



HAL
open science

Modélisation incrémentale et localisation par amers pour la navigation d'un robot mobile autonome en environnement naturel

Stéphane Betge-Brezetz

► **To cite this version:**

Stéphane Betge-Brezetz. Modélisation incrémentale et localisation par amers pour la navigation d'un robot mobile autonome en environnement naturel. Automatique / Robotique. Université Paul Sabatier - Toulouse III, 1996. Français. NNT: . tel-00139789

HAL Id: tel-00139789

<https://theses.hal.science/tel-00139789>

Submitted on 3 Apr 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Année 1996

THÈSE

préparée au

Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS

en vue de l'obtention du

Grade de Docteur de l'Université Paul Sabatier de Toulouse

Spécialité: Robotique

par **Stéphane Betgé-Brezetz**

Ingénieur HEI

Modélisation incrémentale et localisation par amers pour la navigation d'un robot mobile autonome en environnement naturel

Soutenue le 16 février 1996 devant le jury composé de :

Georges	GIRALT	}	Président
Raja	CHATILA		
Michel	DEVY	}	Directeurs de thèse
Michael	BRADY		
Marc	RICHETIN	}	Rapporteurs
Zhengyou	ZHANG		
Maurice	BRIOT	}	Examineurs
Michel	FERRER		
Malik	GHALLAB		

Rapport LAAS N° 96148

Cette thèse a été préparée au Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS
7, avenue du Colonel Roche 31077 Toulouse Cedex

Avant-propos

Ce mémoire est l'aboutissement de plus de trois années de recherche passées dans le groupe "Robotique et Intelligence Artificielle" du LAAS. J'exprime ici toute ma reconnaissance à Monsieur Alain Costes, directeur du LAAS, pour m'avoir accepté au sein de son laboratoire.

L'accueil que Georges Giralt m'a réservé m'a permis de réaliser ces divers travaux dans des conditions particulièrement favorables. Qu'il soit remercié pour la confiance qu'il a su me témoigner et la qualité de ses conseils.

Ma reconnaissance se tourne maintenant vers Raja Chatila, directeur de ma thèse, dont j'ai éprouvé la patience et Ô combien la disponibilité et qui a toujours pris le temps de partager ses réflexions. Un grand merci à Michel Devy, directeur de ma thèse, pour l'attention qu'il m'a consacrée et les échanges enrichissants ... obtenus à toute fin !

Je remercie tous les membres de mon jury, Mike Brady -professeur à l'université d'Oxford-, Marc Richetin -professeur à l'université Blaise Pascal- et Zhengyou Zhang -chargé de recherche à l'INRIA- d'avoir accepté de porter un jugement sur mon manuscrit, Maurice Briot -professeur à l'université Paul Sabatier-, Michel Ferrer -directeur technique de MIDI-ROBOT- et Malik Ghallab -responsable du groupe RIA- d'avoir accepté d'être présents le 16 février et pour l'intérêt qu'ils ont témoigné à ma thèse.

Ces trois années m'auront d'autant plus enthousiasmé qu'elles ont été l'occasion d'activités corrolaires. Bon vent à toute l'équipe de Jacby (Patricia, Sara, Anne, Christophe, Béa, Philippe, Cathy, Olivier, Brigitte, Laurent, ...). Par ailleurs, je sollicite officiellement mon inscription dans la prochaine édition de l'annuaire des anciens doctorants du LAAS. J'attends que Jean-Charles, Christophe, Bertrand, Eric, Vanessa,... de l'ABG m'y rejoignent...

Je n'oublie pas toute l'équipe entière de RIA dont l'ambiance et l'esprit ont donné à ce travail une autre dimension. Merci à tout ceux avec lesquels j'ai partagé toutes ces années et à qui ce travail doit beaucoup : Jackie, Pat, Hanna, Matthieu, Simon, Vincent, Philippe, Fawzi, Sep, Florent, Fred...

Une pensée toute particulière pour ma famille et plus personnelle encore pour Guillou...

Table des matières

Table des figures	v
Présentation Générale	1
I Introduction	3
I.1 Robotique mobile en environnement naturel	3
I.1.1 La problématique	3
I.1.2 Modèle d'environnement naturel	4
I.2 Les projets de recherche	5
I.2.1 Les projets connus	5
I.2.2 Les projets impliquant le LAAS	6
I.2.3 Le projet EDEN	7
Première Partie : Le modèle par objets dans la représentation d'environnement naturel	9
II Modèle d'environnement naturel	11
II.1 Cahier des charges	11
II.1.1 Les besoins	11
II.1.2 Les contraintes	13
II.2 Etat de l'art des modèles d'environnements naturels	15
II.3 Bilan et approche générale proposée	20
III Le modèle par objets	25
III.1 Contexte de l'approche par objets	25
III.2 Définition des concepts utilisés	26
III.3 Des objets particuliers : les amers	30

III.4	Démarche de la construction du modèle par objets	30
Deuxième Partie : Segmentation et modélisation de scène		33
IV	Segmentation d'une scène	35
IV.1	Les capteurs tridimensionnels	35
IV.2	Extraction du sol	38
IV.2.1	Histogramme des normales	39
IV.2.2	Histogramme des distances au sol	41
IV.2.3	Analyse des histogrammes	42
IV.3	Extraction des objets	45
IV.3.1	Identification des contours entre objets	45
IV.3.2	Identification des objets	46
IV.4	Conclusion	47
V	Modélisation d'une scène	51
V.1	Modélisation géométrique	51
V.1.1	Modèle géométrique grossier	51
V.1.2	Modèle géométrique fin	53
V.2	Modélisation topologique	56
V.2.1	Relations topologiques	56
V.2.2	Relations d'occultation	59
V.2.3	Modèle topologique	59
V.3	Sélection et modélisation des amers	61
V.3.1	Caractéristique utilisée : le sommet	61
V.3.2	Sélection des amers	62
V.3.3	Calcul de l'incertitude du sommet d'un objet	63
V.3.4	Un exemple de sélection d'amers	68
V.4	Un détour par un environnement intérieur	68
V.4.1	Segmentation d'une scène structurée	69
V.4.2	Modélisation d'une scène structurée	70
V.4.3	Extraction de repères	71
V.5	Structure du modèle local	74
V.6	Conclusion	75

Troisième Partie : Modélisation par objets de l'environnement	77
VI Modèle par objets et incertitudes	79
VI.1 La carte stochastique	79
VI.2 Construction de la carte	81
VI.2.1 Les outils	81
VI.2.2 Les trois événements élémentaires	85
VI.3 Réalisations expérimentales	88
VI.4 Discussion	88
VI.5 Utilisation de la carte stochastique en environnement naturel	92
VII Construction du modèle de l'environnement	95
VII.1 Reconnaissance des amers et des objets	97
VII.1.1 Approches connues	97
VII.1.2 Approche choisie	98
VII.2 Recalage du robot	101
VII.3 Fusion et construction du modèle global de l'environnement	103
VII.4 Une première illustration de la modélisation incrémentale	106
VIII Trois exemples de modélisation incrémentale	113
VIII.1 Description des expérimentations	113
VIII.1.1 Objectifs recherchés	113
VIII.1.2 Moyens nécessaires	114
VIII.1.3 Démarche expérimentale	116
VIII.2 Première expérimentation	118
VIII.3 Deuxième expérimentation	124
VIII.4 Troisième expérimentation : cas d'un environnement structuré	128
VIII.5 Conclusion	131
Conclusion Générale	135
A Représentation de l'incertitude	139
A.1 Représentation graphique de l'incertitude d'un point	139
A.2 Représentation graphique de l'incertitude d'un repère	140
A.3 Calcul des deux premiers moments d'une distribution uniforme	141

B	Combinaison de transformations incertaines	143
B.1	Composition de deux transformations incertaines	143
B.1.1	Composition : cas 2D	144
B.1.2	Composition : cas 3D	144
B.2	Inversion d'une transformation incertaine	145
B.2.1	Inversion : cas 2D	146
B.2.2	Inversion : cas 3D	146
B.3	Inversion et composition de deux transformations incertaines	147
	Références bibliographiques	149

Table des figures

I.1	Robot Adam dans l'environnement EDEN du LAAS	7
I.2	Plateforme multisensorielle d'Adam	7
II.1	Tableau récapitulatif des réponses apportées au cahier des charges par chaque type de modèle.	21
II.2	Les différentes représentations de l'environnement	22
III.1	Image du site Geroms	26
III.2	Trajectoire en boucle	27
III.3	Une scène naturelle	29
III.4	Algorithme de modélisation incrémentale	32
IV.1	Image vidéo et image tridimensionnelle de la scène "cailloux"	36
IV.2	Image tridimensionnelle par stéréovision	37
IV.3	Principe de l'évaluation des normales	38
IV.4	Résultat de l'estimation des normales de l'image "cailloux"	38
IV.5	Système de coordonnées sphériques utilisé pour extraire le sol.	39
IV.6	Histogramme bi-variable des normales	40
IV.7	Système de coordonnées sphériques pour l'extraction des surfaces verticales.	41
IV.8	Procédure récursive de sélection des points d'un pic	42
IV.9	Histogramme mono-variable H'	43
IV.10	Résultat de l'extraction du sol de la scène "cailloux"	43
IV.11	Cas d'une scène composée de différentes régions	44
IV.12	Autre résultat de l'extraction du sol	44
IV.13	Image de contours	45
IV.14	Régions considérées pour la fermeture des contours	46
IV.15	Point de moyennage	47
IV.16	Résultat de l'extraction des objets de la scène "cailloux"	47
IV.17	Segmentation d'une image stéréo (1)	48
IV.18	Une scène segmentée du terrain EDEN (1)	49
IV.19	Une scène segmentée du terrain EDEN (2)	49
IV.20	Segmentation d'une image stéréo (2)	50

