Mémoire d'Habilitation à Diriger des Recherches

Présenté à l'Université d'Evry Val d'Essonne

Par

Hichem MAAREF

TRAITEMENT DE DONNEES IMPARFAITES DANS LE CADRE DE LA THEORIE DU FLOU : CONTRIBUTION A LA NAVIGATION ET A LA LOCALISATION D'UN ROBOT MOBILE

Soutenu le 24 Janvier 2000 devant la commission d'examen composée par :

Rapporteurs :	Pr. Pierre Borne, LAIL Lille
	Pr. Pierre-Yves Glorennec, IRISA Rennes
	Pr. Florent Chavand, CEMIF-LSC Evry
Examinateurs :	Pr. Laurent Foulloy, LAMII Annecy
	Pr. Dominique Meizel, UTC compiègne
	Pr. Joris De Schutter, PMA Louvain
	Pr. Claude Barret, CEMIF-LSC Evry

A la mémoire de mon père
à ma mère
à mon épouse leïla et mes filles hayfa, cérine, fetna et nehla
à mes frères
en témoignage de ma reconnaissance.

Remerciements

Les travaux présentés dans ce mémoire ont été effectués au sein de l'équipe de recherche portant la dénomination récente de CEMIF-Laboratoire systèmes complexes, dirigée par Monsieur Florent Chavand que je tiens à remercier pour ses discussions précieuses et d'être un des rapporteurs de ce mémoire.

Je remercie vivement Messieurs Pierre Borne, Professeur, Directeur Scientifique de l'Ecole Centrale de Lille, d'avoir accepté d'évaluer ce travail en tant que rapporteur et Pierre-Yves Glorennec, Professeur à l'INSA de Rennes, pour ses discussions, ses remarques pertinentes et d'avoir bien voulu faire partie des rapporteurs.

Je tiens à remercier Messieurs Dominique Meizel, Professeur à l'université de Compiègne, Laurent Foulloy, Professeur à l'université de Savoie et Joris De Schutter, Professeur à l'université catholique de Louvain, d'avoir accepté d'être examinateurs.

Enfin, Monsieur Claude Barret, Professeur à l'université d'Evry, partenaire de recherche de toujours, qu'il trouve ici l'expression de toute ma gratitude et de ma profonde reconnaissance.

Que soient également remerciées toutes les personnes, thésards, stagiaires et membres du laboratoire, qui ont participé au bon déroulement de ce travail que ce soit par leur implication scientifique, leur aide technique ou leur soutien moral.

J'adresse aussi mes vifs remerciements à tous mes collègues et amis que j'ai souvent sollicité au cours de ces dernières années.

Table des matières

Notice individuelle _____ *1*

1- Curriculum vitae _____ **1**

2- Titres universitaires _____ **1**

3- Activités de recherches _____ **1**

4- Activités d'enseignement _____ **6**

5- Responsabilités collectives _____ **6**

5-1-Responsabilité dans le cadre du département GEII :_____ 6

5-2-Activités dans le cadre du laboratoire_____ 6

6- Liste des publications _____ **7**

6-1- Ouvrage _____ 7

6-2- Revues nationales et internationales _____ 7

6-3- Participation à des Congrès avec comité de lecture _____ 7
6-3-1- Participations affichées : _____ 7
6-3-2- Participations orales : _____ 8

6-4- Conférences avec rapports internes:_____ 10

6-5- Conférences sans actes_____ 10

6-6- Rapport interne _____ 10

6-7- Contrats _____ 10

7-Encadrement _____ **11**

7-1-Stages _____ 11
7-1-1-DUT _____ 11
7-1-2-Maîtrise _____ 11
7-1-3-Elèves-ingénieurs _____ 11
7-1-4-DEA _____ 11

7-2-Thèses _____ 11

8-Divers _____ **12**

Chapitre 0 : Introduction _____ *13*

Chapitre I : Optimisation d'un système d'inférence floue _____ *15*

1-Introduction _____ **15**

2- La commande floue _____ **16**

3-Optimisation d'un système d'inférence floue (SIF) _____ **16**

3-1-Problèmes posés _____ 16

3-2- Solutions proposées _____ 17
3-3- Approche dite "neuro-floue" _____ 17

4-Description de la technique d'optimisation adoptée _____18

4-1-Principe _____18

4-2- Réalisation des fonctions d'appartenance _____19

4-3- Calcul des valeurs de vérité _____19

4-4- Algorithme d'apprentissage _____20

5-Application au contrôle _____21

5-1- Architecture de contrôle JEAN _____21

5-2- Architecture de contrôle simplifié mini-JEAN _____21

5-3- Exemple d'application _____22
5-3-1- Présentation du problème _____22
5-3-2- Première expérience _____22
5-3-3- Introduction d'une régression des poids de sortie _____23
5-3-4- Optimisation avec introduction de connaissances _____25
5-3-5- Optimisation complète du SIF _____26

6-Conclusion _____27

Bibliographie I _____ 28

Chapitre II : Aides floues à la navigation d'un robot mobile _____ 29

1-Introduction _____29

2- Robots mobiles utilisés dans le cadre de nos travaux et hypothèses _____30

2-1- Le robot miniature Khepera® _____30

2-2- Simulateur de Khepera _____30

2-3- Le robot RMI _____31

2-4-Hypothèses et contraintes _____31

3- Présentation de la stratégie adoptée _____32

4- Méthode de planification d'itinéraire adoptée dans un environnement connu _____33

4-1- Graphe de visibilité _____33

4-2- Algorithme A* _____33

4-3- Suivi du chemin planifié par contrôle flou _____34
4-3-1- Architecture du système de suivi de trajectoire _____34
4-3-2- Contrôleur flou _____35

4- Algorithme de navigation locale _____36

4-1- Navigation locale par champ de potentiels artificiels _____36
4-1-1- Limitation de la méthode classique _____36
4-1-2- Champ de potentiel artificiel à coefficients adaptatifs _____38

4-2- Navigation locale par combinaison de comportements réactifs élémentaires _____40
4-2-1- Introduction _____40
4-2-2- Navigation sans obstacle concave _____40
4-2-2-1- Suivi du milieu de l'espace libre _____40
4-2-2-2- Convergence vers le but _____41
4-2-2-3- Fusion des deux comportements : Stratégie S1 _____42
4-2-2-4- Application _____42

4-2-3- Navigation avec obstacles concaves 45
4-2-3-1- Suivi de mur 45
4-2-3-2- Expérimentation 46
4-2-4- Stratégie complète de navigation réactive : Stratégie S2 46

5-Fusion des stratégies locale et globale 47

5-1- Principe 48

5-2- Module de prise de décision 48

5-3- Résultats 49

6- Optimisation des modules flous de navigation réactive 51

6-1- Introduction 51

6-2- Principe 51

6-3- Construction du comportement " Suivi du mur " 52
6-3-1- Génération des règles de contrôle de la vitesse angulaire 52
6-3-2- Génération des règles de contrôle de la vitesse linéaire 53
6-3-3- Réaction du SIF optimisé face à des situations nouvelles 54
6-3-4- Méthode des moindres carrés récursifs 56

7-Conclusion 57

Bibliographie II 59

Chapitre III : Localisation absolue par fusion multisensorielle basée sur la théorie des possibilités 60

1- Introduction 60

2- Fusion multisensorielle en robotique mobile 60

3- Fusion d'informations en théorie des possibilités 62

3-1- Combinaison adaptative 62
3-1-1- Règle CADP 62
3-1-2-Généralisation de la règle CADP 63

4- Application et comparaison avec la méthode de Kalman 64

4-1- Cadre expérimental 64

4-2- Principe général de la solution adoptée 66
4-2-1- Association 67
4-2-2-Alignement (solutions locales) 67
4-2-3- Combinaison (solution globale) 68
4-2-4- Modélisation possibiliste des informations capteurs 68
4-2-4-1-Capteur odométrique 68
4-2-4-2- Capteur infrarouge 70

4-3-Résultats des tests et comparaison avec la méthode de Kalman 71
4-3-1-Tests en fin de parcours 71
4-3-2-Simulation 71
4-3-3-Expériences réelles 72
4-3-4- Entrées du système 73
4-3-5- Fonctionnement du système 74
4-3-6- Sorties du système 74
4-3-7 - Autres remarques 74

5-Modification de la règle CADP 75

5-1- Robustesse de la règle CADP 75

5-2- Modification du choix du paramètre m 76

5-3- Obtention d'une règle progressive (RP) 76

6- Application de la règle progressive dans une expérience de localisation 79

6-1- Cadre expérimental 79

6-2- Principe général de la solution adoptée 79
6-2-1- Modélisation des données capteurs 80
6-2-1-1-Capteur ultrasonore 80
6-2-1-2- Capteur odométrique 80
6-2-2- Alignement des données dans le même repère 81
6-2-3-Obtention des solutions locales 81
6-2-4-Solution globale 82

6-3-Résultats 82

6-4- Influence de la modélisation des capteurs 84
6-4-1- Influence de la forme de la distribution 85
6-4-2- Influence de l'étendue 85

7- Conclusion 86

Bibliographie III *88*

Conclusions et perspectives *89*

Notice individuelle

1- Curriculum vitae

MAAREF Hichem
né le 04 juillet 1960 à Mahdia (Tunisie)
Nationalité française
Marié, quatre enfants
Adresse personnelle :
16, avenue Léon Blum
94700 Maisons-Alfort
Tél. : 01.45.18.17.57
Adresse professionnelle :
Université E.V.E, IUT Dept. GEII
22, allée Jean Rostand
91011 Evry Cedex
Tél. : 01.69.47.72.71
e-mail : maaref@cemif.univ-evry.fr

CEMIF-Groupe Systèmes Complexes
40, rue du Pelvoux
91020 Evry Cedex
Tél. : 01.69.47.75.54

2- Titres universitaires

*DEUST Physique/Chimie obtenu en 1982 à la Faculté des sciences et techniques de Monastir (Tunisie).
*Maîtrise EEA obtenue en 1984 à la Faculté des sciences et techniques de Monastir.

*D.E.A. "Chimie-Physique" obtenu à l'Université P. et M. Curie de PARIS VI en 1985 Mention Assez-bien.

*Doctorat de l'Université de PARIS VI soutenu le 29 juin 1990 avec la mention très honorable.
"Caractérisation électronique d'interfaces profondes Al/InP"
Jury : C. BONNELLE, Présidente,
A. VAPAILLE, Rapporteur,
M.BENSOUSSAN, Rapporteur,
C. BENOIT-A-LA GUILLAUME, Examinateur,
F. VERGAND, Examinatrice,
C. BARRET, Examinateur.

3- Activités de recherches

De janvier 1986 à juin 1990, j'ai préparé ma thèse de doctorat de l'université sur l'étude des interfaces métal-phosphure d'indium (InP) dans le cadre d'une collaboration scientifique entre le Laboratoire de Chimie Physique (LCP) de l'Université PARIS VI (Pr. C. Bonnelle) et l'Institut d'Électronique Fondamentale (IEF) de l'Université PARIS XI (Pr. A. Vapaille). En octobre 1990, j'ai été nommé ATER à l'Université Paris XII-IUT d'Evry. Je me suis trouvé alors devant la nécessité d'entreprendre une reconversion de thème de recherche, aucun laboratoire de l'Université Paris XII ne travaillant sur mon ancien domaine. Pour m'insérer dans le Laboratoire de Robotique d'Evry (LRE) et étant attiré par un axe de recherche novateur et très prometteur, j'ai participé en 1991 avec le Pr. C. Barret à la mise en place d'une nouvelle équipe du LRE travaillant sur le thème de l'application des techniques neuronales et de la logique floue à la commande de systèmes. En fait cette époque correspondait au "boum" de la logique floue au Japon, ce qui a motivé le démarrage ou la reprise de recherches dans ce domaine dans plusieurs laboratoires européens et nord-américains. J'ai d'ailleurs participé activement aux travaux du groupe "Commande Symbolique et Neuromimétique" du GR Automatique du CNRS crée en 1992 pour fédérer les travaux de recherche français sur ce sujet. Depuis septembre 1993, j'occupe un poste de Maître de Conférences en 61ème section toujours au sein du Département GEII-IUT faisant partie désormais de l'Université nouvelle d'Evry Val d'Essonne. A la suite de la création de cette nouvelle université le LRE a intégré le CEMIF (Centre d'Etude de Mécanique d'Ile de France) sous forme d'un pôle de recherche baptisé GSC (Groupe Systèmes Complexes). Mes travaux de recherche effectués au sein de ce groupe s'intéressent essentiellement au traitement de données imprécises et incertaines par la théorie du flou et ses applications à la robotique mobile. On entend par théorie du flou le cadre théorique regroupant la théorie des sous-ensembles flous, la logique floue et la théorie des possibilités qui ont des liens très forts. Cette théorie permet d'une part la modélisation de la connaissance humaine pour calquer les capacités d'experts humains à commander des

systèmes mal définis et/ou difficilement modélisables et d'autre part de traiter des informations imparfaites (imprécises et/ou incertaines) dans l'objectif de mieux les représenter et ensuite de les fusionner si nécessaire.

Je commence ci-après par décrire brièvement mon domaine de recherche initial. Ensuite, je présente une synthèse des travaux actuels qui seront détaillés à la suite de cette notice.

J'indiquerai pour chaque sujet et thème de recherche les publications correspondantes dont les références exactes se trouvent dans le paragraphe 6.

En effet, mes travaux de recherche lors de la préparation du doctorat concernent les contacts métal-semiconducteurs et en l'occurrence le phosphure d'indium (InP). Ce domaine avait d'une part un intérêt technologique pour la réalisation de dispositifs électroniques rapides et d'autre part jouait un rôle capital dans la compréhension physique des phénomènes gouvernant la formation de la barrière de Schottky. Les résultats concernant la hauteur de cette barrière et sa sensibilité à divers paramètres étaient quelque peu confus et contradictoires. Par exemple, l'idée de corrélation forte entre la hauteur de barrière et la réactivité du métal déposé avait été avancée par Williams et Brillson. Suivant ces auteurs, il y aurait deux types d'interfaces M-InP : ceux pour lesquels il y a réaction chimique entre le métal et le semiconducteur et qui ont une barrière nulle ou très faible et ceux pour lesquels cette réaction ne se produit pas et qui ont une barrière de l'ordre de 0,5 eV. Ceci a été contredit en particulier par Kendelewicz *et al.* qui n'observent pas de corrélation entre la chaleur de réaction entre différents métaux et InP et la hauteur de la barrière. Plus généralement, depuis des années, s'affrontaient différentes théories tentant d'expliquer la formation de la barrière de Schottky. La plupart faisaient appel à la notion d'état d'interface, mais un grand désaccord régnait sur l'origine de ces états. On peut citer par exemple la notion d'états dus à des défauts structuraux produits par la métallisation et celle d'états liés aux queues de fonctions d'onde des électrons du métal, pénétrant dans le semiconducteur. Les informations directes d'origine expérimentale sur ces états et leurs caractéristiques étaient rares.

Dans ce cadre, j'ai tenté d'effectuer une caractérisation poussée du contact Al-InP en couplant des méthodes électriques et d'analyse physico-chimique et en multipliant les méthodes de préparation des interfaces.
* Les méthodes de caractérisation électrique classiques (I-V, C-V) fournissent principalement comme information la valeur de la hauteur de barrière. Nous obtenons, pour le type N, une barrière comprise entre 0,35 et 0,5 eV en fonction de la méthode de préparation des interfaces utilisées. Par ailleurs, des anomalies de comportement des caractéristiques I-V, C-V sont mises en évidence pour un certain nombre d'échantillons. Des caractéristiques I-V non linéaires (en "N") sont attribuées à des inhomogénéïtés latérales de barrière. Ceci se produit particulièrement après clivage à l'air et après bombardement ionique. Des courbes C-V anormales sont observées sur ces mêmes types d'échantillon. Elles sont dues à une courbure de bande non parabolique provoquée par la pénétration à une distance importante de l'interface de défauts électriquement actifs.
* Une méthode originale développée à l'IEF sous le nom de Spectroscopie de Capacité Schottky (SCS) permet la caractérisation de certains états d'interface situés dans la bande interdite du semiconducteur. Pour les interfaces Al-InP, nous observons la présence d'états bien localisés en énergie et de densité sensible à la méthode de préparation. Deux de ces états ont pu être caractérisés complètement : A_n situé vers E_c - 0,37 eV et C_p situé vers E_V + 0,40 eV. La présence de continuum d'états est aussi mise en évidence, particulièrement dans la moitié inférieure du gap sur les surfaces (100). Ce continuum apparaît aussi, dans le cas du clivage à l'air, superposé aux niveaux localisés.
* La spectroscopie d'émission X induite par électron (EXES) développée au LCP permet d'obtenir les informations sur la structure électronique de l'interface entre un métal et un semi-conducteur. A la différence des autres méthodes telles que UPS, XPS... qui s'utilisent in situ sur les surfaces semiconductrices recouvertes de couches métalliques ultra minces (monocouche). EXES s'applique à des interfaces profondes où le métal (épaisseur de l'ordre de 100 A°) a effectivement des propriétés de matériau massif. L'emploi de cette méthode pour la caractérisation d'interfaces réelles était une première et ouvrait une voie prometteuse pour progresser dans la connaissance des mécanismes physicochimiques et l'évaluation de leur rôle dans la fixation de la barrière de Schottky, d'autant plus que la contrainte de préparation in situ des échantillons étant éliminée, on peut multiplier les méthodes de préparation. Dans le cas de Al-InP nous avons observé que :

i- l'interface clivée sous ultra-vide est la moins perturbée, à la limite de la méthode, c'est-à-dire quelques dizaines d'Angströms;

ii- pour les autres préparations, des perturbations fortes de l'interface se manifestent dans la zone étudiée (une centaine d'Å). Par exemple :
-après clivage à l'air la présence d'oxyde interfacial et d'inhomogénéités se manifeste ;
-les échantillons nettoyés chimiquement comportent une couche superficielle amorphisée ;

-le bombardement ionique élimine cette couche amorphe;

Par ailleurs, lorsque de l'oxyde est présent à la surface de l'InP, la métallisation d'aluminium provoque la réduction de cet oxyde et la création d'Al_2O_3 .

De la confrontation des différents résultats expérimentaux ressortaient deux points particulièrement importants:

1- Malgré la diversité des préparations de la surface et malgré les modifications physicochimiques qui en résultent et qui sont mises en évidence par EXES, la variation de la hauteur de barrière est faible. Il existe donc une cause très forte et commune de "pinning" sur tous les échantillons étudiés. Nous pensons qu'elle est due à la présence de l'état A_n en E_c - 0,37 eV observé à la fois sur les interfaces clivées et nettoyées chimiquement. L'origine physique de cet état reste à élucider mais il nous semble plus plausible de l'attribuer à des défauts au sens de Spicer plutôt qu'à des états induits par le métal du type "états de Heine". En effet, nous avons effectué par ailleurs des études comparatives des caractéristiques électriques et de la distribution des états d'interface pour quatre métaux (Al, Pd, Au, Ag) déposés sur des surfaces d'InP (110) type N clivé sous ultra-vide. Malgré la grande variété du comportement physico-chimique de ces différents métaux vis-à-vis de InP, les valeurs mesurées de la barrière de Schottky varient relativement peu : 0,37 eV pour Al ; 0,41 eV pour Au et Pd et 0,54 eV pour Ag. L'étude des états d'interface par SCS met clairement en évidence la présence de l'état A_n commun à toutes les interfaces. Les variations de barrière observées peuvent être liées à des modifications de la densité (et/ou du profil de distribution) de A_n. L'influence des autres états situés plus loin du niveau de Fermi peut aussi se manifester par des perturbations de la courbure de bandes au voisinage de l'interface.

2- Les résultats de EXES, les anomalies du comportement électrique (C-V en particulier) et dans une certaine mesure l'existence d'états d'interface en équilibre avec le semiconducteur conduisent à rejeter totalement l'idée d'une interface abrupte à l'échelle atomique. Parmi les échantillons étudiés, l'interface la moins diffuse est celle obtenue par clivage sous ultra-vide. Pour toutes les autres, nous sommes arrivés à la conclusion qu'il existe une zone interfaciale perturbée de plusieurs dizaines, voire plusieurs centaines d'Angströms.
Par ailleurs, le nettoyage par bombardement ionique a été effectué à l'aide d'un prototype expérimental baptisé ICSIII, développé par la Société VEECO qui permet l'emploi de faisceaux d'ions de très basse énergie (0 à 200 eV). L'emploi des sources d'ions classiques d'énergie de 0,1 à quelques KeV est impossible pour la surface d'InP qui, trop fragile, est décomposée par élimination de phosphore. Les techniques utilisées prouvent que pour des énergies inférieures à 50 eV, le modèle thermoionique reste valide et que l'élimination de la couche d'oxyde natif n'entraîne pas de défauts de structure catastrophiques, comme cela se produit en utilisant des ions de forte énergie.

Ces travaux m'ont permis de soutenir en juin 1990 une thèse de doctorat de l'Université Paris VI. Ils ont par ailleurs donné lieu à 6 publications dans des revues nationales et internationales et à 5 participations à des colloques et congrès.

Revues nationales et internationales	Congrès avec comité de lecture
6-2-1, 6-2-2, 6-2-3, 6-2-4, 6-2-5, 6-2-6	6-3-1-1, 6-3-1-2, 6-3-1-3, 6-3-2-1, 6-3-2-2

Depuis octobre 1990, j'ai intégré le Laboratoire de Robotique d'Evry (LRE). Dans un premier temps j'ai été amené à réaliser des outils de simulation permettant d'utiliser la logique floue dans la commande de processus et à proposer des architectures de contrôle-commande en les testant sur des exemples canoniques. Ceci fait partie de la démarche entreprise par le groupe de recherche "Commande Symbolique et Neuromimétique". Nous avons ensuite appliqué les systèmes d'inférence floue en robotique mobile telles que le contrôle du glissement d'un engin mobile et la génération de trajectoire de robots mobiles dans des environnements d'intérieurs partiellement connus. En effet, le contrôle des glissements présente un intérêt certain pour les véhicules terrestres, et notamment les véhicules tout terrain. Il devient indispensable d'équiper le véhicule d'un système embarqué qui lui permette de prendre les décisions nécessaires. Le contrôleur flou a pour objet de contrôler les glissements sol-roue en conservant en priorité le cap du véhicule, éventuellement au détriment de la vitesse. D'après les tests effectués (suivi de trajectoire rectiligne et circulaire traversant des types de sols différents), la logique floue apporte une réponse satisfaisante à ce problème.

Congrès avec comité de lecture
6-3-1-4, 6-3-2-3, 6-3-2-4, 6-3-2-5, 6-3-2-6

En ce qui concerne le contrôle de systèmes en robotique mobile, plusieurs méthodes ont été développées et elles sont généralement classées en deux catégories : la planification globale et le contrôle local. L'utilisation de l'une seulement de ces deux stratégies pour un contrôle du mouvement d'un robot mobile dans un environnement complexe s'avère insuffisante et hasardeuse car viennent se greffer sur leurs inconvénients intrinsèques maintes contraintes qui sont typiquement causées par :

- le manque et l'imprécision des connaissances dans notre compréhension du système ; c'est-à-dire que nous sommes incapables de générer des descriptions exactes et complètes de tous les phénomènes qui contribuent au comportement du système et de l'environnement;

- la difficulté d'avoir correctement la représentation de l'environnement et l'estimation de position du robot à cause des erreurs de mesure des capteurs qui sont loin d'être parfaits et qui conduisent à des imprécisions et à des incertitudes.
Pour se rapprocher au mieux d'un comportement humain, le contrôle d'une manière efficace du mouvement d'un robot mobile dans un environnement complexe nécessite forcément une stratégie hiérarchisée :

-un niveau haut de planification de chemin utilisant une description totale voire incomplète ou/et imparfaite du modèle du monde;

-un niveau local tel que le déplacement du robot est basé complètement sur les informations données par les différents capteurs qui couvrent l'entourage proche du véhicule.

La stratégie retenue [Thèse M. Benreguieg] pour contrôler le mouvement d'un robot mobile dans un environnement partiellement connu, est basée sur une méthode hybride pour exploiter les avantages de stratégies de navigation globale et locale. La connaissance à priori de l'environnement est mémorisée afin d'une part de générer un chemin global et d'autre part de pouvoir comparer la scène réelle à la scène mémorisée. Si les données capteurs sont identiques dans les deux scènes, la navigation est réalisée à vitesse maximum grâce à un contrôleur flou. Sinon, le robot se trouve dans une scène localement inconnue. Alors deux méthodes d'actions réflexes étudiées parallèlement n'utilisant que les informations capteurs sont appliquées au robot pour naviguer à vitesse réduite. La première est composée d'un ensemble de stratégies locales mettant généralement en jeu des règles floues. La deuxième utilise la méthode du champ de potentiels artificiels modifiée de manière à éviter les situations de blocage. Enfin, un module flou basé sur la capacité humaine de prise de décision est construit pour incorporer les erreurs capteurs. Il permet d'une part de choisir entre les deux stratégies (globale ou actions réflexes) et, d'autre part, de fournir au mobile une consigne de vitesse fonction du degré de ressemblance entre l'environnement modélisé et l'environnement réel. La validation de ces algorithmes est réalisée en les implantant sur le robot miniature Khepera® muni de six capteurs infrarouge et sur le robot du laboratoire baptisé RMI (Robot Mobile Intelligent) muni de capteurs ultrasonores.

Revues nationales et internationales	Congrès avec comité de lecture
6-2-7, 6-2-8, 6-2-9, 6-2-11	6-3-2-9, 6-3-2-10, 6-3-2-11, 6-3-2-12, 6-3-2-13, 6-3-2-22

Ces travaux ont montré les qualités de robustesse, mais aussi les limites, des contrôleurs floues. Nous sommes donc tournés vers des solutions permettant d'intégrer des stratégies d'autoadaptativité. En particulier, une faiblesse reconnue des régulateurs flous est leur difficulté d'optimisation qui nécessite le réglage d'un nombre important de paramètres. Dans la plupart des applications ce réglage se fait de manière assez empirique. Une voie que nous avons explorée consiste à fusionner les concepts de la logique floue et les architectures des réseaux neuromimétiques [Thèse M. Brunet], technique communément appelée "neuro-flou". Ceci permet de bénéficier des avantages complémentaires des deux techniques : facilité d'apprentissage pour les réseaux de neurones, possibilité d'inclure de la connaissance a priori pour la logique floue. Nous avons obtenu d'intéressants résultats en identification et en commande de processus non linéaires. La validation de ce nouveau type de régulation a motivé l'application de cette technique à la robotique mobile afin de générer automatiquement des règles floues régissant des agents d'actions réflexes. En effet, dans le cas de navigation du robot dans des environnements complexes, les bases de règles peuvent être difficiles à déterminer à cause du grand nombre de situations à considérer. L'ajustage des règles est coûteux en temps car il dépend des connaissances d'un expert. Donc, pour

plus de rapidité et d'optimalité, une mise au point automatique des règles floues est souhaitable. Afin de rendre la génération et l'optimisation de règles floues automatiques en temps réel pour le suivi de mur par exemple, nous avons utilisé l'architecture "neuro-flou" optimisant en ligne une fonction coût caractérisant le comportement désiré. Le problème majeur est que l'on ne parvient pas à obtenir une convergence des poids vers une situation stable. Pour y remédier, une technique dite de régression des poids (weight decay) a été utilisée. Elle consiste à introduire un terme de pénalité dans la fonction coût qui va assurer un comportement de convergence des poids vers des valeurs faibles et quasi-constantes. On peut alors extraire une table de règles dont on peut constater qu'elle est très proche de celle déduite par expertise. On peut donc conclure que le réseau neuro-flou fournit une table de décisions interprétable et cohérente. Cette technique de contrôle peut être utilisée de deux façons. La première consiste à fixer les poids après convergence et alors la structure se comporte comme un contrôleur flou classique avec de bonnes propriétés de robustesse. Dans la seconde, les paramètres sont laissés continûment libres de varier pour pouvoir s'adapter à de nouvelles situations ou à des grandes modifications des caractéristiques physiques du robot.

Revues nationales et internationales	Congrès avec comité de lecture
6-1-1	6-3-1-5, 6-3-2-20, 6-3-2-21, 6-3-2-24, 6-3-2-25 6-3-2-27, 6-3-2-29

Par ailleurs, le contrôle de mouvement d'un robot mobile nécessite l'utilisation de capteurs proprioceptifs et extéroceptifs de manière à pouvoir d'une part se localiser et d'autre part éviter des obstacles imprévus. Mais les données capteurs sont bruitées et entachées d'erreurs. La fusion multisensorielle est utilisée pour améliorer la qualité de l'information. Les informations sensorielles sont traitées le plus souvent dans un cadre probabiliste (approche Bayésienne, filtrage de Kalman,...). Ces méthodes sont efficaces dans le cas idéal où les bruits sont connus, décorrélés et blancs. Par contre dans le cas où les bruits sont mal connus, colorés ou corrélés, l'utilisation de ce genre de techniques devient plus délicate à mettre en œuvre. Rappelons qu'il existe trois notions fondamentalement différentes qui peuvent représenter ce qu'on entend par bruit de mesure : l'incertitude, l'imprécision et la fiabilité. L'incertitude représente le doute sur la conformité de la mesure avec la réalité, l'imprécision est plutôt liée à l'imperfection du capteur et enfin la fiabilité évalue l'impossibilité d'acquisition d'une information à cause de la structure de l'environnement ou d'une défaillance du capteur. Récemment de nombreuses équipes se sont intéressées aux méthodes inspirées du raisonnement humain telle que la théorie des sous-ensembles flous pour pouvoir prendre en compte ces notions. Le travail entrepris se place dans le cadre de la théorie des possibilités qui est issue de la théorie des ensembles flous [Thèse M. Oussalah]. Cette approche se caractérise par la diversité des solutions qu'elle peut apporter au problème de combinaison des informations et elle fournit un formalisme commode et direct pour exprimer et agréger des informations conflictuelles. Le point central du travail est l'utilisation de la règle de combinaison adaptative proposée par D. Dubois et H. Prade pour fusionner les différentes données. Il existe deux modes de combinaison fondamentaux en théorie des possibilités : le mode conjonctif (intersections généralisées) et le mode disjonctif (réunions généralisées). Le mode conjonctif utilise comme opérateurs des normes triangulaires (t-normes, par exemple : minimum, produit etc.) et s'applique dans le cas où les sources sont considérées comme toutes fiables. Le mode disjonctif est basé sur les opérateurs de t-conorme (maximum, somme probabiliste etc.) et correspond aux cas où seulement certaines sources (non identifiées) sont fiables. Le choix du mode de combinaison dépend donc du degré de cohérence entre les sources. La règle de combinaison adaptative permet alors de passer continûment du mode conjonctif au mode disjonctif en fonction de ce dernier. Nous avons analysé cette règle du point de vue des propriétés algébriques qu'elle engendre et des différentes étapes aboutissant à sa formulation ce qui a permis de proposer des variantes de cette dernière afin de valider certaines propriétés. Une méthode de localisation absolue entièrement basée sur la théorie des possibilités a été comparée au filtrage de Kalman en simulation et par des expérimentations réelles. Ces expérimentations réelles ont été menées avec le robot miniature Khepera® qui est doté de deux capteurs odométriques et de 8 capteurs infrarouges. Ces capteurs travaillant en réception permettent de mesurer des distances à partir de diodes émettrices infrarouges jouant le rôle de balises. Par ailleurs, le filtrage de Kalman impose l'utilisation de distributions de probabilité sous forme de gaussiennes pour représenter les informations. Cette modélisation est approximative car, dans la réalité, les différents bruits présents au niveau des mesures des capteurs sont rarement gaussiens. Elle présente par contre l'avantage de la simplicité, car il suffit de deux paramètres pour définir une gaussienne : sa moyenne et sa variance. Les manipulations et les calculs se font donc simplement par des formules numériques depuis longtemps éprouvées. A contrario, la méthode possibiliste mise au point n'impose pas en théorie de forme particulière aux distributions. Elle est donc de ce point de vue plus souple et plus réaliste que le filtrage de Kalman. Cette souplesse se paye par une augmentation de la complexité, car l'enregistrement d'une distribution requiert soit plusieurs intervalles (forme alpha-coupe) soit une suite de points (forme fonctionnelle). D'autre part, la fusion des informations dans la méthode du filtrage de Kalman peut se voir comme une moyenne pondérée par l'incertitude liée aux informations. Le principe consiste à accorder plus de poids aux informations dont l'incertitude est faible (i.e. variance faible). Il s'agit donc d'une fusion par

"minimisation de la variance". La règle de combinaison adaptative utilisée dans la méthode possibiliste s'intéresse plus à la redondance des différentes sources en accordant plus de poids aux informations qui se répètent. Il s'agit donc plutôt d'une fusion par "concordance".
Enfin, la méthode possibiliste pose dans certains cas des problèmes de robustesse vis à vis de la forme des distributions à combiner, qui peuvent engendrer un dysfonctionnement du point de vue de la valeur défuzzifiée de la distribution résultante. Nous avons proposé donc une nouvelle règle de combinaison dite *progressive* qui tient compte de façon explicite de la distance séparant la zone de consensus de chaque point du support et qui est robuste par rapport à la forme des distributions de possibilités qui est évaluée et validée dans le cas de la localisation du robot mobile RMI muni de capteurs ultrasonores.

Revues nationales et internationales	Congrès avec comité de lecture
6-2-10	6-3-1-6, 6-3-2-14, 6-3-2-15, 6-3-2-16, 6-3-2-17 6-3-2-19, 6-3-2-23, 6-3-2-26

4- Activités d'enseignement

De 1986 à 1990, j'ai enseigné en tant que maître-auxiliaire dans les académies de Versailles et de Créteil. J'ai assuré des cours de sciences physiques et de mathématiques à des niveaux variant de la sixième à la terminale. Durant ces années d'enseignement, j'ai suivi des stages de formation pédagogique.

J'ai été nommé comme ATER (192h TD) au Département Génie Electrique et Informatique Industrielle de l'IUT d'Evry à la rentrée universitaire 1990. Durant cette période qui a correspondu pour moi à une réorientation complète de thème de recherche, j'ai complété ma formation d'enseignant en participant aux TP, TR et projets de fin d'année, en parallèle avec mes collègues en poste à l'IUT. J'ai encadré en particulier :
*des Travaux Pratiques en électronique générale, numérique et analogique en 1ère année (AOP, transistors, fonction amplification, logique combinatoire et séquentielle, ...)(80h).
*des "Travaux de Réalisation" (80h) c'est à dire conception et réalisation de projets en électronique analogique et numérique. Mon rôle consiste à encadrer et diriger les étudiants sur le plan théorique et pratique jusqu'à l'obtention de la meilleure solution.
Quelques exemples de TR réalisés : formeur à diodes, générateur de fonctions, dB-mètre, amplificateur à touch control, amplificateur RIAA, amplificateur à grand gain, ...
*des Projets de programmation en langage C (20h en 90/91),
*des Projets d'IAO ou de Circuits logiques cablés (15h par an),
*des projets VHDL (12h).

Depuis septembre 1993, j'occupe un poste de Maître de Conférences en 61ème section toujours au sein du Département GEII. J'ai pu alors diversifier mes enseignements et prendre davantage de responsabilité pédagogique. Je me suis chargé en plus :
*du Cours et des Travaux Dirigés en électricité (50h).
*du Cours et des Travaux Dirigés d'Algorithmique (20h).
*des Travaux Pratiques en électronique en 2ème année (Amplificateur de puissance, Hacheur, Multivibrateurs, PLL, Modulation,...) (80h).

Par ailleurs, le site d'Evry ayant vu s'ouvrir de nouvelles formations, j'ai été sollicité pour participer à leur mise en place. C'est ainsi que j'ai été chargé d'un module (25h) de Cours-TD d'Electronique Analogique en 2ème année de l'IUP en 93/94, d'un TD d'électronique (25h) au Deug MIAS en 96/97, d'un TD (10h) traitant des systèmes numériques en 1ère année en 98/99 et j'ai assuré la préparation à l'UV A d'Electronique (analogique et numérique) du CNAM (100h en cours de soir) en 93/94, 95/96 et 97/98. Depuis l'année dernière je dispense le cours de la théorie du flou (10h) au DESS. Je participe depuis deux ans aux enseignements de l'électronique (30h) à l'école supérieure d'ingénieurs en électronique et électrotechnique (ESIEE).

Je contribue au sein du département GEII au suivi des étudiants de deuxième année pendant les stages de fin d'études en entreprises industrielles et dans les laboratoires de recherche. Et, je suis tuteur d'un groupe d'étudiants de 1ère année pour les encadrer et les informer.

5- Responsabilités collectives

5-1-Responsabilité dans le cadre du département GEII :

J'ai eu la charge de m'occuper pendant l'année 94/95 du recrutement des étudiants au département GEII, ce qui m'a amené à diffuser l'information dans les salons d'étudiants et les établissements d'enseignement secondaire.
Depuis le mois de juillet 1999, j'ai été élu à l'unanimité pour assurer la charge de chef du département GEII pendant les trois années à venir. Les priorités que je compte réaliser sont :
- la lutte contre la baisse des candidatures en formation initiale (contacts renforcés avec les lycées, semestre d'orientation, création d'option telle l'option RLI),
- mise en place d'une licence professionnelle,
- le développement de la formation continue (actions pour les handicapés, apprentissage, formation continue).

5-2-Activités dans le cadre du laboratoire

Depuis mon intégration au laboratoire, je m'occupe du parc informatique du groupe "neuro-flou". Je participe à la rédaction et la mise en forme du rapport d'activité du laboratoire.
A la suite de l'incorporation du LRE au sein du CEMIF (Centre d'Etude Mécanique d'Ile de France) sous forme d'un pôle de recherche baptisé GSC (Groupe Systèmes Complexes), une série de séminaire a été présentée pour faciliter l'intégration des différentes composantes du CEMIF. J'ai dans ce cadre organisé et animé une journée de travail sur les thèmes "Robotique téléopérée - Robotique mobile".
Enfin, J'ai été un membre élu de la commission des spécialistes 61-63ème sections de l'université d'Evry Val d'Essone (EVE) de 1995 à 1998. Je dispose depuis octobre 1999 d'une prime d'encadrement doctorale et de recherche (PEDR).

6- Liste des publications

6-1- Ouvrage

1. Optimization of a Fuzzy Controller for a Reactive Navigation
   H. Maaref, C. Barret, A. Amamou
   Edité en 1999 dans « Computational Intelligence and Applications » par Nikos E. Mastorakis.
   ISBN : 960-8052-05-X

6-2- Revues nationales et internationales

1. Effet d'un bombardement ionique de très basse énergie sur la barrière de Schottky métal-GaAs.
   T. Neffati, C. Barret, G.N. Lu, H. Maaref.
   Rev. Phys. Appl. 23 (1988) 1103.
2. Etude des interfaces Al-GaAs et InP préparées par bombardement ionique de très basse énergie.
   T. Neffati, H. Maaref, C. Barret.
   Le vide, les couches minces, 241 (1988) 267.
3. Distributions électroniques aux interfaces profondes
   F. Vergand, P. Jonnard, H. Maaref, C. Sénémaud, H. Roulet, G. Dufour, C Bonnelle.
   Le vide, les couches minces, 241 (1988) 223.
4. Caractérisation électronique d'interfaces profondes Al/InP.
   C. Barret, F. Vergand, H. Maaref, C. Sénémaud, C. Bonnelle.
   Rev. Phys. Appl. 24 (1989) 439.
5. Etude de contacts métal-InP(n) clivé : barrière de Schottky et états d'interface
   H. Maaref, C. Barret.
   J.Phys. III 1 (1991) 749.
6. Study of Cleaved metal-InP(n) contacts
   C. Barret, H. Maaref.
   Solid State Electron. 36 (1993) 879.
7. Aide floue à la navigation d'un robot mobile dans un environnement partiellement connu
   M. Benreguieg, H. Maaref, C. Barret.
   Revue d'Automatique et Productique Appliquée RAPA, Vol. 8, 1995, pp. 447-452.
8. Aides floues à la navigation d'un robot mobile autonome
   H. Maaref, M. Benreguieg, C. Barret.
   Journal Europèen des Systèmes Automatisés JESA, Vol. 30, 1996, pp. 839-857.
9. Fuzzy navigation strategy: application to two distinct autonomous mobile robots
   M. Benreguieg, P. Hoppenot, H. Maaref, E. Colle, C. Barret.
   Robotica, Vol. 15, 1997, pp. 609-615.

10. Fusion de données capteurs en vue de la localisation absolue d'un robot mobile par une méthode basée sur la théorie des possibilités : Comparaison avec le filtre de Kalman
H.Maaref, C.Barret, M.Oussalah.
Revue Traitement du Signal, Vol.16, n°5, 1999, pp. 345-359.
11. Sensors-based fuzzy navigation of an autonomous mobile robot in indoor environment
H. Maaref, C. Barret.
Control Engineering Practice, sous presse.

6-3- Participation à des Congrès avec comité de lecture

6-3-1- Participations affichées :

1. Distributions électroniques à l'interface Al-InP
F. Vergand, C. Sénémaud, H. Maaref, C. Bonnelle.
Journées de la matière condensée (S.F.P.)
Toulouse, 1988.
2. Etude de l'interface Al/InP par spectroscopie de capacité Schottky et spectroscopie d'émission X induite par électrons.
H.Maaref, C. Barret, F. Vergand, C. Sénémaud, C. Bonnelle.
Journées nationales microélectroniques III-V,
Oléron, 1988.
3. Physicochemical characterization of Al-InP interfaces.
H. Maaref, F. Vergand, C. Barret, P. Jonnard, C. Sénémaud, C. Bonnelle.
ECASIA 89, European Conference on Applications of Surface and interface Analysis
Juan-les-Pins, Oct. 1989 et dans Le vide, les couches minces, supplément au n°248, 1989.
4. A fuzzy rules based antiskating system
C. Barret, V. Cortyl, H. Maaref.
Int. Conf. on Information Processing and Management of Uncertainty, IPMU'94,Paris, Juil.1994.
5. Conception d'un contrôleur flou auto-ajustable
M. Brunet, H. Maaref, C. Barret.
MCEA'95
Grenoble, Sept. 1995.
6. Approche possibiliste pour la fusion d'information de capteurs à ultrasons
M. Oussalah, H. Maaref, C. Barret.
LFA'95,
Paris, Nov. 1995.

6-3-2- Participations orales :

1. Distributions électroniques aux interfaces profondes.
H. Maaref.
Deuxièmes Journées d'Etudes sur l'Epitaxie et la passivation des composés III-V,
Plestin-les-Grèves, 1988.
2. Etude des interfaces Al-GaAs et InP préparées par bombardement ionique de très basse énergie (0-200 eV).
H. Maaref, T. Neffati, C. Barret.
Deuxièmes Journées d'Etudes sur l'Epitaxie et la passivation des composés III-V
Plestin-les-Grèves, 1988.
3. Commande de processus linéaires par un contrôleur flou.
H.Maaref, M.Brunet, W.Nouibat.
First international conference on Electronics and Automatic Control
ICFA ‘92, Tizi-Ouzou , Mai 1992.
4. Fuzzy control of a thermal process
H. Maaref, Ph. Papin, C. Barret.
Int. Workshop on Applied Automatic Control
Waac'93,Prague, Mai 1993.
5. A fuzzy antiskating controller for autonomous vehicle
C. Barret, V. Cortyl, H. Maaref.
Int. AMSE Conf., Systems Analysis, Control and Design
SYS'94, Lyon, Juil.1994.
6. Design of a fuzzy rules based antiskating device
C. Barret, V. Cortyl, H. Maaref.

2° European Congress on Intelligent Techniques and Soft Computing, EUFIT'94,Aachen, Sep.1994.

7. Réalisation d'un suivi de couloir par un robot mobile à l'aide de la logique floue
M. Benreguieg, H. Maaref, M.Mallem, C.Barret.
ICEEE'94, Oran, Oct. 1994.

8. Planification et suivi de trajectoire d'un robot mobile par graphe de visibilité et logique floue
M. Benreguieg, H. Maaref, W. Nouibat, C.Barret.
ICEEE'94, Oran, Oct. 1994.

9. Fuzzy help in mobile robot navigation
H.Maaref, C. Barret.
IEEE IACC'95, Taipei, Mai 1995.

10. Fuzzy helps to control a mobile robot in a partially-known environment
M. Benreguieg, H. Maaref, C. Barret.
EUFIT'95, Aachen, Aout 1995.

11. Association du champ de potentiels et du contôle flou pour la navigation d'un robot mobile
M. Benreguieg, H. Maaref, C. Barret.
LFA'95, Paris, Nov. 1995.

12. Navigation of a mobile robot in fuzzy artificial potential fields
H. Maaref, M. Benreguieg, C. Barret.
IEEE-SMC CESA'96, Lille, Juillet 1996.

13. Control of a medical aid mobile robot based on fuzzy navigation
P. Hoppenot, M. Benreguieg, H. Maaref, E. Colle, C. Barret.
IEEE-SMC CESA'96, Lille, Juillet 1996.

14. Adaptive combination rule: influence of t-norm operators;
M. Oussalah, H. Maaref, C. Barret.
EUFIT'96; Aachen; Sep. 1996.

15. Kalman filter and possibility theory for the localization of a mobile robot;
M.Oussalah, H. Maaref, C. Barret.
EUFIT'96; Aachen; Sep. 1996.

16. Possibility theory and multisensor data fusion
M.Oussalah, H. Maaref, C. Barret.
CIMASI'96, Casablanca, Nov. 1996.

17. Localisation d'un robot mobile : Etude Comparative entre le filtrage de Kalman et la théorie des possibilités
M. Oussalah, C. Barret, H. Maaref.
COMAI'96 3-5 , Tlemcen Algerie, Déc. 1996 et dans Algerian Journal of Technologie, Vol.7, 1997.

18. Localisation par fusion multisensorielle basée sur la théorie des possibilités
H. Maaref, M. Oussalah , C. Barret.
LFA'96, Nancy, 1996.

19. Absolute Position Estimation of a Mobile Robot
M.Oussalah, C.Barret, H.Maaref.
ICARCV'96 , Singapour, 1996.

20. Fuzzy agents for reactive navigation of a mobile robot
C. Barret, M. Benreguieg, H. Maaref.
KES'97, Adelaide, 1997.

21. Design of an Auto-Tuned Fuzzy Controller : Application to the Reactive Navigation of a Mobile Robot
M. Benreguieg, H. Maaref, C. Barret.
SICICA'97, International Federation of Automatic Control, Annecy, Juin 1997

22. Fusion of fuzzy agents for reactive navigation of a mobile robot
M. Benreguieg, H. Maaref, C. Barret.
International Conference on Intelligent Robots, IROS'97, Grenoble, 1997.

