliser des capteurs supplémentaires pour estimer les déplacements réels entre deux actions de localisation successives. Dans ce cadre, les capteurs odométriques semblent être les plus adaptés puisqu'ils sont destinés à évaluer le nombre de rotations des roues.

Une autre solution consiste à modéliser la cinématique du robot mobile, soit de façon continue en posant les équations différentielles du mouvement et en les intégrant, soit de façon discrète en liant les actions du robot aux changements de position possibles.

Enfin, dans ces deux cas, il est possible d'intégrer les aléas de déplacement en introduisant des incertitudes probabilistes, soit dans l'expression d'équations différentielles, soit dans les transitions entre les positions possibles.

Les approches continues sont généralement employées dans la détection, par traitement systématique, d'entités mobiles dans une succession d'acquisitions, suivi d'un véhicule dans séquence d'images par exemple ([Koller93], [Koller94] et [Marchand01]). Les approches discrètes sont, quant à elles, réservées aux approches dont la localisation consiste en la détermination d'un couple position/orientation parmi un ensemble fini de possibilités ([Laroche99]).

10.4.2. Méthodes utilisées en traitement du signal

10.4.2.1. Filtre de Kalman

Le passage de la localisation simple à la détermination des paramètres d'une trajectoire requiert l'utilisation d'une séquence d'acquisition à la place d'acquisitions simples. Le principe général du filtre de Kalman est de prédire la position d'un objet dans une séquence de perception en utilisant une position initiale, déterminée par des méthodes classiques analogues à celle vues dans le chapitre précédent, et un modèle de déplacement paramètré et modifiable.

Dans le cadre d'un suivi à partir d'une séquence d'images, il s'agit de détecter la trajectoire de primitives caractéristiques des entités mobiles (segments de droite, régions, textures, objets complexes, ...). Le filtre de Kalman permet de résoudre le problème du filtrage dans le domaine temporel. Un objet à suivre est caractérisé par un vecteur d'état, comprenant, en autre, la position et les paramètres du mouvement, et par une modélisation des processus aléatoires pouvant influer sur le déplacement. Ces derniers sont en général modélisés par un bruit blanc gaussien. Le suivi s'effectue par estimation des vecteurs d'états successifs de façon récursive.

La méthode du filtre de Kalman est en général décomposé en 5 phases [Giai95] :

- Assignation d'un modèle cinématique à chaque primitive
- Prédiction de la position des primitives dans l'image suivante en utilisant les paramètres du modèle cinématique avec une certaine incertitude.
- Utilisation de l'incertitude sur la prédiction pour déterminer une zone de recherche.
- Recherche du meilleur appariement possible en utilisant une distance normalisée de type distance de Mahalanobis.
- Utilisation des nouvelles mesures pour mettre à jour les paramètres du modèle cinématique.

Le filtre de Kalman est très utilisé en traitement de séquences d'images et plus particulièrement en robotique.[Roumeliotis99]

10.4.2.2. Flot Optique

Le flot optique est une méthode plus précise mais également plus coûteuse en temps de calcul que le filtre de Kalman classique. La principale différence entre les deux approches est la granularité des objets suivis. En effet, plutôt que de chercher un modèle cinématique adapté à chacune des primitives intéressantes, il s'agit ici de paramétrer le déplacement de chacun des pixels de l'image 2D. Ces paramètres sont en général représentés sous la forme d'une carte de déplacement ([DeCarlo00]) ou carte de

mouvement sur laquelle est représenté un vecteur de déplacement pour chacun des pixels en mouvement. (cf. figure 92)

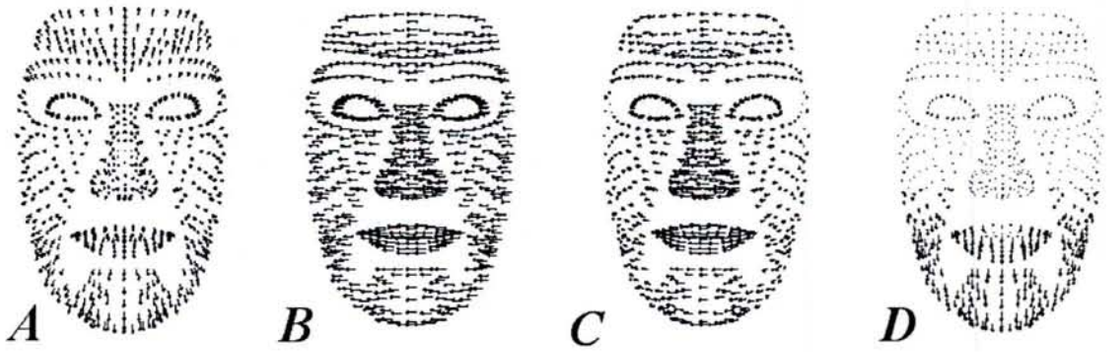

figure 92 : Représentation de carte de mouvement de flot optique pour des mouvements de visage (d'après [DeCarlo00]). *A* : Translation vers l'avant, *B* : Translation à droite, *C* :Rotation vers la droite, *D* :Ouverture de la bouche.