V.1	Estimation de la taille de l'ellipsoïde	52
V.2	Exemple de modèle ellipsoïde	52
V.3	Points 3D et superquadrique	54
V.4	Directions d'observation	55
V.5	Segmentation d'une scène générique	57
V.6	Objets identifiés dans l'image "cailloux"	59
V.7	Modèle topologique de l'objet no 1 de la scène "cailloux"	60
V.8	Modèle topologique de l'objet no 2 de la scène "cailloux"	60
V.9	Interprétation qualitative d'une scène	61
V.10	Planification qualitative de perception	61
V.11	Sélection des amers	62
V.12	Incertitude de forme pour un objet 2D	64
V.13	Incertitude de résolution	65
V.14	Région caractéristique de l'incertitude de résolution	67
V.15	Volume de l'incertitude de résolution	67
V.16	Incertitude de résolution pour un robot vertical	67
V.17	Cas d'un objet dont le sommet est plus haut que le capteur	68
V.18	Un exemple de sélection et de modélisation d'amers	69
V.19	Environnement intérieur composé de boîtes	70
V.20	Image 3D d'une scène structurée	70
V.21	Segmentation d'une scène structurée	70
V.22	Histogramme des normales d'une boîte	71
V.23	Incertitude de résolution d'une face	73
V.24	Sélection des amers d'une scène structurée	74
V.25	Modèle local d'un objet	75
V.26	Modèle local d'une scène	75
VI.1	Estimation itérative du point de linéarisation	84
VI.2	Evolution de la carte: le robot se déplace	85
VI.3	Evolution de la carte: le robot observe un nouvel objet	86
VI.4	Evolution de la carte: le robot reperçoit un objet	87
VI.5	Apport des corrélations pour la construction de la carte	89
VI.6	Evolution de l'incertitude avec et sans corrélations	91
VII.1	Algorithme de modélisation incrémentale	96
VII.2	Exemple d'un table de correspondance	100
VII.3	Principe d'appariement de structure d'objets	102
VII.4	Modèle global d'un objet	104
VII.5	Modèle global de l'environnement	106
VII.6	Exemple de modèle global d'objet	107

VII.7	Première perception	108
VII.8	Carte stochastique (1)	108
VII.9	Le robot se déplace (1)	108
VII.10	Deuxième perception	109
VII.11	Carte stochastique (2)	109
VII.12	Le robot se déplace (2)	109
VII.13	Troisième perception	110
VII.14	Carte stochastique (3)	110
VII.15	Le robot se déplace (3)	110
VII.16	Quatrième perception	111
VII.17	Carte stochastique (4)	111
VII.18	Modèle grossier	111
VIII.1	Le robot Adam dans le site Geroms	114
VIII.2	Démarche suivie pour chaque expérimentation	116
VIII.3	Calibrage d'une expérimentation (1)	117
VIII.4	Calibrage d'une expérimentation (2)	117
VIII.5	Photo et carte d'élévation de référence du site Geroms (1)	119
VIII.6	Segmentations de la partie "A" de Geroms	119
VIII.7	Comparaison des positions de la balise (1)	120
VIII.8	Comparaison de l'erreur de position de la balise (1)	120
VIII.9	Evolution de l'incertitude du robot sans recalage (1)	122
VIII.10	Carte construite et évolution de l'incertitude du robot avec recalage (1)	122
VIII.11	Déplacements relatifs estimés, recalages et nombre d'amers reconnus pour chaque perception.	123
VIII.12	Comparaison de l'erreur de position de la balise en présence de biais sur l'odométrie (1)	124
VIII.13	Photo et carte d'élévation de référence du site Geroms (2)	125
VIII.14	Segmentations de la partie "B" de Geroms	125
VIII.15	Comparaison des positions de la balise (2)	126
VIII.16	Comparaison de l'erreur de position de la balise (2)	126
VIII.17	Evolution de l'incertitude du robot sans recalage (2)	127
VIII.18	Carte construite et évolution de l'incertitude du robot avec recalage (2)	127
VIII.19	Déplacements relatifs estimés, recalages et nombre d'amers reconnus pour chaque perception.	129
VIII.20	Comparaison de l'erreur de position de la balise en présence de biais sur l'odométrie (2)	130
VIII.21	Amers segmentés d'une scène structurée	130
VIII.22	Erreur de translation et de rotation avec et sans recalage	132

VIII.23	Erreur en translation et en rotation de la carte construite	133
VIII.24	Comparaison de cartes obtenues en environnement intérieur	134
A.1	Différentes valeurs de la distribution de χ^2	141
A.2	Distribution de probabilité uniforme et unidimensionnelle	142
B.1	Composition de deux transformations incertaines	143
B.2	Inversion d'une transformation incertaine	146
B.3	Inversion et composition de deux transformations incertaines	147

Présentation Générale

L'*autonomie* d'un robot mobile résulte de sa capacité à pouvoir agir et réagir face à son environnement pour accomplir par lui-même sa tâche. Le robot est alors en interaction permanente avec cet environnement, suivant une boucle continue de "perception / décision / action". Nous nous situons dans le cadre d'une *approche délibérative*: le robot décidera des actions à entreprendre par des raisonnements sur sa tâche d'une part et sur les informations issues de ses perceptions d'autre part. Vis à vis d'un système purement réactif, nous estimons que cette approche permet d'apporter au robot l'efficacité et la robustesse qui sont des qualités indispensables pour mettre en application ces machines.

Mais pour mener ces raisonnements, le robot a besoin de structurer et d'interpréter les informations qui proviennent de ses capteurs. Il doit construire, à partir des données de perception, des *représentations internes* de l'environnement qui lui permettront notamment de prédire le résultat de ses actions et de choisir celles qui seront les plus appropriées pour réaliser sa tâche. Le choix de ces représentations est essentiel. Elles doivent être adaptées à la tâche du robot, mais aussi à l'environnement dans lequel il évolue.

Par ailleurs, la recherche en robotique mobile, forte de son expérience dans les environnements intérieurs, s'est tournée vers les milieux naturels, plus complexes et plus irréguliers. Le champ d'application ouvert dans ce type d'environnement est très vaste (sécurité civile, exploration planétaire ou sous-marine, ...). Mais ces milieux présentent des caractéristiques qui leur sont propres et des contraintes supplémentaires et spécifiques. Les représentations internes qui seront construites devront bien évidemment prendre en compte ces particularités pour être adaptées à cette classe d'environnement.

Notre travail se situe ainsi dans la problématique de la modélisation d'environnements naturels par un robot mobile. Il s'intègre dans un projet de recherche du groupe RIA du LAAS, visant à concevoir un robot d'intervention en milieu naturel. Un tel robot doit pouvoir adapter ses traitements aux caractéristiques de l'environnement qu'il parcourt: plusieurs travaux dans le groupe ont proposés des représentations surfaciques [Fillatreau 94] ou basées sur des régions caractérisant la nature du terrain [Lacroix 95a]. Les travaux présentés dans ce mémoire proposent une représentation originale basée sur la notion d'objets. Le robot identifiera ces objets à travers ses perceptions pour construire un modèle de l'environnement. Celui-ci sera à la fois précis (par les caractéristiques que présentent certains objets) et de niveau plus abstrait (par une description de l'ensemble des objets de l'environnement). Ce mémoire présente les principes de cette approche ainsi que les différentes

méthodes qui ont été mises en œuvre pour construire ce modèle dans des environnements réels. Il est structuré en trois parties.

La première partie analyse les spécificités des environnements naturels et leurs implications dans le choix des représentations. Le premier chapitre étudie le rôle qui revient au modèle dans la navigation autonome du robot. Les informations que ce modèle devra fournir sur l'environnement ainsi que les contraintes qu'il aura à satisfaire seront formulées à la manière d'un cahier des charges. Nous examinerons ensuite les différentes représentations d'environnements naturels proposées dans la littérature. Dans ce contexte, nous décrirons l'approche générale actuellement menée par le groupe de robotique du LAAS et préciserons comment le modèle par objets se situe dans ce projet. Le chapitre suivant déterminera alors les éléments de base sur lesquels s'appuie le modèle que nous proposons et donnera le principe général de sa construction incrémentale.

La deuxième partie porte sur l'analyse d'une scène particulière de l'environnement. Après une brève description des images tridimensionnelles, un premier chapitre traite du problème de la segmentation qui consiste notamment à identifier les objets de la scène. Le chapitre suivant précisera quelles seront les représentations choisies pour décrire ces objets, mais montrera surtout comment certains d'entre eux, appelés amers, pourront être sélectionnés afin de localiser le robot. L'application de l'approche par objets pour des environnements structurés sera aussi étudiée. Les différentes méthodes proposées tout au long de cette partie seront bien évidemment illustrées par de nombreux exemples sur des données réelles.

La dernière partie traite la modélisation incrémentale par objets de l'ensemble de l'environnement. La prise en compte des incertitudes est centrale dans notre approche. Aussi le premier chapitre lui est consacré afin de présenter les outils utilisés (carte stochastique, filtre de Kalman) et d'analyser leur application dans le cas des environnements naturels. Le chapitre suivant décrira les trois fonctions principales du processus de modélisation incrémentale: la reconnaissance, la localisation et la fusion. Enfin ce travail a bien évidemment fait l'objet de nombreuses expérimentations, étape indispensable à toute recherche en robotique. Le dernier chapitre est ainsi consacré à présenter certaines d'entre elles (notamment réalisées grâce au robot mobile Adam¹). Les modèles construits (ainsi que les résultats de localisation) seront analysés en détail à partir de données de référence.

Finalement, après un bilan des contributions apportées par ce modèle pour la navigation autonome du robot, nous concluons ce mémoire de thèse par une brève réflexion sur les différentes perspectives et directions d'extensions possibles de ces travaux. Nous évoquerons notamment les voies prometteuses de la perception active (choix de points de vue) ou d'une perception plus dynamique (capacité du robot d'intégrer *en continu* de nouvelles données capteurs) qui permettrait d'accroître l'autonomie et l'efficacité du robot.