23. Positionong of a mobile robot with landmark-based method
M. Oussalah, H. Maaref, C. Barret.
International Conference on Intelligent Robots, IROS'97, Grenoble, 1997.

24. Navigation of an autonomous mobile robot by coordination of behaviours
M. Benreguieg, H. Maaref, C. Barret.
3rd IFAC Symposium on Intelligent Autonomous Vehicles, IAV'98, Madrid, 1998.

25. Progressive learning : from learning with a miniature robot to reactive navigation of a real mobile robot
H. Maaref, M. Benreguieg, C. Barret.
IMACS Int. Conf.on Computational Engineering in Systems Applications, CESA'98, Hammamet, 1998.

26. Absolute localisation of a miniature mobile robot using heterogeneous sensors : comparison between Kalman filter and possibility theory methods
T. Pannerec, M. Oussalah, H. Maaref, C. Barret.
IMACS Int. Conf.on Computational Engineering in Systems Applications, CESA'98, Hammamet, 1998.
27. Sensor-Based Reactive Navigation Systems Using Auto-Tuned Fuzzy Controllers
H. Maaref, F. Despalles, C. Barret.
IEEE Int. Conf. on Advances in Vehicle Control and Safety, AVCS'98, Amiens, 1998.
28. Vers une règle de combinaison progressive dans le cadre de la théorie des possibilités
H. Maaref, M. Oussalah , C. Barret.
LFA'98, Rennes, 1998.
29. Optimization of a Fuzzy Controller for a Reactive Navigation
H. Maaref, C. Barret, A. Amamou
CSCC'99, Athène, Grèce, 1999.

6-4- Conférences avec rapports internes:

1. Présentation d'un banc d'essai pour la validation d'un contrôleur flou.
H.Maaref.
GR Automatique du CNRS, Pôle automatisation intégrée, Projet Commande Symbolique et Neuromimétique, 11 juin 91.
2. Commande de processus par un contrôleur flou.
H.Maaref, M.Brunet, W. Nouibat.
GR Automatique du CNRS, Pôle automatisation intégrée, Projet Commande Symbolique et Neuromimétique, Rapport 91-2, Novembre 91.
3. Application d'un contrôleur flou à des processus linéaires.
H.Maaref, C. Barret.
GR Automatique du CNRS, Pôle automatisation intégrée, Projet Commande Symbolique et Neuromimétique, Rapport 92-1, Mars 92.
4. Etude de performances d'un contôleur flou appliqué à la commande de processus linéaires
H.Maaref, C. Barret.
GR Automatique du CNRS, Pôle automatisation intégrée, Projet Commande Symbolique et Neuromimétique, Rapport 92-2, Juillet 92.

6-5- Conférences sans actes

1. Aide floue à la navigation d'un robot mobile dans un environnement partiellement connu.
M. Benreguieg, H. Maaref, C. Barret.
Premier Colloque International Productique et Robotique du Sud Atlantique, Bourges-France,1-2 juin 1995.
2. Application de la logique floue à la navigation d'un robot mobile dans un environnement a priori connu.
M. Benreguieg, H. Maaref, C. Barret.
Troisième journée "séminaire jeunes chercheurs en robotique", Laboratoire de robotique de Paris, Avril 1995.
3. Fusion de données capteurs ultrasonores dans le cadre de la théorie des possibilités
M. Oussalah, H. Maaref, C. Barret.
GT5 Fusion de données, Novembre 1995, Paris
4. Assistance au pilotage d'une base mobile, équipée de capteurs ultrasonores,par contrôle flou
P. Hoppenot, M. Benreguieg, H. Maaref, J.C. Lafont, E. Colle.
GR Automatique du CNRS, Pôle robotique. Janvier 1996, Metz.
5. Localisation d'un robot mobile par fusion de données odométriques et ultrasonores
M. Oussalah, H. Maaref, C. Barret.
GR Automatique du CNRS, Pôle robotique. Juin 1996, Evry.
6. Navigation d'un robot mobile basée sur la logique floue
M. Benreguieg, P. Hoppenot, H. Maaref, E. Colle, C. Barret.
GR Automatique du CNRS, Pôle robotique. Juin 1996, Evry.

6-6- Rapport interne

1. Performances d'un régulateur flou appliqué à la commande de systèmes
H. Maaref, C. Barret
LRE , 1993.
2. Réflexions sur le traitement de l'information imparfaite

H. Maaref
Cemif-GSC, 1998.

6-7- Contrats

1. Etat de l'art sur la fusion multisensorielle.
F. Chavand, C. Vasiljevic, H. Maaref.
Rapport CEMIF-Systèmes Complexes, Contrat INRETS UR/DART n°102-97-085, 23 décembre 1997.

7-Encadrement

7-1-Stages

7-1-1-DUT
Dix stages de 91 à 95 dont deux sujets ont été soutenus par les aides de l'ANVAR

7-1-2-Maîtrise

1. "Reconnaissance de primitives de l'environnement à partir de mesures ultrasonores"
N. Harabech, Maîtrise de physique et applications, Paris VII, Septembre 1997.

7-1-3-Elèves-ingénieurs

1. "Application des réseaux Neuro-Flous à la commande d'un processus du premier ordre avec non linéarité"
A. Philippot, mémoire d'ingénieur IIE Juin 1992.
2. "Régulation thermique par logique floue: conception, simulation et implantation"
Ph. Papin, mémoire d'ingénieur INT Mars 1993 - Contrat Chaffoteaux et Maury
3. "Reconstitution géométrique d'obstacles détectés par un robot autonome évoluant dans un milieu partiellement connu"
N. Bourdon, IIE Septembre 1995 (Aide ANVAR).
4. "Navigation d'un robot mobile par techniques neuro-floues"
L. Deswaelle, mémoire d'ingénieur IIE Juin 1996.
5. "Reconstitution géométrique d'obstacles détectés par un robot autonome évoluant dans un milieu partiellement connu".
I. LOU, IUP Juin 1996.
6. "Optimisation d'un SIF par réseau de neurones"
T. Raulin, IIE Septembre 1997.
7. "Localisation absolue d'un robot mobile : Fusion d'informations par la théorie des possibilités"
T. Pannerec, mémoire d'ingénieur IIE Juin 1997.
8. "Navigation de robots mobiles autonomes par des systèmes d'inférence floue optimisés"
F. Despalles, mémoire d'ingénieur IIE Juin 1998.

7-1-4-DEA

1. "Réseau de neurones et logique floue: analyse d'implémentation hardware en vue d'application à la commande de systèmes"
L. Amara Maaref, DEA Sciences Pour l'Ingénieur (SPI) Juillet 1992.
2. "Application de la logique floue à la robotique mobile"
M. Benreguieg, DEA SPI Juillet 1993.
3. "Correcteur PID à supervision floue"
X. C. De Thezac, DEA Robotique Juillet 1994.
4. "Planification de mouvement d'un robot mobile dans un environnement partiellement connu"
M. Oussalah, DEA SPI Juillet 1994.
5. "Application de la logique floue à l'antipatinage de roues de robot"
V. Cortyl, DEA SPI Juillet 1994, Contrat Unité Robotique CEA.
6. "Extraction de règles de commande par un réseau neuro-flou pour la navigation d'un robot mobile"
K. Elmaraabet, DEA SPI Juillet 1997.
7. "Mise en œuvre d'un filtre de Kalman pour la localisation d'un robot mobile"
P. Delaperrelle, DEA Robotique Juillet 1999.

7-2-Thèses

1. "Identification et contrôle de processus par réseau neuro-flou "

M. Brunet (soutenance en décembre 1996)
Directeur C. Barret, encadrement H. Maaref et M. Houkary

2. "Aides floues à la navigation d'un robot mobile"
M. Benreguieg (soutenance en janvier 1997)
Directeur C. Barret, encadrement H. Maaref

3. "Fusion de données par la théorie des possibilités : Application à la localisation d'un robot mobile"
M. Oussalah (soutenance en juin 1998)
Directeur C. Barret, encadrement H. Maaref

8-Divers

1. Rapporteur auprès de la Revue de Physique Appliquée (Janvier 1991).
2. Rapporteur IEEE ICRA'95 (Nov. 1994).
3. Co-chairmain IEEE IACC'95, Taipei, Taiwan (Mai 1995).

CHAPITRE 0

Introduction

Les recherches récentes en robotique s'orientent vers des architectures dotées de moyens de perception, d'action et de décision permettant une certaine autonomie. Cette autonomie est liée aux capacités de génération de trajectoire dans des milieux contraints, aux moyens de prise de décision à partir d'informations capteurs imprécises et / ou incertaines et d'apprentissage à partir de l'interaction robot / environnement.
Plusieurs méthodes de contrôle de systèmes ont été développées en robotique mobile. On peut les classer en deux catégories : celles basées sur la planification globale et celles utilisant un contrôle local. Les méthodes globales, basées sur une connaissance précise du robot et de l'environnement, ont l'avantage d'une part de prouver l'existence d'une solution optimale permettant au robot d'atteindre son but et d'autre part de construire une cartographie d'espace libre. Mais elles ont comme inconvénients d'exiger d'avoir un modèle exact de l'environnement et on ne peut prendre en compte une quelconque variation du modèle du monde. Les méthodes locales peuvent être assimilées à des méthodes réactives ou d'actions réflexes qui sont basées seulement sur l'interaction robot/environnement. Puisque le robot n'a qu'une vue très réduite de l'environnement, ces méthodes ne peuvent garantir le succès de la mission en toute circonstance. L'une ou l'autre de ces deux stratégies (globale ou locale), ne peut être efficace à elle seule pour le contrôle du mouvement d'un robot mobile dans un environnement complexe réel, qui, généralement, ne peut être que partiellement modélisé. D'où l'idée de les fusionner, pour profiter des avantages de chacune d'elles. Cette fusion doit permettre :
- d'utiliser le mieux possible les connaissances disponibles a priori sur l'environnement d'évolution par la méthode globale qui garantit l'optimalité de la mission et un maximum de sécurité pour le robot.
- de recourir à une méthode locale lorsque l'environnement réel du robot diffère de façon significative de l'environnement modélisé.
Donc, un contrôle efficace du mouvement d'un robot mobile dans un environnement complexe doit intégrer des capacités de planification des mouvements et de réactivité face à des situations imprévues. Par ailleurs, il est bien évident que pour mener à bien ce contrôle, il est nécessaire que le robot connaisse sa position par rapport à un repère pris comme référence dans son environnement.
Le travail développé s'efforce de répondre à ces différents objectifs. Nous aurons recours à plusieurs reprises aux techniques de la théorie du flou pour réaliser les modules nécessaires à la navigation et à la localisation des robots. La théorie du flou est le cadre théorique regroupant la théorie des sous-ensembles flous, la logique floue et la théorie des possibilités. En l'occurrence, l'approche basée sur la logique floue semble particulièrement bien adaptée à la commande d'un robot mobile. On a en effet besoin de systèmes de navigation qui soient robustes, qui puissent être portés le plus facilement possible d'un véhicule à un autre et qui soient éventuellement évolutifs en fonction des missions à réaliser. L'absence de nécessité de disposer d'un modèle préalable du processus et l'intégration de savoir-faire heuristique qui caractérisent le contrôle flou répondent parfaitement à ces objectifs. Cependant on ne tire généralement pas complètement parti d'un système d'inférence floue (SIF) quand on se contente d'une mise au point manuelle. En effet, lors de la conception d'un SIF des choix sont effectués de façon relativement empirique et la mise au point d'un système de commande floue peut s'avérer longue et délicate. Par ailleurs, les réseaux de neurones, eux, ont prouvé théoriquement et pratiquement leur aptitude à approcher de très larges classes de fonctions non-linéaires et à identifier des systèmes non-linéaires statiques et dynamiques. Partant de ces constatations, plusieurs chercheurs ont cherché à fusionner les deux approches pour pallier les défauts de l'une par les avantages de l'autre dans le but d'optimiser automatiquement certains paramètres d'un SIF. Cette technique a été baptisée sous le nom générique de “ neuroflou ”. Partant de ce concept, une architecture de commande a été conçue pour permettre de générer et d'optimiser les conclusions de règles floues. Ce point est particulièrement intéressant si on souhaite extraire une base des règles sans avoir recours à une expertise humaine. Il est alors essentiel que la phase d'apprentissage converge vers des paramètres stables et traduisibles en concepts linguistiques. La possibilité d'extraction de connaissance a motivé l'application d’une telle technique à la robotique mobile afin de générer automatiquement des règles floues régissant des agents d'actions réflexes tels que le suivi de mur, le suivi du milieu de l'espace libre et la convergence vers un but.
Dans le sillage du succès des contrôleurs flous, la fin des années soixante-dix a vu également la naissance de la théorie des possibilités. Cette théorie, issue de la théorie des sous-ensembles flous, a pour but de traiter dans un même formalisme mathématique l'imprécision et l'incertitude et d'offrir ainsi un cadre général pour la représentation et la manipulation des connaissances. Un des domaines où l'intérêt de cette nouvelle théorie peut pleinement se manifester est la fusion de données. En effet, les données issues de différentes sources d'informations sont imprécises et incertaines. La fusion des informations provenant des différentes sources permet justement de compenser les limites ou les effets indésirables d'une source quelconque par les données d'autres sources. On obtient ainsi un résultat plus sûr et plus fiable. De plus, notre attitude vis à vis du processus de fusion suivant qu'on adopte une position de prudence ou d'optimisme ou bien une position intermédiaire détermine le niveau de l'imprécision et de l'incertitude correspondant au résultat de la fusion. La robotique

mobile a souvent constitué un carrefour où plusieurs techniques d'intelligence artificielle et de fusion de données sont mises à l'épreuve. Les différents capteurs qui entourent le corps du robot jouent le rôle de sources d'informations. On cherche alors à fusionner les différentes informations capteurs dans le but de rendre le robot indépendant de l'opérateur humain, c'est à dire capable de prendre la décision appropriée suivant la situation qui se présente devant lui. On conçoit que doter un robot d'autonomie n'est pas une opération facile et que la tâche de la perception de l'environnement est une opération essentielle. Elle doit intégrer tous les algorithmes nécessaires à la fusion des données des différents capteurs pour pouvoir utiliser les mesures brutes afin de réaliser par la suite des fonctions comme l'évitement d'obstacle ou la localisation. Les techniques employées pour la localisation, varient généralement suivant le type d'environnement dans lequel le robot est amené à naviguer, les conditions connues sur cet environnement et le type de capteurs dont est muni le robot.
La théorie des possibilités, bien qu'elle présente des avantages théoriques assez fondés, n'a encore été que peu testée sur ce genre de problème. Notre objectif est d'apporter une contribution à l'évaluation de techniques possibilistes dans la fusion de données en vue de la localisation d'un robot mobile.

CHAPITRE I

Optimisation d'un système d'inférence floue

1-Introduction

Afin de mettre en œuvre des systèmes robustes, de faible coût et en adéquation avec le monde réel, on a éprouvé ces dernières années le besoin de formaliser des méthodes empiriques, de généraliser les modes de raisonnement naturels et de construire des systèmes artificiels modélisant les tâches habituellement prises en charge par les humains. Pour répondre à ces besoins, plusieurs techniques dites "modernes" ou "non-conventionnelles" émergeant de l'intelligence artificielle ont été proposées et récemment regroupées par L.A. ZADEH [ZADEH 96] au sein du concept unique de *soft computing*. Ce concept décompose d'une manière didactique ces techniques en deux classes : approches *cognitive* et *non cognitive* qui ne sont pas des techniques concurrentes mais plutôt synergiques et possèdent de nombreuses ressemblances.
Une approche non cognitive est une approche utilisée lorsque le système est trop complexe (le cas d'un espace multidimensionnel) de sorte que l'être humain ne peut ni l'analyser ni expliciter les connaissances s'y rapportant. Dans cette catégorie, on classe toutes les méthodes d'apprentissage supervisé et non supervisé, statistiques et neuronales. Elles nécessitent toutes des données suffisamment nombreuses pour fournir des échantillons d'apprentissage représentatifs. Contrairement à cette dernière, l'approche cognitive exploite les connaissances disponibles sur le système pour les décrire sous forme experte sans avoir besoin de modèles mathématiques. Dans cette classe, on peut inclure par exemple les systèmes experts classiques ou flous, les contrôleurs flous et les techniques basées sur la théorie de l'incertain.
En fait, la description d'un système est fonction de sa complexité et des connaissances dont on dispose qui sont généralement imparfaites. Ces imperfections ont été traitées dans leur globalité jusqu'alors par la théorie des probabilités qui est initialement conçue pour la modélisation des jeux de hasard et les observations de fréquence dans les phénomènes physiques. Mais celle-ci ne permet pas de traiter des opinions subjectives (croyance dans l'occurrence d'un événement) ni de résoudre le problème posé par les connaissances imprécises ou vagues. En effet, étudier ou gérer des systèmes complexes nécessite la prise en compte de certains types d'incertitudes liées à la difficulté des observations, aux imprécisions linguistiques, à la fiabilité tant des observateurs humains que des capteurs et instruments de mesure, à l'utilisation de connaissances empiriques et à l'imprécision du raisonnement humain.
Ceci nous amène à définir et à distinguer certaines notions :
-Imprécision : elle est liée au défaut quantitatif relatif au contenu propre de l'information captée. Dans le cas du raisonnement humain, elle est relative au contenu d'une proposition (énonciation d'un jugement ou d'une affirmation) "mesurer environ 1.75m", "avoir entre 20 et 25 ans",... On ne donne pas de valeur précise mais un intervalle, une "fourchette" (intervalle dont les frontières sont mal définies). Cet intervalle est désigné usuellement en probabilité par intervalle de confiance ou incertitude. En fait "une mesure à 5% près" est plutôt une imprécision qu'une incertitude probabiliste.
-Aspect vague : il est englobé dans la notion d'imprécision mais toutefois il se distingue par son lien à des propositions dont les prédicats sont qualitatifs ou difficilement quantifiables telles que par exemple "forte douleur", "il est presque chauve", "bonne ambiance" ou "avoir à peu près le même âge".
-Incertitude : elle représente le doute sur la véracité de l'information. Par exemple "il est possible qu'il soit chauve", le prédicat (l'attribut de la proposition) "chauve" est considéré connu (localisé), mais c'est la proposition qui est douteuse.
Par ailleurs, l'imperfection de l'information est rarement d'un type unique et dépend fortement du contexte réel. Les notions d'imprécis et d'incertain sont fortement liées et il est souvent difficile de les identifier clairement et séparément car, dans un raisonnement quelconque, l'imprécision sur une information induit de l'incertitude sur le résultat. C'est à dire que plus une information est imprécise, plus sa certitude est grande et que des connaissances imprécises engendrent souvent des incertitudes sur la décision finale à laquelle elles conduisent. Exemple : quelqu'un qui a "environ 18 ans" a t-il le droit de voter ? C'est possible mais pas certain.
En 1965, L.A. ZADEH [ZADEH 65] connu internationnalement pour ses travaux sur la théorie des systèmes, a introduit la notion de sous-ensembles flous (fuzzy sets) généralisant ainsi l'algèbre de la théorie classique des ensembles et aussi la logique trivaluée (0, 1 et ½ représentant un état d'ignorance ou de doute) introduite en 1920 par Lukasiewicz. Ceci permet d'une part de représenter et de manipuler des connaissances imprécisément et/ou vaguement décrites et d'autre part d'établir une interface entre des données décrites symboliquement (avec des mots) et numériquement. Contrairement à sa dénomination cette théorie est très précise et s'appuie sur des fondements mathématiques solides.
En 1978 [ZADEH 78], il introduit la théorie des possibilités qui constitue un cadre permettant de traiter dans un même formalisme les notions d'incertitude et d'imprécision précédemment définies.

2- La commande floue

La commande floue est une application de la logique floue au contrôle des systèmes dynamiques pour lesquels on ne possède pas de modèles satisfaisant. Son principe est simple : il s'agit dans la plupart des cas d'imiter le comportement d'un opérateur humain dans la régulation d'un processus complexe, à l'aide de règles floues.
Cette méthode de conduite de processus est mise en œuvre grâce à un dispositif comportant :
. un certain nombre d'*entrées* (elles se rapportent à des variables d'état du système, dont les grandeurs physiques sont mesurées précisément, à l'aide de capteurs ou sondes) ;
. des *règles floues* assurant le passage des entrées aux sorties ;
. un certain nombre de *sorties* (les différents paramètres faisant l'objet d'une régulation).
Ce dispositif est appelé usuellement contrôleur flou. Un contrôleur flou est en fait un système expert qui fonctionne à partir d'une représentation des connaissances basée sur les sous ensembles flous (SEF). Comme tout système expert il dispose donc d'une *base de faits* (état des paramètres d'entrée du système, dont les grandeurs exactes instantanées sont connues), d'une *base de connaissances* (description des variables et règles floues) et d'un *moteur d'inférence* (c'est au niveau de ce dernier bloc que se déroule le mécanisme de déduction du système, qui débouche sur la détermination des valeurs de sortie du contrôleur flou). Les contrôleurs flous se démarquent cependant des systèmes experts classiques par leur petit nombre de règles, par un traitement doux des contradictions (deux règles aux effets opposés auront un effet compensatoire l'une sur l'autre) et également par leur rôle qui consiste à modifier en permanence les paramètres de commande d'un processus.
Un système flou de commande à n entrées et une sortie (MISO) est caractérisé par :
- une base de connaissances exprimée sous formes de règles du type :

règle i: si X_i est A_{i1} ... et X_n est A_{in} alors Y est B_i

où les X_i sont les variables d'entrée, Y est la variable de sortie, les A_{ij} et les B_i sont des SEF. Les règles sont connectées par des "Sinon". Si les entrées sont évaluées par rapport à p SEF, il y a p^n règles au maximum.
- une méthode d'évaluation des entrées par rapport aux SEF (fuzzyfication) et une méthode de passage du flou au réel pour la sortie Y (défuzzyfication).
- un mécanisme d'inférence permettant d'évaluer l'ensemble des règles et d'en tirer une conclusion.

Ceci peut être illustré par le schéma de la figure I-1.

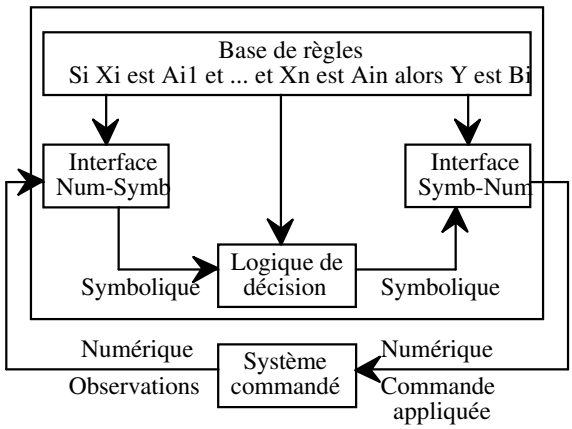


Figure I-1 : Schéma général d'un système de commande floue.

3-Optimisation d'un système d'inférence floue (SIF)

3-1-Problèmes posés

Les contrôleurs classiques traitent des systèmes qui ont des modèles mathématiques précisément définis. Dans la conception d'un SIF on n'a pas besoin de connaître le modèle mathématique précis du système à contrôler. Si toutefois le modèle existe, il peut être utilisé pour la simulation et pour le test de la stratégie de commande. On peut citer quelques autres caractéristiques des SIF :
- Il est possible d'implémenter les connaissances et l'expérience d'un opérateur, concepteur ou expert humain en utilisant des règles linguistiques compréhensibles et faisant appel à des notions imprécises.
- La distinction entre systèmes linéaires et non linéaires n'existe pas.

- Grâce aux fonctions d'appartenance on peut faire le lien entre la connaissance linguistique qualitative et les données quantitatives.

Mais face à ces avantages, le concepteur d'un SIF se trouve affronté à un certain nombres de problèmes. En effet, une base de règles floues est le résultat de l'expression d'une expertise. On cherche à modéliser un système physique, le savoir-faire d'un opérateur ou d'un expert. Or il n'y a pas de méthode systématique et standard pour la transformation des connaissances humaines ou de l'expérience en base de règles et aucune procédure générale pour choisir le nombre optimal de règles. La connaissance des opérateurs humains est souvent incomplète et non systématique. En effet, un opérateur expérimenté prend des décisions qualitatives sur des situations spécifiques dont il n'a qu'une connaissance imparfaite mais, grâce à son expérience il peut s'adapter facilement et rapidement à des situations nouvelles ou imprévues. En pratique, une démarche empirique de mise au point de SIF est coûteuse en temps et ne garantit pas des résultats satisfaisants sur la cohérence et la complétude de la base de règles. Plus généralement la définition d'un SIF suppose des choix structurels et des choix paramétriques:

- Choix structurels :
    1- choix des variables d'entrée et de sortie ;
    2- choix du nombre des fonctions d'appartenance pour chaque variable;
    3- choix des relations entre entrées et sorties de la base de règles ;
    4- choix du nombre de règles.
- Choix paramétriques :
    1- paramètres définissant les fonctions d'appartenance d'entrée des règles ;
    2- paramètres définissant les sorties des règles.

Parmi ces choix, ceux susceptibles d'être effectués le plus facilement par une procédure d'optimisation automatique sont les paramètres des fonctions d'appartenance et éventuellement le nombre de règles.

3-2- Solutions proposées

Le contexte du recours à une technique d'optimisation peut être varié. On peut désirer :

1- identifier un système (ou le comportement d'un opérateur) le plus finement possible.
2- minimiser la complexité d'une base de règles floues en déterminant le nombre minimal de règles et de fonctions d'appartenance pour résoudre correctement un problème donné.
3- optimiser une base de règles floues écrite manuellement.
4- rendre une base de règles floues portable d'un système sur un autre de principe semblable, mais différent quantitativement.
5- obtenir un contrôleur ou un SIF muni d'une faculté d'apprentissage, pour s'adapter en temps réel aux modifications du processus.
etc...

Tous ces cas ne peuvent être résolus par un seul algorithme, c'est pourquoi de nombreuses recherches ont été effectuées et des solutions très variées ont été proposées. Le tableau reproduit ci-après résume quelques grandes classes d'algorithmes existants.

⇨ contrôleur auto-organisé (S.O.C.) [PROCYK 79] [GUELY 89]
⇨ apprentissage par renforcement [LEE 91] [BERENJI 92]
⇨ gradient [GLORENNEC 91] [NOMURA 91] [JANG 93]
⇨ mémoire associative floue [KOSKO 92]
⇨ algorithmes génétiques [BERSINI 92] [GLORENNEC 92]
⇨ recuit simulé [ISAKA 90]
⇨ identification de modèle flou [TAKAGI 85]
⇨ "clustering" + "backpropagation" [TAKAGI 88]

3-3- Approche dite "neuro-floue"

Les similarités et les complémentarités des techniques neuronales et floues ont été souvent discutées dans la littérature [KOSKO 92] [TAKAGI 90]. Les systèmes à réseaux de neurones et à logique floue sont des structures parallèles à caractère paramétrique.

Les formes des fonctions d'appartenance des systèmes flous et les fonctions d'activation des réseaux de neurones peuvent être similaires. Les opération de multiplication-addition des neurones artificiels sont très proches des opérateurs "Sup-Min" du raisonnement approximatif.

- Le raisonnement utilisé dans un système d'inférence floue présente une structure que l'on peut qualifier de "pré-neuronale" : chaque étape bien distincte peut en effet être assimilée au calcul effectué au niveau d'une couche dans un réseau neuronal multicouches.

D'un point de vue pratique, on peut également reconnaître plusieurs points communs :

- Les deux méthodes, appliquées au contrôle, présentent de bonnes qualités de robustesse et sont capables d'interpolation et de généralisation.

- Aucune des deux méthodes ne nécessite de modèle mathématique du système à contrôler et elles peuvent donc toutes deux s'appliquer à des systèmes non linéaires sans complication particulière.
- Elles présentent toutes deux une assez grande facilité de mise en oeuvre.

Cependant, dans les réseaux neuronaux, l'inclusion, l'extraction et la représentation des connaissances sont difficiles. Les avantages des systèmes flous sont que les connaissances sont représentées sous forme de règles linguistiques compréhensives et qu'il sont capables de traiter des informations imprécises. Leurs points faibles sont causés essentiellement par la difficulté à définir avec précision les fonctions d'appartenance et le manque de procédure systématique pour la transformation des connaissances d'un expert en une base de règles.

Ces différentes raisons ont conduit, à partir du début des années 90, à la création d'un domaine de recherche, communément désigné par "neuro-flou", cherchant à fusionner les deux techniques. Il s'agissait de les "combiner" soit de façon à conserver les avantages des deux, tout en évitant les inconvénients, soit en exploitant les analogies entre les deux. Il y a deux principales méthodes :

- doter le système flou de fonctions d'apprentissage ou reproduire le modèle de traitement à partir du contrôleur flou ;
- transcrire le SIF directement dans une structure de réseau de neurones.

Dans la suite de ce chapitre nous présentons un réseau neuro-flou proposé par P.Y. GLORENNEC [GLORENNEC 91] qui présente une grande souplesse d'utilisation et des performances intéressantes en identification [BRUNET 96]. Nous montrons sur un exemple académique comment on peut s'en servir pour optimiser les performances d'un contrôleur flou et extraire automatiquement des règles de commande. Au chapitre suivant, ce type de contrôle sera utilisé pour la navigation réactive d'un robot mobile.

4-Description de la technique d'optimisation adoptée

4-1-Principe

Cette technique est appliquée à un SIF présentant une complète analogie structurelle avec un système d'inférence floue de type "SUGENO". Nous raisonnerons sur un exemple (qui peut facilement se généraliser) à deux entrées x_1 et x_2, dont la connaissance s'exprime sous la forme de règles du type :

règle i : Si x_1 est A_{i1} et x_2 est A_{i2} alors y est W_i

dans lesquelles les conclusions W_i sont des valeurs exactes (non floues), pouvant être appelées poids par analogie aux réseaux de neurones.

La prise de décision est faite à l'issue des étapes suivantes :

- Pour une entrée donnée (x_1,x_2), on calcule les différents degrés d'appartenance aux sous-ensembles flous A_{i1} ou A_{i2} apparaissant dans la partie prémisse des règles. C'est l'étape de fuzzyfication.
- Pour chaque règle, on calcule la valeur de vérité αi telle que :

$\alpha i = ET((x_1 \text{ est } A_1), (x_2 \text{ est } A_2))$

- Enfin, il faut agréger les règles et tirer la conclusion. Les W_i des parties conclusion des règles sont pondérés par les valeurs de vérité des prémisses :

$$y = \frac{\sum_i \alpha_i \times W_i}{\sum_i \alpha_i}$$

Chacune de ces étapes sera représentée par les neurones d'une des couches du réseau neuro-flou (fig.I-2) :

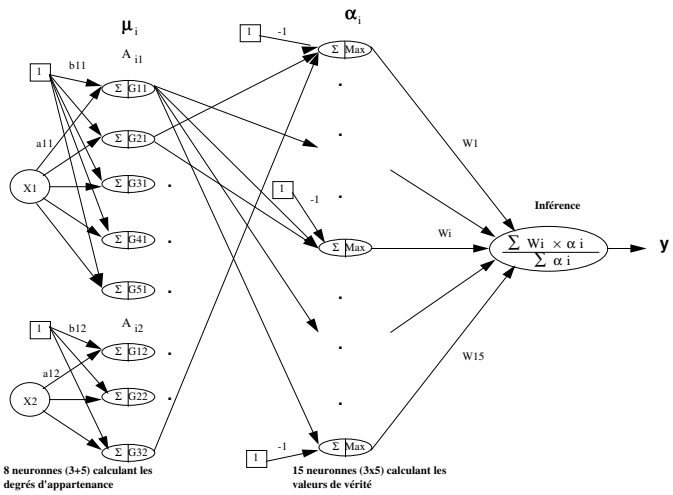
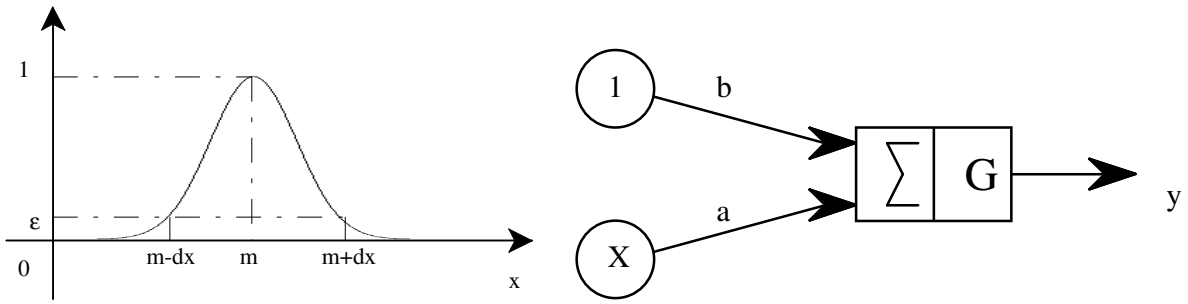


Figure I-2 : Exemple de représentation d'un réseau neuro-flou.

Ce réseau comprend donc deux couches cachées :
- la première calculera les degrés d'appartenance des entrées à leurs SEF : les poids du réseau entre la couche d'entrée et cette couche correspondront aux paramètres des fonctions d'appartenance.
- la seconde calculera les valeurs de vérité à l'aide des poids entre les deux couches cachées.
Les parties conclusion des règles (W_i) correspondront, elles, aux poids du réseau entre la seconde couche cachée et la sortie Y.

4-2- Réalisation des fonctions d'appartenance

Les fonctions d'appartenance sont des Gaussiennes caractérisées par leur centre m et leur largeur dx. Cette largeur est ici fixée à partir de la valeur résiduelle ε de la courbe aux points m+dx et m-dx. L'équation d'une telle Gaussienne est donnée par : $y=\exp(- (ax+b)^2)$ où $a=x - \ln(\varepsilon)/dx$ et $b= - am$
La réalisation sous la forme d'un neurone formel est représentée sur la figure I-3.



a)Fonction d'appartenance. b) Réalisation neuronale d'une Gaussienne.

Figure I-3

La fonction G est la fonction d'activation du neurone. Il s'agit ici d'une sigmoïde :

$$y = \frac{1}{1 + e^{-\frac{(x-a)}{b}}}$$

4-3- Calcul des valeurs de vérité

Dans un contrôleur flou le "ET" entre les prémisses des règles peut être calculé de diverses manières qui répondent à la définition d'une T-norme.
On utilise ici le "ET" de LUKASIEWICZ à cause de sa réalisation simple sous forme neuronale :
$ET(A,B) = \text{Max}(0, \mu_{A(x)}+\mu_{B(y)}-1)$.
La réalisation neuronale de ce "ET" est donnée sur la figure I-4a mais l'algorithme de rétropropagation suppose la dérivabilité des fonctions d'activation des neurones. Au niveau de la deuxième couche cachée, la fonction y=Max(0,X) n'étant pas dérivable en 0, elle sera approximée par une sigmoïde (fig. I-4b). On peut utiliser par exemple la sigmoïde d'équation : $y = 1/(1+ e^{-(X-0.5)/0.227})$.

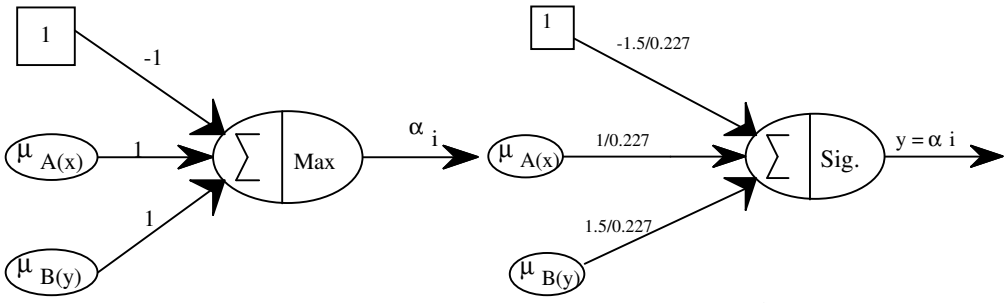


Figure I-4 : a)Implémentation neuronale du ET de Lukasiewicz
b)Approximation de la fonction y = Max(0,X) par une sigmoïde

L'apprentissage d'un tel réseau diffère assez nettement dans sa méthode de l'apprentissage d'un réseau de neurones multicouches traditionnel. Des différences sensibles existent en effet entre les deux structures :
- au contraire d'un réseau traditionnel, les neurones ne sont pas indifférenciés, mais fortement spécialisés ;
- le réseau n'est pas entièrement connecté : toutes les liaisons ne sont pas établies entre les neurones d'une couche et ceux de la couche suivante (par exemple un neurone de la première couche - une des entrées - n'est en liaison qu'avec certains neurones de la seconde couche) ;
- l'initialisation des paramètres n'est pas aléatoire. Les poids a_i et b_i de la couche 1 sont choisis par l'opérateur pour réaliser les fonctions d'appartenance initiales souhaitées. Les poids de la couche 2 définissant le "ET" sont fixes. Les poids Wi de la couche 3 sont généralement initialisés à zéro (génération automatique des règles), mais peuvent aussi servir à introduire de la connaissance a priori sur les conclusions des règles.

4-4- Algorithme d'apprentissage

Plusieurs algorithmes sont utilisables pour optimiser les paramètres ajustables du réseau. Dans nos applications pour lesquelles la simplicité de l'algorithme est un critère important, en fonction des ressources disponibles, nous nous contenterons d'utiliser la méthode de rétropropagation du gradient classique des réseaux neuronaux multicouches, adaptée au cas du réseau neuro-flou utilisé [GLORENNEC 91].
A chaque itération, on modifie donc les poids, en commençant par les poids de la couche 3 pour arriver aux poids de la couche 1. Cette modification des poids se fait dans le sens opposé au gradient de la fonction coût. On répète le processus jusqu'à ce que la fonction coût soit inférieure à une valeur seuil; on dit alors que le réseau a convergé.

Les poids sont modifiés suivant la formule suivante : $W_{ij}^{n}(t) = W_{ij}^{n}(t-1) + \Delta W_{ij}^{n}(t)$

$$\Delta W_{ij}^{n}(t) = -\eta.\delta_i^n.x_j^{n-1} + \alpha.\Delta W_{ij}^{n}(t-1)$$

W_{ij}^{n} : poids synaptique entre le i^{ème} neurone de la couche n et le j^{ème} neurone de la couche n-1 ;

η : gain d'apprentissage ou pas d'apprentissage ; cette valeur gère la vitesse à laquelle les poids vont converger vers leur valeur finale. Plus elle est grande, plus la convergence est rapide. Il faut toutefois éviter de prendre une valeur trop grande, car on risquerait alors de ne pas avoir de convergence du tout (les variations de poids entre deux itérations seraient trop importantes). A l'opposé, si cette valeur est trop faible, la convergence risque d'être extrêmement longue.

α : ce paramètre, appelé "moment", est compris entre 0 et 1 ; l'introduction du terme $\alpha.\Delta W_{ij}^{n}(t-1)$ permet d'éviter les minima locaux en faisant intervenir les variations de poids des itérations précédentes, ce qui contribue à donner un effet d'inertie à la descente du gradient.

x_j^{n-1} : sortie du j^{ème} neurone de la couche n-1.

δ_i^n : terme de dérivée de la fonction coût (i^{ème} neurone de la couche n). Pour les 3 couches on a donc :

$$\delta_i^4 = \frac{d - y_k}{\sum_j x_j^3}$$

$$\delta_i^3 = f'(\sum_j W_{ij}^2 x_j^2)\sum_k \delta_k^4 (W_{ki}^4 - y_k)$$

$$\delta_i^2 = g'(\sum_j W_{ij}^1 x_j^1)\sum_k \delta_k^3 W_{ki}^3$$

avec : d_j : sortie désirée

y_k : sortie effective du $k^{ème}$ neurone

f : fonction d'activation des neurones de la couche 3.

g : fonction d'activation des neurones de la couche 2.

5-Application au contrôle

Dans le contrôle de processus dynamiques, le contrôleur doit reproduire la fonction de transfert inverse du système pour déterminer la commande à fournir. Cela est aisé dans certains cas simples, par exemple quand le système peut être modélisé sous une forme linéaire, mais le plus souvent la structure du système soit est inconnue, soit présente des non-linéarités non modélisables. Nous nous plaçons dans ce cas, c'est à dire que nous supposons n'avoir aucune connaissance a priori sur la processus à contrôler. Le principe de la méthode de contrôle que nous utilisons est celui du "distal control" [JORDAN 91], repris par [RENDERS 94] sous le nom de JEAN (Jordan method Extended for Adaptive Neuro-control).

5-1- Architecture de contrôle JEAN

Cette architecture (fig. I-5) nécessite la présence de deux réseaux qui seront ici neuro-flous :
- un premier réseau pour identifier le processus (Modèle Neuro-Flou) ;
- un second réseau pour contrôler le processus (Contrôleur Neuro-Flou).

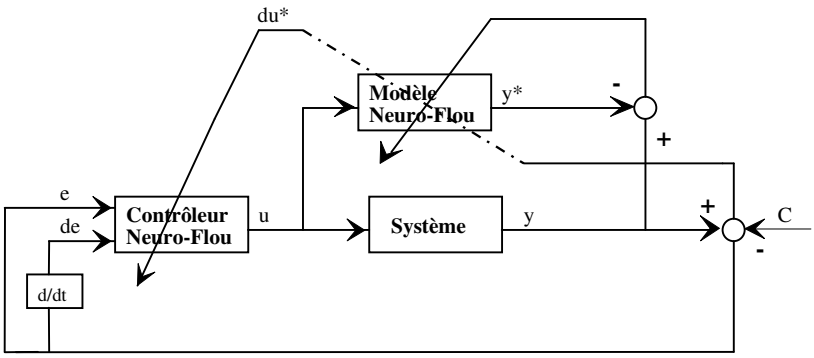


Figure I-5 : Architecture de contrôle JEAN

Cette méthode nécessite deux étapes successives : l'identification du processus par le réseau MNF, puis l'apprentissage du réseau CNF. L'ajustage des poids synaptiques du réseau CNF est effectué en "rétropropageant" à travers le réseau neuro-flou la valeur du* obtenue par rétro-propagation à travers MNF de l'erreur en sortie : e = y-c. Ce qui correspond à une fonction coût égale à l'erreur quadratique.

5-2- Architecture de contrôle simplifié mini-JEAN

Mais en fait, souvent il n'est pas nécessaire de connaître précisément le modèle du processus (ou plus exactement son Jacobien) pour obtenir un contrôle correct. En effet, un modèle imprécis du système n'altère pas la minimisation de la fonction coût, il modifie seulement le trajet pris par l'algorithme d'apprentissage dans l'espace des actions : on ne suivra sans doute pas la plus forte pente, mais on restera sur un chemin descendant [Jordan 91]. Le moyen consistant à utiliser l'approximation du Jacobien par son signe (plus facile à connaître que le Jacobien lui même) [SAERENS 91] est une approche viable pour la commande de systèmes simples. On arrive alors à une architecture (fig. I-6) ne faisant pas intervenir de réseau émulateur. Elle utilise un seul réseau - qui sera ici neuro-flou - pour le contrôleur, dont l'apprentissage s'effectue en rétropropageant directement l'erreur de sortie.

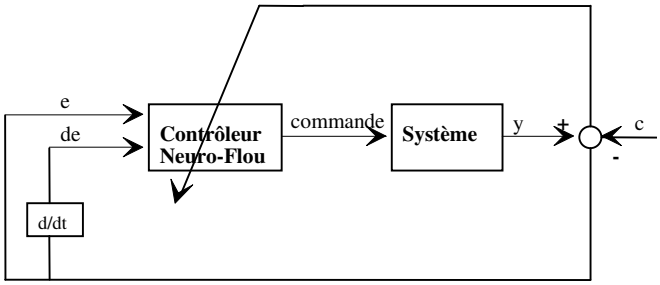


Figure I-6 : Architecture de contrôle simplifiée mini-JEAN

RENDERS [RENDERS 94] baptise "mini-JEAN" cette architecture. Il l'a comparé avec l'architecture JEAN. Il obtient des performances équivalentes, tant pour la vitesse de convergence (vitesse d'apprentissage) que pour l'erreur moyenne en généralisation. Par contre, le temps de calcul est nettement en faveur de mini-JEAN.
Nous avons également observé [BRUNET 96] que, malgré quelques différences dans la conduite de l'apprentissage et dans la commande fournie, il n'existe pas d'écart de performance sensible entre la méthode de Jordan initiale et sa version simplifiée. C'est pourquoi, notre préférence ira à cette méthode simplifiée, efficace, rapide et facile à mettre en œuvre. Il est évident néanmoins qu'elle ne peut s'appliquer qu'à des systèmes dans lesquels le signe du jacobien est constant dans le domaine de fonctionnement balayé.

5-3- Exemple d'application

5-3-1- Présentation du problème

Pour illustrer le potentiel d'une telle technique dans les applications de contrôle des processus, nous choisissons un système du premier ordre, avec une non linéarité. Il s'agit d'effectuer une régulation de la vitesse de rotation d'un moteur en présence de frottement, décrite par l'équation discrète :

$y(t+\Delta t) = y(t) + \Delta t/J\ [u(t)-Kf\ Signe(y(t))]$

avec : y = vitesse de rotation du moteur ; u = couple moteur ; Δt = pas d'échantillonnage ; J = moment d'inertie ; Kf = module du couple de frottement ; Les conditions d'études sont les suivantes: Consigne = ± 0.1 ; Kf = 0.1 ; J = 1 et Δt = 0.05 s.
Afin de bien mettre en évidence le phénomène de non-linéarité lié au signe de **y**, l'objectif sera d'obtenir une rotation alternée (consigne en créneau d'une durée de 10 s).
Le contrôleur neuro-flou utilisé est de type Proportionnel Intégral (CNFPI), de loi de commande du=f(e,de). Les caractéristiques de ce contrôleur sont les suivantes :
- Les fonctions d'appartenance les entées **e** et **de** du contrôleur sont des gaussiennes au nombre de 5 pour chaque entrée, régulièrement réparties sur l'univers de discours (fig. I-7);

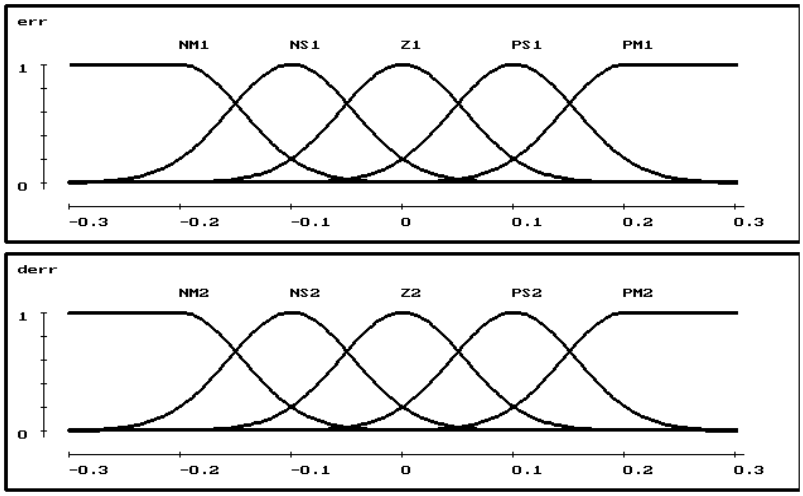


Figure I-7 : Fonctions d'appartenance utilisées pour e et de

-Les valeurs de vérité sont calculées avec le "ET" de Lukasiewicz.