Le flot optique d'un pixel d'une image 2D peut être considéré comme le développement de celui-ci au cours du temps [Gaia95]. Pour des primitives tels que des segments, le flot optique peut être paramétré par deux cartes de mouvement β et υ. La carte β représente les composantes du mouvement normal au segment et υ représente la carte des variations d'orientation du segment (cf. figure 93). Cette paramétrisation est bien évidemment généralisable à d'autres types de primitives.

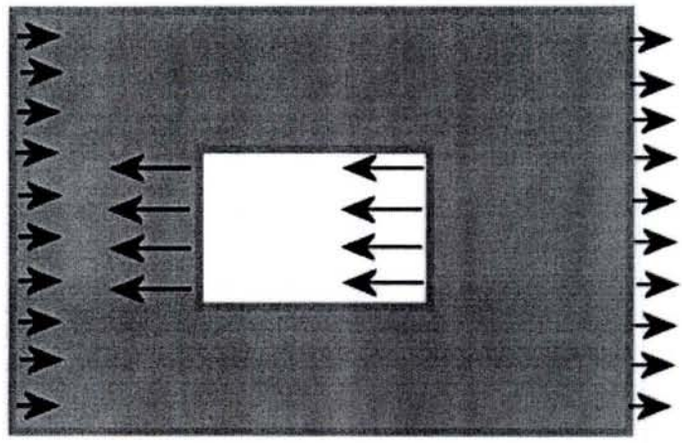

figure 93 : Exemple de carte de mouvement β dans le cas où deux mobiles se croisent

Le flot optique est couramment utilisé dans de nombreuses applications, en particulier des applications de surveillance de flux de circulation ([Mück00], [Nagel98]). Cette méthode est également utilisée dans la détermination des déplacements du dispositif d'acquisition en particulier lorsque celui-ci est placé sur un robot mobile. [Dev97]

10.4.2.3. Les approches Markoviennes

Les approches Markoviennes en robotique mobile sont assez répandues dans la littérature ([Simmons95] et [Fox98]). Ce type d'approche permet d'évaluer la position la plus probable du robot en prenant en compte la nature de ses déplacements En outre, il existe, dans la littérature des algorithmes, très élaborées utilisant le principe de fonctionnement que les approches probabilistes. Parmi celles-ci, nous pouvons citer les méthodes développées par Thrun et all. [Thrun99] utilisant des Processus Décisionnels de Markov couplé à un raisonnement de type algorithme de Monte-Carlo. Dans ce paragraphe, nous avons choisi de détailler la méthode POMDP simple car elle semble plus conforme à notre problématique dans la mesure où elle s'adapte à des situations où l'environnement est partiellement connu, alors que la méthode utilisant les MCMDP (Monte Carlo Markov Decision Processes) correspond plus à une stratégie d'exploration et de cartographie.

Un Processus de Décision Markovien (MDP) est un automate probabiliste : un état de départ et une action donnée peuvent aboutir à plusieurs états différents selon une distribution de probabilités. Un MDP est donc définit de la façon suivante :

- S, un ensemble fini d'états (Dans notre cas correspond à une localisation du robot (position et orientation).
- A, un ensemble fini d'actions (Dans notre cas : avance, tourne à droite et tourne à gauche).
- T[s'| s,a], une distribution de probabilités.

Le résultat d'une action est déterminé par la matrice T qui donne la probabilité du système à se trouver dans l'état ***s'*** sachant que la position de départ était ***s*** et que l'action ***a*** a été effectuée. Dans la formulation MDP, le système connaît toujours son état courant. Malheureusement, cette formulation ne convient à notre problème étant donné que nous cherchons à localiser le robot dans son environnement, nous ne pouvons donc pas connaître sa position à chaque instant. Nous devons donc utiliser une formulation plus générale : les POMDP (Processus de Décision Markoviens Partiellement Observables).