1. Adam est propriété de Framatome et de Matra Marconi Space et est actuellement prêté au LAAS.

Chapitre I

Introduction

I.1 Robotique mobile en environnement naturel

I.1.1 La problématique

Depuis quelques années, la recherche en robotique mobile porte un intérêt tout particulier aux environnements naturels. Ce type d'environnement regroupe tous ceux situés à l'extérieur des bâtiments (voire les bâtiments eux-mêmes s'ils sont endommagés). Aussi qualifiés d'*extérieurs*, ils peuvent être très irréguliers et ne sont pas correctement représentables par des primitives géométriques simples (ligne droite, facette plane). Ils sont caractérisés par une grande variabilité du terrain. Leurs zones navigables sont plus ou moins encombrées, parfois même difficilement praticables et nécessitent alors de fortes capacités de locomotion. De surcroît, nous nous plaçons dans le cas général où l'environnement est initialement inconnu ou très peu connu (le robot disposant dans ce cas d'informations incomplètes ou imprécises).

La perspective des applications potentielles motive la recherche dans ce domaine. Le robot peut être utilisé dans des sites où l'homme ne peut aller, en raison de risques trop importants ou de difficultés d'accès [Giralt 91].

C'est le cas des *robots d'intervention* employés dans un souci de sécurité civile (lutte contre les incendies, traitement de zones endommagées après une explosion et contaminées par des radiations ou des produits toxiques). Le robot pourra intervenir sur le site pour lutter contre le sinistre ou simplement pour identifier les dommages.

Un autre exemple est le *robot d'exploration* qui devra prendre des mesures ou réaliser des expériences dans des sites distants (exploration planétaire ou sous-marine, exploration des volcans...). Depuis la station de contrôle, les scientifiques transmettront des requêtes au robot et sa mission pourra alors s'étendre sur de longues durées ou de longues distances [Giralt 92b].

Notons que pour les applications d'exploration, la cartographie des régions traversées peut consister en elle-même une des tâches à réaliser par le robot.

Cette classe d'applications soulèvent de nombreux problèmes robotiques.

D'abord la capacité du robot à se localiser dans l'environnement (celui-ci étant mal connu ou inconnu et donc bien évidemment non balisé par avance). Par ailleurs, l'aspect non structuré du terrain vient compliquer ce problème de localisation.

La variabilité des environnements naturels doit bien évidemment être prise en compte et le robot devra adapter sa navigation en fonction de la nature du terrain traversé.

Enfin, une contrainte essentielle provient des possibles difficultés de communication entre le robot et sa station de contrôle. Celles-ci sont de deux ordres : d'une part les délais de transfert de l'information qui peuvent devenir non négligeables lorsque le robot est très distant (par exemple parfois plus 40 mn aller-retour entre une station terrestre et un robot sur Mars), d'autre part le débit disponible qui peut être très faible (1 à 2 kilobits par seconde sur des fenêtres de huit heures pour le même type d'application). Pour ces deux raisons et sachant qu'il faut aussi tenir compte d'éventuelles interruptions de communication, la téléopération est rendue très difficile dans ce type d'application.

L'approche développée au LAAS consiste à doter le robot d'un certain degré d'autonomie, en lui faisant prendre à sa charge une part des décisions qui concernent sa navigation. Sa commande se fera à un niveau d'abstraction élevé : celui de la tâche qui est à exécuter. On parle alors de Programmation au Niveau Tâche [Giralt 93]. Dans un contexte de navigation, elle peut être définie au robot de la manière suivante : "aller à tel emplacement", "rechercher tel objet dans l'environnement"... Le robot affinera cette tâche et élaborera par lui-même un plan d'action pour la réaliser. Il raisonnera à la fois sur sa tâche, sur sa connaissance de l'environnement et sur sa configuration courante. Ce raisonnement évaluera les effets de chaque action envisageable et décidera des plus appropriées pour atteindre son but ou s'en rapprocher. Une fois le plan construit, le robot en exécutera les différentes actions (déplacements, perceptions) et en contrôlera le bon déroulement. L'architecture à bord du robot a donc trois niveaux : l'affinement de tâche, le contrôle d'exécution et la couche fonctionnelle.

Le robot dialoguera avec la station de contrôle en lui transmettant des comptes rendus sur la réussite ou non de sa tâche. Cette station constitue le deuxième élément du système global. A partir de buts généraux formulés par un opérateur (par exemple "explorer telle région"), elle planifie dans le temps et l'espace la mission du robot. Cette station supervisera aussi la bonne exécution de la mission et pourra la modifier en cas de problèmes.

1.1.2 Modèle d'environnement naturel

Chacune des opérations menées par le robot nécessite des capacités de perception, de représentation et d'interprétation de l'environnement [Chatila 93a]. Au fur et à mesure de ses déplacements, le robot va donc en construire un modèle. Il transformera pour cela les données sensorielles issues de ses capteurs en informations utilisables par ses composantes décisionnelles. Son autonomie dépendra pour une grande part de la richesse des informations qu'il extraira de cette représentation et des interprétations qu'il en fera.

Le modèle de l'environnement joue également un rôle important dans l'interface entre la station de contrôle (ou l'opérateur) et le robot. C'est grâce à sa représentation interne que ce dernier convertira la tâche abstraite qui lui est soumise en actions à réaliser. Considérons par exemple la tâche "aller à (amer)" qui consiste à chercher et à reconnaître un amer dans l'environnement. Le robot comparera la description donnée de cet amer à l'ensemble de ceux déjà modélisés et présents dans sa représentation interne. Il tâchera alors de le reconnaître sinon poursuivra la recherche.

Nous voyons ici l'importance du modèle de l'environnement pour la navigation du robot et pour sa commande. Le choix des représentations internes est un point clé dans la conception du robot, il participera pour une grande part à son efficacité et à sa robustesse. Ces représentations devront permettre de répondre à certains besoins du robot, mais elles devront aussi prendre en compte les contraintes inhérentes à la nature de l'environnement (peu structuré dans notre cas), aux moyens mis à la disposition du robot pour le percevoir (caractéristiques des capteurs), pour le construire (capacité de calcul) et le stocker (capacité mémoire).

Notre travail se situe dans cette problématique. La première partie de ce mémoire est ainsi consacrée à l'étude du rôle qui revient à la représentation interne de l'environnement dans la navigation du robot. A partir de cette analyse, nous proposerons une représentation particulière bâtie sur la notion d'objets et permettant de modéliser un grand nombre d'environnements naturels. Tout au long de ce mémoire nous montrerons comment le robot peut construire ce modèle et l'utiliser pour réaliser sa tâche.

Ce travail a été réalisé au sein du groupe de robotique du LAAS et a ainsi pu profiter du contexte stimulant de l'équipe et de son expérience acquise par sa participation à de nombreux projets. Nous présenterons dans la section suivante une brève revue des différents projets de robotique mobile en milieu naturel menés dans un contexte international puis plus particulièrement ceux impliquant le groupe de robotique du LAAS.

1.2 Les projets de recherche

1.2.1 Les projets connus

Les premiers projets de robots mobiles avaient trait au suivi d'une trajectoire initialement planifiée sur une carte d'élévation et à l'évitement d'obstacles temps réel (projets ALV [Daily 88], Navlab [Thorpe 88]). Ces projets se sont ensuite étendus en intégrant notamment des fonctions de perception plus complexes comme la modélisation incrémentale de l'environnement, la localisation ou la fusion de capteurs. Ils ont donné naissance à diverses réalisations parmi lesquelles nous pouvons citer :

- **Ambler.** C'est un robot à six pattes télescopiques développé à l'institut de robotique de l'université Carnegie Mellon (CMU). Le projet a permis la réalisation de nombreux progrès dans les domaines de la modélisation 3D de l'environnement à partir de données télémétriques [Krotkov 91], de la planification de déplacement d'un

robot marcheur et de l'intégration matérielle et logicielle d'un système robotique [Krotkov 96].

- **Dante II.** C'est un robot à huit pattes aussi développé à CMU [Krotkov 94]. Son système de navigation possède différents modes allant de la téléopération directe à l'autonomie complète. Il utilise ainsi des caméras pour l'asservissement visuel et un télémètre pour la cartographie du terrain.
- **Robby.** C'est un robot à six roues développé au Jet Propulsion Laboratory (JPL). Il est équipé de stéréovision et se déplace de manière semi-autonome en suivant des points de passage fixés au préalable par un opérateur [Matthies 92].
- **Projet UGV** (Unmanned Ground Vehicle) dirigé par la DARPA et qui réunit de nombreux laboratoires américains.

Face aux contraintes fortes des applications spatiales, la NASA s'est aussi intéressée à la réalisation de robots plus "réactifs" dont l'architecture de contrôle est basée sur une approche comportementaliste ([Gat 92, Gat 94, Miller 92]). Ces robots possèdent vraisemblablement des caractéristiques de robustesse par leur simplicité, mais risquent toutefois de ne pas satisfaire l'ensemble des spécifications que les scientifiques attendent lors d'une mission. Le groupe RIA du LAAS suit ainsi une approche "délibérative" où la machine planifie les actions qu'elle doit entreprendre afin de réaliser la tâche qu'un opérateur lui aura assignée.

1.2.2 Les projets impliquant le LAAS

Le premier projet de robotique mobile en environnement extérieur auquel a participé le groupe RIA (Robotique et Intelligence Artificielle) fut le projet Euréka AMR (Advanced Mobile Robot) [Lacombe 91]. Il a débuté en 1987; l'objectif était de définir, concevoir et développer des robots mobiles d'intervention pour des applications de sécurité civile. Le groupe RIA a principalement contribué à l'implémentation et à l'intégration du sous-système de perception ainsi qu'à la conception de l'architecture du système décisionnel. Durant le projet AMR, les sociétés Matra Marconi Space et Framatome se sont équipées du robot mobile Adam¹ que nous présenterons dans le paragraphe suivant.