5-3-2- Première expérience

Les sorties du réseau sont initialisées à zéro (pas d'introduction de connaissance a priori) et l'optimisation du contrôleur est effectuée entièrement "en ligne". La modification des poids du réseau par rétropropagation du gradient a porté simultanément sur la couche 1 et 3 du réseau, c'est à dire sur la largeur et la position des fonctions d'appartenance des entrées et sur les conclusions des règles. L'apprentissage a été mené sans imposer aucune contrainte sur les poids de la couche 3 et avec une saturation de la commande à ±2 de façon à prendre en compte des conditions physiques réelles.
Nous avons choisi un gain d'apprentissage relativement fort (de l'ordre de l'unité) pour les 20 premières itérations de façon à se rapprocher rapidement de la solution, puis le gain est réduit (de l'ordre de 0.1) pour affiner les résultats, avec un moment d'une valeur de 0.2 pour réduire les oscillations.
Dans la figure I-8,on présente le réponse du moteur en début et fin d'apprentissage.

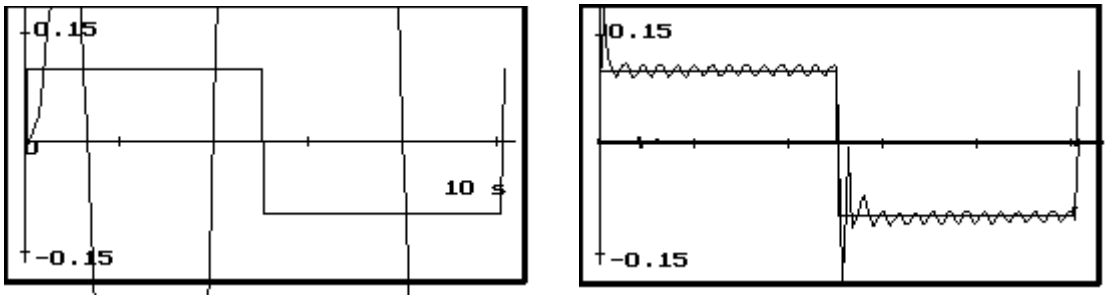


Figure I-8 : Réponse du système contrôlé en début et fin d'apprentissage

On constate l'existence d'oscillations persistantes de la sortie autour de la valeur de la consigne. Si on regarde l'évolution des poids de la couche de sortie du réseau (fig. I-9), on remarque que cette évolution est très chaotique au début de l'apprentissage, puis les valeurs des poids augmentent continuellement même au bout d'un grand nombre de périodes. Par ailleurs la table de ces poids ne présente aucune régularité, aucune symétrie, même pas de zéro central. C'est à dire que pour un processus en quasi équilibre par rapport à la consigne, une variation de commande est fournie, ce qui est illogique et provoque des oscillations constatées sur la figure I-8. Dans ces conditions, chercher une signification linguistique aux règles de commande obtenues n'a aucun sens.

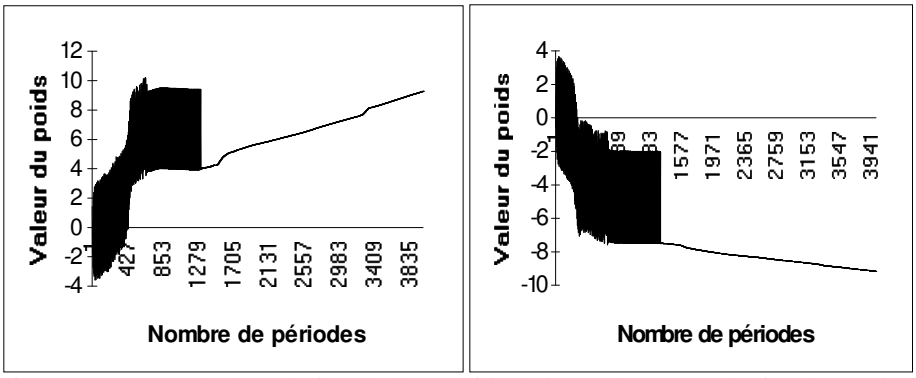


Figure I-9 : Exemple d'évolution de deux poids de la couche 3 (conclusion des règles) au cours de l'apprentissage

Nous constatons donc dans cette expérience deux problèmes majeurs :
- la perte de toute logique dans la "table de règle" ;
- la dérive continue des poids.

Pour traiter le premier problème nous allons décomposer l'apprentissage en deux étapes :
1) optimisation des poids de la couche 3 seule (conclusion des règles).
2) optimisation globale avec contrainte sur les poids de la couche 1 (fonctions d'appartenance des entrées).
Pour traiter le second problème nous allons introduire une modification de la fonction coût utilisée.

5-3-3- Introduction d'une régression des poids de sortie

Une solution simple pour combattre la dérive des poids est de les contraindre par une modification de la fonction coût sous la forme :

$J = E + \lambda \sum w^2$, où λ est une constante qui contrôle l'augmentation des poids et E le terme d'erreur quadratique. En fait cette modification a été utilisée dans [CHOW 95] sous le nom de "weight decay " pour traiter

des problèmes de classification par réseaux de neurones. Cette technique a été analysée [BISHOP 95] dans le cadre de la théorie de l'apprentissage et il a été montré que c'était une façon très simple d'implémenter une méthode de "régularisation" dans un réseau de neurones afin d'optimiser le compromis entre l'erreur d'apprentissage et l'erreur de généralisation.
La modification de l'algorithme de rétropropagation du gradient pour intégrer l'effet du terme de régression des poids dans la fonction coût est simple. En effet, il suffit de dériver la fonction coût par rapport à chaque poids W. $\frac{\partial J}{\partial w} = \frac{\partial E}{\partial w} + 2\lambda w$; ce qui conduit à une modification de ce poids par l'algorithme de rétropropagation du gradient telle que : $w(k+1) = w(k) + \eta\left(\frac{-\partial J}{\partial w}\right) + \alpha\Delta w(k)$; si on pose $\beta = 2\eta\lambda$, l'équation devient :

$$w(k+1) = w(k) + \eta\left(\frac{-\partial E}{\partial w}\right) + \alpha\Delta w(k) - \beta w(k).$$

Comme il s'agit ici d'un réseau neuro-flou, on adapte cette formule en multipliant β par le terme de déclenchement de la règle, à savoir $\frac{\alpha_i}{\sum_i \alpha_i}$. α_i est la valeur de vérité de la partie prémisse de la règle déclenchée.

Choisir β détermine le poids dans la fonction coût du terme de régression par rapport à celui de l'erreur quadratique. S'il est trop petit par rapport au gain, l'effet de la régression des poids ne se fera pas sentir. En revanche, si β est trop important, les poids resteront proches de zéro, car l'effet de diminution des poids sera trop important par rapport à l'augmentation due à la rétropropagation normale. Dans ce cas, le contrôleur neuro-flou n'apprendra rien.
Il est également important que le terme de régression des poids soit proportionnel au déclenchement de la règle. Si on n'introduisait pas ce coefficient, le poids baisserait quand la règle n'est pas déclenchée. Donc si, par exemple, le système évoluait suffisamment longtemps sans déclencher une règle, le poids correspondant tendrait vers 0, et on perdrait alors de l'information.
Dans ces conditions (couche 1 bloquée) et avec des poids de la couche 3 initialement nuls, les résultats obtenus sont très satisfaisants. La réponse fournie par le CNFPI optimisé est très proche de la réponse désirée (créneau) (fig. I-10).

Figure I-10 : Réponse et variation de la commande du système
contrôlé par CNFPI optimisé avec régression des poids

Mais, ce qui est encore plus important, on constate maintenant que les poids évoluent de façon "douce" et tendent vers des constantes (fig. I-11).

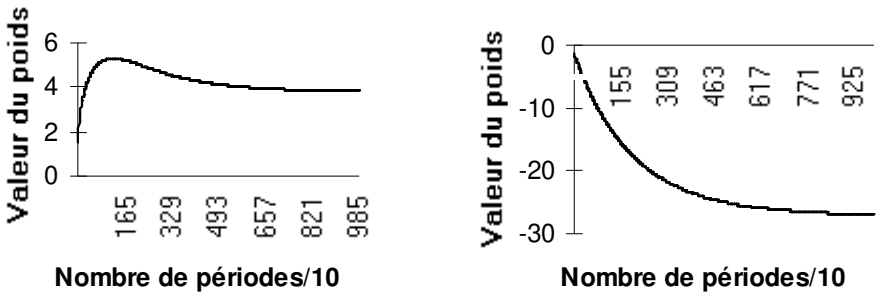


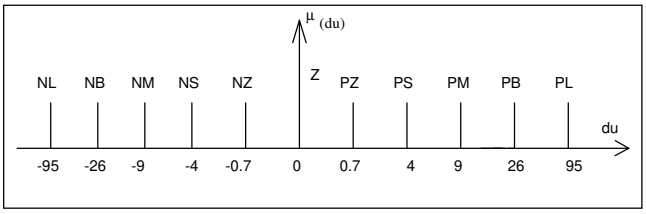
Figure I-11 : Evolution de deux poids de la couche de sortie
au cours de l'apprentissage avec régression des poids

La table I-1 des règles, obtenue après convergence des poids, est donnée ci-après. On note que cette table présente un zéro central et une symétrie parfaite par rapport à ce zéro central, ce qui est parfaitement logique, compte tenu de l'application.

de \ e	NM1	NS1	Z1	PS1	PM1
PM2	-0.01	10.7	0.72	28.2	95.7
PS2	-9.81	0.11	0.67	3.82	26.9
Z2	-0.63	-0.66	0.00	0.66	0.63
NS2	-26.9	-3.82	-0.67	-0.11	9.81
NM2	-95.7	-28.2	-0.72	-10.7	0.00

Table I-1 : Table des poids de sortie correspondant au CNFPI (avec régression des poids)

On peut donner maintenant une interprétation linguistique à cette table. Il faut pour cela convenir d'une échelle de traduction numérique-symbolique. Nous convenons d'attribuer le concept PL (Positive Large) aux valeurs numériques proches de 95, le concept PB (Positive Big) aux valeurs numériques proches de 27, le concept PM (Positive Mean) aux valeurs numériques proches de 9, le concept PS (Positive Small) aux valeurs proches de 4, le concept PZ (Positive Zéro) aux valeurs proches de 0.7 et le concept Z (Zéro) aux valeurs proches de 0, et de même dans le sens négatif (fig. I-12a). On obtient la table de la figure I-12b.



a)Expression linguistique de la commande

de \ e	NM1	NS1	Z1	PS1	PM1
PM2	Z	PM	PZ	PB	PL
PS2	NM	Z	PZ	PS	PB
Z2	NZ	NZ	Z	PZ	PZ
NS2	NB	NS	NZ	Z	PM
NM2	NL	NB	NZ	NM	Z

b) Interprétation linguistique de la Table

Figure I-12

Cette table présente une grande analogie avec la table dite de MCVICAR [MCVICAR 76] qui est traditionnellement utilisée comme base pour la mise au point de contrôleurs PI-flou et qui a une forte structure logique. On a une diagonale de Z et une symétrie par rapport à cette diagonale. Par contre, on constate une répartition un peu différente des termes linguistiques de valeurs faibles.

5-3-4- Optimisation avec introduction de connaissances

Nous débutons l'apprentissage après chargement des poids issus de la table de MCVICAR (fig. I-13a) tels que les termes linguistiques sont représentés par des singletons correspondant aux sommets des fonction d'appartenance de la sortie [BRUNET 95]. L'évolution de l'erreur moyenne par période est quelque peu chaotique en début d'apprentissage, puis elle se régularise et décroît au même temps que les poids convergent vers des

valeurs asymptotiques. On constate que la table obtenue (fig. I-13b) est très proche de celle obtenue au paragraphe précédent. Ceci montre donc que, quelle que soit l'initialisation effectuée, la convergence se fait vers une solution identique et prouve indirectement que l'optimisation ne conduit pas à un minimum local.

de \ e	NM1	NS1	Z1	PS1	PM1
PM2	0	0.33	0.67	0.67	0.67
PS2	-0.33	0	0.33	0.67	0.67
Z2	-0.67	-0.33	0	0.33	0.67
NS2	-0.67	-0.67	-0.33	0	0.33
NM2	-0.67	-0.67	-0.67	-0.33	0

a)Table de Mc Vicar initialisant les poids

de \ e	NM1	NS1	Z1	PS1	PM1
PM2	0.02	10.6	0.72	28.2	95.7
PS2	-9.81	0.10	0.67	3.80	26.9
Z2	-0.63	-0.66	0.00	0.66	0.63
NS2	-26.9	-3.80	-0.67	-0.10	9.81
NM2	-95.7	-28.2	-0.72	-10.6	-0.02

b)Table de poids obtenus après convergence

Figure I-13

L'introduction de connaissance a priori n'a pas d'effet franchement positif sur la rapidité d'apprentissage contrairement à ce qu'on pourrait penser intuitivement. Par contre, cette initialisation permet un démarage de l'apprentissage dans des circonstances favorables pour le processus en lui évitant de forts à-coups de commande.
Remarque : dans le cas où la rétrogropagation ne se fait que sur la couche 3, l'utilisation de fonctions d'appartenance gaussiennes pour les entrées n'est pas indispensable. Nous constatons que l'emploi de fonctions d'appartenance de forme triangulaire équiréparties sur leur univers de discours ne modifie pas le comportement du contrôleur, mais facilite plutôt l'interprétabilité du SIF optimisé.

5-3-5- Optimisation complète du SIF

Après convergence des poids de la troisième couche, nous faisons porter l'apprentissage simultanément sur les couches 1 et 3 du réseau. Nous avons introduit des contraintes en imposant aux centres des fonctions d'appartenance correspondant aux concepts linguistiques "Zéro" de ne pas se déplacer afin de conserver l'interprétabilité du SIF. Une fois que les poids sont stabilisés, le système fournit une réponse et une table de décision très proches de celles déjà obtenues avec un apprentissage ayant porté seulement sur les poids de la couche 3.

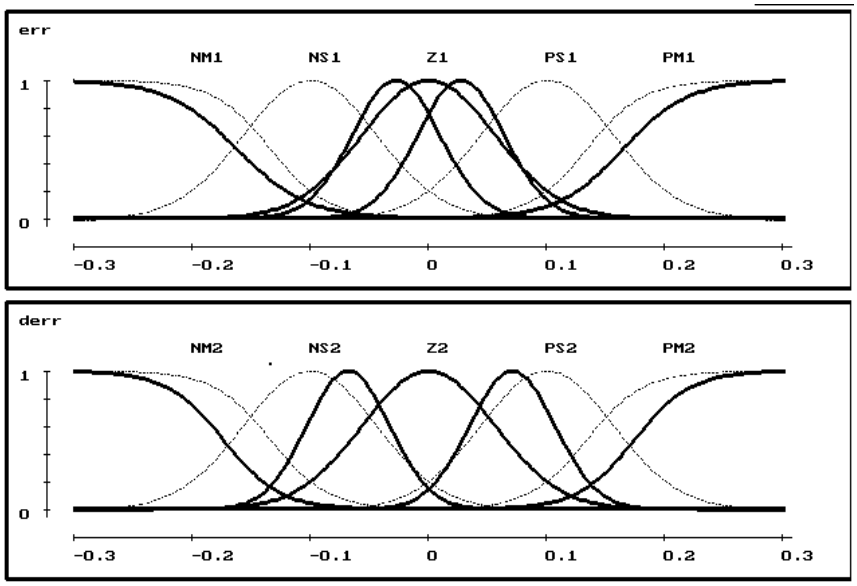


Figure I-14 : Fonctions d'appartenance pour e et de au début (----) et à la fin (——)d'apprentissage

En étudiant l'évolution des centres des gaussiennes au cours de l'apprentissage, on n'observe pas de mouvements désordonnés et, en fin d'apprentissage, tous sont stabilisés (fig. I-14). Pour ce qui concerne le déplacement des fonctions d'appartenance associées à l'erreur e (il en est de même mais moins fortement pour la variation de l'erreur de), nous remarquons que NS1 et PS1 se sont déplacés vers Z1, contrairement aux déplacements de NM1 et PM1. Ceci signifie que pour e le nombre de SEF pourrait très probablement être réduit à trois.

D’après cette étude il est clair que, si le souci majeur est de construire un SIF optimisé avec la possibilité d’interpréter les conclusions des règles pour en extraire de la connaissance, il est peu utile de libérer les poids de la première couche.

6-Conclusion

Il apparaît à l’issue de ce travail préparatoire que le recours à une architecture dite “neuro-floue” est un moyen simple et efficace pour obtenir par apprentissage un contrôleur performant dont le comportement soit interprétable sous la forme de règles de décision. Nous avons pu constater que l’utilisation d’une structure simplifiée, munie de fonctions d’appartenance triangulaires et avec un apprentissage sur la dernière couche seule, permet d'obtenir une loi de commande efficace par apprentissage en ligne, même sans introduire de connaissance préalable au niveau des conclusions des règles.
Le non recours à la rétropropagation du gradient sur la première couche, nous affranchit de la représentation neuronale du SIF à optimiser, ce qui ne justifie plus l’appellation “neuro-floue” usuellement donnée à cette technique. Nous parlerons donc simplement de SIFO (Système d'Inférence Floue Optimisé).

Bibliographie I

[BERENJI 92]: H. R. BERENJI, *A Reinforcement Learning Based Architecture for Fuzzy Logic Control*, International Journal of Approximate Reasoning, No. 6, pp. 267-292, 1992.

[BERSINI 92]: H. BERSINI, *Génération Automatique de Systèmes de Commande Floue par la Méthode du Gradient et les Algorithmes Génétiques*, 2èmes journeés Nationales des Application des Ensembles Flous, pp. 199-209, Nîmes 1992.

[BRUNET 95]: M. BRUNET, H. MAAREF, C. BARRET, *Conception d'un contrôleur flou auto-ajustable,* MCEA'95, pp. 409-414, Grenoble, Sep. 13-14, 1995.

[BRUNET 96]: M. BRUNET, *Identification et contrôle de processus par Réseaux Neuro-flous*, Thèse de doctorat de l'université d'Evry val d'Essonne, Décembre 1996.

[CHOW 94]: M-Y. CHOW, *An analysis of Weight decay as a methodology of reducing three-layer feedforward artificial neural network for classification problems*, ICNN'94, Proceeding of the IEEE Int. Conf. on Neural Networks, Vol. 1, pp. 600-605, June 28-July 2, orlando, Florida, 1994.

[GLORENNEC 91]: P. Y. GLORENNEC, *Un réseau neuro-flou évolutif*, Proc. 4° Int. Conf. Neural Networks and their Applications, pp. 301-314, Nîmes, 1991.

[GLORENNEC 92]: P. Y. GLORENNEC, *Applications des Algorithmes Génétiques pour l'optimisation des Fonctions d'Appartenance d'un Réseau Neuro-Flou*, 2èmes Journées Nationales le Applications des Ensembles Flous, pp. 219-226, Nîme 1992.

[GUELY 89]: F. GUELY, I. TODO, *A New Self-Organizing Control Algorithm and its Application to the Control of a Robot Arm,* 20th International Symposium on Industrial Robots, Tokyo, pp. 137-144, October 1989.

[ISAKA 90]: S. ISAKA, A. V. SEBALD, *An Optimisation Approach For Fuzzy Controller Design,* 1990.

[JANG 93]: J.-S. R. JANG, *ANFIS :Adaptive-network-based fuzzy inference systems.* IEEE Trans. Systems, Man and Cybernetics.

[JORDAN 91]: M.I. JORDAN, Rumelhart D., *Internal world models and supervised learning*, Proc. of the eighth Int. Workschop on Machine Learning, pp.70-74, 1991.

[KOSKO 92]: B. KOSKO, *Neural networks and fuzzy systems*, Prentice-Hall, 1992.

[LEE 91]C. C. LEE, *A Self-Learning Rule-Based Controller Employing approximate Reasoning and Neural Net Concepts*, International Journal of Intelligent Systems, Vol. 6, pp. 71-93, 1991.

[MACVICAR 76]: P.J. MACVICAR-WHELAN, *fuzzy sets for man-machine interaction,* Int. J. Man-Machine Studies, vol. 15, pp. 687-697, 1976.

[NOMURA 91]: H. NOMURA, I. HAYASHI, N. WAKAMI, *A Self-Tuning Method of Fuzzy Control by Descent Method,* IFSA '91, Vol. Engineering, pp. 155-158, Bruxelles, 1991.

[PROCYK 79]: T. J. PROCYK, E. H MAMDANI, *A Linguistic Self-Organizing Process Controller,* Automatica, IFAC, Vol. 15, pp. 15-30, 1979.

[RENDERS 94]: J.M. RENDERS, *Métaphore biologiques appliquées à la commande de processus*, Thèse de Doctorat, faculté des sciences Appliquées, Université libre de Bruxelles, 1994.

[SAERENS 91]: M. SAERENS, *Approche connexionniste de la commande de processus*, Thèse de Doctorat, faculté des sciences Appliquées, Université libre de Bruxelles, 1991.

[TAKAGI 85]: T. TAKAGI, M. SUGENO, *Fuzzy identification of systems and its applications to modeling and control,* IEEE Trans. On Syst, Man and Cyber., Vol. 15, pp. 116-132, 1985.

[TAKAGI 88]: H. TAKAGI, I. HAYASHI, *Artificial neural network driven by fuzzy reasonning,* Int. Work. on Fuzzy Appl., pp. 217-218, 1988.

[TAKAGI 90]: H. TAKAGI, *Fusion technology of fuzzy theory and neural networks, survey and future directions,* In Proc. of the International Conf. on fuzzy Logic and neural Networks, pp. 13-26, Izuka, Japan, July 1990.

[ZADEH 65]: L.A.ZADEH, *Fuzzy sets*, Information Control, 1965.

[ZADEH 78]: L.A.ZADEH, *Fuzzy sets as a basis for a theory of possibility theory*, Fuzzy Sets and Systems, 1978.

[ZADEH 96]: L.A.ZADEH, *The roles of soft computing and fuzzy logic in the conception, design and deployment of intelligent systems,* Proc. Conf. IZUKA'96, pp. 3-4, Japan, 1996.

CHAPITRE II

Aides floues à la navigation d'un robot mobile

1-Introduction

D'importantes activités de recherche se sont développées au cours de ces deux dernières décennies dans le domaine de la navigation de robots mobiles autonomes. Plusieurs applications réussies ont été remarquées dans lesquelles les robots ont montré des caractéristiques de robustesse de navigation dans des milieux autres que des laboratoires, quoique bien structurés : environnements de type domotique [LAKEHAL 95], hôpitaux [SKEWIS 91], ateliers industriels, entrepôts [ARKIN 90], [COX 91], etc... Un des plus grands défis actuels posé à la communauté de recherche est d'étendre le domaine d'application de la navigation autonome à toutes les sortes d'environnements non structurés [PIN 93], [LACROIX 94]. Ceci est dû principalement à la nécessité de remplacer l'intervention humaine dans des milieux hostiles et/ou dangereux (nucléaire, spatial, sécurité civile, etc...).
Pour notre part, notre objectif essentiel est de développer des stratégies de navigation en environnements d'intérieurs à faible coût dans un but domotique [BENREGUIEG 97] et pour l'aide aux handicapés [HOPPENOT 96]. Dans ce contexte, le souci majeur est de construire des techniques de navigation efficaces, telles que la sécurité soit prioritaire par rapport à l'optimalité. La stratégie de navigation doit intégrer le fait que, généralement, à l'intérieur d'un bâtiment, on dispose d'un plan sur lequel sont positionnés les principaux éléments fixes de l'environnement : murs, portes, meubles lourds et fixes, etc … Mais de nombreux éléments non fixes et de position non connue a priori vont s'ajouter à cette structure de base. Dans ces conditions, deux cas extrêmes peuvent se produire. Si l'environnement perçu par le robot correspond à la carte (plan) mémorisée, il pourra suivre à vitesse maximum une trajectoire planifiée en utilisant une méthode globale. Si au contraire l'environnement n'est pas reconnu, un déplacement à vitesse réduite devra être géré par une méthode locale de navigation réactive. Entre ces deux situations limites, une évolution progressive se fera en fusionnant les commandes issues des deux modules en fonction du degré de reconnaissance de la scène mémorisée.
Dans un environnement totalement inconnu, la navigation se fait complètement de manière réactive. Une méthode classique pour faire cela est celle du champ des potentiels artificiels. Mais il est bien connu que cette méthode souffre du problème de gestion des minima locaux provoquant des situations de blocage. Nous proposons une solution basée sur un ajustage automatique du coefficient des forces attractive et répulsive par des règles floues. Néanmoins des problèmes d'oscillations demeurent, particulièrement dans les environnements étroits et les passages de porte, ce qui est gênant pour une robotique de service d'intérieur.
Nous proposons alors une démarche basée sur des systèmes d'inférence floue et inspirée du comportement humain qui consiste à atteindre l'espace libre tout en convergeant vers le but (stratégie S1), ce qui permet au robot d'éviter les minima locaux en passant dans un environnement encombré par le milieu de l'espace libre disponible. Néanmoins, nous rencontrons certaines situations d'échec dans le cas d'obstacles concaves. C'est pourquoi nous développons une deuxième stratégie S2 par coordination de S1 et d'un autre comportement élémentaire de type suivi de mur incluant la création de sous buts intermédiaires. L'idée est en fait d'anticiper pour éliminer une situation potentielle de blocage si le contexte le rend possible (c'est à dire lorsqu'un obstacle «concave» est détecté) au lieu de le constater et d'agir à postériori. Un obstacle sera en fait appelé «concave» lorsque tous les capteurs extéroceptifs utilisés mesurent simultanément des petites distances. En fait, nous préférons déclencher la stratégie S2 même si l'obstacle n'a réellement pas une forme géométrique concave, plutôt que de risquer une situation d'échec avec la stratégie S1. Le schéma ci-après (fig. II-1) résume la stratégie générale adoptée.
Tous les modules de navigation ont été d'abord réalisés avec des SIF standards. Ensuite, afin de rendre la génération et l'optimisation de règles floues automatiques en temps réel, nous avons utilisé la technique présentée dans le chapitre I. Nous proposons une technique d'apprentissage "localisé" qui utilise les propriétés de localité d'un SIF dans l'espace de sortie et non dans l'espace d'entrée ce qui permet de laisser l'optimisation activée en permanence et donc d'avoir un contrôleur flou adaptatif.

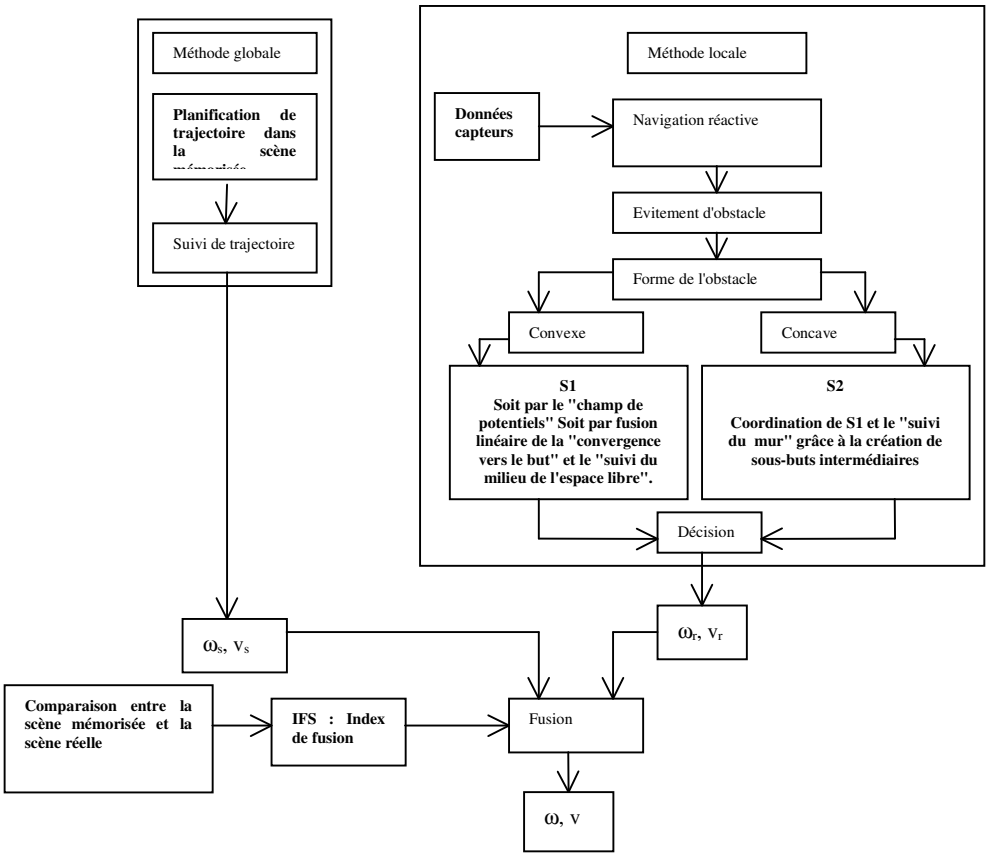


Figure II-1 : Stratégie générale

2- Robots mobiles utilisés dans le cadre de nos travaux et hypothèses

2-1- Le robot miniature Khepera®

L'étude et la mise en œuvre du système de commande ont été réalisées sur un robot miniature nommé Khepera® (fig. II-2) développé à l'Ecole Polytechnique Fédérale de Lausanne (EPFL) [MONDADA 93]. Ce robot (fig. II-3) qui est de dimensions très petites (55 mm de diamètre avec un entraxe de 50.4mm, 30 mm de hauteur et 70g de masse) à servi de valider les différents aspects du système de commande. Ses vitesses linéaire et angulaire maximales sont respectivement de 40 mm/s et de 1.58 rd/s. La vitesse fournie au robot est la vitesse linéaire de chacune des roues. Cette dernière est la somme ou la différence de la vitesse linéaire du robot et la vitesse angulaire du robot multipliée par le rayon du robot r.

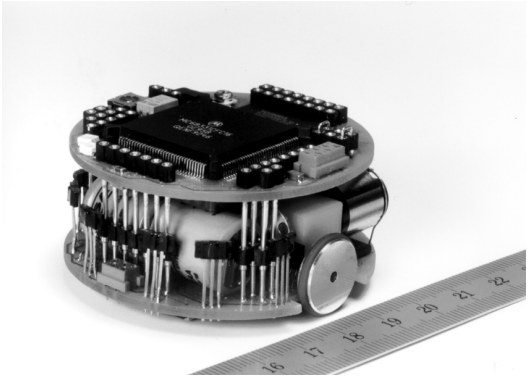


Figure II-2 : Le mini-robot mobile Khepera®.

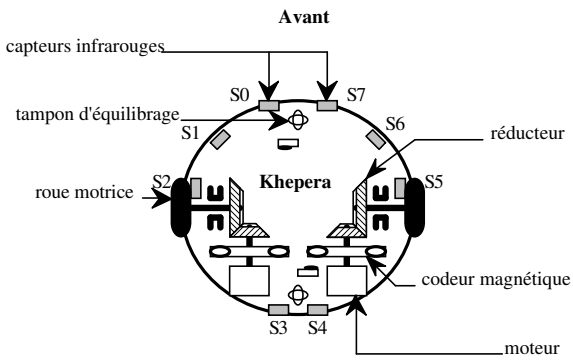


Figure II-3 : La structure physique de "Khepera".

2-2- Simulateur de Khepera

Le simulateur de Khepera (fig. II-4) utilisé a été conçu par MICHEL [MICHEL 95]. Ce simulateur permet de construire des mondes variés pour tester les différents algorithmes. Il prend en compte tous les bruits sur les capteurs infrarouges et sur l'odométrie.

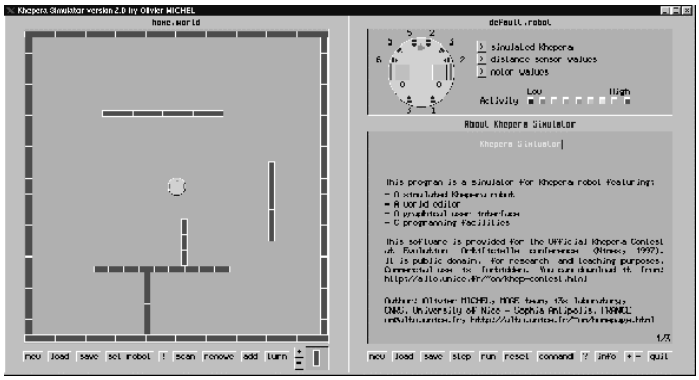


Figure II-4 : Le simulateur utilisé

2-3- Le robot RMI

Un autre robot, "RMI" (Robot Mobile Intelligent) (fig. II-5) développé dans notre laboratoire [HOPPENOT 97], de dimensions plus grandes a été également utilisé pour tester la portabilité et la robustesse des stratégies de commande.

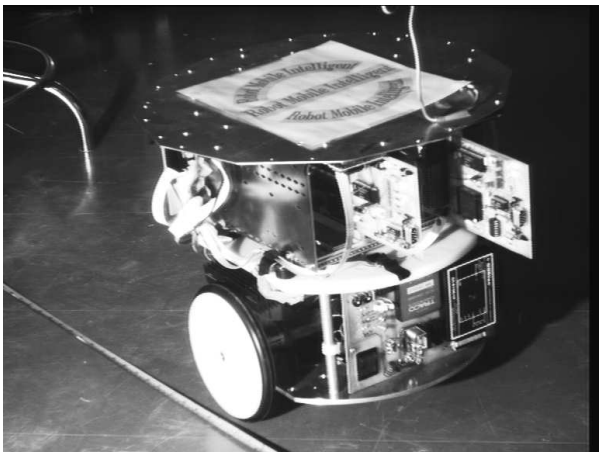


Figure II-5 : Le robot mobile RMI.

RMI est un robot circulaire à deux roues motrices. Il a 20 cm de hauteur et 40 cm de diamètre et peut supporter une charge de dix kilogrammes. La motorisation est assurée par deux moteurs à courant continu couplés à des codeurs incrémentaux (18 impulsions par millimètre de déplacement du robot).
Pour la perception de l'environnement, le robot possède huit capteurs ultrasonores "Polaroïd" autour de lui : sept devant et un derrière. L'angle du cône de mesure est de 15 degrés. La précision est de 3 cm pour une portée de 3m. La portée maximale de ces capteurs est de 10 m.

2-4-Hypothèses et contraintes

Afin de définir les conditions et les limites du bon fonctionnement des algorithmes, ce paragraphe expose les différentes hypothèses retenues pour réaliser cette étude :

- Les robots mobiles que nous utilisons présentent la caractéristique de non holonomie mais non contraignante.

- Le déplacement du véhicule s'effectue uniquement sur un sol plat (mouvement dans le plan horizontal). A tout moment, la position du robot pourra être exprimée par trois paramètres (x, y, θ).

- Le robot se déplace sur un sol adhérent. Le mouvement des roues motrices sur le sol est composé de roulement pur. Ceci nous assure, notamment durant la phase d'expérimentation des algorithmes sur les robots mobiles, de pouvoir mesurer les déplacements à partir des angles de rotation des roues. Ceci nous permet la connaissance, à

chaque instant t, de la position relative du robot par rapport à son point de départ et/ou son but. Dans la pratique, cette opération sera réalisée à l'aide des informations fournies par les encodeurs disposés sur chacune des roues motrices des robots. Il est clair que cette technique d'odométrie est source d'imprécision cumulative. C'est pourquoi, un module de localisation est prévu pour recaler la position du robot par rapport à un repère absolu (Chap. III).
- Les effets d'inertie seront considérés comme négligeables. En effet, étant donné que les vitesses de déplacement et les poids des robots sont faibles, les inerties de ces derniers n'influencent pas leurs mouvements.
- Le robot est considéré comme totalement autonome (c'est à dire, sans intervention humaine d'aucune sorte lors d'une mission).
- On possède une certaine connaissance a priori (plan) sur l'environnement, mais le robot doit pouvoir prendre en compte des modifications significatives (obstacles déformés, déplacés ou nouveaux) pouvant exister dans la scène réelle.

3- Présentation de la stratégie adoptée

Les techniques de planification globale supposent qu'un modèle complet de l'environnement est disponible, dans lequel les obstacles et l'espace libre sont définis. Ces techniques ne sont pas directement applicables dans un environnement partiellement ou totalement inconnu, par manque d'informations globales sur le chemin libre d'obstacles à suivre pour rejoindre le but.
Les techniques de navigation locale sont basées sur des techniques réactives utilisant les informations capteurs. Le principal inconvénient de ces méthodes est de ne pas toujours garantir une solution à cause de la possibilité de situations de blocage, dues au fait que le robot n'a qu'une vue très réduite de l'environnement. Pour plus de sécurité et d'efficacité, il faut multiplier les moyens de perception (par exemple : plusieurs caméras et divers capteurs de distance de natures différentes) pour obtenir des informations pertinentes sur l'environnement du robot. Mais l'analyse de ces informations est très difficile à mettre en oeuvre pour des opérations en ligne. Il faut souvent dégrader l'information pour pouvoir espérer l'interpréter en temps réel, et cela aux dépens de sa précision et de sa richesse.
L'utilisation d'une seule de ces deux stratégies pour un contrôle du mouvement d'un robot mobile dans un environnement complexe s'avère insuffisante et hasardeuse. Pour se rapprocher au mieux d'un comportement humain, le contrôle d'une manière efficace du mouvement d'un robot mobile dans un environnement complexe nécessite forcément une stratégie à deux niveaux :
- un niveau haut de planification de chemin utilisant une description totale, mais éventuellement incomplète ou/et imparfaite du modèle du monde. On essaye ici de tirer parti du fait que, dans le cas d'une robotique d'intérieur, on dispose d'une connaissance a priori sur l'environnement sous la forme d'un plan repérant les principales caractéristiques fixes (murs, portes, meubles fixes,…).
- un niveau local où le déplacement du robot est basé complètement sur les informations données par les différents capteurs qui couvrent l'entourage proche du véhicule.
Donc, pour mener à bien la gestion du mouvement d'un robot mobile dans un environnement partiellement connu, nous proposons d'effectuer une fusion de stratégies globale et locale. Cette fusion exploite la représentation de l'environnement (scène dite "modélisée" ou "mémorisée") qui est sujette à des variations locales par modification de la géométrie et/ou par ajout d'obstacles (scène dite "réelle") (fig. II-6).

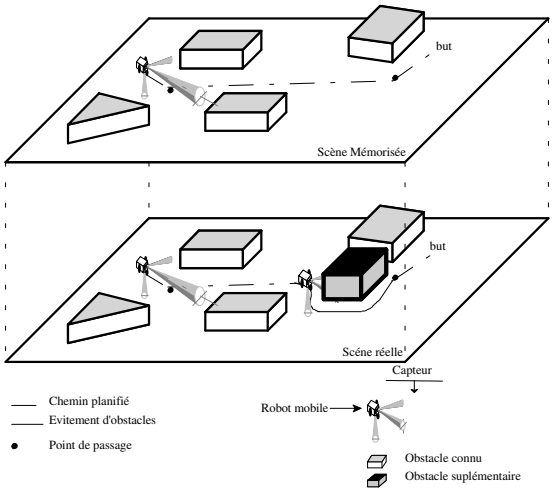


Figure II-6 : Scène mémorisée et scène réelle.

Dans la scène mémorisée, se déplace un robot virtuel muni de capteurs dont les données sont comparées, en temps réel, à celles fournies par les capteurs du robot évoluant dans la scène réelle. En fonction du résultat de cette comparaison, un indice (IFS : Indice de fusion de stratégies) de déclenchement de stratégie locale ou globale est fourni. Cet indice est généré par un module flou de prise de décision prenant en compte les erreurs de mesure des capteurs. Il reflète en quelque sorte la situation du robot par rapport à l'environnement préalablement connu. En l'exploitant, on fusionne les deux stratégies (globale et locale) par une pondération de la commande. Ainsi, si les données capteurs des deux scènes sont très proches, la navigation dans la scène réelle se fera par un suivi de la trajectoire planifiée ; si elles sont différentes on déclenche une stratégie de navigation locale.

4- Méthode de planification d'itinéraire adoptée dans un environnement connu

Nous présentons ici les outils que nous avons retenus et mis en œuvre pour planifier un itinéraire dans un environnement totalement connu. Cet environnement correspond à la scène mémorisée (ou théorique) dans laquelle se déplace le robot virtuel. Vu que les méthodes cellulaire de planification sont coûteuses en espace mémoire et que la méthode du graphe de traversabilité détermine directement un graphe des géodésiques (graphe éliminant les droites qui se croisent à l'intérieur d'un obstacle concave), nous avons choisi la méthode du graphe de visibilité qui offre plusieurs possibilités de chemin. On aura toujours la possibilité d'utiliser le graphe des géodésiques pour éliminer quelques itinéraires possibles. L'algorithme A* est appliqué par la suite pour déterminer le meilleur itinéraire.

4-1- Graphe de visibilité

Le graphe de visibilité est défini à partir de l'espace de configuration, ce qui revient à représenter le robot par un point dans l'espace libre avec des obstacles sous formes dilatées. L'obstacle dilaté correspond à l'espace balayé par le déplacement du robot ajouté à l'espace réel de l'obstacle afin d'éviter les collisions. Le graphe de visibilité est alors constitué des arêtes de ces obstacles dilatés et des tangentes communes à ces derniers. Le phénomène de visibilité en robotique mobile sert à vérifier le passage du robot (ponctuel) entre deux points. Ce passage n'est possible que si ces deux points soient mutuellement visibles.
Le graphe de visibilité détermine tous les chemins possibles passant par des points caractéristiques visibles, appelés des nœuds. Après la détermination de tous les points caractéristiques, on trace le graphe des chemins libres d'obstacles. Ainsi, un segment reliant deux nœuds signifie que le passage du robot entre ces deux points dans l'espace géométrique est libre d'obstacles.
Le graphe ainsi tracé est complètement indépendant des positions de départ et d'arrivée, ce qui permet son utilisation pour n'importe quelle mission dans cet environnement.

4-2- Algorithme A*

Cet algorithme recherche un chemin optimal en minimisant un critère de coût. Ce critère pourrait être le temps, la vitesse du robot, l'énergie consommée, la visibilité ou le nombre de virages, mais nous choisirons de minimiser la distance, car en la minimisant, la majorité des autres facteurs de qualité sont minimisés en même temps [LIMSAKOUNE 91]. Une fonction heuristique est évaluée afin d'orienter la recherche. Cette fonction sera ici la somme de la distance parcourue pour arriver au nœud et d'une estimation du coût restant pour arriver au but (distance à vol d'oiseau). Le chemin résultant relie le point source au point but, soit par une ligne droite, soit par une ligne brisée dont les sommets correspondent aux sommets des obstacles dilatés.
Sur la figure II-7 on présente un exemple d'environnement du mini-robot "Khepera". Les obstacles sont les polygones hachurés, entourés de traits fins représentant la dilatation. Dans notre cas où les robot utilisés sont de forme circulaire, les obstacles sont dilatés d'une distance égale au diamètre du robot avec une symétrie de révolution. Puis, les arcs de cercle sont approchés par des segments de droites.
Le chemin optimal obtenu par l'algorithme A*, après élimination des autres chemins possibles, est le trait pointillé reliant le point source au point but par l'intermédiaire des points de passage indiqués en noir.

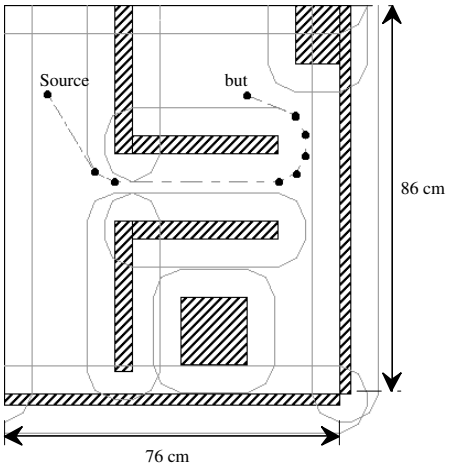


Figure II-7 : Planification d'itinéraire sur un exemple d'environnement réel.

4-3- Suivi du chemin planifié par contrôle flou

La planification de la trajectoire est faite au niveau topologique. L'itinéraire optimal issu du planificateur est un ensemble de points de passage entre le point de départ et le point d'arrivée. Ces points sont choisis comme sous-buts. Le chemin à suivre est ici les segments de droites joignant ces sous-buts. Pour assurer le contrôle du robot entre les points de passage, plusieurs méthodes de commande peuvent être utilisées. Ces méthodes peuvent être classiques [DELAPLACE 91][CORDEWENER 94][SHIN 92] [KAMGA 96] ou floues [ZHANG 93] [NISHIMORI 93].

4-3-1- Architecture du système de suivi de trajectoire

Le module de commande que nous avons développé pour générer la trajectoire entre les points de passage est basé sur le contrôle flou (fig. II-8). Cette génération de trajectoire a été faite dans le but d'effectuer d'une part le déplacement du robot virtuel dans l'environnement théorique et, d'autre part, de contrôler la navigation du robot réel lorsque l'environnement réel est identique à celui mémorisé

Ce module est utilisé pour fournir la vitesse angulaire (ω_s) du robot qui est supposé évoluer à vitesse linéaire constante (v_s). La vitesse angulaire du robot est déterminée à partir de sa position courante par rapport à la trajectoire et est réalisée par la variation relative des vitesses angulaires des roues motrices.

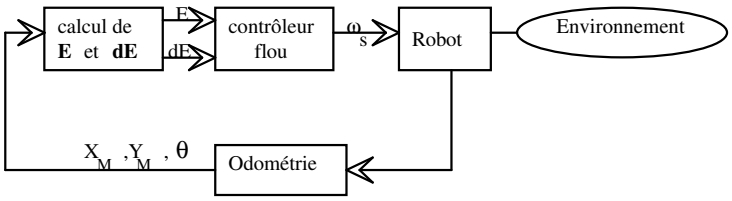


Figure II-8 : Architecture de commande pour le suivi de trajectoire.