Dans ce cas la position actuelle du robot n'est pas connue mais nous disposons d'une nouvelle information sous la forme d'une observation représentative de la position courante.

Un POMDP est donc défini de la façon suivante :

Un MDP (un ensemble d'état S, un ensemble d'actions A et une distribution de probabilité T)

O, un ensemble d'observation.

P[o|s], une distribution représentant la probabilité d'effectuer l'observation o en se trouvant dans l'état s.

Ce modèle fournit des estimations de la position du robot sous la forme de *belief states* (ou états de croyance) qui sont calculés en prenant en compte l'état précédent, l'action effectuée et l'observation réalisée selon la loi décrite par l'équation ci-dessous. Les états de croyance représentent la probabilité, à chaque pas de temps t, de se trouver dans l'état s. Ils sont calculés en prenant en compte les actions et les observations effectuées.

$$B(t,s)=K\times(\sum_{s'\in S}T(s,s',a)\times B(t-1,s'))\times P(o,s)$$

Équation 35

Cette loi peut être décomposée en deux parties :

La première partie, ($\sum_{s'\in S}T(s',s,a)\times B(t-1,s')$), représente l'évolution des *belief states* en prenant en compte les actions effectuées par le véhicule. Elle consiste en l'estimation de la probabilité de suivre une transition entre un état inconnu et l'état ***s***. La probabilité de suivre la transition de ***s'*** vers ***s*** est égale à la probabilité d'être dans l'état ***s'*** au temps ***t-1*** multipliée par la probabilité d'effectuer la transition vers ***s*** sachant que le système se trouve dans la position ***s'***. Il s'agit ensuite d'effectuer la somme sous l'ensemble des états s' possibles.

La seconde partie $P(o,s)$ est une mise à jour du *belief state* en utilisant l'observation réalisée.

Le facteur K est un facteur de normalisation destiné à s'assurer que la somme des probabilités vaut 1.

S.C.D. - U.H.P. NANCY 1
BIBLIOTHÈQUE DES SCIENCES
Rue du Jardin Botanique - BP 11
54601 VILLERS-LES-NANCY Cédex

10.5. Le robot « Simplet »

Le robot « Simplet » a été conçu et construit directement au sein de l'équipe MAIA. Sa conception a pour but d'éviter les problèmes rencontrés avec le robot « Gaston ». En particulier nous voulions maîtriser l'ensemble de ses composants pour faciliter les opérations de maintenance. Nous voulions également pouvoir envisager l'utilisation de plusieurs robots de façon simultanée, ce qui implique l'obligation d'un coût de construction faible. De plus, l'architecture de ce robot se devait d'être modulaire pour permettre des ajouts de capteurs et/ou d'actionneurs tout en conservant la même architecture.

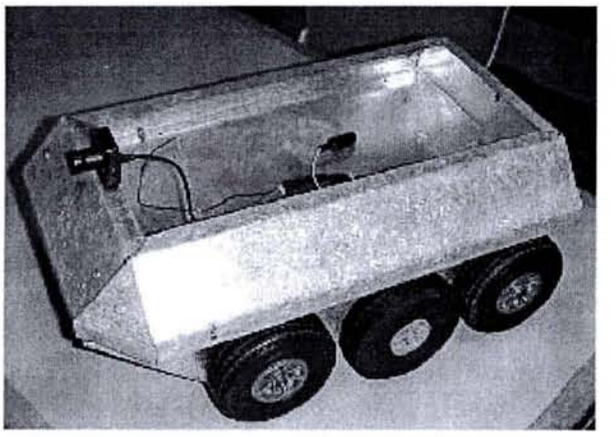

figure 94 : Le robot "Simplet" (Premier Prototype)

Ainsi, la structure du robot est agencée en plusieurs éléments que l'on peut assembler à loisir. Le robot est alors décomposé en plusieurs étages correspondant chacun à une tâche donnée (Propulsion, transmission HF, un étage par type de capteurs).

« Simplet » est un robot de type char à 6 roues, dont deux motrices, qui a la possibilité d'effectuer des rotations sur place. Sa partie propulsion est assurée par deux moteurs pas à pas monter avec des réducteurs. Le contrôle est réalisé par l'intermédiaire d'un microcontrôleur Phillips^{tm} XA-S3© dérivé du processeur Intel^{tm} A51© mais avec des convertisseurs analogiques digitaux supplémentaires.