Le projet VAP (Véhicule Automatique Planétaire) a été engagé en 1989 par le CNES (Centre National d'Etudes Spatiales) dans le but de développer les concepts de base pour l'exploration planétaire par robotique mobile [Giralt 92a]. Le groupe RIA y a participé activement et notamment à travers le consortium RISP (Robots d'Intervention sur Site Planétaire) regroupant des laboratoires appartenant au CNRS, au CEA, à l'INRIA et à l'ONERA².

Dans le cadre des applications spatiales et à la suite du projet VAP, le CNES a entrepris le projet Euréka IARES (Illustrateur Autonome de Robotique mobile pour l'Exploration Spatiale) [Lacroix 95b, Chatila 93c]. L'objectif est de valider les concepts proposés pour les futurs robots d'exploration planétaire, le LAAS participant notamment au système de

1. Advanced Demonstrator for Autonomy and Mobility

2. CEA: Commissariat à l'Energie Atomique, CNRS: Centre National de la Recherche Scientifique, INRIA: Institut National de la Recherche en Informatique et Automatique, ONERA: Office National des Etudes et Recherches Aérospatiales

déplacements autonomes. Prévu pour l'exploration de la planète Mars, ce projet connaît toutefois des retards et des évolutions vers des missions lunaires dus à des modifications programmatique.

1.2.3 Le projet EDEN

Le projet EDEN³ auquel nous avons participé est interne au groupe RIA et a pour but de démontrer la navigation complète d'un robot mobile en environnement naturel [Chatila 93b, Nashashibi 94, Betgé-Brezetz 96]. L'environnement est initialement inconnu et la tâche du robot consiste à se rendre en un point particulier. Cette tâche est alors spécifiée par la commande "aller à (but)", le but consistant à atteindre des coordonnées ou à reconnaître un amer.

Ce projet englobe les différents thèmes liés à la navigation d'un robot mobile en environnement naturel :

- perception et modélisation de l'environnement ;
- localisation du robot ;
- planification de trajectoires sur tout type de terrain ;
- construction d'une architecture globale recouvrant les niveaux décisionnel et fonctionnel ;

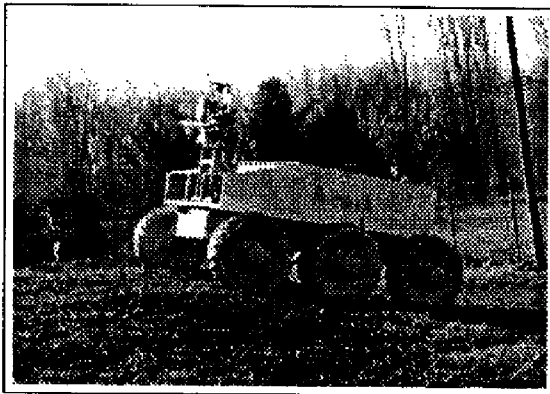


FIG. I.1 – Robot Adam dans le site d'expérimentation du LAAS.

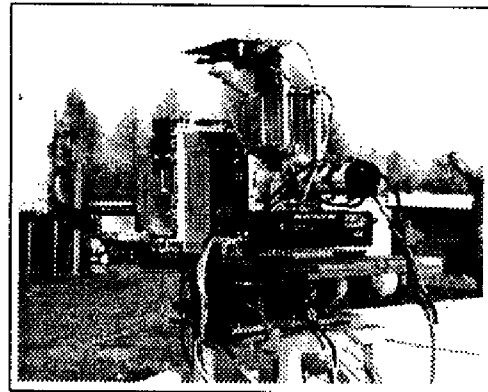


FIG. I.2 – Plateforme multisensorielle d'Adam.

Les méthodes développées dans ce projet ont pu être expérimentées avec le robot Adam prêté au LAAS par les sociétés Matra Marconi Space et Framatome. Ce robot montré figure I.1 dans le terrain d'expérimentation du LAAS est de taille imposante (3m×2m×2m) et possède les caractéristiques suivantes :

- le châssis, composé de six roues motrices, provient de l'institut VNII Transmash (à St. Pétersbourg en Russie) ;

- les capteurs proprioceptifs sont six codeurs (un au niveau de chaque roue) et une centrale inertielle SFIM (dont la dérive est estimée de 1 à 3 deg/h par le constructeur). A partir de ces capteurs, la position et l'orientation du robot sont calculées par le module GNC (Guidance Navigation & Control) développé par Matra Marconi Space.
- le robot est équipé d'une plateforme multisensorielle (cf figure I.2). Celle-ci possède un capteur laser à temps de vol conçu par le société Riegl (Allemagne). La variance sur la profondeur (pratiquement constant pour des distances mesurées entre 1 et 10 m) est de 9cm^2 (soit un écart type de $\pm 3\text{cm}$), la portée maximale étant d'une vingtaine de mètres. Deux moteurs animant d'une part la platine en rotation (selon un axe vertical) et d'autre part un miroir pivotant (selon un axe horizontal) permettent de faire varier la direction du faisceau respectivement en azimut et en site. Le système sensoriel possède aussi quatre caméras vidéo pour développer des algorithmes de vision stéréoscopique et de fusion multisensorielle.

Les capacités de locomotion et de perception d'Adam font de ce robot un outil tout à fait bien adapté pour des expérimentations de navigation en milieu naturel.

Adam a parcouru le site d'expérimentation du LAAS que nous présenterons au cours de ce mémoire. Un autre site, Geroms, a été construit au CNES dans le but d'y développer des applications de robotique mobile spatiale. Nous le présenterons au dernier chapitre dans lequel nous avons regroupé différentes expérimentations que nous y avons réalisées.

Première Partie

Le modèle par objets dans la représentation d'environnement naturel



Chapitre II

Modèle d'environnement naturel

Pour accomplir sa tâche, un robot doit disposer d'une représentation interne de l'environnement dans lequel il évolue. En effet, c'est par des "raisonnements" à la fois sur sa tâche et sur cette représentation qu'il décidera des actions à entreprendre pour parvenir progressivement à son but. Dans ce chapitre, nous nous intéressons ainsi au rôle qui revient à un tel modèle. Dans un premier temps, nous cherchons à définir, sous la forme d'un cahier des charges, quelles doivent être ses caractéristiques.

Au regard de celles-ci, nous étudierons ensuite les différents modèles d'environnements naturels que l'on trouve dans la littérature. Nous détaillerons alors plus particulièrement l'approche choisie dans notre équipe et montrerons comment notre travail s'y intègre.

II.1 Cahier des charges

La fonction du modèle de l'environnement est de délivrer un certain nombre d'informations pour que le robot puisse exécuter les tâches qui lui sont confiées efficacement et en toute sécurité. Ce modèle devra garantir, tout au long du déplacement, la fiabilité des informations fournies. Dans cette partie nous allons ainsi formuler, à la manière d'un cahier des charges, quels peuvent être les *besoins* du robot en information et les *contraintes* que le modèle devra satisfaire pour assurer leur qualité.

II.1.1 Les besoins

Nous dénombrons trois principaux types de besoins que nous analyserons successivement. Ces besoins sont les suivants :

- la détection et représentation des zones navigables pour les déplacements du robot ;
- la localisation du robot par rapport à son environnement ;
- la reconnaissance de son but.

Détection et représentation des zones navigables

Dans les environnements extérieurs, le terrain est généralement très varié : certaines zones sont relativement régulières, tandis que d'autres sont beaucoup plus inégales voire très accidentées. Dans ce contexte, un élément clé de l'efficacité du robot est sa capacité à pouvoir s'adapter au terrain. Il devra ainsi identifier la nature de ses différentes parties, et déterminer celles qu'il pourra traverser pour rejoindre son but. Nous mentionnons ici les travaux réalisés dans notre équipe par S. Lacroix [Lacroix 95a] qui dénombre trois classes de terrains : plat, obstacle et accidenté. Trois différents modes de déplacement sont alors définis suivant la nature du terrain à traverser, chacun de ces modes correspondant à une représentation et à un **planificateur de trajectoire** particuliers. La caractérisation du type de terrain (ou **classification de terrain**) va permettre de sélectionner le mode de déplacement correspondant.

- Dans le cas où l'environnement est libre d'obstacle (*i.e.* plat), le robot peut se déplacer sur un simple asservissement à une consigne de cap ou de position. Un modèle qualitatif grossier suffit pour sélectionner ce mode. Il n'y a pas de planification de trajectoire à proprement parler, une surveillance continue de la zone de déplacement est simplement nécessaire. On parle alors de *mode réflexe*.

- Si la zone de navigation s'apparente à un plan sur lequel reposent des obstacles isolés, le robot utilisera un *planificateur 2D* qui fournit une trajectoire sous forme de courbe plane. Le modèle de l'environnement représentera alors cette région en décrivant par exemple ses frontières et les bords des obstacles par des lignes polygonales.

- Dans le cas plus complexe où les zones navigables sont très irrégulières (*i.e.* accidentées), l'utilisation d'un *planificateur 3D* s'impose. Le robot doit alors disposer d'une représentation surfacique du terrain dans un format compatible avec ce planificateur (carte d'élévation, surface B-splines).

D'une manière plus générale, le choix de l'un de ces trois modes de déplacement est supervisé par un module appelé **planificateur de chemin**. En fonction du but, celui-ci décidera des régions à explorer ou à traverser, et pour ces dernières, du type de planificateur de trajectoire à appeler. Une fonctionnalité importante du planificateur de chemin est le choix des perceptions (*i.e.* des positions et des modalités des capteurs). Ce choix prendra en compte les régions que le robot doit encore modéliser, mais aussi le besoin qu'il a de posséder une bonne estimation de sa position.

Localisation

La capacité de se localiser par rapport à son environnement est un besoin clé de la navigation d'un robot mobile. Des capteurs proprioceptifs (odométrie, centrale inertielle) délivrent une estimation de la position et de l'orientation du robot. Mais ces systèmes sont entachés d'erreur dont les causes principales sont :

- l'intégration des erreurs des codeurs. Ces erreurs qui sont relativement faibles s'accumulent progressivement ;
- une dérive dans le temps pour ce qui concerne la centrale inertielle ;

- le glissement des roues ou autres mécanismes tracteurs. Ce phénomène, qui produit un biais sur la position du robot, est aggravé dans les environnements extérieurs.