Puisque le robot est un solide indéformable, les données de la distance signée E (séparant le point M du robot du segment de droite reliant les points de passage A et D (fig. II-9)) et de la variation dE de cette distance sont suffisantes pour accomplir la tâche.

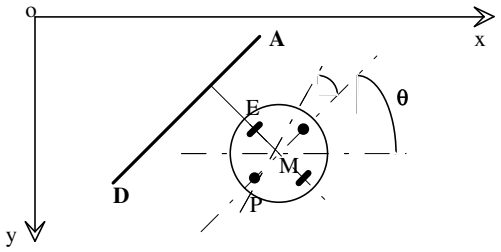


Figure II-9 : Position du robot par rapport à la trajectoire.

Les coordonnées (X_M,Y_M,θ_M) du robot sont données par odométrie directe. L'écart E est déterminé grâce au produit vectoriel PV tel que:

$$PV = |DM| \times |DA| \times \sin(A\hat{D}M)$$

avec :

$$E = |DM| \times \sin(ADM)$$

or :

$$PV = (X_M - X_D) \times (Y_A - Y_D) - (X_A - X_D) \times (Y_M - Y_D)$$

donc :

$$E = PV / |DA|$$

4-3-2- Contrôleur flou

Le coeur du régulateur est constitué d'un ensemble de règles floues de la forme suivante:

règle R_i : "Si E est Négatif Petit et dE est Négatif Petit Alors ω_s est Zéro"

Sinon

règle R_{i+1} : "Si..."

L'ensemble des règles de contrôle est regroupé dans la table II-1 de décision. La signification des termes linguistiques utilisés est la suivante: { (NG) *Négatif Grand*, (NP) *Négatif Petit*, (Z) *Zéro*, (PP) *Positif Petit*, (PG) *Positif Grand*}

dE \ E	**NG**	**NP**	**Z**	**PP**	**PG**
NG	Z	PP	PG	PG	PG
NP	NP	Z	PP	PG	PG
Z	NG	NP	Z	PP	PG
PP	NG	NG	NP	Z	PP
PG	NG	NG	NG	NP	Z

Table II-1 :Table de décision

Ces règles heuristiques obéissent aux principes suivants : Si la variable dE appartient au sous-ensemble Z (dérivée de la grandeur à régler presque nulle), on attribue à la variable de sortie ω le même ensemble (NG, NP, Z, PP ou PG) que celui de la variable d'entrée E. Si, par contre, cette dernière appartient à l'ensemble Z (écart de réglage presque nul), on attribue à la variable de sortie ω l'ensemble opposé (PG au lieu de NG, PP au lieu de NP, NP au lieu de PP et NG au lieu de PG) de celui de la variable d'entrée dE. Ainsi, l'intervention du régulateur diminue la dérivée de la grandeur à régler (en valeur absolue) et on obtient un effet de stabilisation du phénomène transitoire de réglage. Sur la diagonale de la table d'inférence, on attribue à ω l'ensemble Z. En effet, dans ce cas, le système à régler se trouve dans un état transitoire favorable qui ne nécessite pas de modifications importantes pour atteindre le régime stationnaire.

Les fonctions d'appartenance sont de forme triangulaire, équiréparties sur un univers de discours normalisé entre -1 et 1. Chaque univers de discours est partitionné en cinq sous-ensembles flous.

A chaque pas de calcul et selon les valeurs numériques des variables d'entrées, certaines règles de la table seront déclenchées. Chaque règle fournit un sous-ensemble flou associé à la variable ω qui sera obtenue à l'aide des opérateurs logiques employés d'une part pour la conjonction "Et" entre les prémisses et d'autre part pour réaliser l'implication (Si.... Alors...). L'opérateur Min est choisi pour cette étape de composition. Pour obtenir le sous-ensemble flou global inféré par l'ensemble des règles, il est nécessaire d'agréger les résultats partiels. Ceci est réalisé à l'aide de l'opérateur Max, ce qui revient à faire l'union de tous les sous-ensembles. Pour extraire une grandeur physique à la sortie du régulateur, nous avons choisi la méthode dite du "centre de gravité" qui consiste à calculer le barycentre des ω_i pondérés par leurs degrés d'appartenance μ_{ω_i}. Enfin, il faut prévoir des facteurs d'échelles qui se manifestent en tant que coefficients multiplicatifs des variables d'entrée et de sortie pour adapter les univers de discours aux conditions de l'application désirée. Ceci permet l'adaptabilité du contrôleur à l'application, sans modifier l'expression des règles qui conservent leur lisibilité. Ces gains sont déterminés ici de manière heuristique. Nous gardons ces choix pour tous les SIF standards utilisés dans ce travail.

La figure II-10 représente un exemple de résultat expérimental complet avec la méthode décrite dans ce chapitre : planification de l'itinéraire par méthode du graphe de visibilité et l'algorithme A* et suivi de la trajectoire planifiée (segments de droite passant par les points issus du planificateur) par le contrôleur flou.

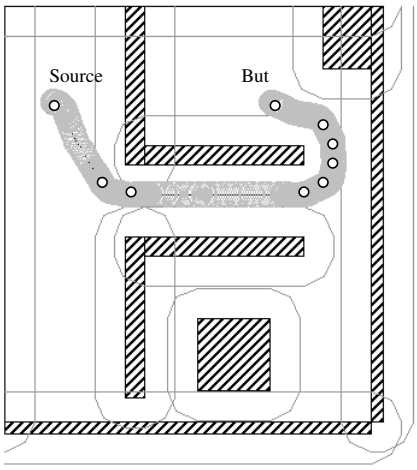


Figure II-10 : Déplacement du robot dans la scène modélisée.

Pour la clarté de la visualisation du résultat, la figure II-10 représente la trajectoire réelle du robot (données odométrique) appariée à un dessin de l'environnement réel.

4- Algorithme de navigation locale

La planification locale du chemin utilise les informations provenant des capteurs extéroceptifs (ultrasonores, vision, télémètre laser, capteurs de proximité, capteurs tactiles,...). Plusieurs méthodes réactives d'évitement d'obstacles ont été développées.

4-1- Navigation locale par champ de potentiels artificiels

4-1-1- Limitation de la méthode classique

Le principe de cette méthode est de considérer le robot comme un point dans l'espace de configuration se déplaçant sous l'influence de forces artificielles [KHATIB 86]. Les obstacles exercent des forces répulsives sur le robot et le but applique une force attractive ; la force résultante détermine la direction de déplacement du robot. Ce qui revient à construire un champ de potentiels artificiels avec un minimum global au point but et des maxima locaux pour les obstacles.

Une force attractive (F_{at}) peut être définie inversement proportionnelle à la distance (d_g) séparant le but du robot. Un coefficient multiplicatif (CG) est ajouté pour permettre le réglage de cette force : $F_{at} = CG \times 1/(d_g)$.

A la mesure de chaque capteur (capt i) on associe une force répulsive (F_{ri}) inversement proportionnelle au carré de la distance (d_{ri}) mesurée. Un coefficient multiplicatif (CO) est ajouté pour permettre le réglage de cette force: $F_{ri} = CO_i \times 1/(d_{ri})^2$.

La force résultante est alors la somme de toutes les forces : $F = F_{at} + \sum F_{ri}$ telles que les force F_{ri} correspondent, respectivement, aux différents capteurs du robot.

La figure II-11a montre l'espace à deux dimensions complètement inconnu par le robot, avec une configuration contenant trois obstacles. Le potentiel attractif est un puits parabolique avec son minimum à l'emplacement du but q_{but} (fig. II-11b).

Le potentiel répulsif est montré sur la figure II-11c. Il est tronqué sur la figure aux frontières des obstacles, car il tend vers l'infini. Le potentiel total et un ensemble de contours équipotentiels sont montrés dans la figure II-11d. Pour facilité la lisibilité et l'interprétabilité des courbes, nous n'avons pas représenté les potentiels des parois de l'environnement.

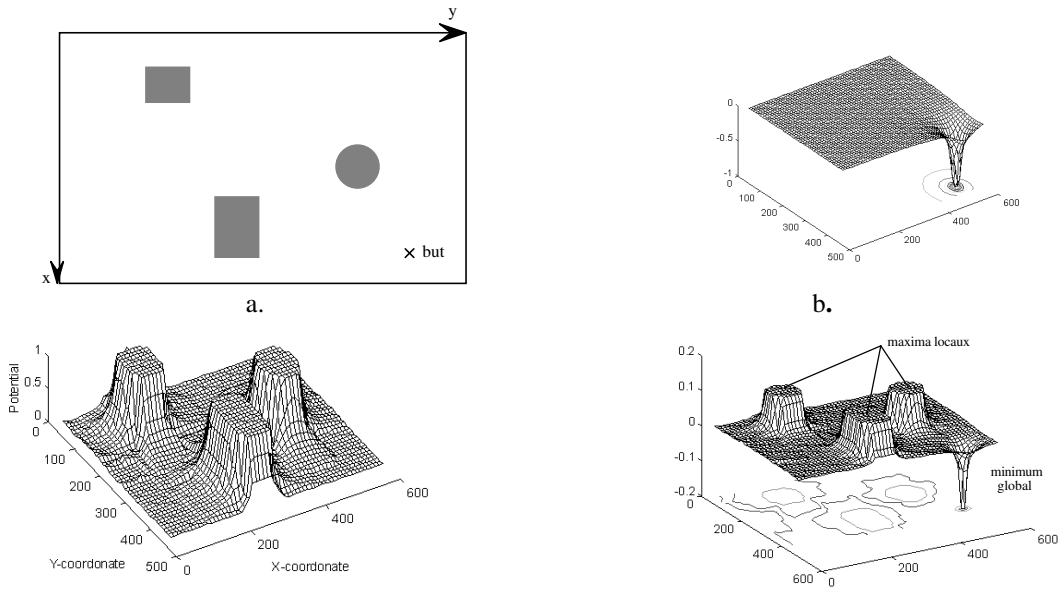


Figure II-11 : Représentation des différents champs de potentiels : (a) un exemple d'environnement à deux dimensions ; (b) potentiel attractif correspondant à l'environnement de la figure a ; (c) potentiel répulsif ; (d) potentiel total et contours équipotentiels.

Cette méthode de navigation locale est très séduisante par sa simplicité, mais elle présente certains inconvénients liés à la présence de minima locaux, à des difficultés de convergence vers le but et à l'existence d'oscillations.
Le défaut le plus connu et le plus souvent cité pour les méthodes de champ de potentiels artificiels est le problème des minima locaux qui produisent l'immobilisation du robot. Ces minima locaux peuvent être créés par certaines configurations d'obstacles qui provoquent une force résultante nulle ($F = 0$) due à l'équilibre entre les forces répulsives et la force attractive. Ce cas se produit dès que le rayon de courbure de la surface d'un obstacle est localement plus grand que celui de l'isopotentielle du champ attractif [TOURNASSOUD 88]. Un exemple de situation de blocage est représenté sur la figure II-12a. La figure II-12b donne la représentation en trois dimensions du champ de potentiel correspondant.

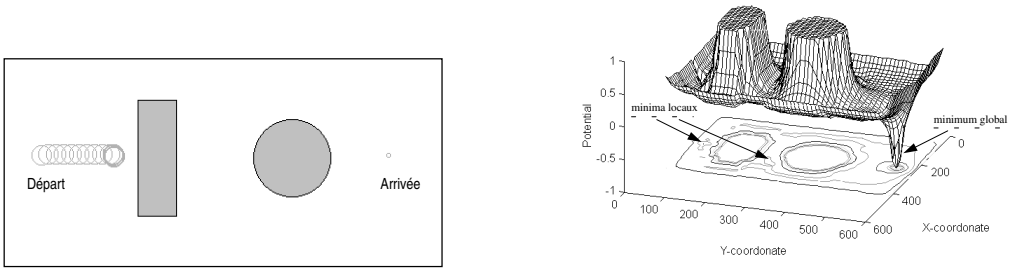


Figure II-12 : (a) Puits de potentiel (coefficients répulsifs identiques et constants COi = 100).
(b) Courbe 3D du champ de potentiel.

D'autre part, le robot peut rencontrer des difficultés pour se diriger vers l'objectif (trajectoire déviée), étant donné les nombreuses sollicitations provoquées par les obstacles autour de lui. Cette situation est fonction d'une part de la formulation de la force attractive et d'autre part du milieu d'évolution (densité d'encombrement et disposition des obstacles), en particulier quand le milieu d'évolution est pourvu d'obstacles entre la position courante et le but ou lorsque les obstacles sont concaves.
La situation précédente peut aussi provoquer des oscillations dans la trajectoire empruntée par le robot et imposée par les lignes de potentiel. En fait, les problèmes de convergence et d'oscillation sont liés car une non convergence entraîne généralement une oscillation et une instabilité de la trajectoire. Ces oscillations sont dues aux variations importantes et rapides de la force résultante. Cela peut apparaître notamment autour de l'objectif si la position finale est obstruée par un obstacle de grandes dimensions ou plusieurs obstacles rapprochés. Dans ce cas, et suivant l'amplitude des forces engendrées, le sens de la trajectoire oscille entre le sens de la force attractive et celui de la force répulsive.

Nous proposons dans le paragraphe suivant une technique basée sur une adaptation en ligne des coefficients de répulsion par des règles floues, afin de tenter de remédier à ces inconvénients.

4-1-2- Champ de potentiel artificiel à coefficients adaptatifs

Dans le cas où on a une connaissance préalable sur l'environnement, on peut s'extraire d'un minimum local en faisant appel au planificateur de mission pour éviter un éventuel minimum local [KOREN 91]. Dans le cas actuel de la navigation locale, nous proposons d'utiliser deux coefficients multiplicatifs des forces répulsives (Crepd pour les capteurs de droite et Crepg pour les capteurs de gauche) adaptatifs en fonction des informations fournies par les capteurs. Cette variation dynamique est assurée par une base de règles du type : " Si le but est droit devant et les distances droite et gauche diminuent, alors il faut dévier à droite ". Ces règles prennent en compte l'orientation ψ du robot par rapport au but et les différentes mesures des capteurs (fig. II-13) sous la forme : règle R_i "**Si** *D* est x_i **et** *G* **et** est y_i **et** Ψ est z_i **Alors** C_{repd} est t_i ».

x_i, y_i, z_i et t_i sont des variables linguistiques de la répartition floue des univers de discours respectif des entrées *D* (valeur minimale des quatre capteurs droit), *G* (valeur minimale des quatre capteurs gauche) et ψ et des sorties Crepd (coefficient répulsif des capteurs à droite) et Crepg (coefficient répulsif des capteurs à gauche) qui sera déduit de Crepd.

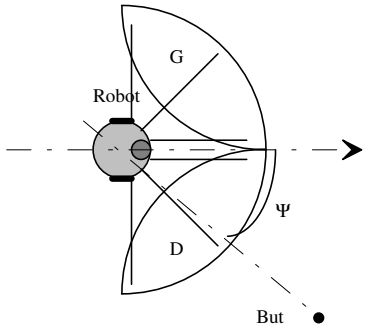


Figure II-13 : Les entrées du contrôleur flou ; D : valeur minimale des quatre capteurs à droite ; G : valeur minimale des quatre capteurs gauche ; ψ : orientation (robot / but).

La forme des fonctions d'appartenance est triangulaire et la somme des degrés d'appartenance pour chaque variable est toujours égale à 1. Les univers de discours sont définis entre $-\pi$ et $+\pi$ pour ψ et entre 0 et d_infl ("distance d'influence " égale ici, dans le cas du robot Khepera, à la portée maximale des capteurs, soit 5cm) pour les entrées (fig. III-14).

Les univers de discours sont partitionnés en trois sous-ensembles flous pour toutes les variables. Les variables linguistiques sont définies par :

P	: *Petit*	NP	: *Négatif*
M	: *Moyen*	Z	: *Zéro*
G	: *Grand*	PP	: *Positif*

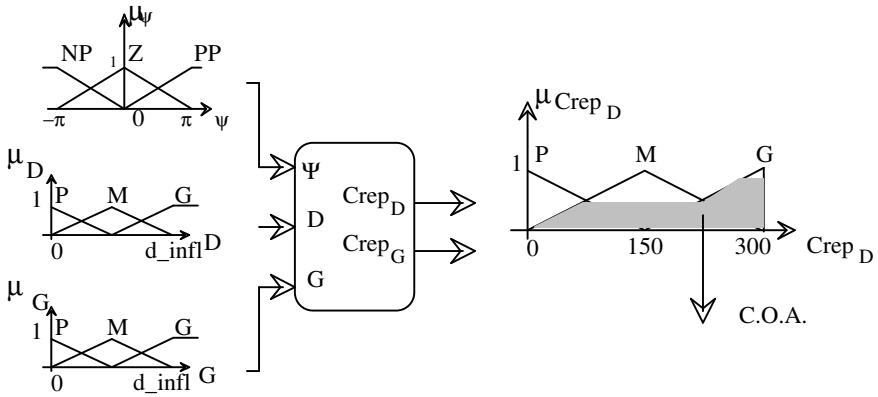


Figure II-14 : Répartition des sous-ensembles flous et fonctions d'appartenance pour le réglage des coefficients (avec $Crep_g = 300-Crep_d$).

Les règles de contrôle obtenues par raisonnement qualitatif et affinées par simulation sont représentées dans la table de décision II-2.

		D/G								
		P/P	P/M	P/G	M/P	M/M	M/G	G/P	G/M	G/G
Ψ	N	G	G	G	G	G	G	M	M	M
	Z	G	G	G	P	M	G	P	P	M
	P	P	P	M	P	P	M	P	P	M

Table II-2 : Table de décision pour le réglage des coefficients**.**

Exemple de règles floues:

- Si (D est *Petit*) et (G est *Petit*) et (Ψ est *Négative*) alors ($Crep_d$ est *Grand*)
- SI (D est *Grand*) et (G est *Petit*) et (Ψ est *Zéro*) alors ($Crep_d$ est *Petit*)

La figure II-15 illustre l'élimination de la situation de blocage de la figure II-12. L'obstacle est situé entre le robot et le point d'arrivée, les mesures capteurs à droite et à gauche sont identiques mais le contrôleur flou favorise dans ce cas précis les forces répulsives à droite par rapport à celles qui sont à gauche.

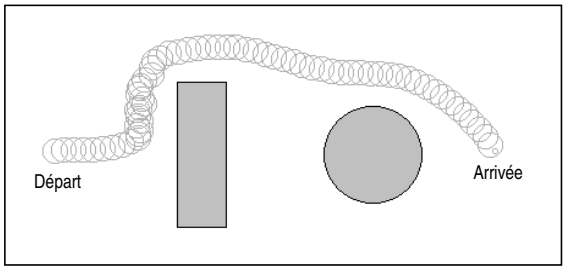
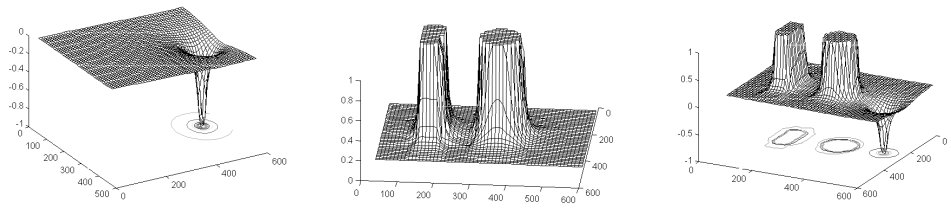


Figure II-15. Elimination du puits de potentiel.

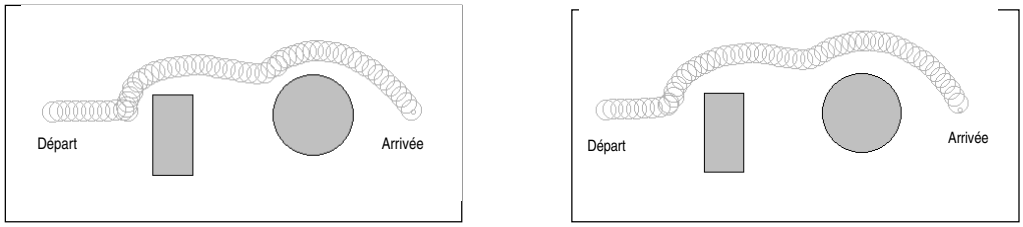
Après scrutation de tout l'environnement de la figure II-12a, nous avons obtenu les courbes de la figure II-16. La figure II-16a représente le potentiel attractif avec un seul minimum global (le but). Le potentiel répulsif est représenté sur la figure II-16b, avec des maxima locaux tronqués aux niveaux des obstacles. La courbe 3D de la figure II-16c montre le potentiel total. Il n'y a qu'un seul minimum global correspondant au but et deux maxima locaux représentant les obstacles. Les minima locaux de la figure II-12b sont éliminés.



a **b** **c.**

Figure II-16 : Représentation des potentiels en utilisant des coefficients gérés par la base de règles floues : (a) potentiel attractif ; (b) potentiel répulsif ; (c) potentiel total.

Par ailleurs, nous avons pu constater, qu'outre l'élimination des blocages dus aux minima locaux, la méthode employée conduisait à une amélioration de la convergence. Ceci est illustré sur la figure II-17. Si on utilise la méthode classique à coefficients fixes, on peut avoir une trajectoire avec des changements de direction trop brutaux (fig. II-17a) ou s'éloignant trop des obstacles (fig. II-17b).



a. b.

Figure II-17 : Réglage des coefficients :
(a) coefficients répulsifs identiques et constants Crepd = 100 ;
(b) coefficients répulsifs identiques et constants Crepd = 300 ;

Cette technique de réglage en ligne des coefficients a permis de remédier au problème de minima locaux, mais le fait d'ajuster les coefficients ne permet pas de contourner le problème des oscillations dans des environnements étroits et passages de porte. C'est pourquoi nous proposons dans la suite une stratégie basée sur des SIF.

4-2- Navigation locale par combinaison de comportements réactifs élémentaires

4-2-1- Introduction

La deuxième approche du problème de navigation locale que nous avons développée est de type comportemental, effectué par fusion d'actions réactives élémentaires.
Le principe général que nous suivrons consiste à essayer d'imiter, même sommairement, le comportement d'une personne voulant atteindre en voiture un point repérable (visuellement, auditivement, ...) dans un lieu dont elle n'a aucun plan. Deux degrés de décision peuvent se dégager du comportement de ce conducteur. Le premier est celui adopté pour éviter au mieux les obstacles situés dans son environnement immédiat (respecter sa position par rapport aux autres véhicules, respecter sa position sur la voie, suivre la chaussée, éviter les voitures stationnées, ...). Ces opérations sont fondées sur sa connaissance de l'environnement local. Elles sont exécutées et menées à court terme. Leur influence est purement locale. Cette première démarche doit être suivie prioritairement pour éviter tout risque de collision. Le second degré de décision permet à ce même conducteur de converger progressivement vers le but. En effet, lorsqu'il arrive dans une situation complexe, telle que intersection, voie sans issue, sens interdit, ..., ce conducteur doit effectuer un choix. C'est de celui-ci que va dépendre en grande partie la suite du chemin parcouru. Toutefois, le conducteur ne dispose pas de beaucoup d'informations pour effectuer ces prises de décision.
Dans cet esprit, nous développerons une stratégie de navigation comportementale basée sur deux principales actions réactives : " suivi du milieu de l'espace libre " et " convergence vers le but ". Cependant, des situations de blocage dans des obstacles concaves nous amènerons à développer un troisième agent " suivi de mur " et à coordonner l'intervention de ces trois actions grâce à une technique de création de sous-buts intermédiaires. Ces comportements seront réalisés dans ce paragraphe avec des SIF réglés heuristiquement. Puis une démarche d'optimisation issue des résultats du chapitre I sera proposée dans le paragraphe 11 de ce chapitre.

4-2-2- Navigation sans obstacle concave

Une stratégie de navigation fusionnant les deux actions élémentaires "suivi du milieu de l'espace libre" et "convergence vers le but", est appliquée dans des environnements composés d'obstacles convexes. Dans ce type d'environnement l'occurrence d'une situation de blocage est très faible.

4-2-2-1- Suivi du milieu de l'espace libre

La méthode consiste à évaluer l'espace encombré dans l'environnement proche du robot. On cherche alors à faire évoluer le robot dans la direction la plus libre. Ainsi, lorsque le robot détecte un rétrécissement de l'espace sur la gauche il dévie vers la droite et vice versa.
Le navigateur utilisé est basé sur un contrôleur flou dont les référentiels d'entrées sont d_n, g_n et f_n, respectivement, les distances normalisées à droite, à gauche et frontale, entre le robot et l'environnement le plus proche :

$$R_n = \frac{R}{R+L} \quad , \quad L_n = \frac{L}{R+L} \quad \text{et} \quad F_n = \frac{F}{\sigma} \; ;$$

D et G représentent les mesures à droite et à gauche données par les capteurs respectifs. σ est la distance à partir de laquelle les obstacles ne sont pas pris en compte. Grâce à cette normalisation, l'univers de discours évolue avec la variation des dimensions de l'espace libre vu par le robot (fig. II-18).

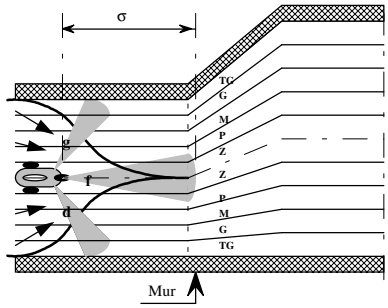


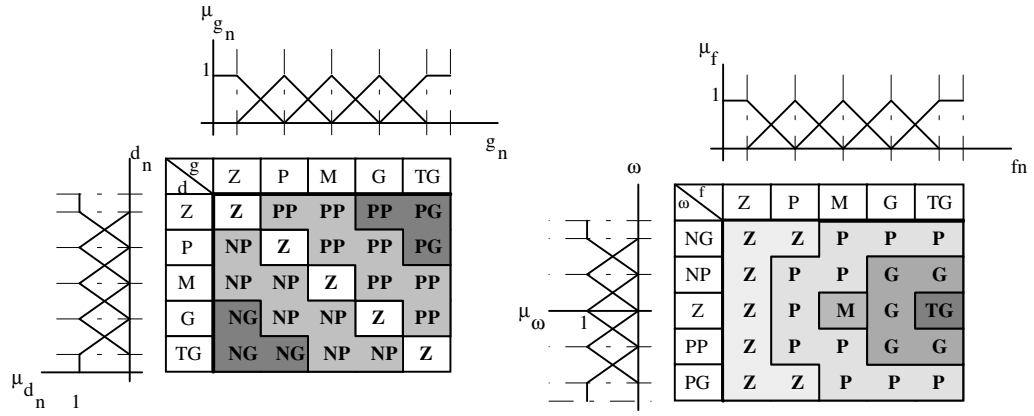
Figure II-18 : Evolution de la partition de l'univers de discours.

Les variables de sortie C_{va} et $C_{\omega a}$ sont des coefficients multiplicatifs des vitesses linéaire et angulaire. Les univers de discours sont normalisés entre –1 et 1 pour $C_{\omega a}$ et 0 et 1 pour les autres variables. Pour des raisons de simplicité et de rapidité de calcul, les fonctions d'appartenances choisies sont de formes triangulaires. La somme des degrés d'appartenance d'une variable aux différents sous-ensembles flous définis sur son univers de discours est toujours égale à 1. Chaque univers de discours est partitionné en cinq sous-ensembles flous. Les répartitions linguistiques sont les suivantes:

- pour les entrées et C_{va} : {Z : Zéro ; P : Petit ; M : Moyen ; G: Grand ; TG : Très Grand} ;

-pour $C_{\omega a}$: { NG : Négatif Grand ; NP : Négatif Petit ; Z : Zéro ; PP : Positif Petit ; PG : Positif Grand}.

L'ensemble des règles floues déduites de l'expérience intuitive d'un conducteur humain est représenté par les cinquante règles regroupées dans les deux tables de la figure II-19 : 25 règles permettent de déterminer $C_{\omega a}$ et 25 autres déterminent C_{va}.



d \ g	Z	P	M	G	TG
Z	**Z**	**PP**	**PP**	**PP**	**PG**
P	**NP**	**Z**	**PP**	**PP**	**PG**
M	**NP**	**NP**	**Z**	**PP**	**PP**
G	**NG**	**NP**	**NP**	**Z**	**PP**
TG	**NG**	**NG**	**NP**	**NP**	**Z**

a

ω \ f	Z	P	M	G	TG
NG	**Z**	**Z**	**P**	**P**	**P**
NP	**Z**	**P**	**P**	**G**	**G**
Z	**Z**	**P**	**M**	**G**	**TG**
PP	**Z**	**P**	**P**	**G**	**G**
PG	**Z**	**Z**	**P**	**P**	**P**

b.

Figure II-19 : Tables de règles pour la détermination des coefficients des vitesses : (a) angulaire $C_{\omega a}$ et (b) linéaire C_{va}.

Exemples de règles utilisées :

- Si (d_n est *Très Grand*) et (g_n est *Très Grand*) alors ($C_{\omega a}$ est *Zéro*) et si (f est *Très Grand*) alors (C_{va} est *Très Grand*).
- Si (d_n est *Très Grand*) et (g_n est *Petit*) alors ($C_{\omega a}$ est *Positive Grande*) et si (f est *Zéro*) alors (v est *Zéro*).

Comme précédemment, nous employons les opérateurs Min pour la composition des variables d'entrée et l'implication floue et Max pour l'agrégation des règles. La défuzzyfication est effectuée par la méthode du centre de gravité.

4-2-2-2- Convergence vers le but

Pour atteindre son but B le robot est attiré par une force qui le guide jusqu'à sa destination. Les actions ($C_{\omega g}$ and C_{vg}) générées par cette force sont modulées par l'inverse de la distance PB entre le centre P du robot et le but (fig. II-20). θ_g est la déviation angulaire nécessaire pour atteindre le but. D est la distance d'influence qui va servir à affiner la stratégie.

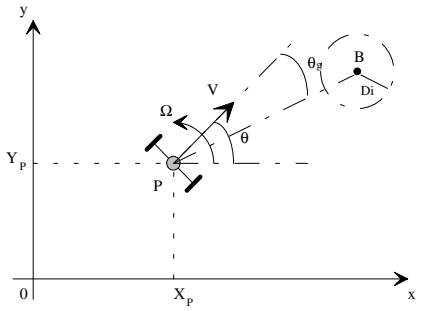


Figure II-20 : Situation de convergence vers le but

Quand le robot est assez loin du but le coefficient de la vitesse angulaire est tel que: $C_{\omega g} = \frac{C_g}{PB} \times \frac{D}{\pi} . \theta_g$.

Le coefficient C_g (compris entre –1 et 1) est choisi de manière à ce que le robot atteigne une vitesse angulaire maximale à $\theta_g < \pi$ afin que son cap ne s'écarte pas trop de la direction de la droite PB.

Mais, si le robot se trouve dans la zone d'influence de diamètre D, le coefficient de la vitesse angulaire est tel que: $C_{\omega g} = \frac{C_g}{\pi} \theta_g$. Ce découpage évite des commandes trop importantes en vitesse angulaire induisant des mouvements brusques du robot.

Par ailleurs, on lie le coefficient de réglage de la vitesse linéaire de convergence vers le but au coefficient de vitesse angulaire par la relation suivante : $C_{vg}= 1-|C_{\omega g}|$. Ceci traduit la règle suivante : plus le robot est orienté vers la direction du but ou plus le robot est loin du but, plus il peut aller vite (sachant que la vitesse est limitée par une vitesse maximale imposée, soit par l'utilisateur, soit par le matériel).

4-2-2-3- Fusion des deux comportements : Stratégie S1

La sécurité du robot est essentielle en navigation réactive. Pour cette raison nous distinguons deux cas:

- Si un obstacle proche du robot est détecté sur les côtés ou en face, l'évitement d'obstacles est prioritaire et l'attraction vers le but est annulée $(C\omega g = 0)$.

-Dans le cas contraire, la vitesse angulaire ω_r appliquée au robot résulte d'une combinaison linéaire des coefficients fournis par les modules de recherche de l'espace libre et d'attraction par le but :

$$\omega_r = (\alpha \times C_{\omega a} + \beta \times C_{\omega g}) \times \omega_{max}$$

où α et β sont des coefficients ajustables par expérimentation pour avoir la meilleure génération de trajectoire possible.

De même, la vitesse linéaire V_r est donnée par :

$v_r = \min(C_{va}, C_{vg}) \times V_{max}$ si PB>D ou $v_r = \min(C_{va}, C_{vg}) \times V_{min}$ si PB<D ,

où Vmax et Vmin sont respectivement le maximum et le minimum choisis pour la vitesse linéaire.

4-2-2-4- Application

La tâche présentée pour illustrer la stratégie précédente consiste à passer par une porte dans un environnement partiellement connu où un point de passage est défini (fig. II-21). Cependant, ici, le robot n'est pas astreint à suivre les segments de droite joignant les sous-buts, comme on le faisait dans une scène totalement connue. L'objectif consiste à utiliser uniquement les deux modules de navigation réactive précédents pour atteindre le but intermédiaire (SB) puis le but final (B), tout en évitant éventuellement les obstacles non modélisés pouvant être présents sur le parcours.

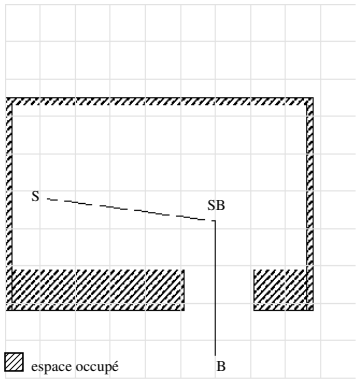


Figure II-21 : Chemin planifié.

Les expérimentations sont effectuées avec les deux robots (Khepera et RMI), dans deux configurations différentes : sans obstacle et avec un obstacle placé juste avant l'accès à la porte.
Pour utiliser la méthode de navigation précédente, il faut disposer de trois entrées normalisées du SIF : frontale (F), gauche (G) et droit (D). Chacune de ces entrées est la valeur donnée par une partie des capteurs infrarouge (Khepera) ou ultrasonores (RMI) (fig. II-22).

entrée frontale	F = Min(IR0, IR7),	F = Min(USd, USe, USf),
entrée à droite	D = Min(IR5, IR6, IR7),	D = Min(USf, USg, USh),
entrée à gauche	G = Min(IR0, IR1, IR2).	G = Min(USb, USc, USd).

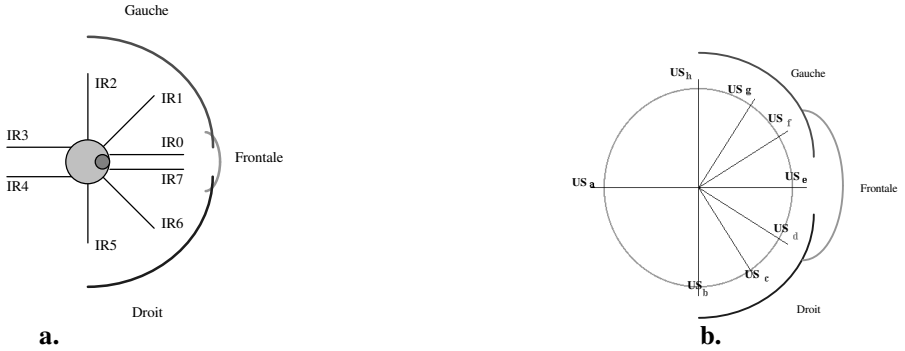


Figure II-22 : Disposition des capteurs: a) infrarouges sur Khepera, b) ultrasonores sur RMI.

Expérimentation sur Khepera

L'environnement de Khepera est composé d'un ensemble de boites de formes polygonales représentant les murs et les obstacles. L'environnement est de 55 cm de longueur et 25 cm de largeur.

1) Sans obstacle

Comme la portée des capteurs infrarouges est limitée à 5 cm, le robot ne peut voir les murs qu'à très faible distance. Les entrées du contrôleur flou appartiennent au sous-ensemble "Grand", par conséquent la déviation appartient au sous-ensemble "Zéro", le robot converge vers le but intermédiaire à la vitesse maximale (fig. II-23a). Arrivant au niveau de la porte, les entrées sont qualifiées de "Moyennes", par conséquent la déviation appartient au sous-ensemble "Zéro". Etant donné que le but est en face du robot, l'attraction par le but va confirmer cette commande.

2) Avec obstacle

Comme illustré dans la figure III-42b l'obstacle est bien identifié par les capteurs. Grâce au contrôleur flou Khepera évite l'obstacle malgré la courte portée des capteurs infrarouges (5cm). Après avoir évité l'obstacle, il converge vers le but intermédiaire.

Expérimentation sur R.M.I.

La zone d'évolution du robot est une partie du laboratoire de quatre mètres de longueur et deux mètres et demi de largeur. La porte a quatre vingt dix centimètres de largeur.

1) Sans obstacle

Dans le cas de RMI la portée des capteurs est importante. De ce fait les murs sont détectés par les capteurs. L'influence de l'attraction vers le milieu de l'espace libre sera donc plus importante que pour Khepera d'où la

trajectoire plus arrondie obtenue (fig. II-23c). Le passage de la porte s'effectue ensuite dans des conditions analogues à celles observées pour Khepera..

2) Avec obstacle

L'obstacle supplémentaire rajouté du côté de la porte étant vu par les capteurs de droite, le contrôleur flou génère une commande qui permet de l'éviter en déviant à gauche vers le milieu de l'espace libre (fig. II-23d). Pour les même raisons que précédemment la trajectoire est plus arrondie que pour Khepera. Puis, une fois l'obstacle dépassé, la commande générée par l'attraction par le but devient de plus en plus grande par rapport à celle de l'évitement, jusqu'au sous-but.

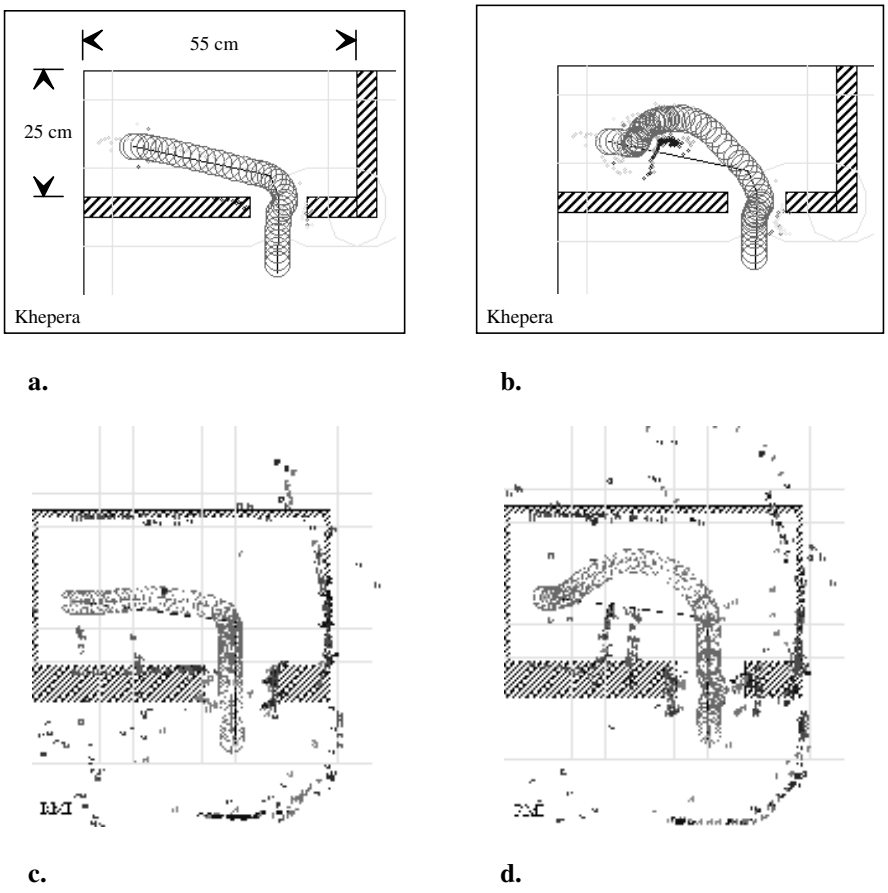


Figure II-23 : Navigation : (a) Khepera sans obstacle inconnu ; (b) Khepera avec un obstacle du côté de la porte ; (c) RMI sans aucun obstacle inconnu ; (d) RMI avec un obstacle du coté de la porte.

Nous pouvons tirer deux conclusions importantes de l'expérimentation. Tout d'abord, le passage de Khepera à RMI s'est effectué sans aucune modification de l'algorithme, ni même sans aucun ajustement des paramètres, (excepté la portée des capteurs σ et la distance d'influence Di). Cette transférabilité immédiate de l'algorithme d'un robot à un autre est évidemment extrêmement intéressante dans la pratique. Elle est liée à l'emploi de règles floues, établies de façon logique en fonction de l'objectif à réaliser, sans aucune référence à un modèle du robot, ni au type et aux performances de ses capteurs.

Le second point est la robustesse de la méthode par rapport aux imperfections des capteurs. Dans le cas de Khepera, les capteurs infrarouges utilisés sont très bruités, de calibration très imparfaite et surtout ont une portée très faible. Malgré l'information pauvre reçue, la navigation se déroule tout à fait correctement. Pour RMI, les impacts des mesures ultrasonores sont nombreux, mais un nombre de mesures aberrantes élevé apparaît, dues au problème des réflexions multiples du faisceau ultrasonore avant retour sur le récepteur. Là encore, sans aucun traitement des mesures, le navigateur utilisé permet un déroulement très satisfaisant de la mission.

Cependant, nous avons pu constater par expérimentation que des échecs subsistent lorsqu'un obstacle concave s'interpose entre le robot et le but (ou sous-but) à atteindre. Ceci se produit d'ailleurs aussi bien avec la méthode du champ de potentiel qu'avec la fusion de comportements réactifs élémentaires. Ceci nous a amené à affiner la stratégie de navigation en introduisant d'autres comportements.

4-2-3- Navigation avec obstacles concaves

Nous considérons qu'un obstacle concave est détecté lorsque les trois données D, G et F mesurent simultanément des distances très faibles. Dans ce cas, une troisième action élémentaire de navigation "suivi de milieu" est développée. La coordination des trois agents est réalisée grâce une technique de création de sous-buts intermédiaires.

En effet, pour sortir de ces situations de blocage, le principe de la solution retenue est de combiner le suivi des parois de l'obstacle concave avec les comportements élémentaires précédents jusqu'à ce que la convergence vers le but soit à nouveau possible. Il s'agit d'une stratégie inspirée de la méthode myopique qui consiste à suivre la paroi d'un obstacle pour le contourner. Nous allons donc doter le robot d'un comportement de type "suivi du mur".

4-2-3-1- Suivi de mur

Le but est de fournir la commande adéquate du robot pour qu'il longe le contour de l'obstacle à une distance de sécurité fixée "d_consigne". Si la distance mesurée à gauche est très grande, le suivi se fait à l'aide des données capteurs droit et frontal (Fig. II-24) et réciproquement.

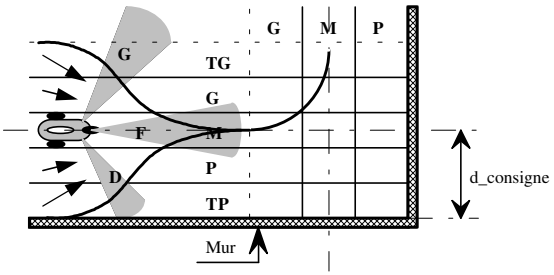


Figure II-24 : Suivi de mur : correspondance entre mesures et sous-ensembles flous.

Cette commande est fournie par un contrôleur flou ayant pour entrées les distances mesurées à droite D et en face F et donnant en sortie la vitesse angulaire du robot ω_r. La forme des fonctions d'appartenance est triangulaire et la somme des degrés d'appartenance pour chaque variable est toujours égale à 1. Les univers de discours sont définis entre $-\omega_{max}$ et $+\omega_{max}$ pour ω, et entre 0 et σ pour les entrées (fig. II-25).

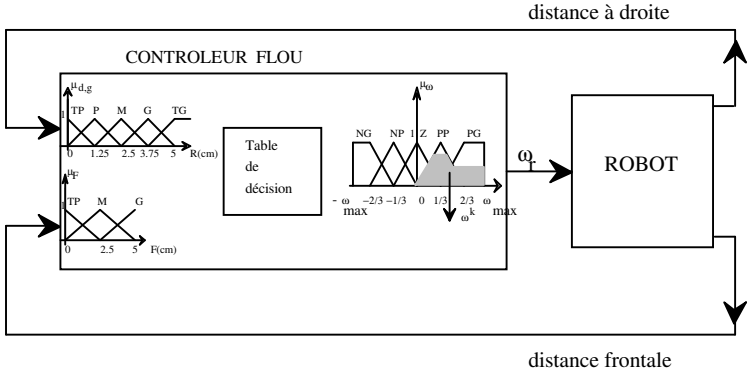


Figure II-25 : Schéma fonctionnel du contrôleur flou dans le cas de Khepera où σ=5cm.

Les univers de discours sont partitionés en cinq sous-ensembles flous pour "D" et "ω_r" et trois pour "F". Les variables linguistiques sont définies par:

TP	: *Très petit*	NG	: *Négatif Grand*
P	: *Petit*	NP	: *Négatif Petit*
M	: *Moyen*	Z	: *Zéro*
G	: *Grand*	PP	: *Positif Petit*
TG	: *Très Grand*	PG	: *Positive Grand*

Toutes les règles de contrôle sont représentées par quinze règles regroupées dans la table de décision II-3. Pour régler ce contrôleur, des facteurs d'échelle ajustés d'une manière empirique sont utilisés. Ce réglage constitue l'inconvénient majeur de cette méthode.

F D	P	M	G
TP	PG	PG	PG
P	PG	PG	PS
M	PG	PP	Z
G	PG	Z	NP
TG	PG	NP	NG

Table II-3 :Table de décision.

Pour le moteur d'inférence, nous utilisons les mêmes opérateurs et la même méthode de défuzzyfication que précédemment.
La vitesse linéaire du robot est calculée indépendamment du contrôleur flou qui gère sa direction. Le principe du calcul est simple: plus le robot est près d'un obstacle, plus sa vitesse diminue (proportionnellement à la distance frontale). Quand il est trop prés, il s'arrête, jusqu'à ce que le contrôleur flou le fasse tourner vers la direction de l'espace libre.