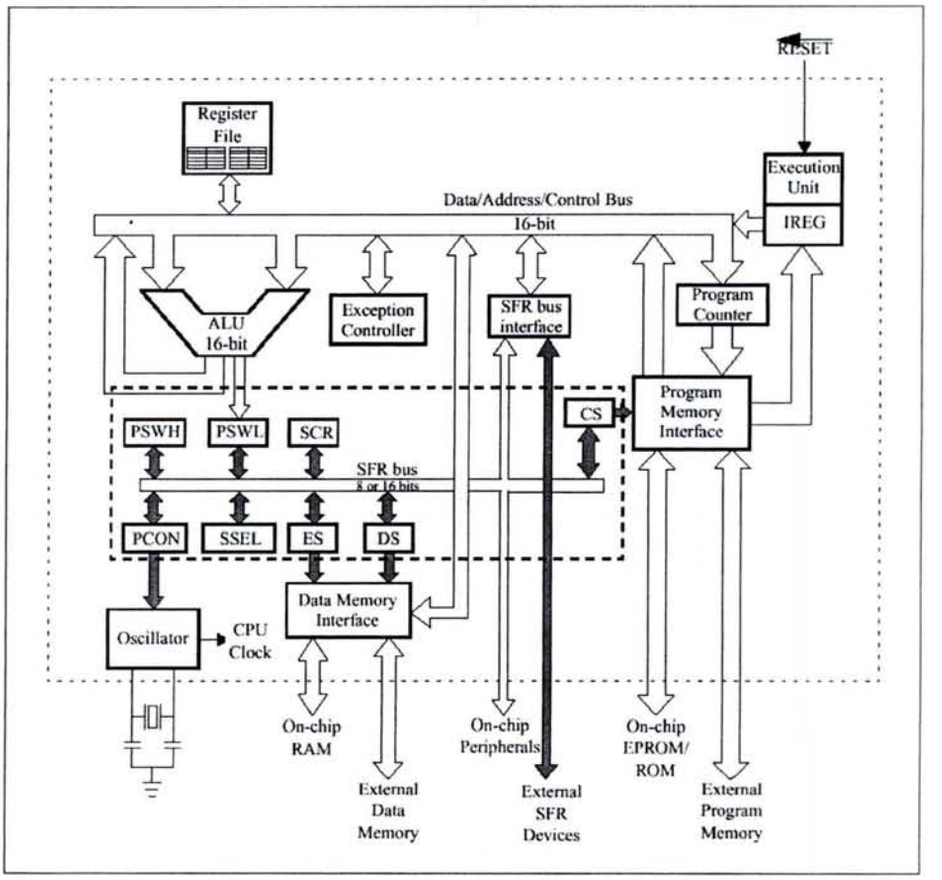

figure 95 : Structure du processeur XA-S3

La transmission d'ordre s'effectue actuellement via un réseau ethernet sans fil. Ce type de transmission nous oblige à avoir recours à un ordinateur portable sur le robot. Cette solution n'est que temporaire puisque nous avons prévu d'effectuer cette transmission directement à partir de couples de barrettes de transmission HF.

La perception est effectuée par plusieurs dispositifs en cours de développement. D'une part, le robot dispose de capteurs de contacts de type moustache qui permettent le traitement de comportements d'urgence. D'autre part, il dispose également de capteurs infrarouges qui ont pour but d'évaluer la distance des obstacles proche du robot. Enfin, « Simplet » dispose également d'une caméra CCD couleur miniature accouplée à un transmetteur HF fonctionnant à une fréquence de l'ordre du GigaHertz.

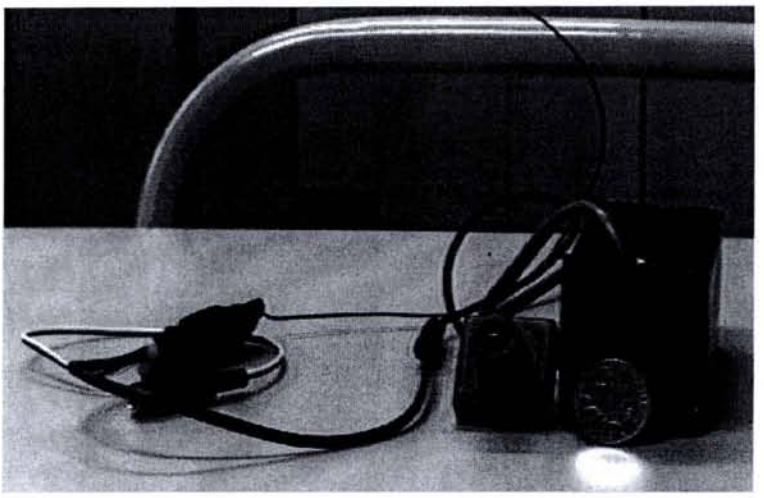

figure 96 : La caméra miniature et son transmetteur

L'acquisition des images s'effectue sur un PC équipé d'une carte d'acquisition du commerce.