Nous verrons que le modèle de déplacement du robot permet de rendre compte des deux premières sources d'erreur (erreur des codeurs et de la centrale inertielle), mais beaucoup plus difficilement des erreurs dues au glissement des roues.

Par ailleurs, les systèmes de localisation globale utilisant des capteurs externes (comme le GPS) sont très répandus dans la navigation maritime mais manquent encore de précision (100m pour un GPS classique, 10m dans le cas des GPS différentiels). De plus, certaines applications rendent leur utilisation impossible (exploration planétaire par exemple).

Aussi une solution consiste à équiper le robot d'un capteur extéroceptif. Le robot exploitera les informations perçues dans l'environnement pour se localiser. Cette fonction sera assurée par le modèle de l'environnement qui permettra lui de détecter, de représenter et de reconnaître des points de repères particuliers (appelés **amers**). Nous soulignons ici l'importance des deux fonctionnalités de modélisation et de reconnaissance d'amers pour la navigation du robot.

Reconnaissance du but

Le robot doit bien entendu se rendre compte qu'il a atteint son but. Si celui-ci s'exprime sous forme de coordonnées, cela se fera simplement sur la base de l'estimée de sa position et de son incertitude. Mais ce but pourrait aussi être décrit au robot par une forme ou un ensemble de formes particulières à reconnaître dans l'environnement (ce qui suppose leur connaissance a priori). La fonction de modélisation devra alors s'acquitter de cette tâche en mettant en œuvre les algorithmes de détection et de reconnaissance de ces formes.

II.1.2 Les contraintes

Le robot prendra en partie ses décisions sur la base des informations fournies par le modèle. Il est par conséquent indispensable pour la sécurité du robot et pour la réussite de sa mission que ce modèle en garantisse la fiabilité. Cela se traduit par des contraintes que le modèle de l'environnement devra satisfaire en permanence. Nous dénombrons trois contraintes majeures :

- la prise en compte des incertitudes ;
- l'adaptabilité du modèle et la richesse de ses représentations ;
- la cohérence globale du modèle.

Prise en compte des incertitudes

Avant tout, nous remarquons qu'il serait plus rigoureux d'utiliser le terme d'imprécision que celui d'incertitude. Toutefois, ces deux termes sont souvent confondus dans la littérature, et nous ferons aussi leur confusion dans le reste de ce mémoire.

Les capteurs du robot, qu'ils soient extéroceptifs (télémètre laser, caméra) ou proprioceptifs (odométrie, centrale inertielle), fournissent tous des mesures entachées d'erreur. Aussi, il est indispensable que le modèle de l'environnement, qui est construit à partir de ces mesures, prenne en compte ces imprécisions. Il convient alors de choisir une représentation de l'erreur de mesure, et d'établir, pour chaque capteur, un modèle de son imprécision.

Dans le cas des capteurs de type télémètre laser, ce modèle d'incertitude peut être construit sur la base d'essais statistiques réalisés pour différentes valeurs de distance. La modélisation des parties intrinsèques du capteur (parties mécaniques, moteurs, ...) permettra ensuite de calculer l'incertitude finale des mesures 3D [Nashashibi 93a]. Pour quantifier les artefacts possibles, des modèles plus complets (mais aussi plus complexes) peuvent être élaborés en considérant également l'angle d'incidence du faisceau ou le type de surface sur laquelle se font les mesures.

Pour les systèmes de stéréovision l'incertitude sur les données est aussi fonction de la distance et des paramètres intrinsèques du système (position relative des caméras, précision des centres optiques, ...).

En revanche, dans le cas du capteur odométrique, il devient plus difficile d'obtenir un modèle réaliste de l'erreur. Il est envisageable de modéliser les erreurs provenant des codeurs et de la centrale inertielle, mais il est beaucoup plus difficile de modéliser celles dues aux glissements des roues. En effet, ces glissements sont fonction du coefficient de frottement qui dépend de la nature du terrain et est très difficile à estimer dans le cas d'un terrain naturel. Afin de ne pas introduire d'information erronée dans la représentation de l'environnement, une solution consiste à se contenter d'un modèle de l'odométrie qui surestime l'incertitude (mais en restant toutefois réaliste). Nous reviendrons sur ce point à diverses reprises au cours de ce mémoire.

Adaptabilité du modèle et richesse de ses représentations

Les environnements naturels sont très divers. Ils peuvent varier continûment d'un terrain plat sans obstacle à un terrain très accidenté. Comme nous l'avons précisé, le modèle doit permettre de différencier ces diverses classes pour adapter ses primitives de représentation.

D'autre part, pour un type particulier de terrain, les besoins ne sont pas toujours les mêmes. Telle région peu accidentée doit simplement être traversée par le robot. Un modèle représentant les parties navigables et quelques points de repère sera dans ce cas suffisant. Deuxième cas de figure : dans une autre région *de même nature*, le robot doit reconnaître une forme particulière. Un modèle plus riche s'avère alors nécessaire pour décrire l'environnement de façon plus détaillée afin d'y reconnaître cette forme. Le modèle de l'environnement devra donc posséder différents niveaux de représentation. Le robot choisira alors celui qui convient le mieux pour réaliser sa tâche.

A cela s'ajoute la notion de gestion du modèle, essentielle pour le robot qui doit parcourir une grande distance. Des contraintes, notamment de taille mémoire, imposent de ne pas conserver toute l'information extraite au cours du déplacement. On distingue alors la partie située au voisinage du robot du reste de l'environnement. Le modèle de la partie avoisinant le robot pourra posséder des représentations riches afin de répondre

dynamiquement à ses besoins (éviter d'obstacles par exemple). Le modèle du reste de l'environnement possédera, de son côté, l'information minimale pour répondre à des besoins comme le traitement des impasses ou les retours en arrière.

Cohérence globale du modèle

Le modèle de l'environnement est construit au fur et à mesure des déplacements du robot, intégrant itérativement les différentes perceptions locales. La position du robot n'étant connue qu'avec incertitude, des problèmes vont se poser quant au modèle global obtenu. Ce dernier devra être suffisamment souple pour conserver sa cohérence tout au long des déplacements du robot et ce malgré ses erreurs de localisation. Nous reviendrons sur cet aspect qui constitue une des priorités de notre travail.

D'autre part nous avons vu que plusieurs représentations complémentaires peuvent coexister dans le modèle. Les informations qu'elles fournissent sur une même partie du terrain doivent bien évidemment être concordantes. Cette autre propriété de cohérence concerne donc celle des différentes représentations entre elles.

11.2 Etat de l'art des modèles d'environnements naturels

Au regard des besoins et contraintes définis précédemment, nous étudions maintenant les différents travaux développés dans le domaine de la modélisation d'environnement naturel. Mais tout d'abord, rappelons brièvement les différents niveaux de représentation d'un environnement [Chatila 85] (un modèle particulier pouvant se situer à l'un ou à plusieurs de ces niveaux) :

1. **le niveau géométrique.** Il contient les descriptions de la géométrie de la surface du terrain ou de certaines de ses parties.
2. **le niveau topologique.** Il représente les relations topologiques entre certaines parties de l'environnement qui présentent des caractéristiques spécifiques (lieu).
3. **le niveau sémantique.** C'est le niveau le plus abstrait. Il donne à chacune de ces parties le nom d'une catégorie (arbre, buisson, idots). Cette classification est réalisée sur la base de connaissances *a priori* fournie au modèle (la liste de ce qui peut se trouver dans l'environnement, des attributs caractéristiques correspondant et de leur nom).

En ce qui concerne les travaux actuels de modélisation d'environnements extérieurs, nous pouvons dénombrer les quatre grandes familles suivantes :

- les modèles surfaciques ;
- les modèles basés sur des cellules ;
- les modèles discrets ;
- les modèles qualitatifs.

Ces modèles sont situés à différents niveaux de représentation, partant des modèles de plus bas niveau pour terminer sur des représentations abstraites de l'environnement. Nous précisons que cette classification ne revêt pas un caractère absolu, mais elle permettra de distinguer les intérêts propres à chaque type de représentation. Nous allons les examiner indépendamment les uns des autres.

Les modèles surfaciques

Ces modèles représentent la surface du terrain sous forme de cartes numériques ou de fonctions analytiques.

Les cartes numériques, encore appelées cartes d'élévation ou modèles numériques du terrain (MNT) représentent l'altitude Z de la surface en chaque point d'une grille (X, Y) . Elles sont construites par interpolation des données 3D, l'incertitude sur l'altitude Z étant calculée à partir de l'incertitude des points de mesure.

I. S. Kweon et T. Kanade [Kweon 92] ont proposé une méthode de construction de MNT à partir d'images tridimensionnelles. L'estimation de la transformation entre la carte locale et le modèle global est d'abord basée sur l'extraction et la reconnaissance de maxima de courbure. Cette estimation s'affine ensuite à l'aide d'une méthode iconique et d'un algorithme de descente de gradient. Une fois la transformation calculée, la carte locale est fusionnée au modèle global par réestimation de la valeur et de l'incertitude de chaque altitude.

Cette méthode nécessite toutefois un modèle idéal de capteur. Aussi des travaux, réalisés dans notre groupe par F. Nashashibi [Nashashibi 93a], ont permis de s'affranchir de cette condition en proposant une technique différente basée sur une interpolation directe des données cartésiennes. La transformation entre la carte locale et globale est aussi estimée à partir d'une approche à la fois iconique et basée sur des caractéristiques. La fusion des cartes s'effectue selon la même méthode, au prorata des incertitudes des élévations.

Citons ici les travaux d'Asada [Asada 90] qui construit aussi un modèle numérique de l'environnement mais en ignorant les incertitudes du capteur.