4-2-3-2- Expérimentation

Un exemple d'expérimentation de la méthode, sur le mini-robot Khepera, est montré sur la figure II-26. Les murs et les obstacles sont représentés par les portions en grisé et les déplacements du robot par une succession de cercles noirs.

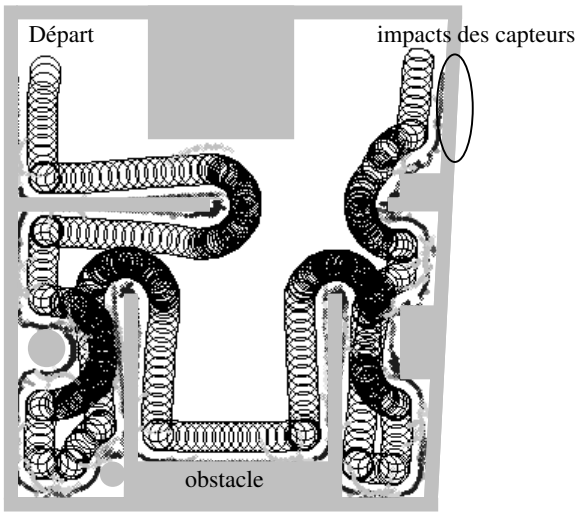


Figure II-26 : Expérimentation.

4-2-4- Stratégie complète de navigation réactive : Stratégie S2

Il s'agit maintenant de combiner cet agent de suivi de mur avec la stratégie précédente qui permettait une navigation sûre dans un milieu sans obstacle concave. La détection d'un obstacle concave engendre la création d'un but intermédiaire de transition "SG1" et déclenche la stratégie de suivi de mur du côté droit. Si le robot s'éloigne de la cible au delà d'une distance plus grande qu'une distance seuil L choisie, il revient, à l'aide de la stratégie S1 "convergence vers le but + suivi du milieu de l'espace libre", au but intermédiaire de transition "SG1" précédent qui a été mémorisé. Il longe alors l'obstacle de l'autre côté, avec la même distance seuil L. Ainsi de suite jusqu'à la détection d'un espace libre qui permette d'atteindre le but (fig. II-27).

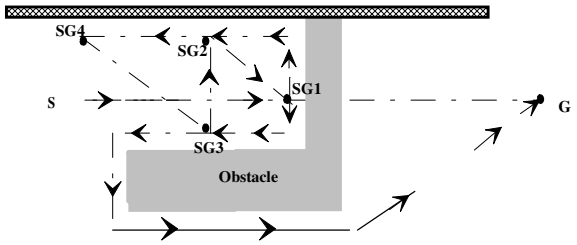
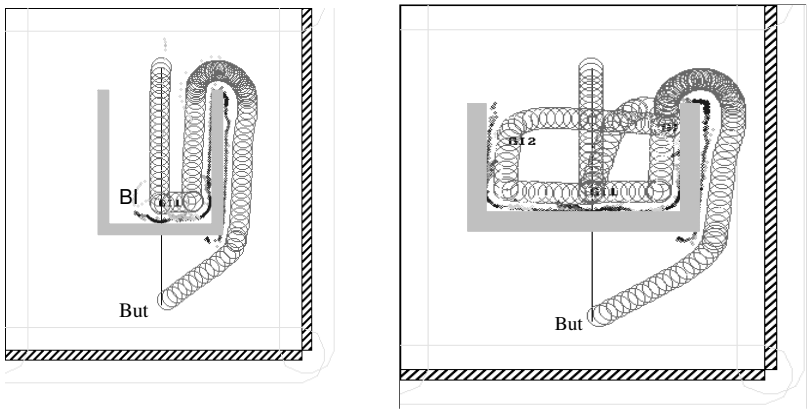


Figure II-27 : Evitement d'obstacles concaves et création de buts intermédiaires.

Nous avons testé cette technique de navigation sur le robot Khepera. Sur la figure II-28a, un seul but de transition "BI" a été créé ; car le seuil L de déplacement autorisé a été assez grand. Sur la figure II-28b, le seuil L est plus petit : trois buts intermédiaires de transition ont été créés avant de converger vers le but final.



a. **b.**

Figure II-28 : Contournement d'un obstacle concave par Khepera face à un obstacle concave
a) un seul but intermédiaire b) trois buts intermédiaires.

La figure II-29 montre un exemple de navigation de Khepera dans un environnement complexe comportant une succession d'obstacles concaves et convexes séparant le robot de son but. Dans ce cas, en fonction des observations des capteurs, un contournement d'obstacle détecté comme concave est mis en jeu (zones Z1 et Z3) ou bien la navigation simple par suivi de milieu de l'espace libre et attraction vers le but est effectuée (zones Z2 et Z4).

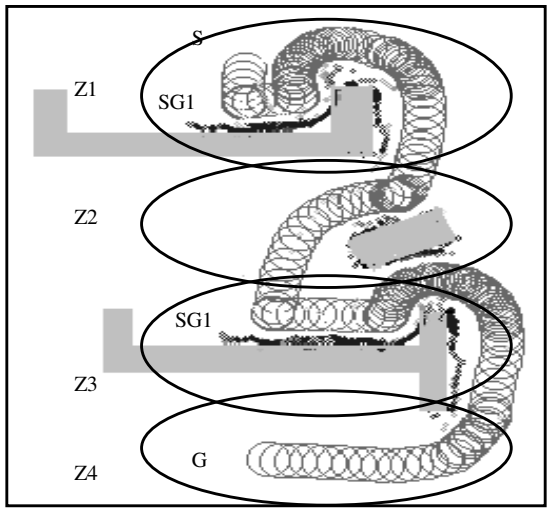


Figure II-29 : Exemple d'environnement complexe

5-Fusion des stratégies locale et globale

Les questions qui se posent pour cette fusion sont : quand faut-il appliquer l'une ou l'autre des deux stratégies globale ou locale ? Est-il possible de les appliquer en même temps et comment ? Pour répondre à ces

questions nous avons développé un module de prise de décision qui se base sur ce que le robot perçoit de son environnement. Mais la perception est entachée d'erreurs. Pour absorber cette imprécision et pour assurer un passage doux entre les modes de contrôle du robot nous avons employé des règles floues pour réaliser un module décisionnel qui, à partir d'une comparaison en temps réel entre deux environnements (mémorisé et réel), fournit un indice permettant une fusion progressive.

5-1- Principe

Un robot virtuel, muni de capteurs virtuels, se déplace dans l'environnement mémorisé, pendant que le robot réel navigue dans l'environnement réel. Une comparaison, à chaque instant, est faite entre ces deux environnements. Un SIF, exploitant les informations capteurs des deux robots (réel et virtuel), fournit un indice qui reflète en quelque sorte la situation du robot par rapport à l'environnement préalablement connu. Cet indice appelé IFS (Indice de Fusion de Stratégies) sert à pondérer le recours à la navigation par suivi de trajectoire planifiée ou à la navigation réactive.

A un environnement complètement connu ou non contraignant (c'est à dire enlèvement ou éloignement d'un obstacle connu) correspond un suivi de la trajectoire planifiée ; les commandes appliquées sur le robot sont alors les vitesses angulaire (ω_s) et linéaire (v_s) fournies par le contrôleur de suivi de trajectoire. Celles correspondant à un environnement complètement inconnu (ω_r, v_r) sont issues des modules de navigation réactive. Dans un environnement partiellement connu, la fusion des commandes est faite selon :

$$\omega = \omega_s \times IFS + \omega_r \times (1 - IFS) ;$$
$$v = v_s \times IFS + v_r \times (1 - IFS).$$

où ω et v sont les commandes appliquées au robot.

5-2- Module de prise de décision

Ce module qui fournit l'indice de fusion de stratégies IFS utilise une prise de décision par règles floues en comparant les informations des capteurs entre les deux scènes. Les entrées de ce module de prise de décision sont les écarts entre (fig. II-30) :

- les données des capteurs latéraux (capteurs parfaits) du modèle mémorisé et celles des capteurs réels (données entachées d'erreurs) (Δd et Δg).
- la position absolue du robot mobile dans la scène réelle et dans la scène mémorisée (Δp) (c'est à dire la distance entre les deux robots, réel et virtuel).

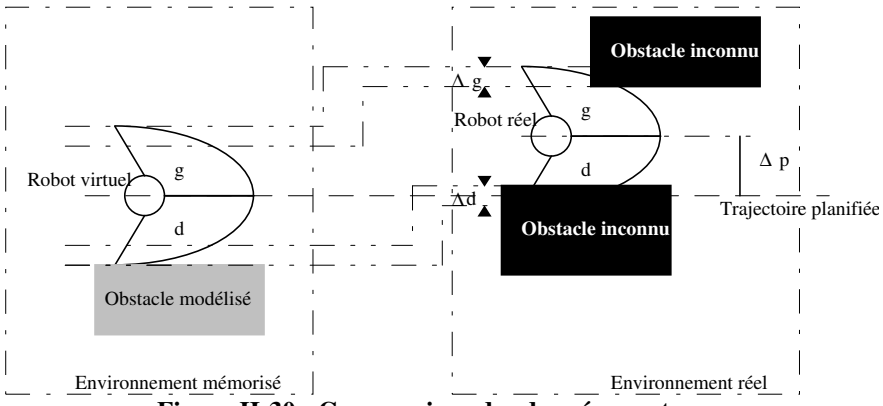


Figure II-30 : Comparaison des données capteurs.

L'indice IFS fourni par ce module reflète le degré de ressemblance de la scène réelle avec le modèle en mémoire. Cette variable de sortie IFS est partitionnée en deux sous-ensembles flous de label Sp correspondant au déclenchement du suivi de trajectoire planifiée et Sr pour le déclenchement des actions réflexes permettant la navigation locale. Les 2 SEF ont la forme de demi gaussienne.

Les univers de discours des variables d'entrée Δd, Δg et Δp sont décomposés en trois sous-ensembles flous (fig. II-31). Les étiquettes utilisées sont *N* (Négatif), *Z* (Zéro), *P* (Positif), *M* (Moyen), *G* (Grand).

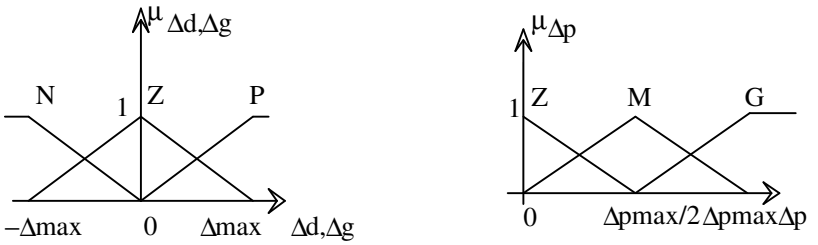
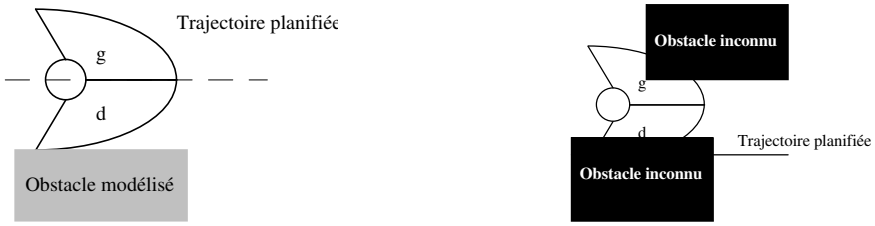


Figure II-31 : Sous-ensembles flous définis pour les variables Δd, Δg et Δp.

Ces règles floues sont de la forme suivante :
Si Δd est *zéro* et Δg est *zéro* et Δp est *zéro* alors le suivi de trajectoire planifiée est déclenché (**IFS=**Sp) (cas de la figure II-32a).
Si Δd est *négative* et Δg est *négative* et Δp est *grand* alors la navigation réactive locale est déclenchée (**IFS=**Sr) (cas de la figure II-32b).



a. **b.**

Figure II-32 : Exemples d'environnement local a) complètement connu; b) complètement inconnu.

Une table II-4 de règles est ainsi définie. Cette table contient l'ensemble des combinaisons possibles existant entre Δd, Δg et Δp. Ces combinaisons symbolisent la relation suivante : "*La stratégie de suivi de trajectoire planifiée est d'autant plus déclenchée que les valeurs Δd, Δg et Δp sont plus petites*".

		Δ d / Δ g								
		N/N	**N/Z**	**N/P**	**Z/N**	**Z/Z**	**Z/P**	**P/N**	**P/Z**	**P/P**
Δ p	**Z**	Sr	Sr	Sr	Sr	**Sp**	**Sp**	Sr	**Sp**	**Sp**
	M	Sr	Sr	Sr	Sr	Sr	**Sp**	Sr	**Sp**	**Sp**
	G	Sr	Sr	Sr	Sr	Sr	Sr	Sr	Sr	**Sp**

Table II-4 : Table d'inférences.

Le SIF utilisé est de type "MAMDANI", (opérateur Min pour le ET et l'implication, Max pour le Sinon et la défuzzyfication par la méthode du centre de gravité). Cette opération permet de déterminer une valeur nette permettant de déterminer les commandes à appliquer au robot au pas de déplacement suivant.

5-3- Résultats

La figure II-33 montre des résultats expérimentaux de la méthode implémentée sur le robot Khepera. Lorsque l'environnement réel est, soit identique (fig. II-33b) au modèle de la figure II-33a, soit non contraignant (fig. II-33c), la navigation est réalisée par le suivi du chemin planifié à vitesse maximum. Sinon (fig. II-33d), l'ensemble des stratégies d'actions réflexes est déclenché à partir du point P1. La vitesse est d'abord fortement réduite car un obstacle inconnu a été détecté. Ensuite, elle augmente graduellement car la scène localement perçue ressemble de plus en plus à la scène mémorisée, jusqu'à ce que le robot atteigne le but intermédiaire P2.

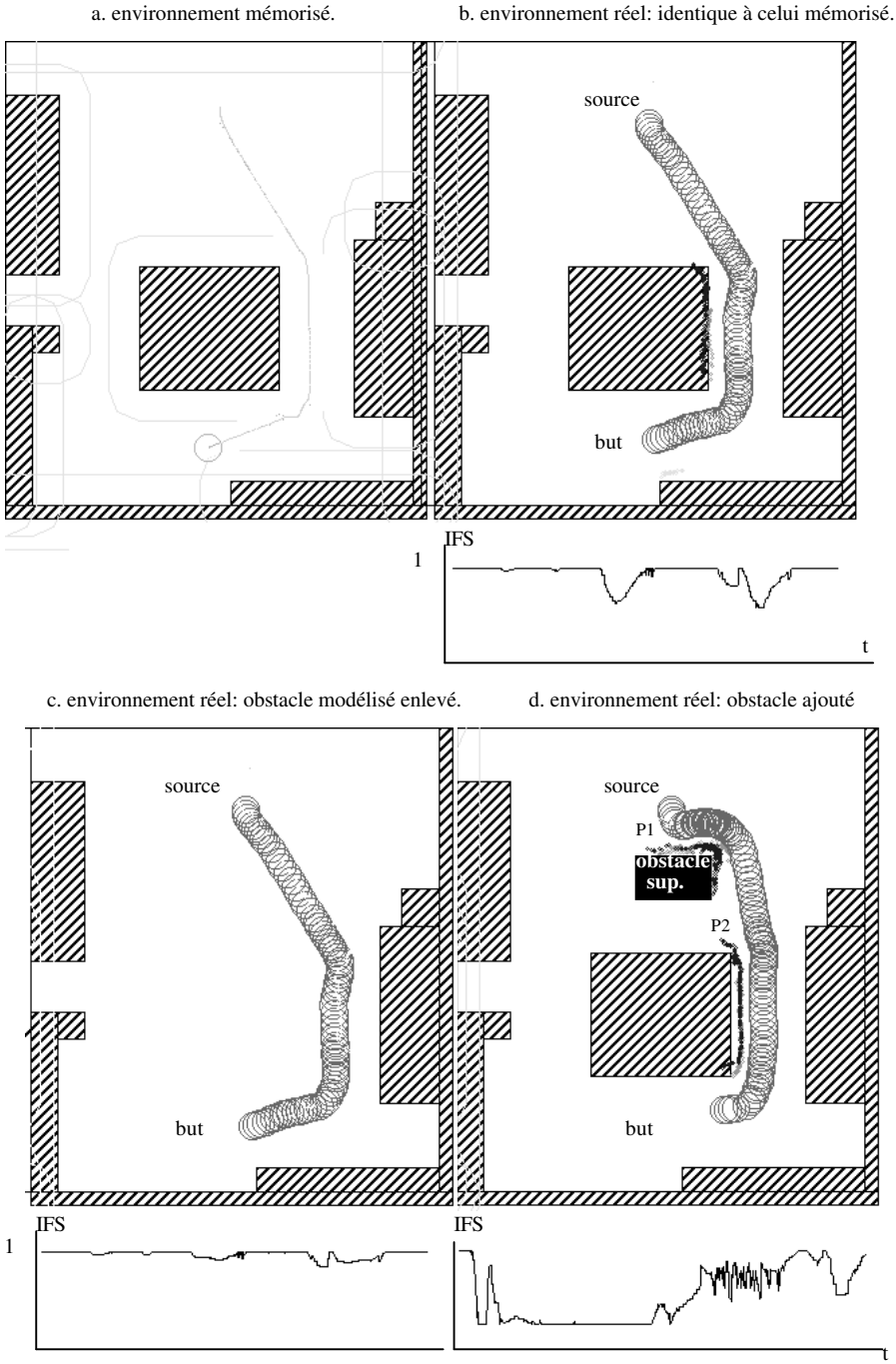


Figure II-33 : Exemples de résultats

On peut observer maintenant que, si on incorpore l'obstacle inconnu de la figure II-33d dans la base de données, la trajectoire optimale générée après planification (fig. II-34) est très proche de celle effectuée par les actions réflexes, ce qui montre qu'à part la réduction de vitesse, le recours à la navigation locale n'est que très peu pénalisant.
Par ailleurs, nous remarquons qu'un écart existe entre les impacts des capteurs et les parois des obstacles ce qui est certainement dû au glissement du robot. C'est pourquoi un module de localisation est nécessaire pour corriger les données odométriques et recaler la position du robot. Ce module sera développé au chapitre III.

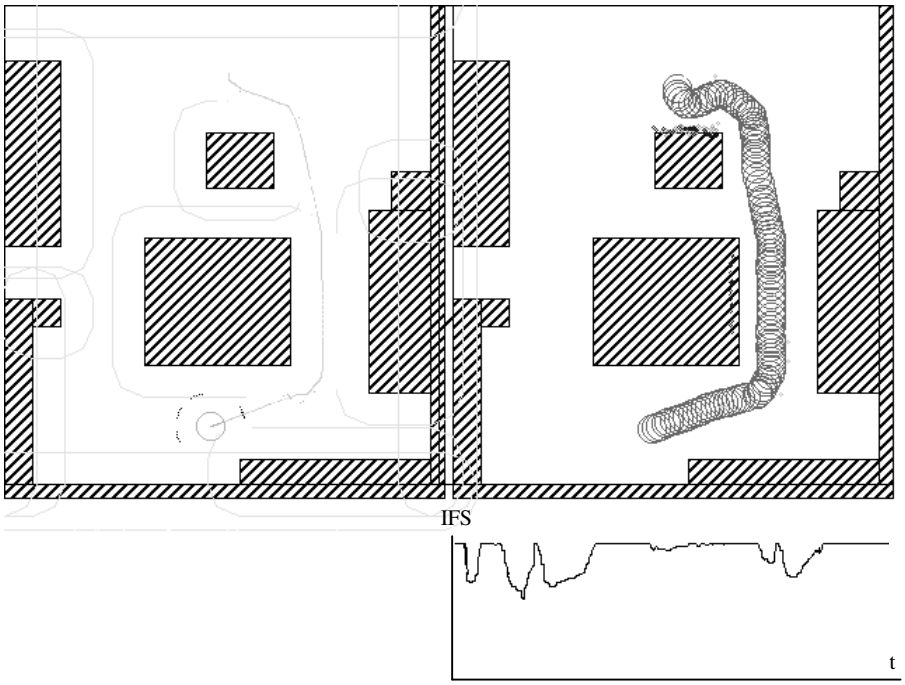


Figure II-34 : Résultat avec obstacle incorporé.

Le travail fait ici est basé sur des SIF réglés empiriquement. Nous allons dans le paragraphe suivant, utiliser la technique décrite au chapitre I pour optimiser en ligne la conception des différents modules de navigation.

6- Optimisation des modules flous de navigation réactive

6-1- Introduction

Notre motivation initiale est d'une part d'éliminer les aspects empiriques de la construction de SIF pour réaliser les actions réflexes de navigation réactive, à savoir le « suivi du mur », le « suivi du milieu de l'espace libre » et la « convergence vers le but » et d'autre part de réaliser les objectifs suivants :
- améliorer les performances du système en accélérant la mise au point et permettre une adaptation à d'éventuelles modification du comportement du robot (dégradation, panne, ...) ;
- conserver la lisibilité des règles afin d'en vérifier la logique ;
- être capable de passer d'un robot à l'autre (ou de la simulation au réel) avec un minimum de réapprentissage et un minimum de risques.

6-2- Principe

Pour chaque action réflexe, deux SIFO sont utilisés, l'un pour générer la commande en vitesse angulaire ω_r et l'autre pour la vitesse linéaire v_r. A vitesse linéaire donnée, la vitesse angulaire est d'abord générée et, après convergence, les règles pour contrôler la vitesse linéaire sont déduites. L'architecture utilisée est de type mini-JEAN introduite au chapitre I (fig. II-35).

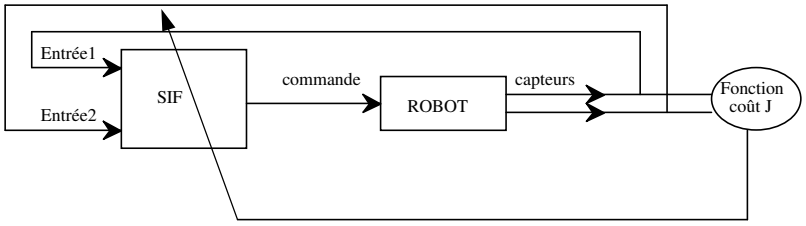


Figure II-35 : Architecture générale d'apprentissage

L'apprentissage est effectué entièrement en ligne en minimisant une fonction coût J (intégrant une erreur quadratique et un terme de régression des poids) pour construire et ajuster les règles floues, soit sur le robot miniature réel Khepera, soit avec son simulateur dédié. Les deux solutions donnent des résultats très voisins.
En s'appuyant sur les résultats du chapitre I, l'algorithme de descente de gradient avec régression des poids et optimisation seulement de la partie conclusion des règles, a été adopté pour satisfaire les objectifs fixés.
Dans la suite, seule l'action «suivi de mur» est détaillée. Les deux autres actions "suivi du milieu de l'espace libre" et la "convergence vers le but" sont construites avec la même technique (pour plus de détails voir [DESPALLES 98]).

6-3- Construction du comportement " Suivi du mur "

6-3-1- Génération des règles de contrôle de la vitesse angulaire

Nous utilisons ici la méthode d'optimisation par descente du gradient sur une fonction coût combinant une erreur quadratique et un terme de régression des poids. L'erreur prise en compte dans la fonction coût est donnée par $\varepsilon_{\omega}=Min(Y,F)-d_{_consigne}$, ou Y est soit D (suivi du mur sur le côté droit) soit G (suivi du mur sur le côté gauche) et $d_{_consigne}$, est la distance de consigne (pour Khepera cette distance est égale à 3cm). L'environnement représenté sur la figure II-36 sert de «piste d'apprentissage». Il a été choisi pour que toutes les situations, correspondant au déclenchement des différentes règles, se présentent. Partant avec une table vide (poids nuls), on met le robot au début de l'opération près du mur. Il fait une acquisition des distances, à partir de ses capteurs, calcule l'erreur à minimiser et actualise les règles déclenchées. Arrivant devant une autre situation, d'autres règles seront déclenchées et actualisées en temps réel ; les poids qui leur correspondent seront ajustés. Au fur et à mesure que l'apprentissage progresse, le robot est de plus en plus capable de faire face à de nouvelles situations.

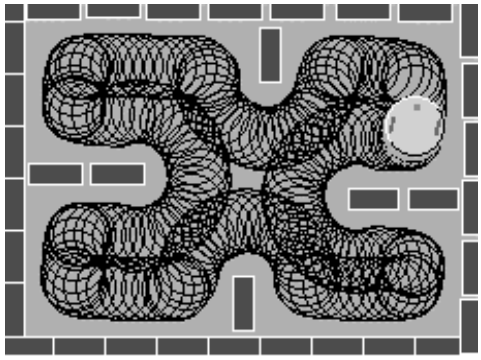


Figure II-36 : Piste d'apprentissage de l'action "suivi de mur"

Si on utilise la méthode décrite au chapitre I dans laquelle le coefficient β du terme de régression des poids est simplement choisi par essais successifs, on constate après convergence que le contrôleur peut fournir une vitesse angulaire supérieure à la vitesse angulaire maximum du robot qui est de 1.58 rad/s. Nous proposons ici une technique de réglage adaptatif du coefficient β décrit dans l'algorithme suivant :

```
Début
    pour toutes les n itérations faire
        si le maximum de la valeur absolue des poids est supérieur à 1.58 rad/s alors
            le coefficient de régression β est incrémenté de Δβ
        sinon si le maximum de la valeur absolue des poids est inférieur à 1.58 rad/s alors
            le coefficient de régression β est décrémenté de Δβ
        fin si
    fin pour
Fin
```

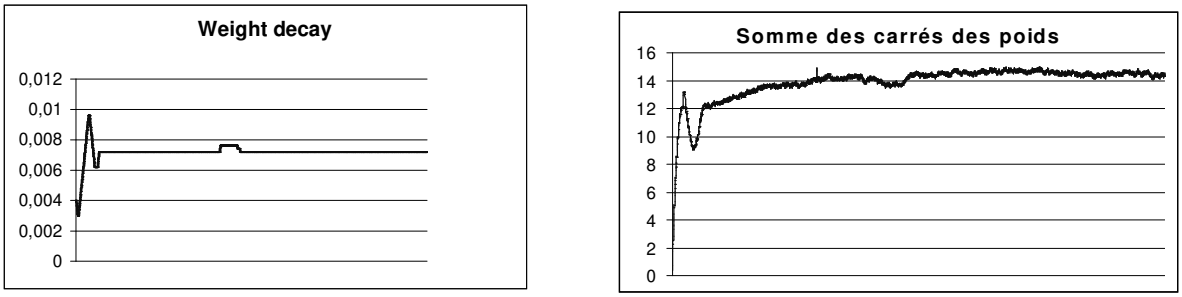
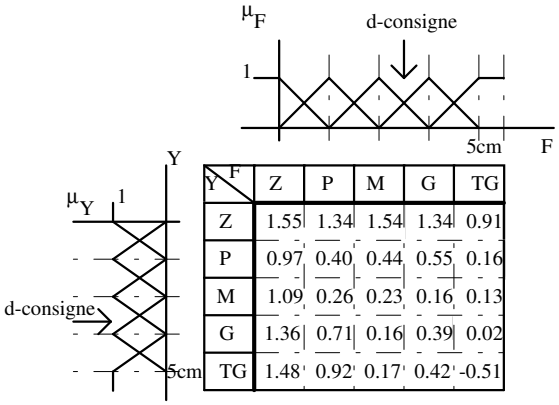


Figure II-37 : Evolution a) du coefficient de régression des poids et
b) de la somme des carrés des poids dans le temps

La somme des carrés des poids (figure II-37) converge vers une valeur quasi constante une fois que le coefficient de régression est stabilisé. En démarrant l'apprentissage avec une valeur petite (environ 0.004), l'influence du gain est prépondérante et les poids évoluent rapidement, puis l'ajustage de β permet d'obtenir des poids proches des contraintes physiques du robot c'est à dire ne dépassant pas la valeur de la vitesse maximale.
Dans ces conditions, les poids convergent rapidement (1 à 3 tours de piste) vers les valeurs de la table de la figure II-38 où les termes linguistiques des entrées sont les suivants : Z (Zéro), P (Petit), M (Moyen), G (Grand) et TG (Très Grand).



Y \ F	Z	P	M	G	TG
Z	1.55	1.34	1.54	1.34	0.91
P	0.97	0.40	0.44	0.55	0.16
M	1.09	0.26	0.23	0.16	0.13
G	1.36	0.71	0.16	0.39	0.02
TG	1.48	0.92	0.17	0.42	-0.51

Figure II-38 : Table de décision de la vitesse angulaire (rd/s)

Nous pouvons maintenant extraire de la table de poids obtenue des règles de commande sous une forme linguistique (ceci dans le cas d'un suivi à droite) (figure II-39). Pour cela, nous attribuons le terme linguistique PG aux valeurs supérieures à 1, PM aux valeurs entre 1 et 0.4, PP aux valeurs entre 0.4 et 0.1, Z aux valeurs entre 0.1 et -0.1, et de même pour les valeurs négatives. On constate que cette table est assez proche de celle donnée au paragraphe 4-2-3-1, ce qui signifie que la stratégie extraite automatiquement par le système d'inférence floue n'est pas très éloignée de celle proposée par l'expert humain [BENREGUIEG 97].

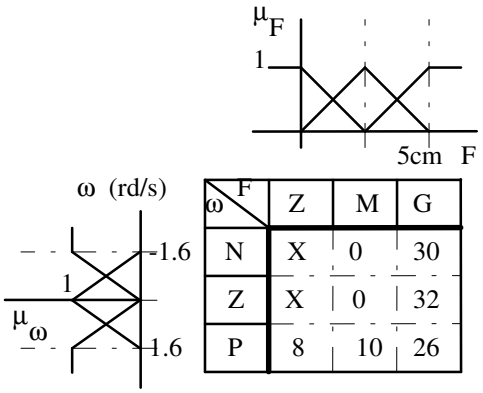
D\F	**Z**	**P**	**M**	**G**	**TG**
Z	PG	PG	PG	PG	PM
P	PM	PM	PM	PM	PP
M	PG	PP	PP	PP	PP
G	PG	PM	PP	PM	Z
TG	PG	PM	PP	PM	NM

Figure II-39 : Interprétation linguistique de la table de décision de la vitesse angulaire

6-3-2- Génération des règles de contrôle de la vitesse linéaire

La même structure a été utilisée pour générer les règles de contrôle de la vitesse linéaire v. Les prémisses du SIF sont la vitesse angulaire ω_r et la distance frontale F. La fonction coût est réalisée avec l'erreur

suivante ε_v=40-max(|v+rω|,|v-rω|)-(1-F/5)*40. Cette erreur à minimiser est choisie d'une part pour maximiser la vitesse linéaire de chaque roue du robot et d'autre part pour la réduire si la distance frontale décroît.
Les termes linguistiques pour ω_r sont définis par : N (négative), Z (approximativement zéro) et P (positive) ; et pour F ils sont : Z (approximativement zéro), M (moyen) et G (grand). On obtient alors la table de la figure II-40.



ω \ F	Z	M	G
N	X	0	30
Z	X	0	32
P	8	10	26

Figure II-40 : Table de décision de la vitesse linéaire

Les valeurs numériques obtenues pourraient être comme précédemment traduites en termes symboliques pour vérifier la cohérence logique des règles. En fait, vu sa simplicité, on constate très facilement que le comportement est logique et se traduit par : plus la vitesse angulaire du robot augmente et plus un obstacle frontal est proche, plus on réduit la vitesse linéaire. Les cases avec une croix ne sont jamais activées car, si le robot tourne à droite, c'est qu'il n'a pas de mur devant lui.

6-3-3- Réaction du SIF optimisé face à des situations nouvelles

Si on place maintenant le robot dans un environnement complexe en figeant les poids obtenus (pour étudier le comportement en généralisation), on constate que le robot a suffisamment appris et qu'il est donc capable d'effectuer un suivi de mur convenable en tout circonstance, comme le montre la figure II-41.

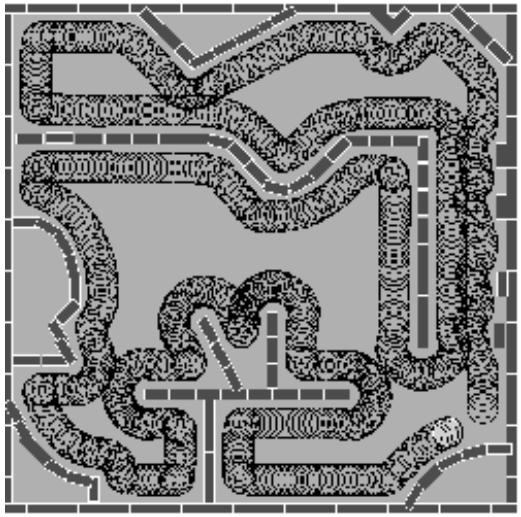


Figure II-41 : Piste de généralisation pour le " suivi du mur "

Toujours en figeant les poids, on peut tester la robustesse du contrôleur flou obtenu. Si par exemple on introduit sur les données capteurs un bruit allant jusqu'à 20% du signal utile, le résultat est similaire à celui de la figure II-41. Nous avons également simulé une dégradation d'une caractéristique physique du robot correspondant par exemple à une crevaison de la roue droite. Ceci entraîne une diminution du diamètre de la roue qui peut être simulée par un ajout d'un biais à la vitesse linéaire de cette roue. Avec des faibles valeurs du biais, le comportement reste satisfaisant, mais néanmoins, progressivement, la forme de la trajectoire se dégrade, même dans des situations simples (fig. II-42).

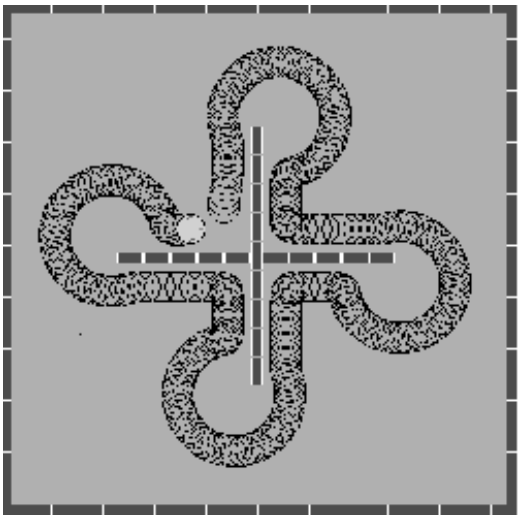


Figure II-42: Trajectoire du robot avec biais et poids fixés

La solution pour remédier à cela est naturellement de relancer la procédure d'apprentissage en libérant les poids pour qu'ils évoluent en continu afin de prendre en compte ces modifications. Après quelques de tours de l'environnement de la figure II-42, les poids convergent vers des nouvelles valeurs qui prennent en compte le biais et le robot est alors capable de suivre une bonne trajectoire (fig. II-43).

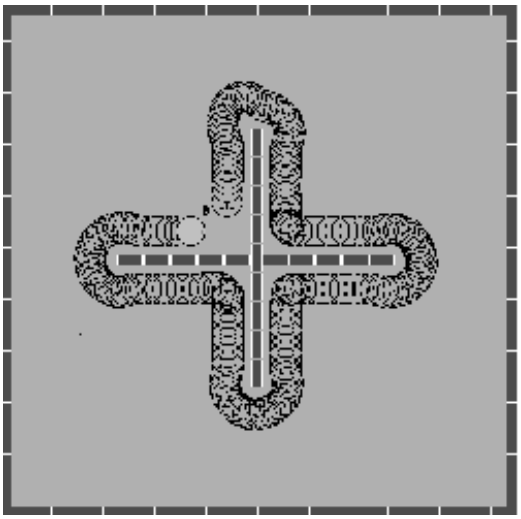


Figure II-43 : Trajectoire du robot avec biais et réapprentissage

Si maintenant nous annulons le biais, les poids reconvergent rapidement vers leurs valeurs initiales ce qui confirme les constatations faites au chapitre I.
On constate donc une bonne adaptativité du contrôleur à des changements de condition. Cela permettra si nécessaire de réutiliser un contrôleur existant (optimisé sur un simulateur ou un robot donné) pour le porter sur un autre robot et le réoptimiser en ligne sans risque au démarrage de l'apprentissage d'effectuer de mauvaises trajectoires qui pourraient être dangereuses avec un gros robot [MAAREF 98].
On peut aussi se poser la question suivante : peut-on laisser l'apprentissage s'effectuer en permanence ou doit-on l'arrêter après convergence des poids comme nous l'avons fait jusqu'à présent ? Si on laisse le robot évoluer dans un environnement "riche", l'apprentissage peut être permanent sans conséquence. Par contre nous avons laissé évoluer Khepera dans une pièce carrée n'exigeant comme mouvement élémentaire, que "aller tout droit" et "tourner à gauche". Replacé dans un environnement plus complexe on constate qu'il ne "sait" plus tourner à droite. Ceci peut paraître contradictoire avec le fait qu'un SIF est par essence un système à caractère local : à chaque pas de temps, seules quelques règles (4 dans notre cas au maximum) sont activées. Il n'y a donc a priori lors de l'optimisation qu'une partie de la table, correspondant à une zone spécifique de l'espace d'entrée, qui est modifiée. Mais en fait, compte tenu du faible nombre des règles (25), du bruit des capteurs et du mouvements du robot, on comprend bien qu'il est impossible de garantir que la modification nécessaire de certaines cases ne s'accompagne pas de la modification non voulue d'autres cases entraînant l'effet de "perte de mémoire" constaté

expérimentalement. Pour contourner cette difficulté, nous proposons de procéder de la façon suivante. Au lieu d'utiliser les propriétés de localité du SIF dans l'espace d'entrée, ce qui s'avère insuffisant, nous allons exploiter les propriétés de localité directement dans l'espace de sortie. En effet, on peut relier des zones de la table de décision à des mouvements particuliers du robot. En fait, ici, nous utiliserons une partition de la table en 3 zones (fig. II-44) qui correspondent à 3 situations du robot par rapport à la consigne : autour de la consigne (zone 2) et loin de la consigne du côté droit(zone 1) ou du côté gauche (zone3). Chacune de ces situations est elle-même associée à un mouvement de base : avancer et/ou tourner légèrement (zone 2), tourner fortement à gauche (zone 1) et tourner fortement à droite (zone 3). Il suffit alors en fonction du contexte observé par les capteurs, de n'autoriser –pour l'apprentissage- la variation des poids que dans certaine(s) zone(s).
Par exemple, avec les 3 règles élémentaires suivantes :
- si F et D < 1.5cm, le robot est du côté droit de la consigne, alors on ne laisse varier que les poids de la zone 1,
- si F et D > 4.5cm, seuls les poids correspondant à la zone 3 sont libres de varier,
- sinon, on laisse varier les poids de la zone 2.
Tout phénomène de " perte de mémoire " disparaît et nous avons pu laisser Khepera évoluer dans différents environnement sous apprentissage permanent.

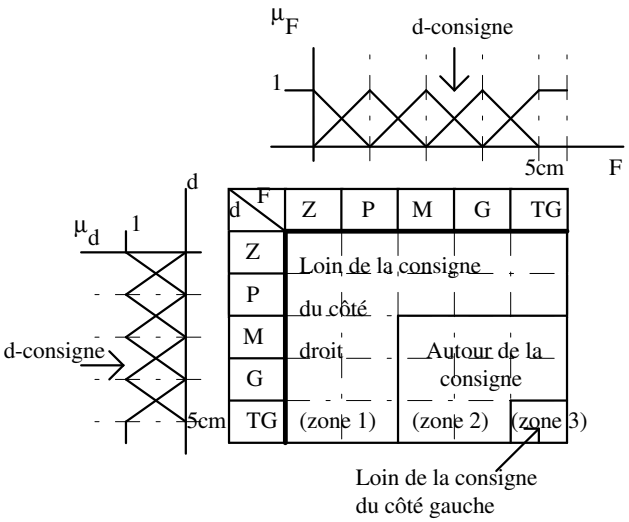


Figure III-44 : Partition de l'espace de sortie (suivi du mur sur le côté droit)

6-3-4- Méthode des moindres carrés récursifs

Comme l'optimisation n'a été portée que sur la partie conclusion des règle, on a donc affaire à un système linéaire en les paramètres à régler, ce qui nous a incité à effectuer des essais avec une techniques classiques éprouvée telle que la méthode des moindres carrés récursifs. Cette méthode fournit une solution en ligne dans la recherche de paramètres qui minimisent une expression quadratique de l'écart entre la solution désirée et la solution obtenue avec ces paramètres. Elle fournit des résultats satisfaisants et la vitesse d'apprentissage est très rapide car chaque poids atteint très vite sa valeur limite, ce qui permet d'avoir presque instantanément le comportement souhaité [DESPALLES 98]. Mais, le robot n'est capable de suivre convenablement le mur d'un circuit que si la richesse de l'environnement ne diffère pas trop de celui de départ, car une fois que les poids ont atteint leurs valeurs limites, ils se figent et le robot ne peut plus réapprendre. Il faut donc que le système soit stable : le parcours ne doit pas évoluer et les caractéristiques du robot ne doivent pas varier. De plus, la table des poids ne peut pas être optimisée par rapport à la vitesse maximale du robot.
On pourrait penser utiliser une technique [WALTER 94] d'estimateur des moindres carrés récursifs avec facteur d'oubli qui permet d'oublier les mesures anciennes qui correspondent à une situation révolue (pour pouvoir s'adapter, il faut savoir oublier). Cette méthode est efficace dans le cas où les paramètres du système sont lentement variables, mais quand ces paramètres varient par sauts comme dans notre application, cette technique est très peu adaptée.

7-Conclusion

Les méthodes de contrôle de systèmes en robotique mobile ont été développées et elles sont généralement classées en deux grandes catégories : la planification globale et le contrôle local. Les méthodes de planification globale sont généralement graphiques, précises et peuvent générer des chemins optimaux sans collisions, mais elles nécessitent que l'espace d'évolution du robot soit parfaitement connu. Les méthodes locales peuvent être assimilées à des méthodes d'actions réflexes, complètement basées sur les informations des capteurs extéroceptifs. C'est-à-dire qu'il n'y a qu'une prise en compte de l'interaction directe robot- environnement. Une localisation absolue du robot n'est pas nécessaire, seule la géométrie relative robot / obstacle doit être évaluée. Le principal inconvénient de ces méthodes est de ne pas toujours garantir le succès de la mission.
Nous nous sommes placés dans ce travail dans la situation usuelle où les déplacements d'un robot mobile se font dans un environnement partiellement connu et nous avons proposé une approche fusionnant ces deux types de navigation (globale et locale). La connaissance *a priori* de l'environnement est mise en mémoire, ce qui permet la comparaison en ligne des données capteurs du robot réel à celles d'un robot virtuel se déplaçant dans l'environnement mémorisé. Un système à base de règles floues a été développé pour effectuer cette comparaison. Le résultat de cette comparaison est un indice permettant la fusion de la navigation par suivi de trajectoire planifiée avec celle par actions réflexes avec un passage doux de l'une à l'autre. Il permet aussi la modulation de la vitesse du robot, qui est élevée dans les endroits les plus sûrs (zones connues) et faible dans les endroits à risque de collision (zones inconnues).
La planification de trajectoire dans l'environnement mémorisé est effectuée par une méthode graphique, celle du graphe de visibilité, que nous avons choisie pour son faible coût en temps et en espace mémoire. L'algorithme A* a été associé au graphe de visibilité pour choisir le chemin optimal, au sens de la distance euclidienne la plus courte. Un contrôleur flou a été réalisé pour effectuer le suivi de chemin planifié par le robot. L'utilisation d'un tel contrôleur, de conception et d'implémentation simples, dont les règles sont déterminées à partir de l'expertise humaine, a permis d'obtenir des résultats satisfaisants. Cette génération de trajectoire a été faite dans le but de générer d'une part le déplacement du robot virtuel dans l'environnement théorique et, d'autre part, de contrôler la navigation du robot réel lorsque l'environnement réel est identique à celui mémorisé.
Pour la navigation locale, nous avons développé indépendamment deux techniques. La première, classique, est celle du champ de potentiel artificiel. Elle est pénalisée par des possibilités de blocage dans des minima locaux et d'oscillations dans des milieux encombrés. Un réglage automatique en ligne des coefficients, basé sur l'emploi de règles floues, a permis d'une part d'éliminer les puits de potentiels et d'autre part de générer des trajectoires mieux lissées. Mais le problème des oscillations demeure délicat à résoudre.
C'est pourquoi nous avons développé une deuxième méthode basée sur la combinaison de comportements élémentaires, pour la plupart basés sur des systèmes d'inférence floue. Nous avons tout d'abord créé et fusionné un comportement de suivi du milieu de l'espace libre vu par le robot et un comportement de convergence vers le but. L'expérimentation a été faite sur les deux robots mobiles Khepera et RMI, pour un passage de porte. Cela nous a permis de montrer la robustesse de la méthode, en particulier vis à vis de mesures effectuées par des capteurs de qualité relativement médiocre. Un autre avantage, à porter au crédit de l'emploi de la logique floue, est l'extrême simplicité du portage de la commande d'un robot à l'autre. A aucun moment on ne fait appel à un modèle du robot et le fait de travailler avec des grandeurs réduites permet de passer d'un robot à l'autre en adaptant seulement la portée des capteurs utilisés, mais sans modification du logiciel, malgré la différence de taille et celle des systèmes de perception.
Cependant, la navigation avec seulement ces deux comportements échoue si un obstacle concave sépare le robot de son but. Pour résoudre ce problème, un troisième comportement élémentaire, de type "suivi de mur" a été développé. Associé à une création de buts intermédiaires de transition, il permet au robot de contourner les obstacles concaves, avant de se rediriger vers son but.
Le recours à des systèmes d'inférence floue pour générer les comportements élémentaires s'est révélé à la fois simple et efficace. Cependant on peut toujours craindre que les règles déduites d'une simple expertise humaine soient plus ou moins largement sous-optimales. Nous avons choisi d'utiliser une méthode d'optimisation par descente de gradient que nous avons présenté plus en détail au chapitre I. Cette méthode est utilisée pour effectuer l'optimisation de paramètres du SIF caractérisant les conclusions des règles floues. Ce point est particulièrement important si on souhaite extraire une base de règles des données expérimentales sans avoir recours à une démarche empirique. Il est alors essentiel que la phase d'apprentissage converge vers des poids stables et traduisibles en concepts linguistiques. L'emploi d'une fonction coût regroupant une erreur quadratique et un terme de régression des poids nous a permis d'atteindre cet objectif. Nous pouvons ainsi obtenir des SIF optimisés, en partant de table de décision vide ou préinitialisée, par apprentissage en ligne. Il est possible de contrôler la logique des règles de décision obtenues après optimisation en retraduisant les résultats numériques en concepts linguistiques. Nous avons pu vérifier par ailleurs que les contrôleurs flous ainsi obtenus avaient une très bonne robustesse par rapport au bruit sur les signaux d'entrée. Néanmoins, si on laisse le contrôleur flou s'adapter en permanence à des modifications de l'environnement ou des caractéristiques du robot, nous avons

constaté que cela pouvait conduire à des "pertes de mémoire", c'est à dire à la disparition de la capacité du robot à exécuter certaines tâches élémentaires. Pour remédier à cela, nous proposons une technique d'apprentissage localisé qui utilise les propriétés de localité d'un SIF dans l'espace de sortie et non dans l'espace d'entrée. Cette technique nous permet de laisser l'optimisation activée en permanence et donc d'avoir un contrôleur flou réellement adaptatif.