Le contrôle et la perception du robot peuvent être déportés sur n'importe qu'elle machine du laboratoire grâce à deux couples de client/serveur Ethernet.

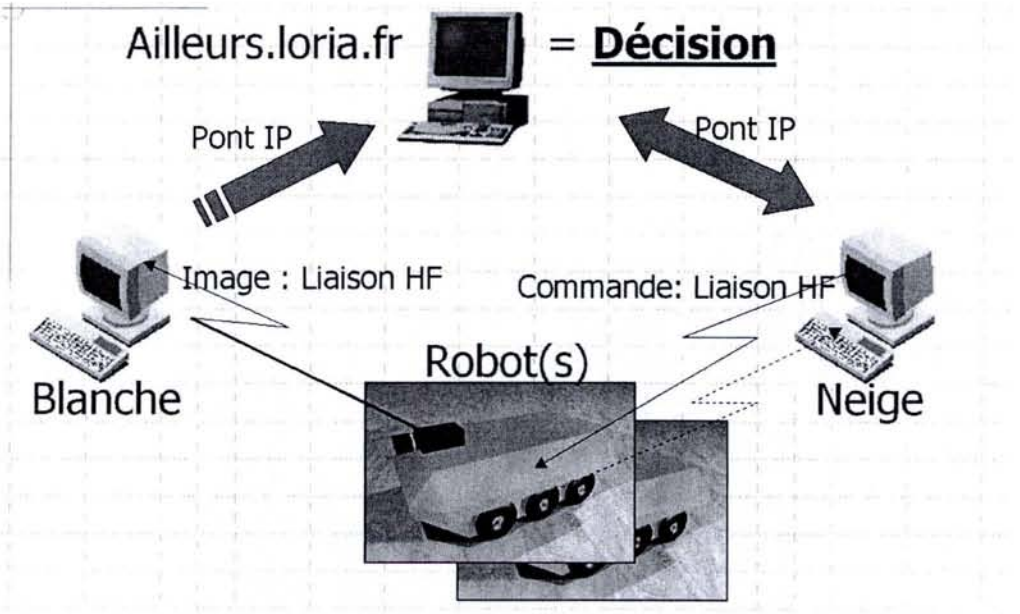

figure 97 : Contrôle et perception à travers le réseau du LORIA

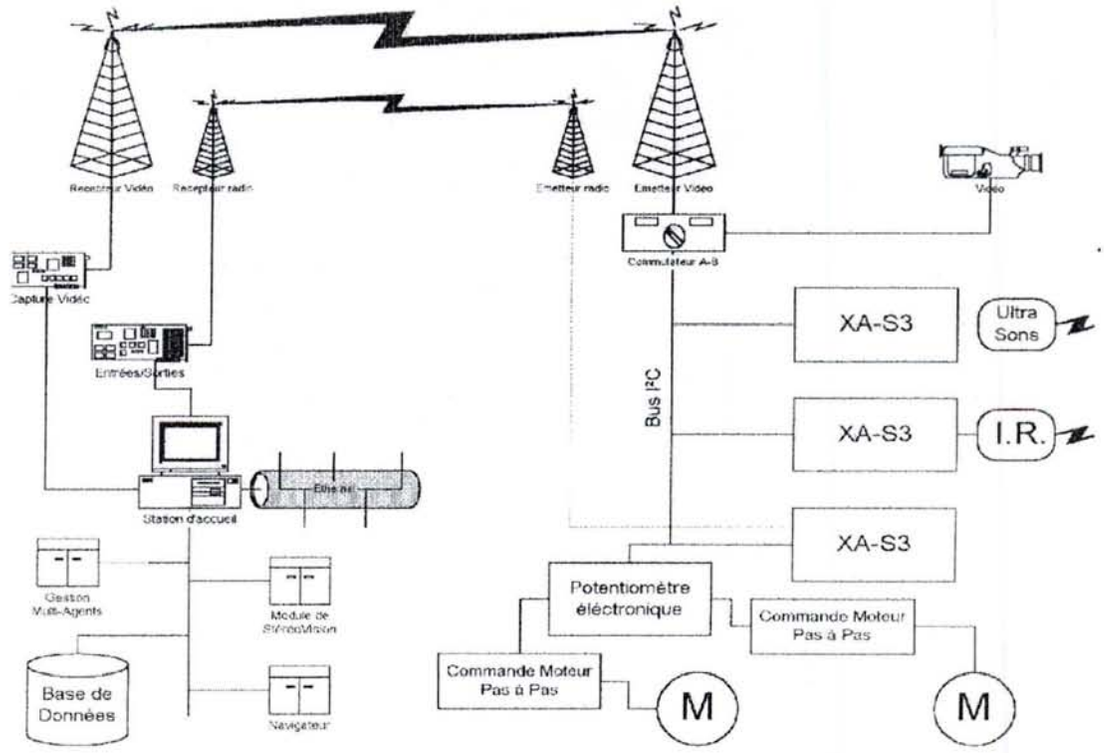

figure 98 : Architecture complète permettant le contrôle du robot "Simplet"

Le robot « Simplet » est toujours en cours de développement. Un nouveau prototype à été construit afin de pouvoir placer les réducteurs sur les moteurs pas à pas. L'ancien châssis va sûrement migrer vers un autre type de propulsion à base de moteur à courant continu contrôlés par des cartes qui seront communes avec celles utilisées dans le robot « Gaston ». La transmission HF et les capteurs infrarouges sont également en cours de développement.

figure 99 : "Simplet" version 2.0

Modèle multi-agent réactif pour la fusion de percepts : application à la localisation et au suivi en robotique mobile

Ce travail s'inscrit dans un mouvement récent, les Systèmes Multi-Agents dont l'un des buts est de montrer que des systèmes intelligents complexes peuvent être élaborés à partir d'éléments réactifs simples et autonomes interagissant entre eux et avec leur environnement.
L'objectif de cette thèse est de mettre en oeuvre ce principe de conception de systèmes intelligents à un cadre applicatif complexe. Ainsi, nous voulons montrer qu'il est possible de construire un système multi-agents situés et réactifs devant résoudre un problème complexe concret inscrit dans une démarche d'interprétation du monde réel à travers ses caractéristiques et sa dynamique.
En ce qui concerne le cadre applicatif, nous avons choisi de traiter un problème récurrent en robotique mobile : la localisation. Nous voulons proposer une architecture permettant de localiser et de suivre une population hétérogène ou homogène de robots mobiles à l'aide de percepts fournis par des capteurs dont le type, le nombre et la position ne sont pas définis *a priori*.