Les méthodes surfaciques, proches des cartes d'élévation, représentent le terrain à l'aide de fonctions analytiques. P. Fillatreau a proposé dans notre équipe l'utilisation des surfaces B-splines [Fillatreau 94]. Ces primitives présentent des propriétés intéressantes de déformation locale qui leur permettent de modéliser plus finement certaines parties du terrain. Elles peuvent être construites à partir de cartes d'élévation pour simplifier le traitement des occultations et offrent alors l'avantage de réduire le volume mémoire occupé.

Regardons maintenant comment ces méthodes répondent aux besoins et contraintes établis précédemment.

L'identification des parties navigables du terrain peut se faire par le choix de critères d'homogénéité (planéité, courbures). La recherche des zones navigables est alors fréquemment réalisée en appliquant un algorithme de type grossissement-fusion partant de régions élémentaires satisfaisant les critères précédents [Nashashibi 93a, Asada 90, Hebert 86].

Ces modèles permettent aussi le recalage du robot sur la base d'appariements de ca-

ractéristiques particulières et/ou de méthodes iconiques. Toutefois, dans le cas des cartes d'élévation, le recalage en orientation se fait simplement par corrélation. Dans le cadre de l'estimation du registre entre deux surfaces décrites par des points, citons les travaux de Z. Zhang [Zhang 94] qui propose une méthode adaptée pour des terrains de forme quelconque. Cependant, les modèles surfaciques nécessitent une première estimation fiable de la position du robot, ce qui limite leur utilisation pour la localisation.

D'autre part, la capacité de représentation d'une information symbolique ou sémantique par un modèle surfacique reste faible. Il sera par conséquent difficile de décrire le but à reconnaître. On demeure avec ce type de modèle à une représentation bas niveau de l'environnement (niveau géométrique).

Enfin, ces méthodes permettent de prendre en compte des incertitudes des capteurs; mais, en raison d'une certaine rigidité, elles peuvent poser des problèmes de cohérence à long terme (notamment dans le cas d'un trajet en forme de boucle). De plus le volume mémoire que ce type de modèle requiert est relativement important et il semble difficilement envisageable de maintenir ces représentations sur une grande distance.

Les modèles basés sur des cellules

Dans ce type de modèle l'environnement est discrétisé en cellules (réparties sur un plan horizontal dans la plupart des cas). A chacune de ces cellules est associé un vecteur d'attributs, la différence de ce modèle par rapport au précédent est que ce vecteur ne représente pas nécessairement l'altitude du terrain. La grille formée par ces cellules peut constituer un maillage régulier (carré, rectangulaire, hexagonal, ...) ou irrégulier.

Les "occupancy grids" (grilles d'occupation) proposées par A. Elfes et L. Matthies [Matthies 88, Elfes 91] en sont une réalisation caractéristique. Ils utilisent des informations provenant de différents capteurs (ultrasons, stéréovision) et un modèle bayésien afin d'estimer la probabilité d'occupation d'une cellule ou de la remettre à jour si elle a déjà été observée. Lorsque chaque cellule est indépendante, la représentation équivaut à un champ de Markov d'ordre 0. A. Elfes a ensuite généralisé la notion "d'occupancy grids" à celle "d'inference grids" (grilles d'inférence) pour que le robot réalise des planifications de perception dynamique [Elfes 92]. Le vecteur stochastique de chaque cellule est alors enrichi d'attributs d'observabilité et d'atteignabilité par le robot.

Dans notre équipe, S. Lacroix [Lacroix 95a] a élaboré une procédure de classification rapide de données 3D provenant d'un terrain quelconque. Il utilise un maillage régulier dans l'espace du capteur (correspondant à la discrétisation angulaire d'une image laser ou à la plaque CCD d'une caméra), et obtient alors des grilles de tailles variables dans le plan support du terrain. A chaque cellule de la grille est associé un vecteur stochastique d'attributs (densité, variance sur l'altitude des points, normale moyenne, ...). Une procédure de classification bayésienne permet d'identifier les classes des cellules (plat, accidenté, obstacles ou inconnu). Les cellules connexes de même classe sont ensuite regroupées pour former des régions homogènes qui serviront à la constitution d'un graphe topologique de l'environnement. L'utilisation de la théorie de la décision permet au robot de mettre en œuvre des stratégies de perception et de déplacement pour rejoindre son but.

Ce modèle se situe au niveau géométrique, mais pourrait aussi se placer au niveau topologique si l'on considère le graphe des régions. Il est tout à fait approprié à la détection et à la classification des zones navigables, fournissant au planificateur de chemin l'information dont il a besoin. D'autre part, l'outil bayésien permet de traiter les incertitudes de classification de manière rigoureuse.

En revanche, ce type de représentation est relativement peu précis. Il est à lui tout seul insuffisant pour détecter et reconnaître les amers de l'environnement, ne pouvant donc assurer la localisation du robot. Pourtant, la position de ce dernier doit être connue avec précision car ces modèles (comme les modèles surfaciques) sont basés sur une structure rigide (généralement à deux dimensions). Ils présentent ainsi un caractère "continu" qui leur autorise peu de variations. Les erreurs de localisation ne sont en réalité tolérées que par les représentations "discrètes".

Les modèles discrets

Les deux approches précédentes représentent l'environnement d'un seul tenant, assimilant tout au terrain. Or nous savons par notre propre expérience qu'il existe des éléments particuliers, isolés et de surcroît précis, que l'on peut retrouver à maintes reprises dans l'environnement. Une autre approche consiste alors à formaliser cette connaissance a priori, puis à la transmettre au robot afin qu'il détecte, modélise et reconnaisse par lui-même ces éléments singuliers.

Dans cette optique, nous mentionnons les nombreux travaux en environnement intérieur qui se servent de primitives géométriques simples appelées "caractéristiques" (en anglais "features"). Ces environnements ont l'avantage de posséder des structures régulières (lignes droites, plans) qui facilitent la définition de telles caractéristiques. Certains auteurs utilisent les murs ou les coins perçus au moyen de capteurs ultrasoniques [Leonard 92], les segments ou les portes dans des coupes laser [Moutarlier 89a, Devy 95a], les plans dans des images laser [Nashashibi 93b], les segments dans des images vidéo [Lebegue 92], les segments et facettes obtenus par stéréovision et fusion multisensorielle [Ayache 89, Ayache 90], la fusion multisensorielle ayant par ailleurs été largement abordée dans la littérature [Grandjean 91b, Moutarlier 89b, Durrant-Whyte 87, Chatila 85].

La modélisation en environnement naturel est un domaine plus récent de la recherche en robotique mobile. De plus, ce type d'environnement ne présente pas de structure régulière ; aussi les approches basées sur des caractéristiques y sont moins nombreuses. Parmi celles étudiées nous trouvons toutefois les maxima de courbure [Goldgoff 89], les courbes de niveau ou les lignes de plus grande pente [Kweon 91], les maxima d'élévation [Sutherland 94] (ces maxima pouvant d'ailleurs s'extraire à partir des modèles surfaciques [Fillatreau 94]).

Ces caractéristiques restant toutefois "ponctuelles", d'autres auteurs préfèrent plutôt considérer des parties isolées qui correspondent à des formes volumiques.

J.Y. Kheng [Kheng 91] propose une approche intéressante basée sur une vue panoramique qui est construite à l'aide d'une caméra vidéo. Le robot identifie des zones particulières dans la scène ("distinctive features") en considérant des paramètres distinctifs (hauteur, forme, couleur, ...) visibles depuis la plus large région possible. Dans un second

temps, un calcul de disparité permet d'enrichir la représentation de ces zones par des informations de distance [Li 92].

A partir d'une carte d'élévation, M. Asada cherche à extraire les obstacles d'une scène extérieure [Asada 88]. Il les classe ensuite en objets naturels ou artificiels, et représente chacun d'eux par le morceau de MNT correspondant. Un travail équivalent [Hirata 92] a été réalisé sur des images de scènes routières prises par une caméra couleur. Le système extrait des éléments caractéristiques (route, arbres, voitures...) à partir de connaissances a priori. Il donne ensuite une interprétation de la scène (ce qui relève du niveau sémantique).

A. Bobick [Bobick 92] cherche quant à lui à représenter des objets naturels à différents niveaux, en fonction de l'information disponible et des besoins du robot. Il parle alors d'espace des "représentations" et utilise des primitives de type "BLOB-2D", "BLOB-3D", superquadriques ou "stick".

M. Hebert [Hebert 92] extrait pour sa part des amers isolés dans des images de profondeur, les utilisant bien sûr pour le recalage du robot. Il utilise un modèle de type surface déformable pour les représenter. Toutefois, il émet au préalable des hypothèses sur la dimension des amers pour les segmenter, et la méthode proposée ne prend pas encore en compte les incertitudes issues des données et l'influence réciproque des amers du modèle.

Ce type de représentation trouve sa validité dans le cas d'environnement extérieur composé d'obstacles isolés. Ces modèles sont fort intéressants pour leur variabilité de représentation. Citons à ce propos les travaux de A. P. Pentland [Pentland 91, Pentland 86] qui décompose l'environnement "en parties", construisant pour chacune un modèle basé sur un agencement de superquadriques.

Ces modèles discrets semblent tout à fait pertinents pour résoudre des problèmes de recalage, de localisation et de reconnaissance de but. Les incertitudes peuvent être prises en compte de manière rigoureuse en propageant progressivement celles associées au capteur. Ces modèles étant basés sur des parties isolées, la réestimation de l'une d'entre elles, mais aussi de l'ensemble, peut se faire aisément sans remettre en cause tout le modèle. Ces représentations offrent ainsi des propriétés de souplesse et de cohérence tout à fait séduisantes. Notons que ce type de modèle se situe aux niveaux géométrique et topologique, mais peut aussi s'étendre au niveau sémantique si chaque partie est identifiée par un nom. En revanche, un modèle composé uniquement de tels éléments paraît insuffisant pour planifier des trajectoires ou pour représenter des environnements très accidentés.