Bibliographie II

[ARKIN 90]: R. C. ARKIN, R. R. MURPHY, *Autonomous Navigation in a Manifacturing Environment*, IEEE Transactions on Robotics and Automation. Vol. 6. N° 4. , pp. 445-454, August 1991.

[BENREGUIEG 97]: M. BENREGUIEG, *Aides à la navigation d'un robot mobile dans un environnement partiellement connu*,Thèse de doctorat de l'université d'Evry val d'Essonne, Jan. 1997.

[CORDEWENER 94]: L. CORDEWENER, D. MEIZEL, *On-line Speed Monitoring of Mobile Robots Tasks*, Eng. Applic. Artif. Intell. Vol. 7, No. 2, pp. 151-160, 1994.

[COX 91]: I. J. COX, *Blanche - an experiment in guidance and navigation of an autonomous robot vehicle*, IEEE Transaction on robotics and automation, Vol. 7, N°2, pp. 193-204, April 1991.

[DELAPLACE 91]: S. DELAPLACE, *Navigation a coût Minimal d'un robot mobile dans un environnement partiellement connu*, Thèse de doctorat de l'Université Paris VI, 21 Novembre 1991.

[DESPALLES 98]: F. DESPALLES, *Navigation de robots mobiles autonomes par des systèmes d'inférence floue optimisés*, Mémoire de stage d'ingénieur IIE, Juin 1998.

[HOPPENOT 96]: P. HOPPENOT, M. BENREGUIEG, H. MAAREF, E. COLLE, C. BARRET, *Control of a medical aid mobile robot based on fuzzy navigation*, IEEE / IMACS Int. Conf. CESA'96, Robotics and Cybernetics, pp. 388-393, Lille, 10-13 Juillet 1996.

[HOPPENOT 97]: P. HOPPENOT, *Contribution de la robotique mobile à l'assistance aux personnes handicapées*,Thèse de doctorat de l'université d'Evry val d'Essonne, Nov. 1997.

[KAMGA 96]: A. KAMGA, *Instrumentation et commande d'un robot mobile industriel*, Thèse de doctorat de l'Université de Picardie Jules Verne-Amiens, 16 Octobre 1996.

[KHATIB 86]: O. KHATIB, *Real time obstacle avoidance for manipulators and mobile robot*, Int. Jour. of Rob. Res., Vol.5, No. 1, pp. 90-99, 1986.

[KOREN 91]: Y. KOREN, J. BORENSTEIN, *Potentiel Field Methods and Their inherent Limitations for Mobile Robot Navigation*, Proceeding of the 1991 IEEE International Conference on Intelligent Robots and Automation, pp. 1398-1404, Sacramento, California, April 1991.

[LACROIX 94]: S. LACROIX, R. CHATILA, S. FLEURY, M. HERRB, T. SIMEON, *Autonomous Navigation in Outdoor Environnment: Adaptive Approach and Experiment*, Proceeding of the IEEE International Conference of Robotics and Automation, pp. 426-432, 1994.

[LAKEHAL 95]: B. LAKEHAL , *Commande d'un robot mobile destiné à des applications domotiques*, Thèse de doctorat de l'Université Paris XII, 25 Janvier 1995.

[LIMSAKOUNE 91]: K.LIMSAKOUNE, *Etude de la modélisation des robots mobiles et de leur environnement: Application au développement d'un logiciel CAO optimisant une mission*. Thèse de Doctorat de l'Université de Paris 6, UPMC, Paris, Mai 1991.

[MAAREF 98]: H. MAAREF, F. DESPALLES, C. BARRET**,** *Sensor-Based Reactive Navigation Systems Using Auto-Tuned Fuzzy Controllers*, IEEE Int. Conf. on Advances in Vehicle Control and Safety, AVCS'98, Amiens, 1998.

[MONDADA 93]: F. MONDADA, E. FRANZI, P. LENNE, *Mobile robot miniaturisation: a tool for investigetion in control algorithms*, 3. int. symposium on Experimental Robotics, Kyoto, Japan, 1993.

[NISHIMORI 93]: K. NISHIMORI, S. HIRAKAWA, H. TOKUTAKA, *Fuzzification of Control Timing in Driving Control of a Model Car*, 2° IEEE, ICFS, pp. 297-302, San Fransisco, 1993.

[PIN 93]F. G. PIN, H. WATANABE*, Using Fuzzy Behaviors for the Outdoor Navigation of a Car with Low-Resolution Sensors*, Proceeding of the IEEE, Int. Conference on Robotics and Automation, pp. 548-553, 1993.

[SHIN 92]: D. H. SHIN, S. SINGH, J. J. LEE, *Explicit Path Tracking by Autonomous Vehicles, Robotica*, Vol.10, pp. 539-554, 1992.

[SKEWIS 91]: T. SKEWIS , J. EVANS, V. LUMELSKEY, B. KRISHNAMURTHY, B. BARROWS, *Motion Planning for Hospital Transport Robot*, Proceedings of the IEEE International Conference on Robotics and Automation, pp. 58-63, Sacramento, California, April 1991.

[TOURNASSOUD 88]: P. TOURNASSOUD, *Géométrie et intelligence artificielle pour les robots*, HERMES, 1988.

[WALTER 94]: E. WALTER, L. PRONZATO, *Identification de modèles paramétriques : à partir de données expérimentales,* Masson, 1994.

[ZHANG 93]: J. ZHANG, P. BOHNER, *A Fuzzy Control Approach for Executing Subgoal Guided Motion of a Mobile Robot in a Partially-Known Environment*, Proceeding of the IEEE International Conference of Robotics and Automation, pp. 545-550, 1993.

CHAPITRE III

Localisation absolue par fusion multisensorielle basée sur la théorie des possibilités

1- Introduction

L'un des principaux problèmes en robotique mobile est la localisation précise d'un véhicule dans un environnement connu ou partiellement connu. En effet, la capacité de se localiser est le préambule minimal pour toute application robotique, quel que soit son degré de complexité. Malheureusement, l'imprécision des capteurs odométriques ne permet pas un positionnement suffisamment correct du robot. Les recherches actuelles portent donc sur l'utilisation de capteurs infrarouges ou ultrasonores pour mesurer des distances par rapport à des objets connus de l'environnement et affiner ainsi la position grossièrement obtenue par les capteurs odométriques. On parle alors de localisation "absolue" par opposition à la localisation "relative" qui n'utilise que des informations de vitesse et d'accélération intégrées sur le temps et pour laquelle l'imprécision augmente progressivement.
Les méthodes classiquement utilisées pour la localisation absolue sont basées sur des approches probabilistes. Or, les limitations de ces approches ont été largement mises en évidence, en particulier dans des cas d'informations pauvres [DUBOIS 94]. La théorie des possibilités peut être utilisée pour remédier à ces limitations. L'idée de l'utiliser pour la fusion de données a été proposée récemment par DUBOIS et PRADE [DUBOIS 92] et étudiée dans un cadre théorique. Peu de réalisations concrètes ont été testées pour traiter des données physiques. On peut citer les travaux de DEVEUGHELE [DEVEUGHELE 96] pour le traitement d'image et ceux de DELPLANQUE [DELPLANQUE 98] pour le guidage de véhicules sous-marins.
Nous commençons à présenter dans ce chapitre les différentes étapes intervenant dans un processus de fusion, puis nous détaillerons le processus de fusion dans le cadre de la théorie des possibilités avant de décrire les deux applications réalisées pour corriger la position d'un robot mobile dans son repère du monde [OUSSALAH 98].

2- Fusion multisensorielle en robotique mobile

Commençons à énoncer des définitions de base :
Intégration de données : c'est l'utilisation de la synergie de données dans le but d'aider un système à réaliser une tâche.
Fusion de données : c'est une partie de l'intégration qui combine des données pour permettre d'en obtenir une nouvelle qu'il serait impossible d'avoir autrement. Selon une autre définition, la fusion de données est un procédé intégrant les données de sources multiples afin de produire les représentations les plus spécifiques et les plus pertinentes sur une entité, activité ou événement.
La fusion multisensorielle a été initialement développée surtout dans un contexte militaire pour des objectifs tels que la localisation des cibles ennemies, la fusion d'images radar, l'autonomie de véhicules mobiles, etc... Les méthodes employées ont recours à des techniques diverses issues de l'intelligence artificielle, l'estimation stochastique, la reconnaissances de formes, etc...
Dans un système multicapteur, la fusion permet :
- de compenser les limites d'un capteur par les observations d'autres capteurs ;
- d'exploiter l'effet de redondance des informations capteurs. On obtient ainsi un renforcement ou un affaiblissement des informations suivant qu'elles s'accordent ou non ;
- d'exploiter l'effet de complémentarité des données en considérant que certains domaines peuvent être observés par un capteur, mais pas d'autres.

Quoique le concept de fusion de données soit relativement récent dans la littérature scientifique, il est utilisé depuis fort longtemps dans la nature : ainsi les animaux exploitent l'information auditive et visuelle pour se protéger contre d'éventuels risques courants. De même, pour localiser un emplacement quelconque, ils combinent plusieurs images visuelles des endroits avoisinant le lieu considéré.
On peut trouver dans la littérature plusieurs manières de classer les différentes étapes ou types de fusion. Cette différence provient principalement du niveau où l'opération de fusion est accomplie, de l'objectif de cette opération, du type de sources (ou capteurs) et de l'application considérée.
On peut par exemple reprendre la classification générale proposée par DASARATHY [DASARATHY 97] qui considère trois principaux niveaux de fusion :
- au niveau des données (ou des pixels en traitement d'images) ;
- au niveau des caractéristiques (ou des symboles) ;
- au niveau de la décision.

Il faut noter cependant qu'il y a souvent ambiguité dans le langage utilisé dans la mesure où on peut considérer soit l'entrée, soit la sortie du dispositif de fusion. On peut ainsi avoir en entrée-sortie différents cas tels que, par

exemple, données → données, données → caractéristiques, caractéristiques → caractéristiques, caractéristiques → décision et décision → décision. Il est bien évident par ailleurs que ces niveaux peuvent s'enchaîner.
Un exemple du premier niveau peut être un système télémètrique à plusieurs capteurs dont la sortie (après fusion) est une donnée numérique la plus représentative, obtenue, par exemple, par la moyenne (ou moyenne pondérée) de toutes ou de certaines données.
Pour le second cas, on peut citer, par exemple, un système télémètrique dont la sortie est cette fois-ci une entité géométrique tel que coin, mur, arête,...
Pour le troisième, à partir de la reconnaissance de certaines caractéristiques de l'environnement, on décide d'une action à entreprendre (parmi un certain nombre d'actions possibles) pour le robot.
DASARATHY a également évoqué un quatrième niveau qui est la fusion temporelle, c'est à dire la fusion de données acquises à des temps différents. Ce niveau de fusion ne se situe pas dans la même hiérarchie que les trois précédents, parce qu'en réalité la fusion temporelle peut aussi bien intervenir au niveau des données qu'au niveau des caractéristiques ou de la prise de décision.
Du point de vue des techniques employées, le premier niveau utilise par exemple des méthodes d'estimation stochastique où la valeur renvoyée est une valeur numérique : moyennes, moyennes pondérées, moindres carrés, filtre de Kalman dans le sens où celui ci effectue aussi une moyenne pondérée tout en minimisant la variance de l'état du système, estimateurs à maximum de vraisemblance, etc...
Le deuxième niveau utilise principalement des techniques de reconnaissance de formes, de classification de données, de regroupement (clustering), etc...
Pour le troisième, on peut citer comme méthodes appropriées :
- les méthodes d'analyse d'hypothèses multiples telles que les inférences bayésiennes où, le plus souvent, l'hypothèse maximisant la probabilité à postériori est retenue ;
- les tests de comparaison de moyennes et de variances permettant de valider une constatation expérimentale sur le choix d'une hypothèse particulière ;
- la méthode de Dempster-Shafer où, par exemple, chaque hypothèse se trouve supportée par un ensemble focal et la décision se portera sur l'ensemble focal ayant la plausibilité ou la crédibilité maximale (ou une combinaion convexe des deux).
Pour un niveau hiérarchique donné, l'opération de fusion s'effectue en plusieurs étapes [HAL 97] :
i) Alignement : les observations ou les données sont ramenées dans un même référentiel spatial et temporel.
ii) Corrélation : détermination des relations entre les différentes données.
iii) Association ou mise en correspondance : chaque mesure se trouve associée à l'entité correspondante (le résultat de l'étape de corrélation est évidemment utilisé). Cette étape permet aussi de rejeter les données aberrantes suivant un critère sur la matrice de covariance ou autre.
iv) Combinaison : seules les données obtenues après alignement et qui sont en accord avec l'étape d'association sont combinées pour obtenir une meilleure représentation de l'estimation correspondant à l'attribut avec lequel i) a été réalisée. On notera par ailleurs que l'étape ii) peut être imbriquée à l'intérieur de iii). Par conséquent, trois étapes sont nécessaires pour toute opération de fusion : alignement, mise en correspondance (ou association) et combinaison.
On utilisera également les notions suivantes :
- fusion statique : quand le résultat de l'opération de fusion est obtenu indépendamment des états antérieurs. On exploite alors uniquement les données de l'instant courant.
- fusion dynamique : quand le résultat tient compte des états antérieurs. Tout processus de fusion ayant une formulation itérative rentre dans ce contexte. Cette fusion peut être située au même niveau que la fusion temporelle.
Nous entendons par localisation la "meilleure" estimation de la position d'un véhicule mobile dans un repère absolu. Dans la littérature, on peut trouver plusieurs façons de classer les différentes techniques de localisation. Cette diversité provient principalement de la fonction de localisation et des contraintes imposées par l'environnement dans lequel le robot est amené à naviguer. Nous proposons ici une relecture de ces méthodes sous l'angle de la fusion de données. Cette analyse est motivée par le fait que, dans toutes les méthodes citées dans le littérature, le résultat de la localisation est obtenu en utilisant soit un seul capteur dont la mesure est répétée dans le temps (c'est à dire qu'on effectue une fusion temporelle), soit plusieurs capteurs de même type ou de types différents.
D'autre part, suivant la connaissance de l'environnement sur lequel on opère, on peut distinguer deux manières d'effectuer la localisation :
- Localisation dans un environnement complètement inconnu : le plus souvent, la localisation est effectuée tout en reconstruisant l'environnement, c'est à dire en reconnaissant des entités telles que murs, coins, etc. Le niveau de fusion employé ici est alors la fusion des caractéristiques (les deux autres niveaux peuvent également être utilisés pour générer des résultats intermédiaires).
- Localisation dans un environnement connu ou partiellement connu : dans ce cas la reconstruction n'est pas forcément nécessaire et seule une estimation des coordonnées du robot à partir des informations proprioceptives

et/ou extéroceptives est effectuée. Le niveau de fusion le plus approprié est alors le premier niveau (fusion de données). Dans la suite nous nous restreindrons à cette situation.

3- Fusion d'informations en théorie des possibilités

Supposons que l'information reflétée par chaque source i soit supportée par une distribution de possibilités. Soit N le nombre de sources (i allant de 1 à n). Ces distributions peuvent être combinées conjonctivement ou disjonctivement, ce qui correspond respectivement à une intersection ou une union des distributions sources :

$\forall s \in S, \pi_{\wedge}(s) = \bigcap_{i=1}^{n} \pi_i(s)$ et $\pi_{\vee}(s) = \bigcup_{i=1}^{n} \pi_i(s)$.

Les opérateurs d'union et intersection peuvent en général être n'importe quel opérateur de t-conorme ou t-norme. La combinaison conjonctive suppose que les sources sont fiables, ou du moins certains d'entre elles. La combinaison disjonctive, quant à elle, est moins contraignante en terme de fiabilité, dans le sens où elle nécessite seulement qu'une seule source soit fiable. Evidemment, pour le cas particulier où on sait à l'avance qu'il n'y a aucune source fiable dans le processus de fusion, une autre attitude raisonnable consiste à choisir l'ignorance totale comme résultat de combinaison. Ceci indique tout simplement qu'on ne peut tirer aucune information d'un tel cas. Le résultat de la combinaison conjonctive peut être non normalisé. Cela vient du fait que l'intersection entre les différentes distributions ne contient pas toujours un élément dont le degré de possibilité soit égal à 1.
Nous venons ainsi de voir que les modes de combinaison conjonctive et disjonctive sont tributaires de l'hypothèse de fiabilité des sources. Considérons une source supportée par la distribution π_i et supposons que l'on connaisse le degré de fiabilité α de cette source. Alors, on pourra représenter cette nouvelle information par la distribution π'_i : $\forall u,\ \pi_i'(u) = \max(\pi_i(u), 1-\alpha_i)$.
Cette formulation est en fait un cas particulier de la qualification " α-certitude " [DUBOIS 91], [YAGER 92]. Dans les cas limites, on retrouve bien le cas classique. Si α_i=1, la source est complètement fiable, il n'y a alors aucun changement par rapport à l'état initial ($\pi_i'(u) = \pi_i(u)$). Si α_i =0, on n'a aucune fiabilité sur la donnée de la source et on retrouve alors l'ignorance totale.
Ainsi, combiner de façon conjonctive l'ensemble des sources dont la fiabilité de chacune est connue, revient à :

$\forall u,\ \pi'(u) = \min_i \pi'_i = \min_i \max(\pi_i(u), 1-\alpha_i)$

C'est la combinaison *min. pondérée* [DUBOIS 86].
Par contre chercher la combinaison disjonctive de ces distributions n'a pas vraiment un grand intérêt dans la mesure où on sait par avance la fiabilité des sources.
Un autre type de combinaison, intermédiaire entre les deux précédents, consiste à considérer le cas où on sait qu'il y a K sources parmi les N qui sont fiables, sans toutefois que l'on sache lesquelles. Cette combinaison se résume à considérer tous les sous-ensembles à k éléments (qui sont susceptibles d'être les K sources fiables). On combine les éléments de ces sous-ensembles conjonctivement (car considérés comme fiables), puis on combine disjonctivement le résultat de chacune de ces combinaisons intermédiaires (attitude de prudence pour ne pas perdre un de ces sous-ensembles censé contenir l'information vraie). On aboutit alors à la formulation suivante, appelée *combinaison quantifiée* :

$\forall x, \pi_{(k)}(x) = \max_{K \subseteq N, |K|=k} \min_{i \in K} \pi_i(x)$

3-1- Combinaison adaptative

L'utilisation d'un seul mode de combinaison figé n'est pas toujours souhaitable. Il serait en effet intéressant d'évoluer du mode conjonctif quand les sources s'avèrent fiables et non-conflictuelles vers un mode disjonctif lorsqu'elles s'avèrent par contre conflictuelles (ou non fiables). Dans ce but, DUBOIS et PRADE ont développé une règle de combinaison adaptative que nous appellerons par la suite règle « CADP » (Combinaison Adaptative de DUBOIS et PRADE) [DUBOIS 92].

3-1-1- Règle CADP

Cette règle est de la forme :

$$\pi(x) = \max\left(\frac{\min(\pi_1(x), \pi_2(x))}{\sup_x[\min(\pi_1(x), \pi_2(x))]}, \min(\max(\pi_1(x), \pi_2(x)), 1 - \sup_x[\min(\pi_1(x), \pi_2(x))])\right)$$

On reconnaît ici une distribution conjonctive normalisée, avec un niveau de fiabilité égal à la plus grande intersection des deux distributions, et dont l'étendue est restreinte par le support des deux sources (zone de consensus). La figure III-1 donne un exemple de la combinaison des distributions $\pi_1$etπ_2 par cette règle.

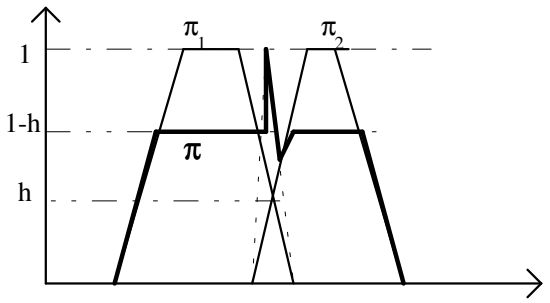


Figure III-1 : Exemple de combinaison par la règle CADP

La quantité $1-\sup_x[\min(\pi_1(x),\pi_2(x))]$ désigne le terme se rapportant au conflit. Il représente le complément de la hauteur maximale de l'intersection des deux sources π_1 et π_2 (le niveau h sur la figure). Ainsi, si les noyaux des deux distributions s'intersectent, c'est à dire, $\sup_x[\min(\pi_1(x),\pi_2(x))]=1$, la règle précédente devient : $\pi(x)=\min(\pi_1(x),\pi_2(x))$ (16). On retrouve alors la combinaison conjonctive. Si, par contre, ce terme tend vers zéro sans toutefois s'annuler (sinon l'expression est indéterminée), la règle tend alors vers la combinaison disjonctive : $\pi(x)=\max(\pi_1(x),\pi_2(x))$ (17).

Le terme $\sup_x[\min(\pi_1(x),\pi_2(x))]$ traduit la certitude accordée à la distribution résultante, ou bien son degré de confiance.

Le rapport $\frac{\min(\pi_1(x),\pi_2(x))}{\sup_x[\min(\pi_1(x),\pi_2(x))]}$ permet de normaliser le résultat de la combinaison conjonctive des deux distributions.

Le terme $\min(\max(\pi_1(x),\pi_2(x)),1-\sup_x[\min(\pi_1(x),\pi_2(x))])$ limite l'étendue du conflit ou par opposition, du degré de confiance (le degré de confiance est égal au complément du facteur de conflit), au support des deux distributions.

Le terme $\min(\max(\pi_1(x),\pi_2(x)),\ 1-\sup_x[\min(\pi_1(x),\pi_2(x))])$ limite l'étendue du conflit, ou par opposition, du degré de confiance (le degré de confiance est égal au complément du facteur de conflit), au support des deux distributions

3-1-2-Généralisation de la règle CADP

Afin de généraliser la règle CADP à plusieurs sources, DUBOIS et PRADE [DUBOIS 94] ont proposé l'utilisation de la combinaison quantifiée. Le principe est de déterminer deux valeurs n (optimiste) et m (pessimiste) entre lesquelles on suppose que l'ensemble des sources fiables se trouve. Ces valeurs correspondent respectivement au nombre maximum de sources (distributions) ayant une intersection non nulle de leur support - c'est à dire susceptibles d'être en concordance-, et au nombre maximum de sources (distributions) ayant une intersection non nulle de leur noyau - c'est à dire en concordance totale-.

De plus, on suppose qu'une fois ces deux valeurs déterminées, on ne sait pas distinguer les sources qui sont fiables de celles qui ne le sont pas. On utilise alors une combinaison quantifiée. La règle généralisée s'écrit :

$$\pi(x)=\max\left(\frac{\pi_{(n)}(x)}{h(n)},\min(\pi_{(m)}(x),1-h(n))\right)$$

avec :

$h(n)=\sup[h(K),\ |K|=n]$,

$h(K)=\sup_x\left[\min_{i\in K}\pi_i(x)\right]$,

$m=\sup[\ |K|,\ h(K)=1]$,

$n=\sup[\ |K|,\ h(K)>0]$,

et $\forall x,\ \pi_{(k)}(x)=\max_{K\subseteq N,\,|K|=k}\ \min_{i\in K}\pi_i(x)$.

h(K) désigne la plus grande intersection entre les sources appartenant au sous-ensemble K de l'ensemble N des sources. h(n) exprime la plus grande intersection non nulle du plus grand nombre de sources possibles. Ce terme traduit aussi la concordance entre les différentes distributions. Pour N=2, on retrouve la formulation initiale de la règle CADP.
La figure III-2 montre un exemple d'application de la règle CADP généralisée. Les distributions à combiner sont π_1, π_2 et π_3. Il n'y a que deux distributions dont l'intersection des noyaux est non nulle, donc m=2, et trois distributions (plus grand nombre possible) dont les supports s'intersectent, donc, n=3. Dans la figure III-2b sont illustrées les combinaisons quantifiées $\pi_{(3)}$ et $\pi_{(2)}$ correspondant respectivement à n et m. Enfin, le résultat final est donné sur la figure III-2c.

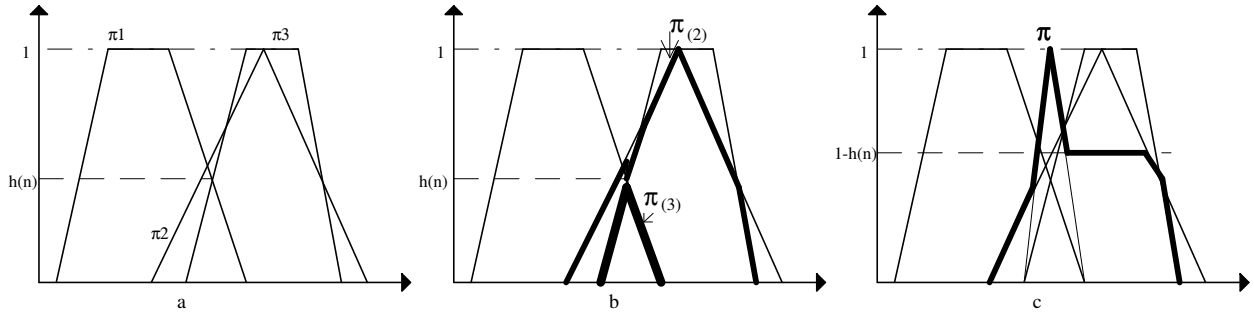


Figure III-2 : Exemple de combinaison par la règle CADP

4- Application et comparaison avec la méthode de Kalman

4-1- Cadre expérimental

Nous nous proposons de tester la fusion possibiliste pour déterminer une meilleure estimation de la position absolue du robot Khepera dans un environnement balisé. Ces balises sont constituées par un ensemble de diodes infrarouges dont le signal est reçu par les capteurs du robot. Après avoir décrit le matériel et le contexte de l'expérience, nous détaillerons l'algorithme de fusion avec les étapes d'alignement, d'association et de combinaison. Pour cette application, nous attacherons une grande importance à la modélisation des capteurs infrarouges et odométrique. Ensuite, nous expliquerons les tests effectués en statique et en dynamique, aussi bien en simulation qu'en implémentation réelle. Dans les deux cas, l'influence du nombre de balises et de l'imprécision initiale seront étudiés. Les résultats de cette application seront ensuite comparés avec ceux obtenus en utilisant le filtrage de Kalman et une discussion comparative des caractéristiques des deux algorithmes sera effectuée.
Les expériences sont réalisées sur le robot miniature Khepera relié à un ordinateur de type PC. La communication se fait par le port série et consiste à envoyer des commandes moteur au robot et à lire les valeurs des capteurs. Les capteurs extéroceptifs infrarouges de Khepera (utilisés en réception, leur portée est de l'ordre de 20cm) sont employés pour détecter les signaux issus de balises actives constituées par des diodes infrarouges. Ces balises sont fixes, de position connue et leur activation est commandée par le PC via le port parallèle. La figureIII-3 résume le montage utilisé.
Le principe des expériences est de faire naviguer le robot et de comparer au bout d'un certain temps sa position réelle et la position après correction par les méthodes testées. L'écart constitue l'erreur de localisation.

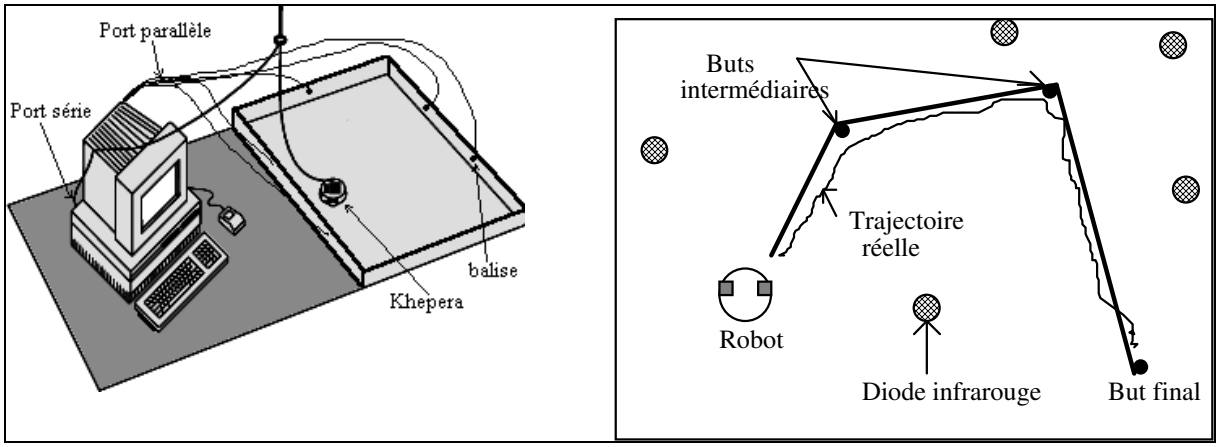


Figure III-3 : Montage utilisé pour les expérimentations

Dans ces expériences, chaque diode émettrice est excitée successivement pendant un temps très court, avec un temps mort de quelques dixièmes de secondes entre l'éclairement de deux diodes. Ceci assure l'absence d'interférences entre les signaux mesurés. Par ailleurs, la mise en correspondance écho-capteur est triviale puisqu'on sait à chaque instant quelle balise est actionnée et quel capteur est déclenché (le plus excité).
Par ailleurs, pour des raisons qui sont dues aussi bien à la faible puissance émise par la diode qu'aux limites des récepteurs de Khepera, les observations ne permettent pas d'obtenir directement des informations sur la direction du robot. En conséquence, par la suite, on ne s'intéresse qu'à corriger les coordonnées x et y du centre du robot. Son orientation ne sera estimée que par l'odomètre.

Par ailleurs, pour pouvoir appliquer les équations du filtre de Kalman, il faut déterminer les bruits associés aux modèles d'état et d'observation. Pour modéliser le bruit d'état nous avons besoin de comparer le résultat issu des équations du modèle d'état, qui représente l'évolution de l'odométrie, et les positions réelles du robot. On tient compte ainsi aussi bien de l'erreur sur le modèle mathématique que des glissements des roues du robot. Afin que les mesures soient décorrélées, les estimations trouvées seront en général légèrement majorées.
En fait, pour représenter le bruit des capteurs odométriques, nous avons utilisé le modèle proposé par OPDERBECKE [OPDERBECKE 94]. Ce modèle suppose que chaque capteur odométrique est soumis à un bruit blanc d'écart-type σ défini par :

$$\sigma(l) = \sqrt{\sigma_0^2 + \left(\varepsilon_0 \times l\right)^2}$$

avec l la distance parcourue par la roue, σ_0 une constante caractérisant l'erreur de quantification des codeurs répartie entre plus ou moins un demi-incrément et ε_0 une constante de proportionnalité avec la distance.
Cette formule est ensuite utilisée pour déterminer l'évolution théorique des écarts type dans le cas d'un mouvement rectiligne :

- Sur l'axe de la trajectoire, $\sigma_x(k) = \sigma(l_0)\sqrt{\frac{k}{2}}$;

- Sur l'axe perpendiculaire, $\sigma_y(k) = \frac{\sqrt{2}}{D} \times l \times \sigma\left(l_0\right) \times \sum_{n=1}^{k} \sqrt{n}$;

avec k le nombre d'itérations et l_0 la distance parcourue pendant une itération.
Pour valider le modèle, on fait naviguer le robot sur plusieurs trajets rectilignes en répétant plusieurs fois la mesure. On pourra ainsi déterminer la variance (sur l'axe du déplacement et l'axe perpendiculaire) pour chaque distance en assimilant l'erreur à l'écart entre la donnée du modèle odométrique et la position réelle du robot. En effectuant le lissage des données trouvées par les formulations précédentes, on peut déduire une estimation des paramètres σ_0 et ε_0.
La modélisation du bruit d'observation doit tenir compte d'une part des conditions externes (principalement les conditions de luminosité) et de l'imprécision avec laquelle sont connues les variables d'état (position du robot). Pour ce faire, une campagne de mesures est effectuée pour différentes distances (à chaque fois la mesure considérée est la moyenne d'une dizaine de mesures). Ces mesures sont réalisées aussi bien dans des conditions de luminosité normale que dans le noir, ce qui nous permet d'éliminer à chaque fois la composante liée au bruit de lumière. On détermine aussi (par répétition plusieurs fois de la campagne de mesures) à chaque distance la

variance associée. La fonction de lissage permettant de donner la variance pour chaque distance, sera majorée pour tenir compte de l'effet d'imprécision sur les variables d'état.
Ces variances ont également été utilisées dans les applications de simulation pour bruiter artificiellement les données.

4-2- Principe général de la solution adoptée

La figure III-4 représente l'algorithme général de la méthode. Il est composé de trois parties :

1- L'estimation odométrique (mesures proprioceptives) : les capteurs odométriques fournissent des déplacements élémentaires des roues du robot l_k^g et l_k^d dans le repère mobile (lié au robot) entre deux instants d'échantillonnages. Ces distances sont représentées par des distributions de possibilités grâce à l'étape de modélisation décrite dans le paragraphe 4-2-4-1. Afin d'estimer l'état du système c'est à dire la configuration du robot (définie dans un plan par ses coordonnées x, y et son cap θ) par rapport à son environnement, il est nécessaire de définir un modèle d'évolution du véhicule. Pour un robot mobile, plusieurs modèles d'état sont possibles selon les approximations faites sur la trajectoire du robot (approximation par un arc de cercle, un segment, etc..). Le modèle d'évolution que nous avons utilisé est le suivant :

$$\begin{cases} x_{k+1} = x_k + \dfrac{l_k^g + l_k^d}{2} \times \cos(\theta_k) \\ y_{k+1} = y_k + \dfrac{l_k^g + l_k^d}{2} \times \sin(\theta_k) \\ \theta_{k+1} = \theta_k + \dfrac{l_k^d - l_k^g}{D} \end{cases}$$

où D désigne l'entraxe (distance entre les deux roues).
Ces équations supposent que la trajectoire parcourue par le robot entre deux instants successifs est assimilée à un segment de droite pour l'accroissement linéaire et un arc de cercle pour $\Delta\theta_k$.
Chaque variable coordonnée est déterminée sous forme d'une distribution de possibilités par application du principe d'extension de Zadeh. La mise en œuvre directe de cet outil pose des problèmes pratiques à cause de la nécessaire discrétisation des domaines des variables. Nous avons donc préféré utiliser la méthode de DONG [DONG 87a] [DONG 87b] pour manipuler les distributions de possibilité. Cette méthode est basée non pas sur la discrétisation des univers de discours, mais sur celle de l'intervalle [0, 1] des degrés de possibilité, c'est-à-dire sur une représentation sous forme d'alpha-coupe des distributions.
2- La gestion des observations (mesures extéroceptives) : chaque balise permet d'obtenir une estimation sur les variables d'état (coordonnées x et y de la position absolue du robot définies dans le même repère, qui est celui où sont définies les coordonnées des balises) en tenant compte de la donnée odométrique. En effet, après une étape de calibration, chaque balise renvoie une distance représentée par une distribution de possibilités grâce au modèle adopté décrit au paragraphe 4-2-4-2. Une étape de sélection est au préalable appliquée pour éviter les situations conflictuelles (§4-2-1). Ensuite, ayant l'estimation odométrique, chaque distance est transformée par l'étape d'alignement (§4-2-2) dans le repère absolu en coordonnées x et y et ceci à l'aide du principe d'extension de Zadeh de sorte que ces coordonnées soient représentées par des distributions de possibilités.
3-La fusion : la méthode possibiliste développée permet une correction de la position estimée d'après les capteurs odométriques en fonction des observations fournies par les capteurs infrarouges. Le processus de correction consiste à utiliser la règle de fusion adaptative présentée au paragraphe 3 pour combiner ces positions et la position estimée.
Nous précisons par la suite où et comment dans ce schéma sont effectuées les opérations principales : association, alignement et combinaison.

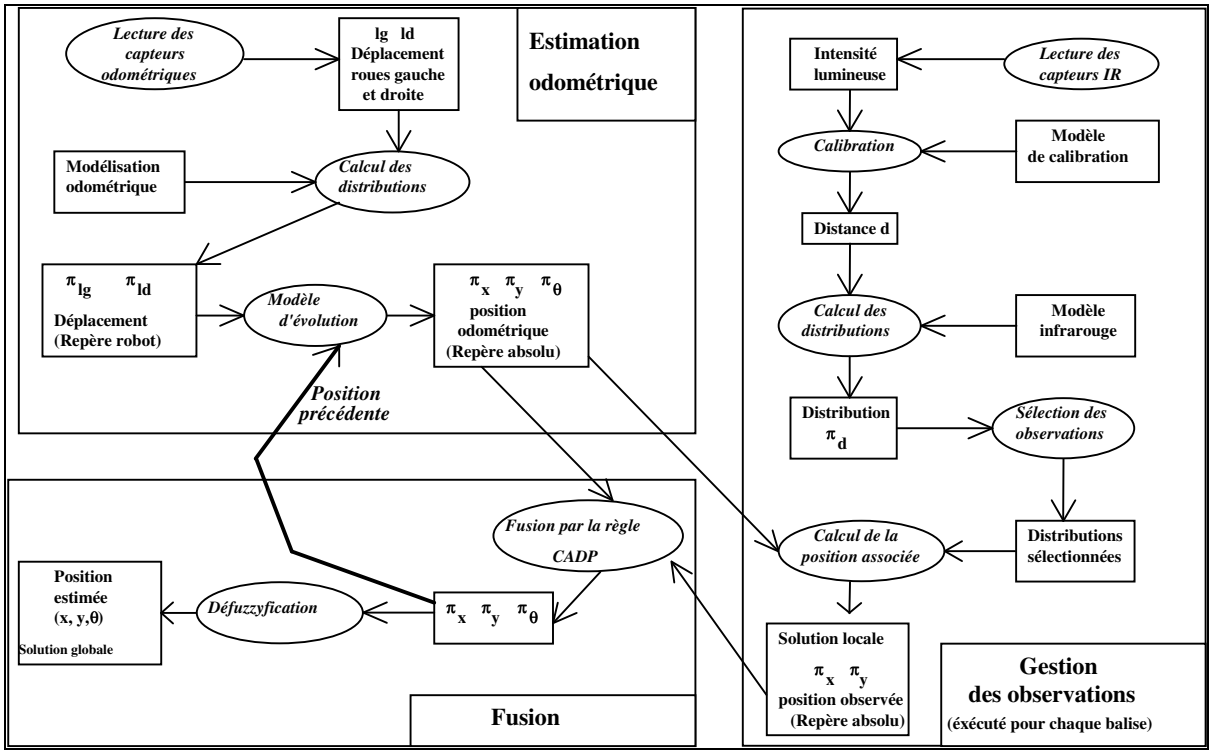


Figure III-4 : Algorithme général de la méthode

4-2-1- Association

L'étape d'association ou de mise en correspondance des données avec les capteurs correspondants est directement contrôlée par l'utilisateur. En effet, via le port parallèle, on peut décider du capteur et de la diode à déclencher. De plus, les diodes sont actionnées successivement.

Cependant, dans le but d'homogénéiser le résultat avec celui de la méthode de Kalman, nous avons introduit une phase de sélection des données (correspondant à la fonction «sélection des observations» de la figure III-4). Dans cette dernière, seules les données dont les supports des distributions de possibilités s'intersectent sont considérées. Autrement dit, toutes les situations conflictuelles sont éliminées à priori.

4-2-2-Alignement (solutions locales)

Le problème est de donner une indication sur la position du robot à partir de l'information de distance (d) fournie par une diode (correspondant à la fonction «calcul de la position associée» de la figure III-4). Pour ce faire, nous nous servons de la position odométrique M_0 (donnée dans un repère absolu) du robot. Nous choisissons comme nouvelle position du robot M (solution locale) issue de l'information fournie par la diode, le point M aligné avec M_0 et situé à une distance d de la diode (fig. III-5). Ce choix correspond à la position la plus proche de la position odométrique située à la distance d de la diode. De façon plus formelle, si on désigne par (x_a,y_a) les coordonnées de la balise A par rapport au repère absolu Oxy, (x_0,y_0) les coordonnées de la position odométrique dans ce même repère et d la distance renvoyée par le capteur infrarouge (diode), la position de M de coordonnées x et y représentant la solution locale (position du robot) vue par la balise A est donnée par :

$$\begin{cases} x = x_a + \dfrac{(x_a - x_o) \times d}{\sqrt{(x_o - x_a)^2 + (y_o - y_a)^2}} \\ y = y_a + \dfrac{(y_a - y_o) \times d}{\sqrt{(x_o - x_a)^2 + (y_o - y_a)^2}} \end{cases}$$

Les calculs sont faits pour chaque balise disponible en utilisant le principe d'extension de Zadeh. Dans l'équation précédente, les variables sont manipulées sous la forme de distributions de possibilités hormis les coordonnées (x_a,y_a) qui sont exprimées par des singletons.

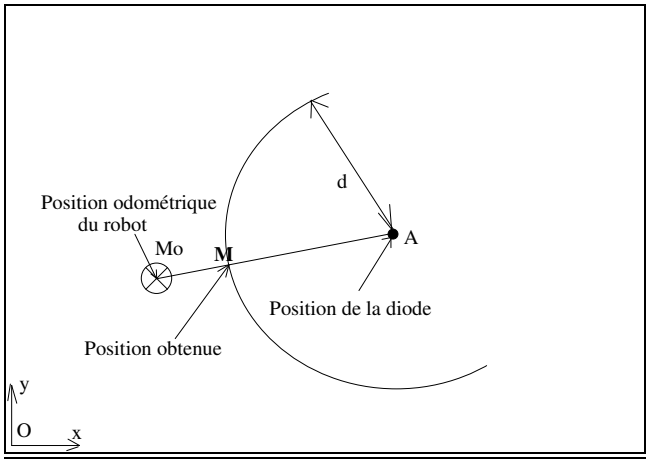


Figure III-5 : Illustration de la première estimation de la position à partir de la position **de la diode et de la position odométrique.**

4-2-3- Combinaison (solution globale)

Après avoir déterminé les solutions associées à chacune des diodes (position absolue du robot vue par une diode) indépendamment les unes des autres, l'étape suivante consiste maintenant à fusionner ces solutions individuelles pour en extraire une solution globale (correspondant à la fonction «fusion par la règle CADP» de la figure III-4). Cette combinaison est effectuée par le biais de la règle CADP de combinaison adaptative. On obtient ainsi une distribution de possibilités correspondant à la position sur x et une correspondant à celle sur y (fig. III-6). Afin d'extraire une valeur numérique de ces distributions, on effectue une défuzzyfication par la méthode du centre de l'aire.

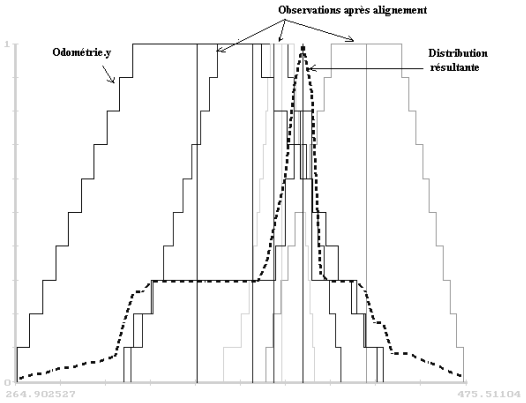


Figure III-6 : Exemple de distributions correspondant à l'estimation de la position sur y

4-2-4- Modélisation possibiliste des informations capteurs

Le but de la modélisation est de donner une représentation des données capteurs dans le cadre de la théorie des possibilités associant à chaque mesure du capteur une distribution de possibilités. Cette distribution représente le degré de possibilité avec lequel la mesure d est égale à la vraie valeur de la distance d_v. Naturellement, plus le capteur est de bonne qualité, plus la distribution π_{dv} est spécifique. La démarche adoptée consiste à effectuer une analyse statistique des données capteurs à partir des relevés expérimentaux.

4-2-4-1-Capteur odométrique

Considérons la distribution de possibilités sous forme d'ensembles emboîtés (ou représentation en alpha-coupes) de la figure III-7.

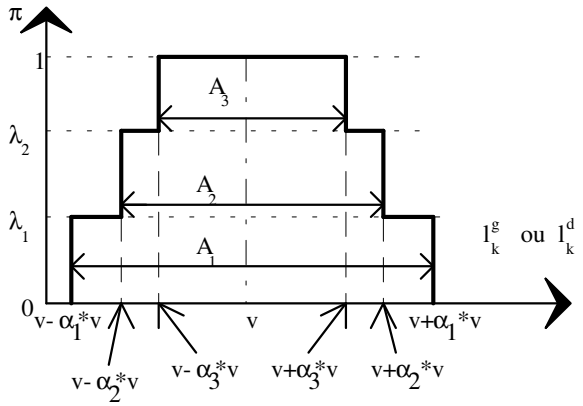


Figure III-7 : Distribution choisie pour les mesures odométriques

En pratique le problème revient à trouver les ensembles A_i correspondant à un certain choix des niveaux d'alpha-coupes. Pour notre cas, afin de simplifier la manipulation et la représentation, nous avons restreint notre choix par les contraintes suivantes :

i) limiter à trois le nombre de niveaux d'alpha-coupes ;

ii)- obtenir des ensembles A_i les plus distincts possible ;

iii)- obtenir des distributions symétriques ;

iv) avoir des bornes des ensembles A_i proportionnelles au terme central v (pour tenir compte du fait que l'imprécision d'une mesure odométrique augmente avec la distance parcourue).

Nous avons alors effectué une série de tests statistiques. Ces derniers consistent à faire parcourir au robot en ligne droite des distances connues (de 5 à 60 cm), chaque opération étant répétée plusieurs fois. Nous relevons, dans chaque cas, la position réelle du robot et sa position odométrique (x, y et θ).