L'enjeu d'une telle approche est la conception d'un dispositif intelligent d'un point de vue global à partir d'agents aux comportements réactifs simples. En effet, l'intelligence d'une tel système peut être considérée comme un phénomène émergent dû à l'organisation spatio-temporelle issue des interactions. Le point clef est alors la conception et la mise au point des comportements élémentaires afin d'obtenir un résultat émergent correspondant aux caractéristiques désirées.
Notre travail porte à la fois sur la conception d'une architecture adaptée : formalisation du problème via l'environnement, définition du processus de résolution, interprétation des résultats et sur son évaluation en simulation et dans un cadre réel.

Mots-clés : Système Multi-Agent, Agents Réactifs et Situés, Localisation, Suivi, Robotique Mobile, Fusion

Reactive multi-agent model for data fusion : application to the localization and tracking problem in mobile robotics.

The aim of this work is to show how a complex system, that has to resolve a real task, can be designed from single entities with small decision abilities. This approach stems from the multi-agent systems and, more particularly, from the reactive-agent systems.
As for the real task, we choose to deal with the localization and tracking problem. The localization is a tough, wide spread task in mobile robotics. It can be defined as the action to determine the position of an object considering a reference coordinate system.
By contrast with the classical algorithms, which usually stem from the signal or the image processing, we propose a reactive architecture based on interaction phenomenon inspired by Physics.
The aim is to design a localization and tracking device able to focus on a variable number of targets that are moving in a structured and partially known environment. Furthermore, the device has to be able to accomplish its task by using a variable number of sensors, static or mobile in the environment.
Our works deals with both the design of the reactive-agent architecture (choice of the world representation, definition of the resolution process, interpretation of the resulting spatio-temporal organization) and its application and evaluation with real robots.

Keywords : Multi-Agent System, Reactive and Situated Agents, Localization, Tracking, Mobile Robotics, Fusion