Les modèles qualitatifs

Les approches précédentes sont des représentations de l'environnement basées sur des primitives géométriques ou numériques. D'autres auteurs envisagent des représentations basées sur plusieurs niveaux de description. Le premier niveau permet d'identifier les lieux de l'environnement, tandis que le niveau le plus abstrait exprime les relations d'accessibilité entre ces lieux (notion de topologie). On parle pour ce dernier de "carte cognitive" ou de "modèle qualitatif".

B. Kuipers et Y. Byun ont développé ces idées dans un contexte d'exploration en environnement intérieur inconnu [Kuipers 88, Kuipers 91]. Ils définissent un lieu comme étant

le maximum local d'une mesure de distinction (déterminée par les caractéristiques visibles d'un point de vue) ; les relations topologiques correspondent aux chemins possibles entre ces lieux et sont représentées par des arcs. Ils exploitent à la fois des informations géométriques et topologiques pour modéliser l'environnement, reconnaître les lieux et localiser le robot.

D. Lawton et T. Levitt [Levitt 90] ont étudié la modélisation qualitative dans le cas où des amers assimilables à des points peuvent s'extraire de l'environnement. Les lieux sont ici définis par des régions ayant pour frontières les segments formés par deux amers. Une relation topologique caractérise le passage de la frontière d'un tel lieu. A partir de ce modèle qualitatif, D. Dai [Dai 93] a mis au point des stratégies de navigation sans mesure exacte des positions des amers.

D'autre part, on peut aussi apparenter le graphe topologique des régions proposé par S. Lacroix à une description qualitative de l'environnement. Citons enfin dans notre équipe les travaux menés actuellement par H. Bulata [Bulata 96] qui visent à décrire un environnement structuré en termes de lieux.

De tels modèles qualitatifs permettent la planification d'un chemin, mais nécessitent des représentations de plus bas niveaux si l'on veut planifier des trajectoires. De même la localisation qualitative à partir de ces modèles n'est pas toujours suffisante. En revanche un modèle haut niveau, défini sur la base de relations topologiques, permet de décrire le but avec plus de généralité. Il sera aussi plus aisé, à partir de ces relations, de maintenir la cohérence du modèle global sur de longs déplacements du robot.

II.3 Bilan et approche générale proposée

Pour mieux comparer tous ces modèles, nous regroupons très sommairement dans un même tableau (cf figure II.1) l'ensemble des remarques faites précédemment. Nous y précisons comment chaque modèle permet de répondre aux besoins et aux contraintes que nous avons fixés avec la légende simplifiée suivante :

- aucune croix : le modèle ne peut pas remplir cette fonctionnalité ;
- "x" : le modèle peut la remplir, mais n'est toutefois pas adapté pour ;
- "xx" : le modèle convient parfaitement pour remplir cette fonctionnalité.

A la lecture de ce tableau, nous constatons que les modèles étudiés apportent chacun certaines réponses au cahier des charges qui a été défini. En revanche, aucun ne semble satisfaire pleinement tous les besoins et les contraintes d'un modèle d'environnement naturel. Certaines représentations sont bien adaptées pour telle classe de terrain, mais peuvent être trop précises ou trop volumineuses dans d'autres cas, ou même révéler des problèmes de cohérence à long terme. Inversement, telle autre représentation peut maintenir sa cohérence malgré de longs déplacements, mais s'avère insuffisante pour planifier des trajectoires ou localiser le robot avec précision.

Aussi une solution consiste à utiliser conjointement les différents types de représentation, pour bénéficier de leur complémentarité. C'est cette approche qui a été adop-

	Modèles surfaiques	Modèles basés sur des cellules	Modèles discrets	Modèles qualitatifs
BESOINS				
Zones navigables				
Planification de chemin		xx		xx
Planification 3D de traj.	xx			
Planification 2D de traj.	x	xx	x	
Rech. de zones navigables	x	xx	x	x
Localisation				
Calcul de la position	xx	x	xx	x
Rech. d'amers	x	x	xx	x
Reconnaissance du but	x	x	xx	x
CONTRAINTES				
Incertitudes	x	xx	xx	x
Souplesse du modèle		x	xx	xx
Cohérence globale		x	xx	xx

FIG. II.1 – Tableau récapitulatif des réponses apportées au cahier des charges par chaque type de modèle.

tée au LAAS dans le cadre du projet EDEN [Devy 95b, Lacroix 94, Betgé-Brezetz 94b, Chatila 93b]. La figure II.2 montre les différentes représentations qui coexistent actuellement dans le modèle de l'environnement. Les flèches indiquent les données qui doivent être utilisées pour la construction d'un modèle.

La première représentation, toujours construite, est constituée de l'ensemble des modèles suivants : le "bitmap", la carte des régions et leur graphe de connexion. Comme nous l'avons vu, ces modèles sont basés sur une classification rapide du terrain et permettent de définir les sous-buts à atteindre successivement et le mode de navigation qu'il convient d'utiliser.

Les deux autres représentations ne seront pas construites systématiquement. Cela dépendra de la nature du terrain et des besoins du robot.

Dans le cas de zones accidentées, c'est un modèle de type MNT qui sera construit afin de planifier des trajectoires 3D. Un modèle B-splines permettra ensuite d'extraire du MNT des zones d'intérêts pour la procédure de recalage par corrélation.

Les zones peu accidentées (*i.e.* correspondant à un sol régulier avec des obstacles) seront représentées par un modèle basé sur une description de l'environnement en objets. Ce modèle représentera leur forme (modèle géométrique) ainsi que leur disposition relative (modèle topologique). C'est plus précisément pour ce type de modèle que se situe notre contribution. Si l'on se réfère à la classification précédente, ce modèle appartient bien évidemment à la catégorie des modèles discrets. Au cours de ce mémoire, nous tâcherons de montrer que ce modèle est tout à fait bien adapté à la localisation du robot ou à la reconnaissance de son but. Nous verrons comment les incertitudes des capteurs peuvent être prises en compte de manière rigoureuse dans tout le processus de modélisation (à

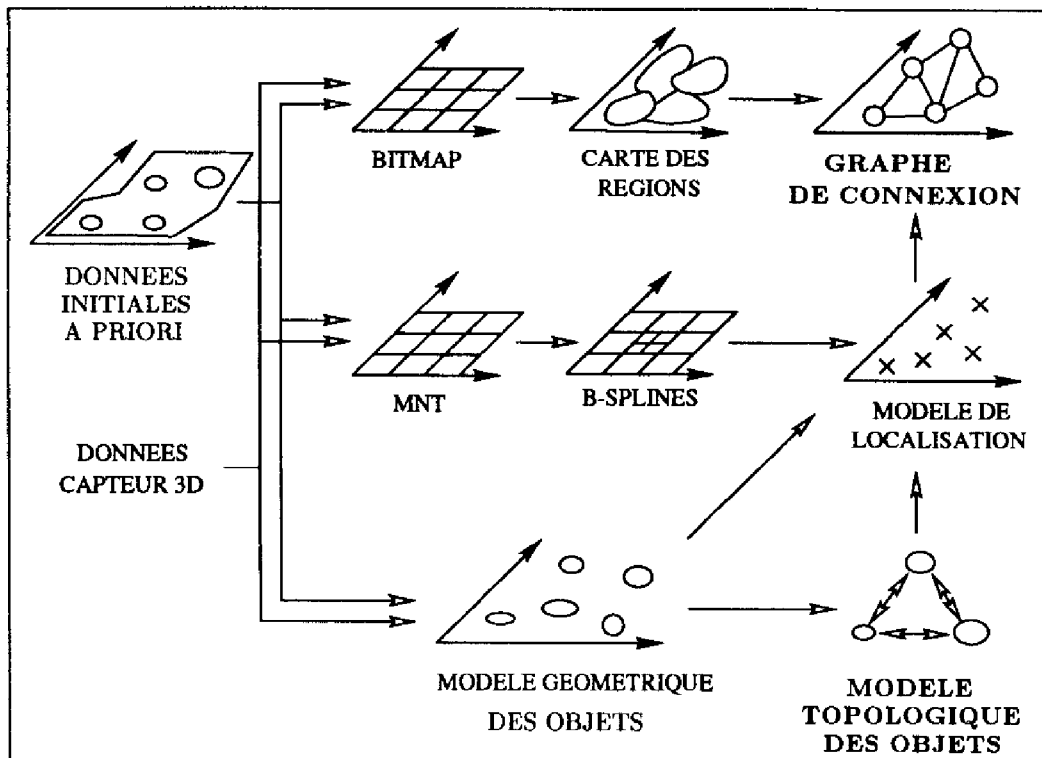


FIG. II.2 – Les différentes représentations de l'environnement. Les deux représentations de "graphe de connexion" et de "modèle topologique des objets" se situent au niveau topologique, tandis que les autres restent à un niveau géométrique.

l'aide du filtre de Kalman). De plus, nous montrerons qu'un des principaux intérêts de cette représentation discrète est de garantir sa cohérence malgré des erreurs importantes sur la position du robot. Mais auparavant, il nous faudra répondre aux questions suivantes :

- comment déterminer les objets de l'environnement ?
- comment les représenter ?
- comment utiliser ces objets pour le recalage, la localisation, l'extraction d'amers et la reconnaissance de but ?

Des réponses seront proposées dans les prochains chapitres et notamment à la première question dès le prochain chapitre, qui s'attachera à déterminer de manière théorique ce qui peut être appelé "objet" dans un environnement naturel.

Une autre question reste cependant ouverte relative à l'ensemble des modèles présents dans le système: la cohérence de ces représentations entre elles. Il se peut en effet que deux modèles caractérisent une *même zone* de l'environnement (par exemple le modèle "bitmap"

et le modèle objet). Il est alors nécessaire de garantir la cohérence des informations que chacun fournit. Si ces informations ne concordent pas (par exemple un modèle indiquerait la présence d'un obstacle, l'autre non), il convient de régler le conflit en réévaluant ces modèles. Dans le cas inverse (les deux modèles indiquant la présence d'un obstacle), la confiance que l'on a sur la nature de cette zone devra augmenter (améliorant ainsi la confiance dans les deux modèles).