On obtient ainsi pour chaque valeur vraie d_v un ensemble de données odométriques mesurées, rangées suivant le tableau III-1 ci-dessous :

Valeur vraie	Données odométriques associées
dv_1	d_{11} d_{12} d_{13} ... d_{1n_1}
dv_2	d_{21} d_{22} d_{23} ... d_{2n_2}
.	. . . .
dv_i	d_{i1} d_{i2} d_{i3} ... d_{in_i}
.	. . . .
.	. . . .
dv_n	d_{n1} d_{n2} d_{n3} ... d_{nn_n}

Tableau III-1 : Mesures odométriques

Nous avons en premier lieu déterminé le centre v de la distribution. Il suffit pour cela d'effectuer pour chaque valeur dv_i la moyenne des différentes données odométriques associées. On obtient ainsi un décalage systématique qui, sur l'axe du déplacement, est pratiquement constant en valeurs relatives pour tout dv_i et est égal à 3.5 %[1] , c'est à dire : $v = d_{moy} - 0.035x\ d_{moy}$.

En termes de données statistiques, il est raisonnable de supposer qu'on est d'autant plus certain sur l'occurrence d'un événement que ce dernier se répète pendant les différentes séries de tests.

Ainsi, la détermination de la distribution de possibilités revient à celle des deux niveaux de possibilités $\lambda_1$et λ_2 ($\lambda_2 > \lambda_1$) (le troisième étant le niveau 1) et des paramètres α_1, α_2 et α_3 délimitant les ensembles α–coupes correspondants, tels que pour chaque dv_i :

- 100 % des données associées $[d_{ij}.... d_{in_i}]$ sont à l'intérieur de l'intervalle $[dv_i(1-\alpha_1)\ dv_i\ (1+\alpha_1)]$.
- $(1-\lambda_1)$x100 % de $[d_{ij}.... d_{in_i}]$ sont à l'intérieur de $[dv_i(1-\alpha_2)\ dv_i(1+\alpha_2)]$.
- $(1-\lambda_2)$x100 % de $[d_{ij}.... d_{in_i}]$ appartiennent à $[dv_i(1-\alpha_3)\ dv_i(1+\alpha_3)]$.

[1] Ce paramètre tient compte aussi bien du biais systématique des données odométriques que de l'influence exercée par l'inertie du fil reliant le robot Khepera au PC.

En outre, la contrainte i) impose que α_1-α_2 et α_2-α_3 soient aussi larges que possible.
On teste alors toutes les combinaisons à deux éléments des neuf niveaux allant de 0.1 à 0.9 (programme itératif). La meilleure solution obtenue est λ_1=0.3 et λ_2=0.6. En termes de confiance, cela veut dire que :
- Dans 40 % des cas, la vraie valeur se trouve à l'intérieur de l'ensemble délimité par v-α_3.v et v+α_3.v .
- Dans 70 % des cas, cette valeur est à l'intérieur de l'ensemble délimité par v-α_2.v et v+α_2.v.
- Dans tous les cas, la vraie valeur est à l'intérieur de l'ensemble délimité par v-α_1.v et v+α_1.v.

ν est la donnée obtenue à partir de la lecture odométrique après élimination du décalage systématique.

Les paramètres (α_1,α_2,α_3) sont fixes et identiques pour l_k^g et l_k^d. Nous avons trouvé les valeurs suivantes : α_1=2.6 %, α_2=1.4 % et α_3=0.6%.

4-2-4-2- Capteur infrarouge

Il faut tout d'abord calibrer chaque capteur de sorte que sa sortie corresponde à une distance. Plusieurs tests ont été effectués à différentes distances en relevant chaque fois une moyenne d'une dizaine de mesures. Ceci permet de réduire au maximum l'influence des bruits de mesure. Nous obtenons la courbe de calibration où une approximation polynomiale de degré quatre permet un bon lissage des points (fig. III-8).

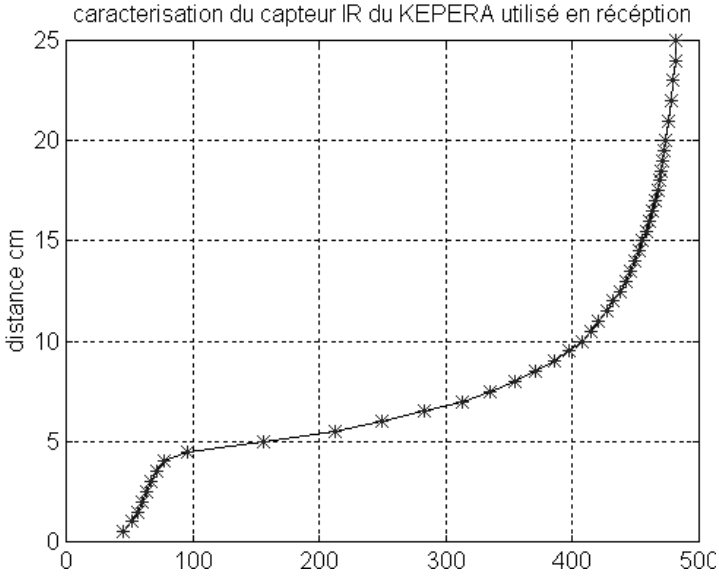


Figure III-8 : Calibration d'un capteur infrarouge :distance en cm en fonction de la réponse capteur (0 à 500).

Afin de déterminer la distribution de possibilité correspondant à chaque distance, une nouvelle série d'acquisitions est effectuée en utilisant cette fois-ci les résultats issus de la calibration. A l'inverse des relevés odométriques, les résultats des tests sont très proches les uns des autres (évidemment pour une même distance). Le premier critère i) du paragraphe précédent relatif à la séparation la plus nette possible entre les différents sous-ensembles A_i est alors très difficile à vérifier pour plusieurs niveaux d'alpha-coupes. C'est pourquoi nous avons préféré conserver ici une simple représentation trapézoïdale (fig. III-9).
Cette représentation signifie que, dans la majorité des cas, la distance vraie est située dans un intervalle de 0.5% autour de la distance mesurée (moyenne de 10 mesures) et qu'en aucun cas la mesure vraie n'est à l'extérieur d'un intervalle de 1% autour de la mesure.

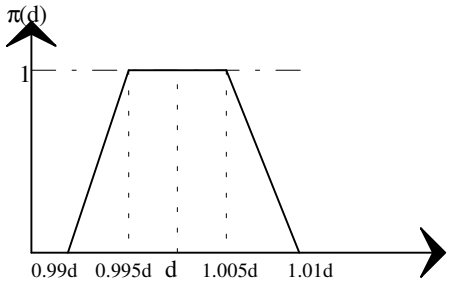


Figure III-9 : Distribution de possibilités obtenue pour un capteur IR. d est la distance renvoyée par le capteur.

4-3-Résultats des tests et comparaison avec la méthode de Kalman

Afin de valider l'approche décrite précédemment, nous avons effectué une série de tests avec le robot miniature Khepera, aussi bien en simulation qu'en implémentation réelle. Puis, les résultats sont comparés avec la méthode du filtrage de Kalman considérée comme la méthode de référence.
La simulation est très proche du cadre expérimental et offre l'avantage de disposer de données parfaites dans le sens où les erreurs qui affectent ces données sont conformes à la modélisation utilisée. Ainsi, nous disposons d'une trajectoire idéale générée par un ensemble de sous-buts intermédiaires. La trajectoire réelle est obtenue à partir des commandes appliquées sur la position estimée. Ces commandes se résument aux vitesses linéaire et angulaire appliquées au centre du robot (desquelles on déduit les vitesses des roues gauche et droite) nécessaires pour atteindre le sous-but. Les mesures relatives aux capteurs infrarouges sont simulées à partir de la distance en pixel entre la position réelle et la position de la diode, augmentée d'un bruit gaussien. La position odométrique est simulée à partir du modèle cinématique du robot. Nous disposons alors à chaque instant de la position odométrique estimée et de la position réelle. La position initiale du robot est déterminée à partir de la position vraie ajoutée d'un bruit gaussien centré dont la variance σ^2 définit l'erreur initiale.
Les études portant sur l'utilisation du filtrage de Kalman pour la localisation absolue d'un robot mobile sont nombreuses. La méthode que nous avons utilisée ici est la version séquentielle non-linéaire manipulant des distributions de probabilités gaussiennes représentant la méthode la plus usitée [DURIEU 96].
Dans l'approche de Kalman, σ représente l'écart type initial sur x et y permettant de déterminer la matrice de variance-covariance initiale du modèle d'état relatif à l'odométrie. De même, dans l'approche possibiliste, σ représente le paramètre déterminant la distribution de possibilités initiale, représentée sur la figure III-10, avec lequel le modèle odométrique est initialisé. La distribution est centrée sur la valeur d (coordonnée en x ou y) obtenue à partir de la valeur vraie ajoutée d'un bruit gaussien centré et de variance σ^2. Autrement dit la variance initiale σ^2 détermine aussi bien le noyau de la distribution initiale que son support.
L'erreur finale est représentée par la distance entre la position estimée par l'algorithme et la position vraie.

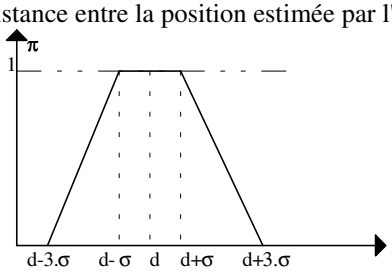


Figure III-10 : Représentation de la distribution associée à la position initiale

Plusieurs scénarios de tests sont possibles suivant qu'on décide de faire une localisation statique (en fin de parcours) ou dynamique (à chaque déplacement élémentaire). Nous étudierons particulièrement l'influence du nombre de balises, des conditions initiales, du nombre d'itérations en comparant le résultat obtenu par la méthode possibiliste et celui obtenu dans les mêmes conditions en utilisant la méthode du filtrage de Kalman.

4-3-1-Tests en fin de parcours

Le test consiste à laisser naviguer le robot jusqu'à son but final en se référant uniquement à son relevé odométrique, puis à appliquer l'algorithme de localisation pour déterminer l'erreur en fin de parcours. C'est à dire que le processus de fusion n'est déclenché qu'une fois que le robot est complètement immobilisé. Par conséquent, si on considère comme instant t=0 celui où le robot s'est arrêté, le modèle d'état du filtre de Kalman associé correspond à un modèle statique (sans aucun mouvement).
L'erreur relative à ce modèle dépend ainsi uniquement de la position initiale X_O, qui correspond ici à la position en fin de parcours, déduite à partir du modèle odométrique, qui, lui, est régi par les équations du modèle d'évolution (§4-2) où l'état initial k=0 représente cette fois-ci l'instant du début de mouvement du robot. Les équations d'observations correspondent à la distance euclidienne entre chaque balise et la position courante du robot.
On notera que cette situation est différente de la fusion statique des données extéroceptives dans le sens où la donnée odométrique se trouve incorporée et permet alors de déterminer une estimation de la position du robot, même si une seule balise est disponible, alors qu'il en faut en moins trois dans le cas de la première situation.

4-3-2-Simulation

La figure III-11 ci-après illustre la différence entre les performances des deux méthodes (la méthode possibiliste est meilleure pour une ordonnée positive). On voit clairement que la méthode possibiliste n'est supérieure au filtre de Kalman que dans le cas où l'erreur initiale est importante et le nombre de balises faible. Ce cas correspond effectivement aux interprétations intuitives des deux formalismes. En effet, le filtre de Kalman (ou d'une façon plus générale une méthode probabiliste) est bien adapté quand l'information est riche (beaucoup

de balises, peu d'incertitude) alors que la théorie des possibilités est bien adaptée pour les situations pauvres (moins de balises, plus d'incertitude).

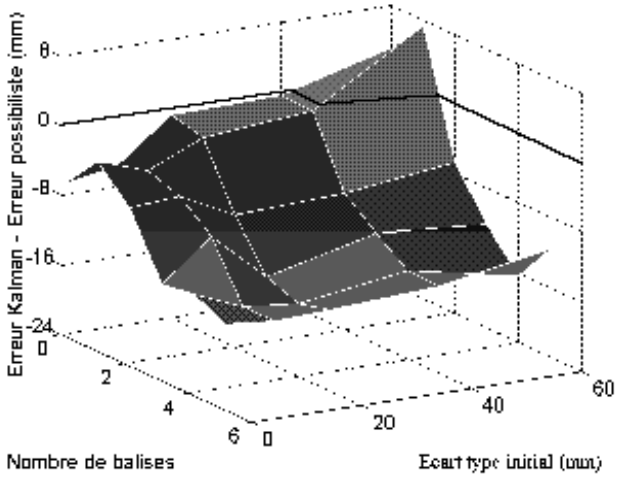


Figure III-11 : Comparaison Kalman/Possibilités en simulation avec correction finale.

Nous pouvons comparer également les variances associées aux deux méthodes. Dans l'approche de Kalman, la variance du résultat est donnée par la diagonale de la matrice de variance-covariance après correction. Dans l'approche possibiliste, nous appliquons la transformation possibilités → probabilités logarithmique de KLIR [KLIR 88] pour déterminer la distribution de probabilités associée à chacune des distributions de possibilités résultantes sur x et y. Ensuite, nous déterminons la variance associée à chacune de ces distributions de probabilités.
On peut ainsi noter que :
- Les variances des deux approches ne sont comparables que dans le cas d'un faible nombre de balises et d'une variance initiale importante, ce qui est conforme aux résultats précédents sur les erreurs de localisation.
- La variance du résultat est toujours inférieure à la variance initiale dans le cas de l'approche de Kalman et diminue avec le nombre de balises utilisées. Ceci est dû au fait que la méthode de Kalman est un estimateur à variance minimale dans le sens où la variance du résultat diminue après chaque itération. En outre, comme les erreurs du modèle d'observation sont fixes et sont donc, le plus souvent, inférieures à la variance initiale σ^2, la variance va également décroître avec le nombre de balises (le résultat le plus précis, i.e., ayant la plus petite variance, se trouve privilégié).
Dans l'approche possibiliste, ce phénomène n'est pas garanti. En effet, ni la règle de fusion, ni la transformation possibilités→probabilités ne font intervenir un quelconque critère de minimisation de la variance. Seule la concordance entre les différentes sources est prise en considération par le résultat final.

4-3-3-Expériences réelles

De la même manière que précédemment, on fait parcourir une trajectoire au robot à partir d'une position initiale. A la fin de son parcours, on détermine sa position en fusionnant sa position odométrique avec les observations des balises infrarouges. On fait également varier le nombre de balises actionnées. On obtient les résultats des figures III-12 et III-13. La figure III-12 montre l'influence de l'erreur initiale sur l'estimation. Les courbes sont la moyenne des résultats obtenus après 50 essais sur une trajectoire rectiligne, les cinq balises étant utilisées en fin de parcours. La figure III-13, quant à elle, montre l'influence du nombre de balises dans le cas d'une erreur initiale importante. Ces résultats confirment globalement les résultats de simulation. Cependant, dans la figure III-12, on observe une diminution des performances du filtre de Kalman dans le cas où l'erreur initiale est très faible. Ceci est dû au fait que, comme σ est faible, il est en de même pour la matrice de variance-covariance, et le filtre de Kalman accorde alors une importance prépondérante au résultat de la prédiction au détriment des observations.

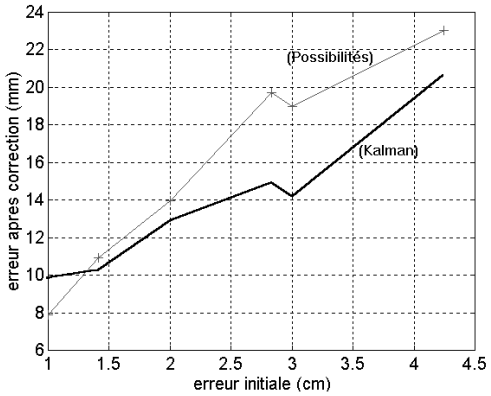


Figure III-12 : Comparaison Kalman/Possibilités en fonction de l'erreur initiale dans le cas où cinq balises sont utilisées.

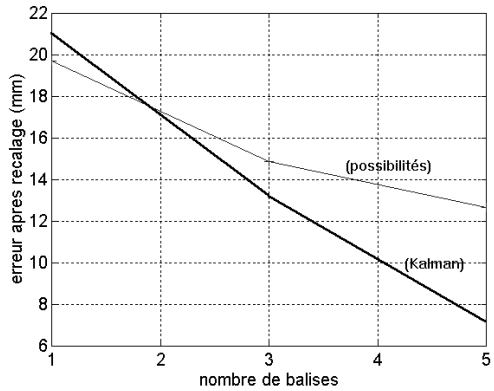


Figure III-13 : Comparaison Kalman/Possibilités en fonction du nombre de balises utilisées pour la correction en fin de parcours pour une variance initiale importante (3600).

Dans le cas des tests en ligne et contrairement au cas précédent, le processus de fusion est déclenché cette fois ci à chaque déplacement élémentaire et on détermine l'écart entre la position ainsi estimée et la position vraie. Il est clair, en simulation, que ces conditions sont très favorables à la méthode de Kalman. En effet, cela vient du fait que cette méthode est un outil fait par excellence pour l'utilisation en ligne. Même avec un faible nombre de balises, comme on répète la mesure et l'acquisition plusieurs fois, l'information s'enrichit de plus en plus, et ainsi on ne peut plus parler de situation pauvre comme dans le cas précédent. Dans le cas d'expériences réelles, la trajectoire est une ligne droite et le nombre de balises déclenchées à chaque instant peut varier de un à cinq. De plus, la correction ne s'effectue pas tout le long de la trajectoire, mais sur une courte portion située dans la zone d'influence des diodes qui a été déterminée par une étape préalable, en testant la réponse des capteurs. Les résultats confirment grossièrement ceux de la simulation.
Pour analyser plus en détail certains résultats obtenus, nous proposons ici une vision comparative des deux méthodes en nous intéressant successivement aux entrées, au mode de fonctionnement et aux sorties des deux systèmes.

4-3-4- Entrées du système

Du point de vue de l'homogénéisation des données à l'entrée du système, les deux formalismes sont très similaires. Dans la méthode possibiliste, sont combinées la donnée odométrique et les solutions locales. Chacune de ces solutions locales résulte de la mise en correspondance de la donnée odométrique avec l'observation relative à cette solution. Les entrées sont les distributions correspondant à l'odométrie et aux données balises qui seront utilisées pour déterminer les paramètres à estimer sous forme de distributions. Pour le filtre de Kalman, les entrées sont l'état prédit du vecteur d'état et l'observation à l'instant k.
On remarque d'après le système d'état que la solution relative à l'observation k est dépendante de la solution relative à l'observation k-1. Cela veut dire que, si on considère chaque solution locale comme un expert, alors l'expert i ne donne son avis sur sa solution qu'après avoir consulté tous les autres experts qui ont déjà donné leur avis. Ceci constitue une différence de taille avec la méthode possibiliste, où les différents avis sont donnés indépendamment les uns des autres.

Il s'agit ainsi de deux visions différentes du problème. Le méthode possibiliste offre plus de liberté et de souplesse, alors que la méthode de Kalman adopte une démarche plus prudente. Cependant, il est facile de voir que, dans le modèle de Kalman utilisé ici, une erreur sur une observation peut affecter toutes les entrées du système, surtout si l'observation erronée est la première observation. La cohérence des observations est donc une condition nécessaire, car, dans le cas contraire, le filtre effectuera une moyenne pondérée avec une donnée éloignée des autres. Par contre, le modèle possibiliste, à cause de cette propriété d'indépendance, n'est guère sensible à ce phénomène. En revanche, si on a des mesures en bonne cohérence, le filtre de Kalman possède un caractère de renforcement des solutions locales qui est un atout. C'est ce qui se passe avec la plupart des expériences réalisées ici avec Khepera.

4-3-5- Fonctionnement du système

Dans le filtre de Kalman, les données sont traitées afin d'obtenir la meilleure estimation de l'état du système minimisant la variance sur l'état estimé. Ainsi, il effectue une correction de la prédiction proportionnelle à l'écart entre la valeur mesurée et la mesure prédite à partir du modèle d'observation en fonction des erreurs d'observation et d'état. Par ailleurs, si la matrice de covariance d'état est faible, la meilleure estimation est fournie par la prédiction. En effet, dans ce cas, la matrice de covariance d'état est moins importante que celle de l'observation. L'observation tend alors à être toujours ignorée par le filtre. De même si l'erreur de mesure est trop faible l'estimation du filtre tend à ignorer le résultat de la prédiction, pour se satisfaire uniquement de l'estimation tirée du vecteur d'observation. En d'autres termes, si la variance d'observation est grande par rapport à celle d'état, le filtre n'accorde que peu de confiance aux observations. Inversement, si la variance d'état est grande par rapport à celle d'observation, le filtre accordera alors plus de confiance à l'observation.
Pour résumer, le filtre agissant comme une moyenne pondérée entre la prédiction et l'observation, une bonne estimation est obtenue dans le cas où les deux conditions suivantes sont satisfaites :

- Elimination des observations aberrantes.
- Cohérence des observations avec le vecteur d'état du point de vue de leurs matrices de variance-covariance associées.

Dans le modèle possibiliste, la combinaison des informations réalisée par le biais de la règle adaptative accorde une importance primordiale à la zone de consensus entre les différentes entrées. C'est à dire que, contrairement au filtre de Kalman, la méthode ne s'intéresse pas à la notion de variance ou de précision des distributions initiales.
Par ailleurs les méthodes basées sur la théorie des probabilités capturent plutôt la notion de variabilité aléatoire et elles sont incapables d'agréger des informations ou des opinions conflictuelles : par exemple dans le cas d'informations provenant de plusieurs capteurs distincts, de natures diverses dont certains fonctionnent mal, la moyenne pondérée est contestable.

4-3-6- Sorties du système

Un dernier élément de comparaison entre l'approche possibiliste et celle de Kalman concerne les sorties des deux méthodes. Dans la méthode de Kalman, les sorties sont l'estimation de l'état du système et la variance de cette estimation (matrice de covariance). D'un autre côté, dans la méthode possibiliste, les sorties sont données sous forme de distributions de possibilités. Autrement dit, on n'a pas directement une valeur numérique. Il s'agit là encore de deux visions différentes. Dans le premier cas, le calcul se limite aux seules données numérique et aux variances associées, alors que, dans le second, c'est un ensemble de valeurs (distribution) qui est traité du début jusqu'à la fin du processus. Si on veut obtenir une valeur numérique, on est amené à choisir une valeur unique à l'intérieur de cet ensemble. En règle générale, la méthode utilisée pour extraire cette valeur est une procédure de défuzzification, par exemple en cherchant le centre de gravité de la distribution. Cependant, ces procédures, bien qu'elles soient généralement efficaces, donnent parfois des résultats difficilement justifiables en particulier si la distribution résultante est multimodale. Dans ce cas on pourrait défuzzifier séparément chaque pic de la distribution. On peut obtenir alors plusieurs valeurs ayant un degré de possibilité égal à 1. Le débat sur le choix d'une valeur plutôt qu'une autre reste encore ouvert. Par exemple, en utilisant une nouvelle information extérieure, on peut confirmer ou infirmer une de ces valeurs. Une perspective intéressante dans ce genre de situation consiste à considérer un système de défuzzification à plusieurs sorties (suivant le nombre de pics dans la distribution multimodale associés des degrés de possibilité correspondants).
Pour obtenir une indication sur la précision de l'estimation possibiliste, on peut considérer par exemple la spécificité ou la non-spécificité de la distribution de possibilités finale, ou encore effectuer une transformation possibilités/probabilités et chercher la variance associée à la distribution de probabilité résultante.

4-3-7 - Autres remarques

Du point de vue de l'implémentation algorithmique, la méthode de Kalman est aisée, car elle se réduit à de simples opérations d'arithmétique matricielle. La méthode possibiliste, telle que nous l'avons mise en œuvre, est beaucoup plus lourde. Cette complexité vient principalement de l'application du principe d'extension, même

en utilisant la méthode de Dong. Afin d'alléger la méthode possibiliste, on pourrait par exemple utiliser une représentation paramétrique des distributions de possibilités et tolérer certaines simplifications du point de vue de la modélisation des données.
Un autre problème que pose la méthode possibiliste utilisée est la gestion de variables liées, car l'extension de la méthode de Dong à ce cas n'est pas aisée [PANNEREC 97].
La méthode de Kalman correspond à une approche asymétrique (conditionnement séquentiel) puisqu'elle utilise des probabilités conditionnelles. La méthode possibiliste est basée, elle sur une approche symétrique où les solutions locales sont combinées de façon parallèle.
Enfin, des expériences sont en cours pour évaluer les performances des deux méthodes en fonction de la modélisation des erreurs.

5-Modification de la règle CADP

5-1- Robustesse de la règle CADP

La règle CADP pose dans certains cas des problèmes de robustesse vis à vis de la forme des distributions à combiner, qui peuvent engendrer un dysfonctionnement du point de vue de la valeur défuzzifiée de la distribution résultante. En effet, considérons les cas académiques illustrés sur la figure III-14 où nous avons tracé pour les quatre cas le résultat de la combinaison par la règle CADP. Les deux premiers concernent la combinaison de distributions de forme trapézoïdales et les deux derniers des distributions de formes triangulaires.

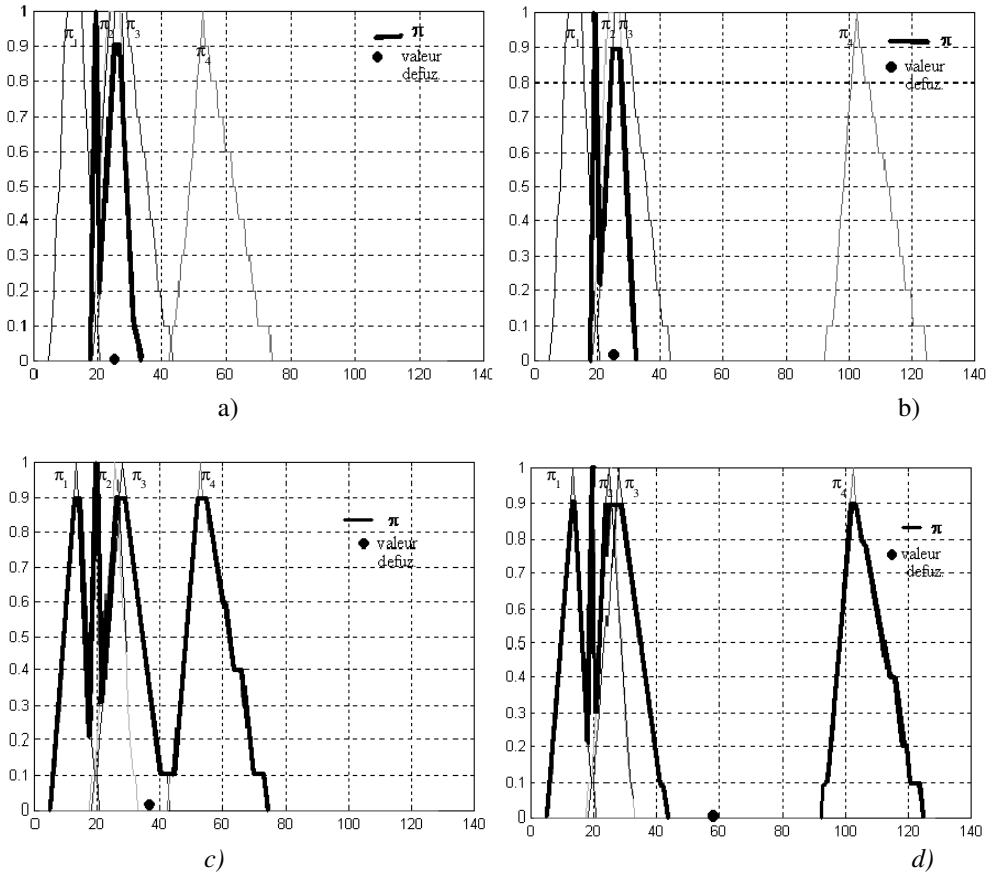


Figure III-14. Exemples particuliers de combinaison par la règle CADP.
a) distributions trapézoïdales et π_4 très proche des autres distributions
b) distributions trapézoïdales et π_4 éloignée
c) distributions triangulaires π_4 très proche des autres distributions
d) distributions triangulaires et π_4 éloignée

On peut remarquer les faits suivants :
- Dans les figures a) et b), où les distributions π_1, π_2 et π_3 sont trapézoïdales, la valeur de m et n sont respectivement 2 et 3. La dernière distribution π_4, n'est pas prise en compte par le processus de fusion, quelle que

soit sa distance aux autres distributions. La valeur défuzzifiée de la distribution reste alors inchangée quelle que soit la position de π_4.
- Dans les figures c) et d), où les distributions sont de forme triangulaire, les valeurs de m et n deviennent respectivement 1 et 3. Cette fois la dernière distribution est toujours prise en compte par la règle de combinaison. La valeur défuzzifiée se trouve alors fortement influencée par la dernière distribution et elle peut tomber dans une zone où le degré de possibilité est nul pour toutes les distributions (fig. III-14d).
Ce manque de robustesse de la règle de combinaison par rapport à une faible modification de la forme des distributions d'une part, et, d'autre part, la non prise en compte de certaines données par rapport à une zone de consensus peut être très gênante pour trois raisons essentielles.
- Des données aberrantes, dues à un mauvaise mesure, ou à un mauvais appariement, ou tout simplement à un quelconque dysfonctionnement d'un des capteurs, peuvent engendrer une distribution de possibilités éloignée du corps des autres distributions. Il est alors souhaitable de pouvoir éliminer cette donnée pour ne pas dégrader le résultat de la fusion.
- Cependant il est souhaitable que ce rejet soit progressif en fonction de l'éloignement par rapport à la zone de consensus afin de ne pas éliminer systématiquement une distribution en faible désaccord avec les autres.
- La détermination précise de la distribution de possibilités est généralement une tâche difficile et, souvent, on utilise une forme approximative. Il est alors important que l'opération de fusion soit robuste vis à vis de variations raisonnables de la forme des distributions. Ce n'est pas le cas ici puisque le passage d'une modélisation trapézoïdale à une modélisation triangulaire modifie considérablement le résultat de la combinaison.
Afin de prendre en compte les trois points précédents, tout en restant dans l'esprit de la règle CADP, la solution que nous proposons consiste à modifier le paramètre m de la règle. En effet, ce paramètre régit l'étendue du support de la distribution résultante. Autrement dit, la décision de tenir compte ou non de la dernière distribution lui revient. Le paramètre n, quant à lui, intervient au niveau du choix de la zone de concordance. Il détermine ainsi l'emplacement du pic de la distribution résultante.
Ainsi si m=1, le support de la distribution finale s'étend au support de toutes les distributions présentes (y compris celle qui est la plus éloignée). Si m=n, le support se réduit à la zone de concordance du plus grand nombre de sources. Par conséquent, si on attribue au paramètre m, dans le cas des figures a) et b), la valeur 1 (c'est à dire qu'on diminue d'une unité la valeur initiale de m) le résultat de la combinaison sera quasi similaire à celui obtenu sur les figures c) et d). Inversement, si on augmente la valeur de m d'une unité, dans le cas des figures c) et d), on retombe, à peu de chose près, sur les cas a) et b). C'est pourquoi nous proposons d'élargir le choix de ce paramètre.

5-2- Modification du choix du paramètre m

Cette alternative consiste, tout en restant dans l'optique de la règle de DUBOIS et PRADE, à remettre en cause la valeurs de m. Cela veut dire que l'on tolère l'existence d'une erreur sur la valeur de m. On admet par exemple que la valeur de m peut être en fait m_0, m_0+1 ou m_0-1. La règle sera alors réécrite en remplaçant $\pi_{(m)}$ par $\pi'_{(m)}$, où le second membre tient compte du triplet (m_0, m_0+1 ou m_0-1) :

$$\pi(x) = \sup\left(\frac{\pi_{(n)}(x)}{h(n)}, \min(\pi'_{(m)}(x), 1 - h(n))\right)$$

Le problème est maintenant de savoir comment tenir compte de ces trois valeurs dans les distributions quantifiées correspondantes.

5-3- Obtention d'une règle progressive (RP)

L'idée est de substituer à $\pi_{(m)}$ la distribution $\pi'_{(m)}$ telle que :

$$\pi'_{(m)} = \alpha_1 . \pi_{(m-1)} + \alpha_2 . \pi_{(m)} + \alpha_3 . \pi_{(m+1)}$$

où les α_i sont tels que $\sum_{i=1}^{3} \alpha_i = 1$ et $\alpha_i \geq 0$.

Cette équation peut être sous normalisée par rapport à $\sup_x(\pi_{(m-1)}(x), \pi_{(m)}(x), \pi_{(m+1)}(x))$. Cette situation n'est cependant pas gênante car, d'une part, les trois distributions $\pi_{(m-1)}$, $\pi_{(m)}$ et $\pi_{(m+1)}$ peuvent elles-mêmes être sous-normalisées, et, d'autre part, la normalisation de la distribution finale est assurée par le terme $\frac{\pi'_{(n)}}{\sup_x \pi'_{(n)}(x)}$.

La réalisation d'un effet de progressivité implique l'idée que les paramètres α_1, α_2 et α_3 soient fonction de la distance par rapport à la zone de consensus (au moins deux sources en accord).
Nous proposons alors la démarche suivante :

Soit d_0 une distance au-delà de laquelle la réjection totale d'une distribution est souhaitée ; c'est à dire que la donnée correspondante est considérée comme erronée si la distance de sa distribution à la zone de consensus est supérieure à d_0.
Cette distance est comptée à partir des extrémités du support de la zone de concordance. Celle-ci n'est définie que si les deux conditions suivantes sont satisfaites :
- n est strictement supérieur à 1, sinon, cela signifie qu'il n'y a pas de zone de consensus.
- n est strictement supérieur à m, parce que, dans le cas contraire, cela signifie que $\pi_{(m+1)}=0$.
Par ailleurs, si m=1, la distribution $\pi_{(m-1)}$ est toujours nulle et le terme α_1 ne sera pas considéré.
La distance entre chaque point de l'univers et cette zone de consensus peut être exprimée en considérant les extrémités du support de la zone de consensus définie par $\pi_{(n)}$. De façon plus formelle, soient s_o et s_1 tels que:

$[s_o \; s_1]=(\pi_n)_{\bar{0}}$ (strict ensemble alpha coupe de niveau 0 correspondant au support de $\pi_{(n)}$) ;

la distance d(s) du point s sur l'univers de discours à cette zone peut alors être formulée comme :

$d(s)=\min(|s-s_o|, |s-s_1|)$ avec $\alpha_2 = \min(\frac{d}{d_o},1)$ et $\alpha_1 = \alpha_2 = \frac{1}{2}(1-\alpha_3)$.

Nous montrons dans les figures III-15 (a, b, c et d) un exemple de combinaison par cette méthode. La distance seuil d_o choisie est égale à 160. Le caractère progressif de la prise en compte de la dernière distribution est clairement montré. Contrairement à la règle CADP, on remarque que cette règle de combinaison est parfaitement robuste vis à vis de la forme de la distribution (passage de trapézoïdal à triangulaire).

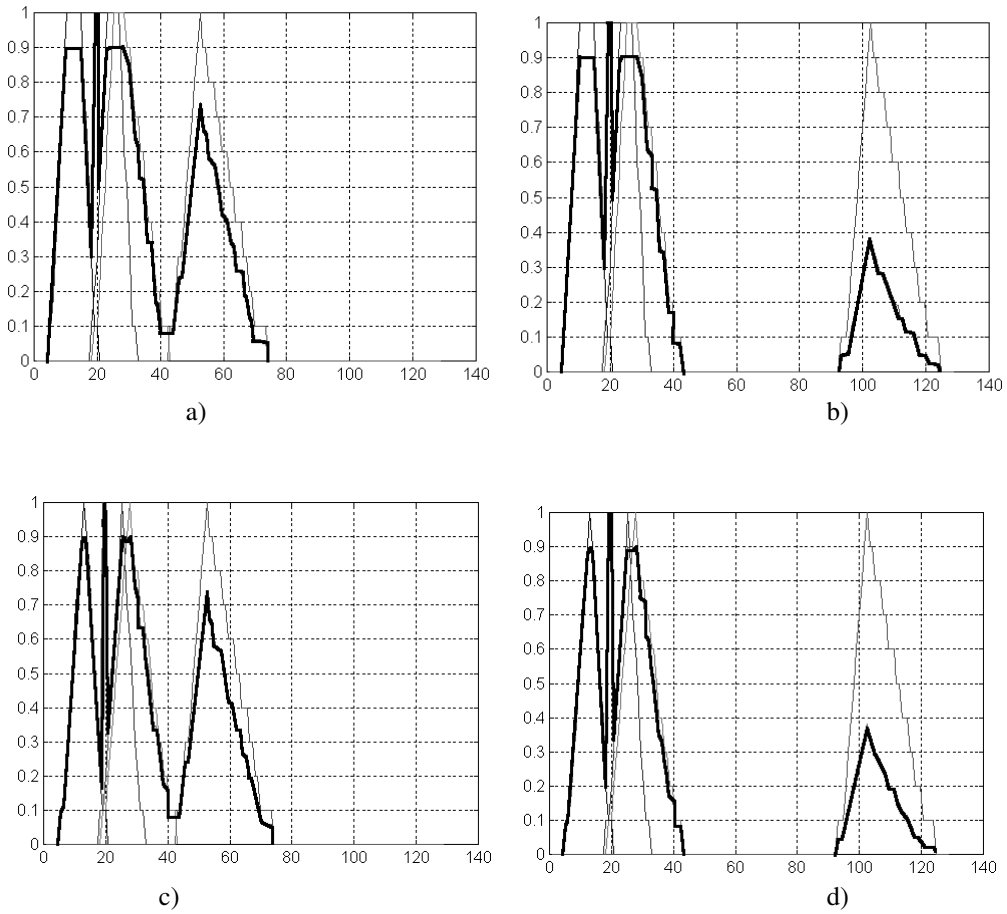


Figure III-15 Combinaison avec la règle progressive RP
a) b) Cas de distributions trapézoïdales (d_0=160)
c) d) Cas de distributions triangulaires (d_0=160)

Un autre choix de la distance d_0 conduirait évidemment à un autre résultat du point de vue de la distribution résultante, et donc de la prise en compte ou non de la dernière distribution.
Par exemple pour d_0=100 et d_0=25 le résultat de la combinaison est visualisé sur la figure III-16.

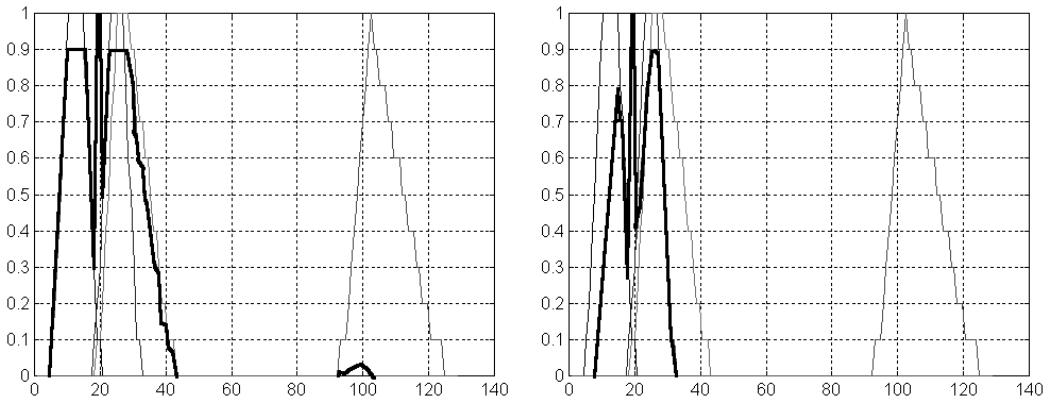


Figure III-16 : Combinaison avec la règle progressive RP pour :
a) d_0=100
b) d_0=25

Pour illustrer le fonctionnement de cette règle de façon plus générale, nous avons fait glisser la dernière distribution le long de l'axe des x en s'éloignant vers la droite (fig; III-17a).
Dans la figure III-17b, nous avons comparé le résultat de la défuzzification avec les deux règles (x=0 correspond à la position des courbes sur la figure III-17a).

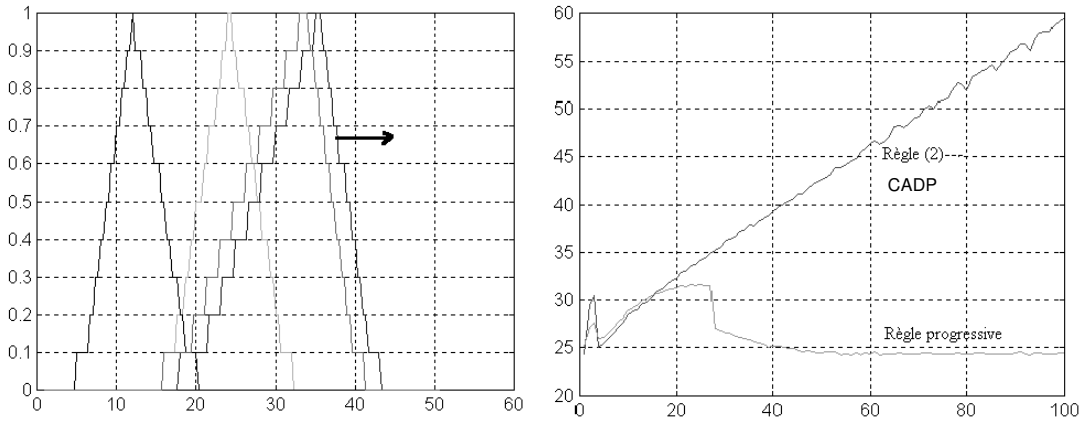


Figure III-17.
a) Prototype de test. Eloignement progressif de la dernière distribution par rapport à la zone de concordance
b) Résultat de la défuzzification de la distribution résultante à chaque pas de déplacement et comparaison avec le résultat de la règle CADP de DUBOIS et PRADE

On peut remarquer que :
- Pour des distributions de possibilités de forme triangulaire, la dérive obtenue avec la règle initiale en fonction du désaccord d'une des distributions est maintenant éliminée. On observe une légère prise en compte de cette distribution avant le seuil d_0, puis son influence disparaît. Le décrochement un peu brutal observé sur la figure III-17b est dû à l'emploi de la fonction Min comme fonction de saturation. Une autre fonction pourrait adoucir la transition.
- Pour des distributions de possibilités de forme trapézoïdale, le problème est très différent, puisque la règle CADP élimine la distribution en désaccord. On constate que la règle progressive donne des résultats assez voisins, avec une légère évolution due essentiellement au déplacement du pic dans la zone de consensus et au choix de la valeur du seuil d_0.
- Par ailleurs, les valeurs défuzzifiées sont maintenant très peu sensibles à la forme des distributions de possibilités.

6- Application de la règle progressive dans une expérience de localisation

L'utilisation de capteurs ultrasonores est assez fréquente en robotique mobile principalement à cause de leur faible coût. Cependant, ce gain se paye par la pauvreté de l'information fournie. En effet, la réponse du capteur est la distance entre le robot et le point d'impact le plus proche du faisceau US. Mais ce faisceau est divergent avec un angle de plusieurs dizaines de degrés. On ne peut donc pas connaître la direction réelle de l'impact à l'intérieur de ce cône. Par ailleurs, dans un certain nombre de cas, le faisceau subit plusieurs impacts avant son retour, entraînant une erreur de mesure considérable. Il est donc intéressant, dans ce contexte d'informations pauvres et éventuellement conflictuelles de tester l'intérêt d'une méthode basée sur la fusion dans le cadre de la théorie des possibilités. Après avoir décrit le cadre expérimental, ainsi que le schéma général de l'expérience, nous expliquons les grandes lignes de l'approche adoptée.

6-1- Cadre expérimental

Nous considérons qu'on connaît une carte de l'environnement précise, mais peu sûre (obstacles inconnus). C'est à dire que nous supposons que les murs délimitant l'environnement, ainsi que la position initiale du robot, sont connus avec précision dans un repère absolu, mais qu'un obstacle non modélisé peut être rencontré. La tâche du robot consiste à passer par quelques points intermédiaires fixés dans l'environnement, de le traverser et de finir par passer une porte.

- Nous utiliserons le robot RMI dont nous admettons que les capteurs sont positionnés sur la circonférence formant la plate-forme du robot.
- Le robot navigue sans glissement sur des distances relativement courtes. L'odomètre fournit des données de bonne précision qui seront considérées par la suite comme "parfaite" et fournissant donc la position exacte du robot. Cette position sera appelée par la suite "position vraie".
- Pour tester l'algorithme de fusion utilisé, nous procédons tout d'abord à une dégradation volontaire (par logiciel) des données odométriques reçues : nous supposons que le diamètre d'une roue est modifiée de λ % (par exemple à la suite d'un dégonflement). La position issue de cette information dégradée sera appelée par la suite "position odométrique".
- Le dispositif de fusion reçoit les deux types d'informations suivantes :
  * les mesures des capteurs ultrasonores,
  * la donnée odométrique dégradée (position absolue en x, y et θ).
- La sortie du système de fusion fournit une position estimée (par rapport au même repère absolu) du robot en x, y et θ sous forme de distributions de possibilités dont on peut éventuellement extraire par défuzzification des valeurs réelles.

6-2- Principe général de la solution adoptée

L'idée générale est d'opérer en deux temps. Tout d'abord, on va considérer l'information fournie par chacun des N capteurs ultrasonores (N étant le nombre de capteurs renvoyant une donnée non erronée) pour obtenir N estimations possibles de la position du robot. Cette phase, qui conduit à ce que nous appelons les "solutions locales", utilise la position odométrique et chaque donnée capteur qui doivent être ramenées dans le même repère (procédure dite d'alignement et de mise en correspondance). Puis, les N solutions locales sont combinées pour obtenir les distributions de possibilités qui définissent ce que nous appelons la "solution globale" (procédure dite de combinaison).

Le principe général est indiqué sur la figure III-18. x, y et θ sont l'abscisse, l'ordonnée et l'orientation du robot dans un repère absolu π_x, π_y et π_θ sont les distributions de possibilités associées à ces trois grandeurs.
Nous présentons successivement dans ce qui suit :

- la modélisation des données capteurs par des distributions de possibilités.
- le processus d'alignement des données dans le même repère temporel et spatial,
- l'obtention des solutions locales,
- l'obtention de la solution globale.

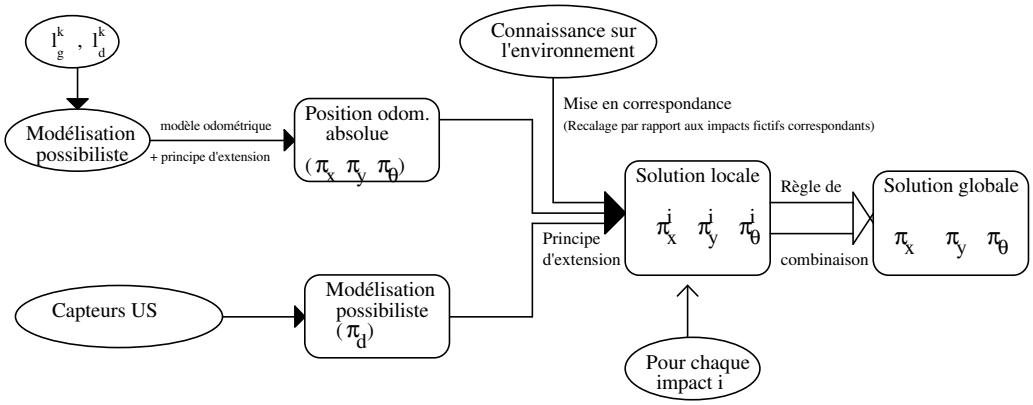


Figure III-18 : Algorithme général

6-2-1- Modélisation des données capteurs

Dans un premier temps les observations anormales sont éliminées. Une observation, représentée par la distance renvoyée par le capteur ultrasonore, n'est considéré que si cette distance est inférieure à un certain seuil. Ce seuil est approximativement égal à la distance entre les deux coins extrêmes de la scène, ce qui correspond à la plus grande distance que doit renvoyer un capteur dans un tel environnement. Les distances qui lui sont supérieures sont dues, soit à un fonctionnement anormal du capteur, soit à des réflexions multiples.
La modélisation des données capteurs (ultrasonores et odométrique) concerne leur représentation sous forme de distribution de possibilités. Dans cette application, nous ne cherchons pas à estimer de façon précise les paramètres de ces distributions en faisant une campagne de mesures spécifique. Par contre, nous étudierons la robustesse du mécanisme de fusion par rapport à leur forme.

6-2-1-1-Capteur ultrasonore

Soient ρ^i_m et α^i_m respectivement la donnée représentant la mesure de distance fournie par le capteur i et l'angle du faisceau central de ce capteur par rapport à l'axe du robot. Nous supposons que la position de chaque capteur par rapport à un repère lié au robot est connue avec précision. L'imprécision entachant α^i_m provient alors uniquement du cône d'émission du faisceau ultrasonore. Nous utilisons comme distributions de possibilités π_ρ et π_α des trapèzes centrés respectivement sur ρ^i_m et α^i_m. Le support correspond à la zone dans laquelle on est certain que la valeur vraie se trouve. Le noyau correspond à l'imprécision extrinsèque entachant la mesure. En se référant à l'expertise courante de ce type de capteurs, les valeurs numériques choisies sont représentées sur la figure III-19.

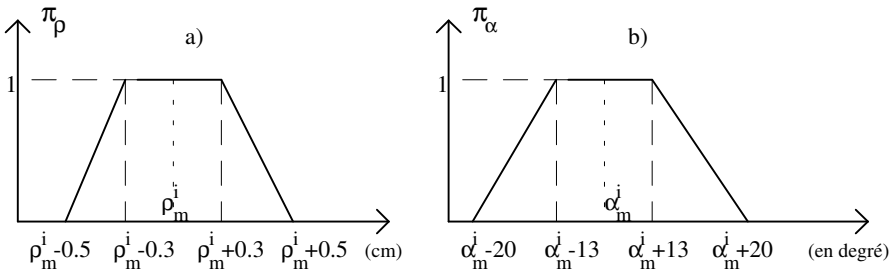


Figure III-19 : Modélisation possibiliste des données ultrasonores. (a: en distance, b: en orientation)

6-2-1-2- Capteur odométrique

Pour les données odométriques, il s'agit de fournir au dispositif de fusion une distribution de possibilités qui intègre le fait que la donnée peut être dégradée sans que cette dégradation soit connue à priori. Nous utilisons -de façon assez arbitraire- une distribution trapézoïdale qui est donnée sur la figure III-20. On note que les dimensions de la distribution (support et noyau) sont exprimées en fonction du déplacement relatif pour tenir compte du fait que la dégradation croit avec la distance parcourue par le robot. Le choix d'une variation linéaire est une hypothèse justifiée ici par la faible valeur de la distance parcourue entre deux instants consécutifs. De la même manière que dans le cas du capteur ultrasonore, les paramètres de la distribution reflètent de façon grossière et très approximative notre état de connaissance sur la donnée d'imprécision qu'on peut associer à la mesure odométrique. On détermine ainsi à partir des déplacements élémentaires l^g_k, l^d_k respectivement des roues gauche et droite les distributions de possibilités correspondantes (repère relatif lié au robot). Ensuite, en utilisant

le modèle cinématique du robot, on détermine la position absolue exprimée sous forme de distributions de possibilités sur x, y et θ obtenues par application du principe d'extension de Zadeh (méthode de Dong).

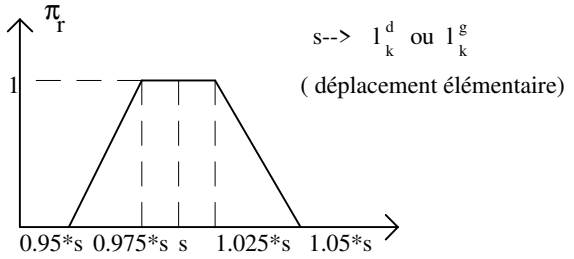


Figure III-20 : Modélisation odométrique

6-2-2- Alignement des données dans le même repère

Il faut tout d'abord exprimer les données odométriques et les mesures US dans le même repère. A un instant k, la position odométrique $(x_k\ y_k\ \theta_k)^T$ est définie par rapport à un repère absolu. Les murs sont également définis avec précision par rapport à ce même repère.
L'impact C_i du capteur ultrasonore n°i (S_i) dans ce repère est donné par :

$$\begin{cases} x_{C_i} = x_k + (r + \rho_m^i).\cos(\theta_k + \alpha_m^i) \\ y_{C_i} = y_k + (r + \rho_m^i).\sin(\theta_k + \alpha_m^i) \end{cases}$$

où r est le rayon du robot.
Connaissant les distributions π_{x_k}, π_{y_k}, π_{θ_k}, $\pi_{\rho_m^i}$ et $\pi_{\alpha_m^i}$, on peut, en appliquant le principe d'extension de Zadeh à l'aide de la méthode de Dong, obtenir les distributions de possibilités correspondant aux coordonnées x-y du point d'impact C_i (x_{C_i} et y_{C_i}).

6-2-3-Obtention des solutions locales

L'association des impacts avec l'environnement est effectuée en considérant les murs connus de la scène. Pour illustrer cette démarche, considérons le schéma de la figure III-21.

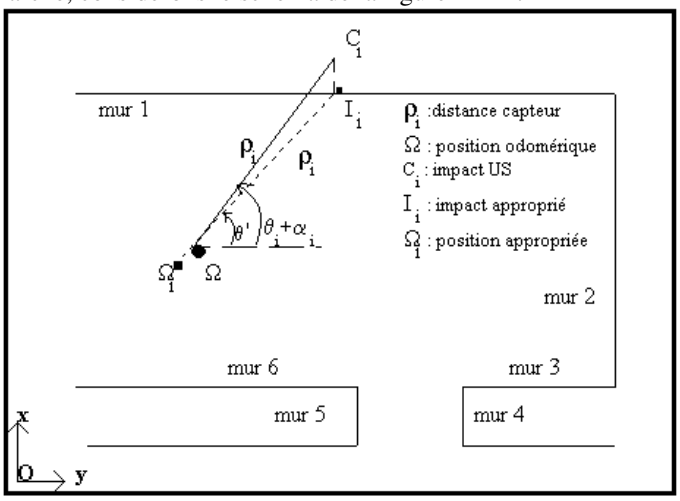


Figure III-21 : Obtention d'une solution locale

Soit C_i le point d'impact obtenu à partir de la procédure précédente. Ce point devrait théoriquement se trouver sur un des murs de la pièce. Ce ne sera en général pas le cas à cause de l'imprécision des mesures. Nous cherchons le mur le plus proche du point C_i par projection orthogonale et nous considérons que ce point (I_i sur la figure III-21) est la meilleure approximation du véritable point d'impact. Une procédure d'élimination des impacts aberrants est en outre utilisée. Nous définissons pour cela un seuil δ (qui est égal pour notre cas à une quinzaine de centimètres) tel que, si la distance C_iI_i est supérieure à δ, l'impact n'est pas pris en compte, car on considère dans ce cas qu'il provient soit d'un obstacle non modélisé, soit d'une réflexion multiple.
La solution locale donnant la position Ω_i du robot associée au capteur i est alors obtenue en reportant la distance ρ_i sur la direction $I_i\ \Omega$. Les coordonnées de Ω_i sont ainsi données sous forme de distributions de possibilités

obtenues en appliquant le principe d'extension. En effet, en considérant par exemple le cas de la figure III-21, les coordonnées de Ω_i par rapport au repère Oxy sont données par :

$$(I_i)_x = (C_i)_x = \Omega_x + (r+\rho_i).\cos(\theta_i+\alpha_i)\ ;\ (I_i)_y = a\ ;\ (\Omega_i)_x = (I_i)_x - \rho_i.\cos(\theta')\ ;\ (\Omega_i)_y = (I_i)_y - \rho_i.\sin(\theta')\ ;$$

$$\text{avec : } \cos(\theta') = \frac{(I_i)_x - \Omega_x}{\sqrt{((I_i)_x - \Omega_x)^2 + ((I_i)_y - \Omega_y)^2}}\ ;\ \sin(\theta') = \frac{(I_i)_y - \Omega_y}{\sqrt{((I_i)_x - \Omega_x)^2 + ((I_i)_y - \Omega_y)^2}}\ ;$$

où a désigne l'ordonnée du mur 1 par rapport au repère Oxy, r le rayon de la plate-forme, θ_i l'orientation du robot par rapport à l'axe horizontal Ox du repère absolu et α_i la position angulaire du point d'impact C_i par rapport à un repère lié au robot.

Etant donné que les estimations $\rho_i, \theta_i, \alpha_i, \Omega_x$ et Ω_y sont données sous forme de distributions de possibilités, il en est de même pour $(I_i)_x, (\Omega_i)_x$ et $(\Omega_i)_y$. Ceci signifie que, par exemple, le point I_i indiqué sur la figure III-21 est un représentant d'un ensemble de points ayant des degrés de possibilités décrits par la distribution associée à $(I_i)_x$. Les points correspondant au noyau de la distribution sont complètement possibles, puis au fur et à mesure que l'on s'éloigne de cette zone, la possibilité de l'occurrence des points correspondants diminue.

Afin de tenir compte de l'angle d'ouverture du faisceau et du fait que l'erreur odométrique accumulative sur la composante θ induit des erreurs sur les coordonnées de l'impact C_i qui sont d'autant plus importantes que la distance ρ_i est grande, nous procédons de la manière suivante :

- D'abord, nous cherchons la plus petite distance ρ renvoyée par les capteurs ultrasonores. Supposons arbitrairement qu'il s'agit de capteur n°1, c'est à dire la distance ρ_1. Nous appliquons alors la procédure précédente pour déterminer successivement C_1, I_1 et la solution locale Ω_1 correspondant à la position estimée du robot telle qu'elle est vue uniquement par ce capteur.
- Pour les autres distances ρ_i qui sont plus grandes que ρ_1, nous faisons pivoter le segment S_iC_i de $\pm\theta_{ouv}$ avec un pas de discrétisation de 1°, c'est à dire que nous prenons plusieurs points dans le cône d'émission du capteur. Nous obtenons alors pour chacun de ces points un point d'impact C_i^k (k correspond au point pris sur le cône d'émission) et son projeté I_i^k ainsi que la solution locale Ω_i^k exprimés également sous forme de distributions de possibilités.
- Parmi ces points (C_i^k), nous conservons celui dont la solution locale associé (Ω_i^k) est la plus proche du point Ω_1[1]. L'impact $I_i^{k'}$ correspondant au point obtenu $C_i^{k'}$ peut ainsi être différent du projeté I_i de C_i.

En final on dispose des évaluations des coordonnées des points Ω_i conservés, chacune sous forme d'une distribution de possibilités.

6-2-4-Solution globale

Dans cette étape finale, toutes les estimations précédentes comprenant les différentes positions Ω_i résultant de la phase d'alignement et d'association (c'est à dire les solutions locales) sont combinées entre elles. Pour extraire une valeur numérique représentative de la distribution considérée, il suffit d'appliquer une des méthodes de défuzzification. De plus, la procédure de fusion choisie permet également d'effectuer une correction sur l'angle d'orientation θ. En effet, le fait de modifier la position du point d'impact par la procédure citée revient aussi à effectuer une correction de l'angle θ d'une valeur égale à $C\hat{\Omega}I$. La méthode de défuzzification utilisée est celle du centre de l'aire.

6-3-Résultats

Nous montrons sur la figure III-22, l'itinéraire réel suivi par le robot. Deux types d'impacts sont également visualisés. Les premiers, désignés par des rectangles pleins représentent ceux obtenus à partir de la position réelle, les autres désignés par "+" représentent les impacts obtenus à partir de la position odométrique, c'est à dire ceux qui sont à la disposition de l'utilisateur. Nous avons également représenté par des positions "*" certaines positions particulières où l'application de l'algorithme, même avec la phase de mise en correspondance conduit à une situation de conflit qui est due, soit à un mauvais appariement, soit à la présence de l'obstacle non modélisé qui apparaît en bas à gauche de la pièce.

[1] Pour obtenir les distances $\Omega_1\Omega_i^k$, nous procédons à la défuzzification des coordonnées de Ω_1 et Ω_i^k données sous forme de distributions de possibilités.

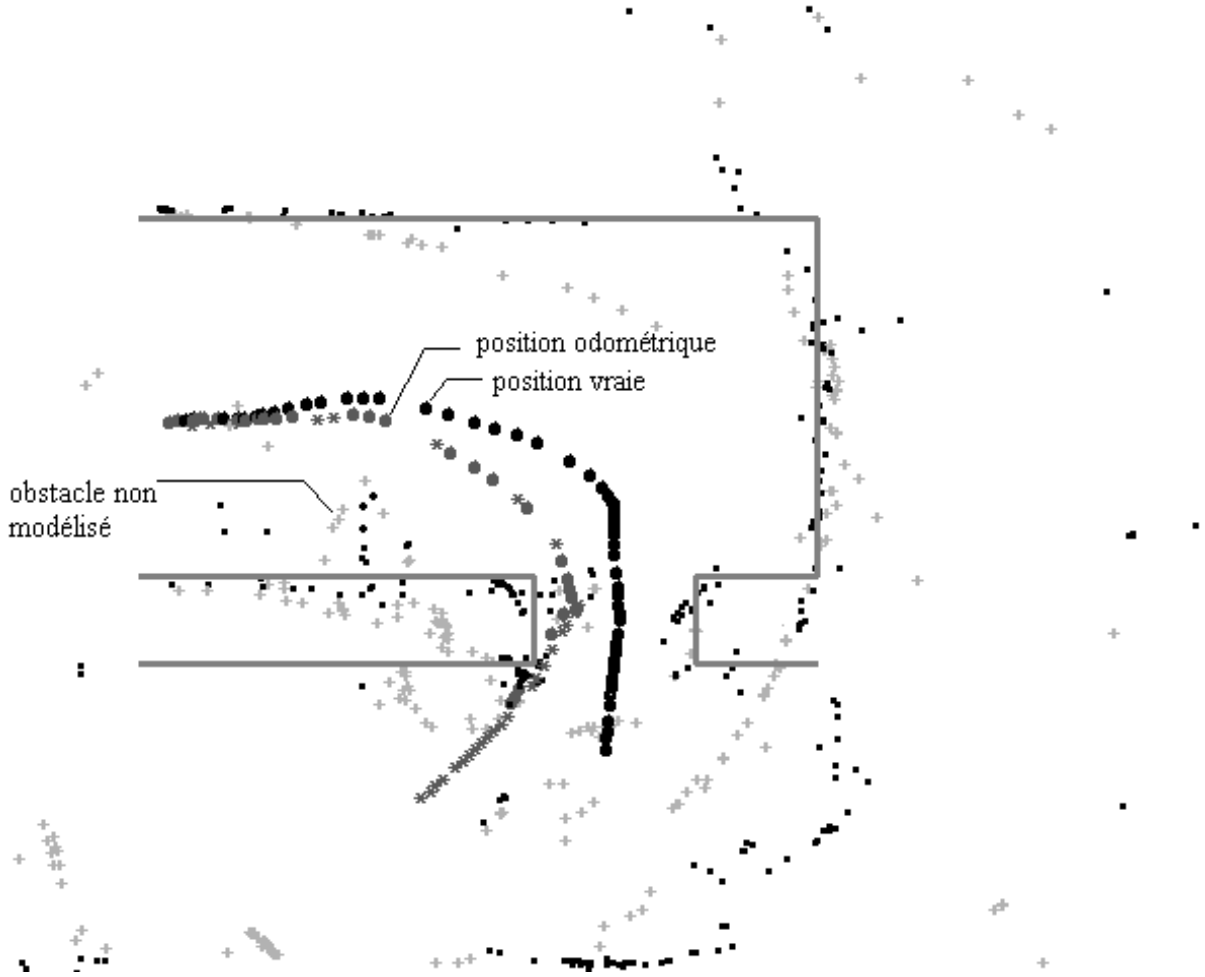


Figure III-22 : Expérience avec RMI

* : points de la trajectoire où l'application de l'algorithme conduit à un mauvais appariement

■ : points d'impacts ultrasonore à partir de la position odométrique vraie (sans dégradation)

\+ : points d'impacts à partir de la position odométrique

La règle RP fournie des résultats satisfaisants dans une situation (par exemple après un faible déplacement du robot à partir de sa position initiale) non conflictuelle dans le sens où les distributions engendrées après l'étape d'alignement et d'association sont en accord.

Dans le cas (fig. III-23) représentant un état de conflit à cause de la présence d'un mauvais appariement (impact C_1) dû à une réflexion multiple non décelée par l'opération de seuillage, nous visualisons les distributions résultant de l'application de l'algorithme de fusion. Nous montrons sur la figure III-24 (pour x et y), les distributions obtenues après l'étape d'alignement et d'association, ainsi que les distributions résultantes par l'application de RP.

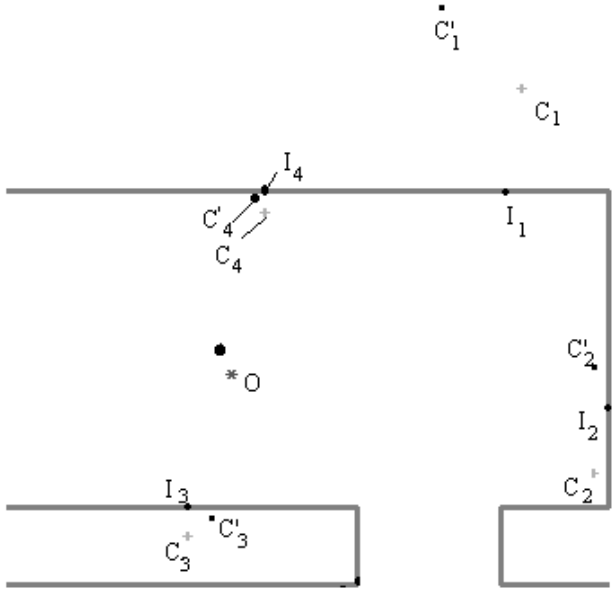
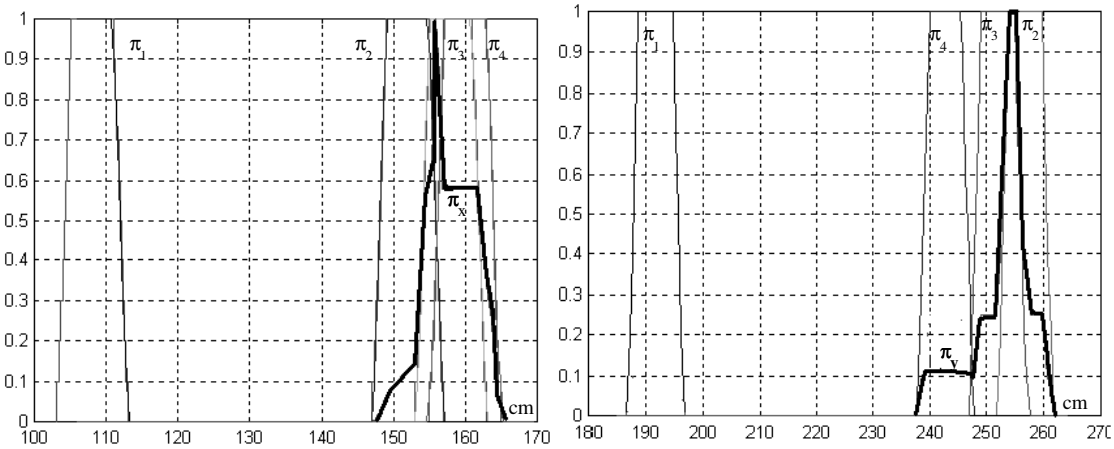


Figure III-23. Identification d'un cas de mauvais appariement.



a) b)

Figure III-24 : Résultat de la combinaison par la règle pour :
a) la coordonnée x et b) la coordonnée y

Nous constatons les points suivants :
- La distribution résultant de l'impact C_1 n'est pas prise en compte par l'opération de fusion à cause de la distance seuil d_0 choisie (d_0 est égale à l'étendue de la zone de concordance).
- La forme de la distribution résultante est directement liée à la distance seuil choisie. Ainsi, on peut, d'une part, jouer sur le niveau au-delà duquel on souhaite la prise en compte ou non de la distribution en désaccord et, d'autre part, régler l'importance de la zone de concordance ou de consensus en terme de surface engendrée.

Nous nous sommes intéressés aussi à la localisation du robot en chaque point de la trajectoire. C'est à dire que le processus de fusion décrit précédemment est appliqué à chaque nouvelle acquisition de données odométriques. On a pu constater que l'estimation de la position du robot est satisfaisante pratiquement tout au long de la trajectoire et que la présence d'un obstacle non modélisé ne perturbe pas cette estimation parce que les situations conflictuelles sont gérées de façon convenable par la règle progressive. Un défaut de correction se manifeste en fin de trajectoire et s'explique très bien car d'une part, l'erreur odométrique y est maximale et, d'autre part, le robot sortant de la pièce, il devient de moins en moins possible d'apparier des échos ultrasonores avec les murs.

6-4- Influence de la modélisation des capteurs

Les distributions de possibilités que nous avons utilisées pour représenter les données capteurs sont très empiriques et on doit se poser la question : que se passe t-il si la modélisation des capteurs est modifiée ?

Autrement dit : la forme de la distribution joue t-elle un rôle dans le processus de fusion et le résultat est il robuste par rapport à cette forme? Nous avons vu dans le §5 que la règle progressive se comporte a priori beaucoup mieux que la règle initiale de DUBOIS et PRADE. Nous allons donc comparer les résultats obtenus avec ces deux règles.
Nous distinguons dans la suite l'influence de la forme de la distribution, à étendue donnée et celle de l'étendue pour une forme donnée.

6-4-1- Influence de la forme de la distribution

Nous avons repris l'exemple de la situation conflictuelle représenté sur la figure III-23. La figure III-25 montre le résultat de la combinaison par les deux règles pour la composante y en donnant aux distributions de possibilités une forme triangulaire.

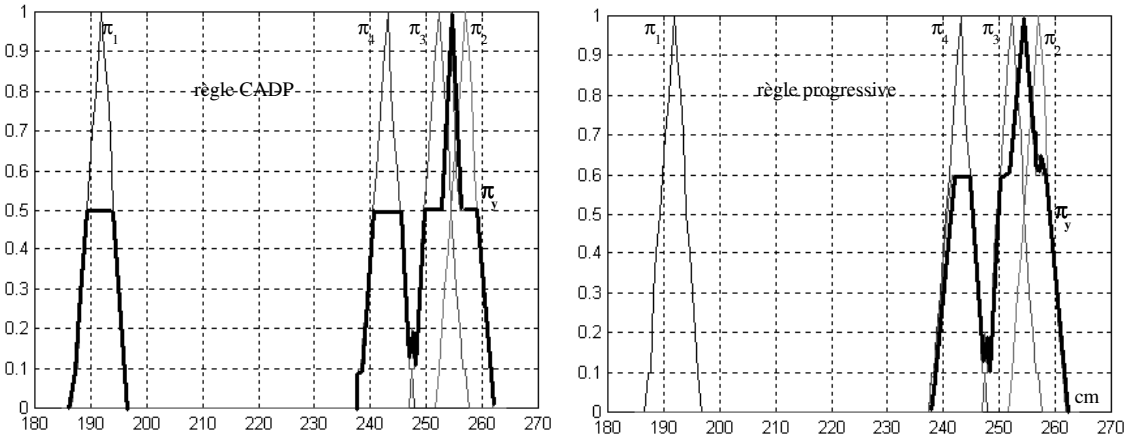


Figure III-25 : Résultat de la combinaison sur y dans le même cas que la figure III-23 avec des distributions triangulaires.

Si le résultat est assez voisin avec des distributions trapézoïdales, la différence est très marquée avec des distributions triangulaires. Dans le cas de la règle CADP, comme maintenant m=1, la distribution en désaccord π_1 est prise en compte au degré 1-h. La figure III-26 montre les erreurs après correction en x, y obtenues par application des deux règles de fusion dans le cas où m=1. On constate la supériorité de la règle progressive, particulièrement dans les cas de conflit important.

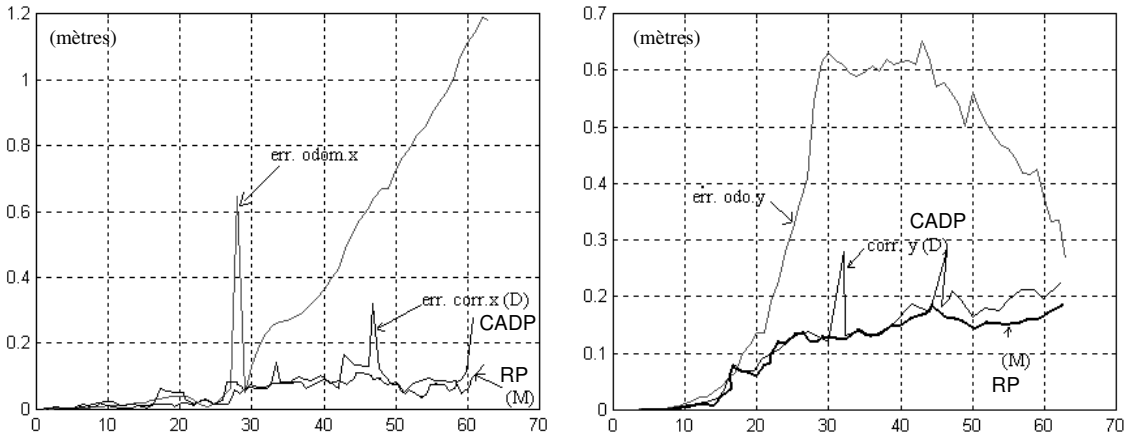


Figure III-26 : Erreur absolue sur x et y avec les deux règles CADP et RP dans le cas de modélisation triangulaire.

6-4-2- Influence de l'étendue

On s'intéresse maintenant à la répercussion qu'aurait une dilatation des paramètres de la distribution (support et/ou noyau) sur les performances de la localisation. On se pose comme condition que le résultat de la localisation est acceptable si les erreurs absolues moyennes sur x et y sont toutes les deux inférieures à 15 cm. Le but est alors de savoir à quel degré cette condition reste vérifiée quand on fait varier l'étendue des noyaux et/ou supports pour différents taux de dégradation de la donnée capteur (pourcentage du dégonflement d'une roue par

rapport à l'autre) [OUSSALAH 99]. En d'autres termes, jusqu'à quel niveau peut-on dégrader l'odométrie, tout en espérant obtenir une localisation acceptable en jouant sur l'étendue des distributions initiales ?
Des différents tests expérimentaux effectués, on déduit que :
i) Si on modifie simultanément les distributions odométriques et ultrasonores, l'analyse des résultats est difficile. Cependant, on constate que, si on modifie la distribution odométrique correspondant à l'orientation, cela engendre un effet négatif.
ii) Par contre, si on modifie seulement les distributions relatives aux ultrasons, le processus de fusion induit un comportement rationnel dans le sens où il existe toujours une valeur limite maximale du taux de dégradation correspondant à une valeur précise de dilatation pour lequel la condition reste vérifiée.
On a pu aussi noter les remarques suivantes :
- Avec des distributions trapézoïdales les deux règles peuvent être utilisées indifféremment, sans que ceci change significativement le taux de dégradation maximale. Par contre, dans le cas de modélisation triangulaire, la règle progressive donne des résultats nettement meilleurs.
- L'introduction d'un obstacle non modélisé dans la scène ne change pas significativement le résultat. Ceci est dû au fait que l'algorithme de fusion développé utilise uniquement les données disponibles à l'instant correspondant à la position courante du robot sans se soucier des états antérieurs. Ainsi, l'obstacle modifie uniquement, éventuellement, l'estimation correspondant aux instants où cet obstacle est décelé par les capteurs ultrasonores. On insiste sur le mot "éventuellement", car, le plus souvent, au moins sur une composante, cet état est éliminé par la règle de combinaison, puisqu'il induit une distribution très écartée des autres distributions. Ceci montre également la robustesse de notre algorithme de fusion.
-Les performances de cet algorithme sont quasiment toujours supérieures à celles d'un algorithme classique inspiré de la méthode de COX [COX 91] et développé sur le même robot dans [HOPPENOT 97] où toutes les données disponibles jusqu'à l'instant présent sont utilisées pour donner une estimation sur la position. Ceci est probablement dû au fait que nous utilisons une procédure permettant le rejet (systématique ou progressif) des données aberrantes (en désaccord avec les autres données), ce qui force le résultat à une position de compromis avec toutes les mesures disponibles.

7- Conclusion

L'objectif de ce chapitre était de tester les capacités de la théorie des possibilités à traiter des problèmes de fusion de données. Le contexte applicatif opté pour illustrer ces capacités étant la localisation d'un robot mobile.
Nous avons tout d'abord défini les principales étapes d'un processus de fusion. Un processus de fusion nécessite deux étapes fondamentales :
- l'alignement des données issues des différentes sources dans un même repère spatial et temporel ;
- la combinaison proprement dite de ces données transformées entre elles.
De l'analyse du comportement de la règle de combinaison adaptative usuelle proposée par DUBOIS et PRADE nous avons mis en évidence deux types de dysfonctionnement:
- la non prise en compte progressive d'une distribution en désaccord avec les autres ;
- la non robustesse du résultat de la fusion par rapport à la forme des distributions.
Nous avons donc proposé une modification de cette règle pour la rendre progressive et robuste.
Deux applications simples de localisation d'un robot mobile ont été traitées. Le but recherché ici n'est pas d'obtenir à tout prix la performance, mais plutôt de tester et valider une démarche en cherchant ses avantages et ses faiblesses.
La première application a été réalisée avec un robot miniature disposant de capteurs infrarouges et d'un odomètre. Il évolue dans un environnement muni de balises actives repérables par les capteurs infrarouges. Le but des expériences réalisées était ici de comparer la méthode possibiliste à celle du filtrage de Kalman, considérée comme la référence pour de tels problèmes. A l'issue de ces travaux, nous constatons que dans les situations d'informations riches, le filtrage de Kalman est sans conteste supérieur. Par contre, dans le cas de situations pauvres, la méthode possibiliste peut parvenir à de meilleurs résultats.
La deuxième application porte sur un robot d'exploration d'intérieur, muni d'un odomètre et de capteurs ultrasonores, qui évolue dans un environnement connu, mais susceptible de présenter des obstacles non modélisés. La méthodologie suivie consiste tout d'abord à mettre en correspondance les impacts des capteurs avec les murs de la pièce et à recaler le robot par rapport à chacune de ces données ultrasonores, puis à fusionner l'ensemble. Nous avons pu mettre en évidence l'efficacité de la méthode de fusion possibiliste mise en œuvre et sa robustesse par rapport aux obstacles non modélisés et par rapport aux caractéristiques des distributions de possibilités associées aux capteurs. Nous avons montré également l'effet positif des modifications apportées à la règle de combinaison dans le cas d'informations conflictuelles.
En plus de sa capacité spécifique de bien gérer les conflits, une méthode possibiliste de fusion n'impose aucune contrainte sur les bruits et la linéarité du processus et n'est pas sensible à l'ordre dans lequel les observations sont

traitées. Cependant, le recours à la théorie des possibilités n'est pas sans poser certains problèmes car la complexité algorithmique reste élevée, la gestion des variables liées n'est pas aisée et le résultat final n'est pas fourni sous forme d'une estimation munie de sa variance, mais globalement par une distribution de possibilités dont il faut extraire une information de type "moyenne" et une information de type "spécificité" caractérisant l'étendue de la distribution.

Bibliographie III

[COX 91]: I.J. COX , *Blanche- an experiment in guidance and navigation of an autonomous robot vehicle*, IEEE Trans. On Robotics and Automation, Vol. 7, pp. 193-204, 1991.
[DASARATHY 97]: V. B. DASARATHY, *Sensor fusion potentiel exploitation -innovative architecture and illustrative applications*, Proc. of IEEE Vol. 85, pp. 24-39, 1997.
[DELPLANQUE 96]: M. DELPLANQUE, *L'application des techniques modernes de traitement de l'information pour le guidage de véhicules sous-marins* Thèse de Doctorat de l'Université des sciences et technologie de Lille, Mars 1998.
[DEVEUGHELE 93]: S. DEVEUGHELE, *Etude d'une méthode de combinaison adaptative d'informations incertaines*, Thèse de Doctorat de l'Université de Compiègne, 1993.
[DONG 87a] W.M. DONG et F.S. WONG, *Fuzzy weighted averages and implementation of the extension principle*, Fuzzy Sets and Systems Vol. 21, pp 183-199, 1987.
[DONG 87b] W.M. DONG et H.C. SHAH, *Vertex method for computing functions of fuzzy variables*, Fuzzy Sets and Systems Vol. 24, pp.65-78, 1987.
[DUBOIS, 86]: D. DUBOIS, H. PRADE, *Weighted minimum and maximum opérations in fuzzy sets theory,* Informations Sciences, 1986
[DUBOIS, 91]: D. DUBOIS, H. PRADE, *Fuzzy sets in approximate reasoning: Part 1 : inference with possibility distributions,* Fuzzy Sets and Systems Vol. 25, pp. 143-202, 1991.
[DUBOIS, 92]: D., DUBOIS, H., PRADE, *Combination of fuzzy information in the framework of possibility theory*, In: "*Data fusion in robotics and machine intelligence"*, M. A. Abidi. and R. C. Gonzalez, (eds), Academic Press, pp.4481-505, 1992.
[DUBOIS, 94]: D. DUBOIS, H. PRADE, *Possibility theory and data fusion in poorly informed environment,* Control Eng. Practice Vol. 2, pp. 812-823, 1994
[DURIEU 96]: C. DURIEU, M. J. ALDON, D. MEIZEL, *La fusion de données multisensorielle pour la localisation en robotique mobile,* Traitement de Signal, Vol. 13, n°2, pp. 144-165, 1996.
[HALL 97]: D. HALL, J. LLINAS, An introduction to multisensor data fusion, Proc. of IEEE, pp.112-148, pp. 6-23, 1997.
[HOPPENOT 97]: P. HOPPENOT, *Contribution de la robotique mobile à l'assistance aux personnes handicapées*,Thèse de doctorat de l'université d'Evry val d'Essonne, Nov. 1997.
[KLIR 88]: G. J., KLIR, T. A., FOLGER, *Fuzzy sets uncertainty and information,* Prentice Hall, 1988.
[OPDERBECKE 94]: J.OPDERBECKE, *Localisation d'un robot mobile par fusion de données multisensorielles dans un environnement structuré*, Thèse de Doctorat de l'Université Paris VI, 1994.
[OUSSALAH 98]: M. OUSSALAH, *Fusion de données par la théorie des possibilités : Application à la localisation d'un robot mobile,* Thèse de Doctorat de l'Université d'Evry, Juin 1998.
[OUSSALAH 99]: M. OUSSALAH, H. MAAREF, C. BARRET, *Multisensor data fusion in the framework of possibility theory,* Soumis à IEEE Trans. on Fuzzy Systems.
[PANNEREC 98]: T. PANNEREC, *Localisation absolue d'un robot mobile : Fusion d'informations par la théorie des possibilités*, Mémoire de stage d'ingénieur IIE, Juin 1997.
[YAGER 92]: R.R.YAGER, *On the specificity of possibility distribution,* Fuzzy Sets and Systems, Vol. 50, 1992.

Conclusions et perspectives

Les travaux présentés dans ce document peuvent être regroupés dans deux thèmes principaux : le premier concerne la navigation d'un robot mobile par des méthodes basées sur des systèmes d'inférence floue et le second traite de la fusion de données capteurs en vue de la localisation absolue à l'aide d'une approche issue de la théorie des possibilités.

Le premier thème a porté essentiellement sur l'emploi de la logique floue en robotique mobile et un système de navigation semi-réactive a été développé. En effet, la commande floue s'avère être une solution attractive du point de vue intégration de l'aptitude des experts humains, ce qui permet une navigation anthropomorphique, non métrique et souple.
Dans le cas où le robot se trouve dans un environnement inconnu, deux méthodes d'actions réflexes n'utilisant que les informations capteurs, ont été étudiées parallèlement et appliquées pour naviguer à vitesse réduite.
La première, est celle du champ de potentiels artificiels. Son avantage est sa simplicité d'implémentation et la rapidité de son exécution, mais elle possède des limites telles que les minima locaux et les oscillations dans des milieux encombrés. Pour y remédier, un réglage automatique des coefficients basé sur des techniques de la logique floue a été réalisé. Ceci a d'une part éliminé les puits de potentiels et d'autre part généré un meilleur lissage de trajectoire. Mais le problème des oscillations dans les endroits étroits et passages de porte n'est pas résolu de cette manière.
Nous avons donc développé une deuxième méthode constituée d'une fusion d'un suivi du milieu de l'espace libre, vu par le robot, et d'une convergence vers le but. Ces comportements ont été implémentés grâce à des SIF sur deux robots mobiles Khepera et RMI, pour un passage de porte. Cette expérimentation a donné des résultats intéressants du point de vue de la robustesse vis à vis des mesures capteurs et de la versatilité, sachant que RMI est dix fois plus grand que Khepera et que les capteurs sont de natures différentes. Cependant, cette solution ne permet pas au robot de s'échapper de certaines situations de blocage créées lors du contournement d'obstacles concaves. Pour résoudre ce problème, un autre comportement élémentaire, de type suivi de mur a été développé. Associé à la création de buts intermédiaires de transition, ce dernier permet au robot d'éviter les obstacles concaves. Par la coordination de tous ces comportements élémentaires, la méthode proposée assure la navigation dans des environnements inconnus avec présence de tous types d'obstacles (convexes ou concaves).
Néanmoins, dans le cas de navigation du robot dans des environnements complexes, les bases de règles peuvent être difficiles à déterminer à cause du grand nombre de situations à considérer. L'ajustage des règles est coûteux en temps car il dépend des connaissances d'un expert. Donc, pour plus de rapidité et d'optimalité, une mise au point automatique de la partie conclusion des règles floues est souhaitable. Afin de rendre la génération et l'optimisation de règles floues automatiques en temps réel, nous avons utilisé une méthode basée sur une descente de gradient optimisant une fonction coût caractérisant le comportement désiré. Il est alors essentiel que la phase d'apprentissage converge vers des poids stables pour qu'ils soient traduisibles en concepts linguistiques. L'emploi d'une fonction coût regroupant une erreur quadratique et un terme de régression des poids nous a permis d'atteindre cet objectif. Nous pouvons alors obtenir des SIF optimisés, en partant de tables de décision vides ou préinitialisées, par apprentissage en ligne. Il est possible de contrôler la logique des règles de décision obtenues après optimisation en retraduisant les résultats numériques en concepts linguistiques. Nous avons pu vérifier par ailleurs que les contrôleurs flous ainsi obtenus avaient une très bonne robustesse par rapport au bruit sur les signaux d'entrée.
Mais, si nous laissons le contrôleur flou s'adapter en permanence à des modifications de l'environnement ou des caractéristiques du robot, nous avons constaté que cela pouvait conduire à des "pertes de mémoire", c'est à dire à la disparition de la capacité du robot à exécuter certaines tâches élémentaires. Pour remédier à cela, nous proposons une technique d'apprentissage "localisé" qui utilise les propriétés de localité d'un SIF dans l'espace de sortie et non dans l'espace d'entrée. Cette technique nous permet de laisser l'optimisation activée en permanence et donc d'avoir un contrôleur flou réellement adaptatif ayant une grande capacité de généralisation. Il serait intéressant pour cette application d'effectuer une optimisation structurelle (nombre de fonctions d'appartenance des variables d'entrée) sans perdre la possibilité de vérifier l'interprétation logique des règles.
Dans un environnement partiellement connu une approche a été proposée pour contrôler le mouvement d'un robot mobile. La connaissance *a priori* de l'environnement est mise en mémoire ce qui permet la comparaison en ligne des données capteurs de la scène réelle à celles de la scène mémorisée. Si elles sont identiques, la navigation dans la scène réelle se fera par le suivi de la trajectoire planifiée à vitesse maximum. Sinon, le contrôle de mouvement est réalisé grâce à un ensemble de stratégies d'actions réflexes du type précédent. Dans ce cas, la vitesse du robot est modulée en fonction du degré de ressemblance entre la scène localement perçue et celle en mémoire. Ce degré est fourni par un module flou qui compare les informations entre les deux scènes en prenant en compte les imprécisions liées aux données réelles des capteurs.

En faisant le bilan de ce thème, nous constatons que l'algorithme actuel de fusion des comportements élémentaires est certainement trop rigide. Il faudrait lui donner plus de souplesse pour passer d'une stratégie à une autre. Cela pourrait se faire, là encore, en ayant recours à des règles floues. Une autre idée pourrait être de pondérer les différents comportements par des coefficients adaptatifs en fonction des contraintes de la mission. Ceci pourrait se faire en ayant recours à des techniques de type apprentissage par renforcement. D'autre part, nous n'avons fait qu'effleurer les problèmes de mise à jour de l'environnement. Une bonne stratégie de mise à jour pourrait permettre de lancer le robot dans un milieu totalement inconnu, puis progressivement d'enrichir la base de données et d'augmenter la part de navigation selon un chemin planifié. Enfin, bien qu'intuitivement rien n'empêche que notre stratégie puisse gérer la présence d'obstacles mobiles, des travaux complémentaires doivent être entrepris pour vérifier les conditions de prise en compte de ce cas.

Le deuxième thème a porté sur l'emploi de la théorie des possibilités en fusion de données. Les techniques de fusion de données ont connu un essor remarquable depuis quelques années. L'opération de fusion permet d'extraire à partir d'un ensemble de sources une information plus fiable, au sens de la redondance et de la complémentarité, en tenant compte de l'imprécision et de l'incertitude supportées par les sources initiales. Les méthodes probabilistes demeurent les plus populaires pour traiter une telle situation, mais l'utilisation de théories différentes s'est répandue, telles que la théorie des sous-ensembles flous, la théorie des possibilités et la théorie de l'évidence. En effet, la théorie des possibilités est un outil très souple qui d'une part offre des cadres très ouverts (mais avec une multiplication des paramètres) et, d'autre part, permet de s'affranchir de la propriété d'additivité des probabilités qui limite les possibilités de modélisation, pour tenter de traiter, entre autres, le conflit entre les sources. Cette théorie est mise à profit dans les situations dans lesquelles nous pouvons dire qu'un évènement est possible, sans pour autant être vraiment capable d'y associer une probabilité. Le travail effectué concerne, d'une part, l'extraction de distributions de possibilités représentatives des données et, d'autre part, le développement de règles de fusion adaptatives. La méthode possibiliste que nous avons mise au point n'impose pas en théorie de forme particulière aux distributions. Elle est donc de ce point de vue plus souple et plus réaliste que le filtrage de Kalman (pris comme référence). Par rapport à ce dernier, la méthode possibiliste repose sur une approche différente pour résoudre le problème. En effet, la fusion des informations dans la méthode du filtrage de Kalman peut se voir comme une moyenne pondérée par l'incertitude liée aux informations. Le principe consiste à accorder plus de poids aux informations dont l'incertitude est faible (i.e. variance faible). Il s'agit donc d'une fusion par "minimisation de la variance". Par contre la règle de combinaison adaptative utilisée dans la méthode possibiliste s'intéresse plus à la redondance des différentes sources en accordant plus de poids aux informations qui se répètent. Il s'agit donc plutôt d'une fusion par "concordance". Elle n'impose aucune contrainte sur les bruits et la linéarité du processus et n'est pas sensible à l'ordre dans lequel les observations sont traitées. A partir d'expériences de comparaison (dans un cas simple) entre la méthode possibiliste et celle du filtrage de Kalman, on peut tirer les conclusions suivantes :
- dans les situations d'informations riches, le filtrage de Kalman est sans conteste supérieur ;
- par contre, dans le cas de situations pauvres, la méthode possibiliste peut parvenir à de meilleurs résultats.
Cependant nous avons pu constater que la règle CADP pose dans certains cas des problèmes de robustesse vis à vis de la forme des distributions à combiner. Nous proposons donc une nouvelle règle de combinaison dite *progressive* qui tient compte de façon explicite de la distance séparant la zone de consensus de chaque point du support et qui est robuste par rapport à la forme des distributions de possibilités. Cette règle a été évaluée et validée dans le cas de la localisation d'un robot mobile munis de capteurs ultrasonores évoluant dans un environnement partiellement connu.
Certains points doivent être améliorés pour rendre plus efficace la méthode possibiliste de fusion de données :
- la complexité algorithmique doit être réduite ;
- la gestion des variables liées n'est pas aisée ;
- le résultat n'est pas fourni sous forme d'une estimation munie de sa variance, mais globalement par une distribution de possibilités dont il faut extraire une information de type "moyenne" (par une méthode de défuzzyfication) et une information de type "spécificité" caractérisant l'étendue de la distribution.
Enfin, la règle progressive proposée a été mise en place afin de manipuler des données numériques, en l'occurrence des mesures capteurs dans le cadre de notre application robotique. Elle nécessite d'être améliorée pour qu'elle puisse être générique et applicable à d'autres domaines, tel que par exemple la fusion d'opinions d'experts.
Ces différents points doivent faire l'objet de travaux complémentaires, afin que leur résolution, même partielle, permette à des méthodes possibilistes de progresser et de se développer.