Nous voyons ici toute la nécessité et l'intérêt d'un **gestionnaire de modèle** dont la fonction serait de détecter et de gérer les éventuels conflits, ou d'affiner les représentations. C'est un domaine de recherche particulièrement intéressant et qui reste encore à étudier.



Chapitre III

Le modèle par objets

Différentes représentations complémentaires cohabitent dans le modèle. Nous avons vu que certaines sont toujours construites, tandis que d'autres le sont suivant la nature du terrain. Le modèle par objets que nous proposons fait partie de cette deuxième catégorie. Dans ce chapitre, nous allons définir à quel type d'environnement ce modèle est adapté, et l'intérêt qu'apporte cette représentation particulière. Il nous faudra ensuite donner une description précise des éléments sur lesquels notre approche s'appuiera (*i.e.* les objets bien entendu, mais aussi le sol et les amers). Enfin nous présenterons la démarche générale qui permettra au robot de construire incrémentalement un modèle par objets de l'environnement.

III.1 Contexte de l'approche par objets

Nous avons vu que les modèles surfaciques étaient bien appropriés pour représenter des terrains très accidentés. Ils posent toutefois des problèmes de cohérence à long terme et restent encore insuffisants pour décrire le but du robot quand celui-ci n'est pas donné par ses coordonnées. En revanche, ce type de modèle, fondé sur une discrétisation régulière, est mal adapté lorsque l'environnement se "simplifie". C'est notamment le cas pour les terrains composés d'une surface uniforme sur laquelle se trouvent des obstacles isolés (comme par exemple la région montrée figure III.1). Une représentation surfacique d'un tel environnement semble dans ce cas trop volumineuse et surtout ne profite pas d'une certaine structure qui réside dans ce type d'environnement.

Aussi nous cherchons à établir un modèle adapté à cette dernière classe d'environnement. La représentation choisie ne devra fournir que l'information nécessaire pour ne pas être trop volumineuse et se focalisera ainsi sur certaines zones d'intérêts appelées objets. D'autre part, elle devra garantir sa cohérence tout au long des déplacements du robot, malgré les erreurs de localisation. Cela passe tout d'abord par une gestion rigoureuse de

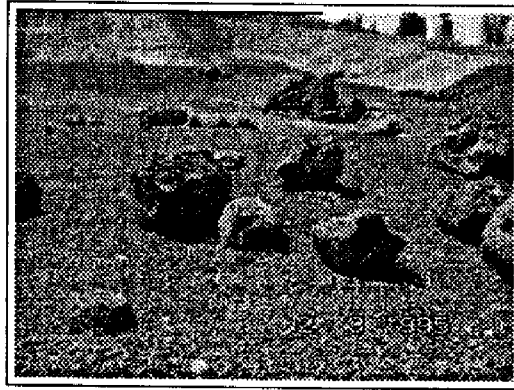


FIG. III.1 – Image prise sur le site Geroms du CNES. On peut décomposer cette partie en une zone uniforme sur laquelle se trouvent des rochers épars.

l'incertitude de ses primitives. Ensuite, le tableau de la figure II.1 (p. 21) montre que la cohérence d'un modèle augmente avec son niveau d'abstraction. Nous tâcherons de mettre à profit cette remarque en intégrant aussi au modèle des informations de plus haut niveau.

Ce modèle devra notamment permettre de gérer le problème des trajets en boucle. La figure III.2 montre en trait continu la trajectoire qu'un robot a réellement suivie, et en pointillés la trajectoire qu'il estime avoir exécutée. Les points noirs donnent les positions d'arrêt et de perception du robot. Malgré les recalages successifs, on peut supposer qu'il n'a pas pu corriger entièrement sa position et qu'il revient, par une autre direction, au voisinage de son point de départ avec une certaine erreur résiduelle (représentée sur la figure II.1 par l'écart entre les deux extrémités des flèches P_n et P'_n). A cette dernière position, le robot reperçoit des amers qu'il avait observés au début du déplacement. Le modèle doit bien évidemment permettre de réestimer correctement la position du robot ; mais il doit aussi, une fois ce recalage effectué, être entièrement remis à jour. En effet, ayant réestimé P'_n , la position P'_{n-1} doit l'être à son tour, ou plus exactement toutes les parties de l'environnement qui ont été perçues et modélisées à cette position. Et ainsi de suite... C'est un problème typique de cohérence de modèle.

Les modèles discrets permettent d'effectuer ce type de correction. En effet, pour de tels modèles, la réestimation de certaines parties peut se faire aisément sans remettre nécessairement en cause son ensemble. La représentation par objets que nous proposons, faisant partie de cette catégorie, permettra de résoudre ce problème.

III.2 Définition des concepts utilisés

L'approche de modélisation par objets est basée sur l'idée simple que, nonobstant leur aspect accidenté, un grand nombre d'environnements extérieurs possède une surface régulière qui occupe une partie importante du terrain (voire majoritaire). L'intuition de cette idée provient de la définition que le langage courant donne au mot "sol". Le reste de

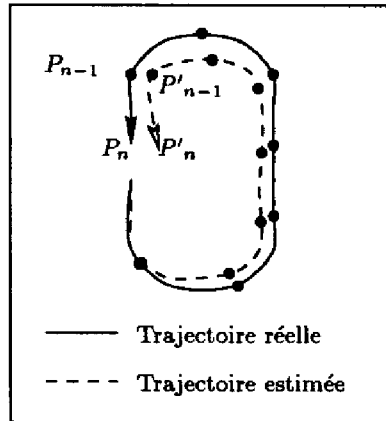


FIG. III.2 - *Problème de cohérence du modèle : la trajectoire en boucle.*

l'environnement est constitué de parties isolées, de forme quelconque, que nous appelons communément "objets".

Ces deux notions abstraites de **sol** et d'**objets** forment la base de notre modèle de l'environnement, mais elles ne sont pour l'instant qu'intuitives. Il nous faut en établir une description précise pour déterminer comment le robot extraira par lui-même le sol et les objets de ses perceptions. Pour être rigoureux, nous proposons tout d'abord de définir ce que nous entendrons par environnement, scène et perception (définitions 1, 2 et 3).

Définition 1 (Environnement) *L'environnement est le milieu dans lequel évolue le robot.*

Définition 2 (Scène) *Etant donné une position du robot et le champ de vue de son capteur, une scène est la partie de l'environnement que le robot peut percevoir.*

Définition 3 (Perception) *Une perception est constituée de l'ensemble des données prises dans la scène par le capteur.*

Notons que dans nos définitions, la scène est liée à l'environnement, tandis que la perception l'est au capteur. Toutefois, ces deux notions sont très proches, et il nous arrivera ainsi de les confondre par la suite. A partir de là, nous pouvons donner une définition précise du sol et des objets [Betgé-Brezetz 94a]:

Définition 4 (Sol) *Le sol correspond à une (ou plusieurs) surface majoritaire dans l'environnement et dont les normales ont des directions équivalentes.*

Définition 5 (Objets) *Une fois le sol déterminé, les objets constituent le reste de l'environnement. Ce sont des parties connexes de matière et isolées les unes des autres par des espaces vides.*

Ces deux dernières définitions appellent quelques remarques.

- D'une part, nous n'avons pas défini le sol à partir des capacités de locomotion du robot. Aussi cette définition (comme d'ailleurs celle des objets) n'est pas fonction d'un type particulier de robot, ce qui lui permet d'être assez générale. Toutefois, on observera que le sol correspond, dans la majorité des cas, aux surfaces sur lesquelles le robot naviguera de manière préférentielle. Notons par ailleurs que l'hypothèse de l'équivalence des directions des normales traduit d'une certaine manière une faible granularité du sol.

- Le sol est majoritaire dans l'environnement. Mais il est possible que, dans une scène particulière, il ne soit pas prépondérant (par exemple pour les perceptions qui ont lieu proches de zones très encombrées). Dans ce cas, la condition de prédominance du sol pourrait être remplacée par une condition relative à son orientation. Par exemple, dans le cas où des zones voisines ont déjà été modélisées, on peut imposer au sol de la perception étudiée d'avoir une orientation similaire. D'autre part, certains robots ont une estimation de la verticale (*i.e.* de la direction de la gravité); c'est le cas pour le robot Adam, avec lequel nous avons mené nos expérimentations, qui est équipé d'une centrale inertielle. Une contrainte sur l'horizontalité du sol (relativement à cette verticale) constituerait une seconde condition sur son orientation. Dans les environnements intérieurs, il conviendrait de prendre en compte cette contrainte pour distinguer les murs du sol.

- Des objets particuliers peuvent se situer *au-dessous* de la surface du sol. Ils seront tout simplement appelés des "trous".

- Aucune hypothèse n'a été avancée sur la forme des objets. La propriétés essentielles permettant de les définir c'est la continuité de la matière (propriété de connexité). Mais à plus long terme, chaque objet pourrait être subdivisé en différentes parties [Biederman 87]. Chacune de ces parties serait représentée par une primitive de base [Solina 94, Gupta 93, Pentland 91], pour former par la suite un graphe qui décrirait leur configuration. Si le robot possède une base de données d'objets déjà définis, il pourra reconnaître celui qu'il perçoit dans cette base. Il lui affectera éventuellement ensuite une étiquette sémantique [Connell 87].

La figure III.3 en donne un exemple. Elle représente une image où l'on supposera que le robot a détecté le sol (majoritaire et d'orientation uniforme) et un objet (*i.e.* la région foncée qui constitue un ensemble connexe de matière). Il décompose ce dernier en une partie cylindrique et une partie sphérique. La disposition de ces deux parties correspondant à la connaissance *a priori* que le système a des arbres, le nom "arbre" serait affecté à cet objet. Mais nous précisons ici que nous nous limiterons, dans ce mémoire, à

une représentation simple des objets. Aussi, la prise en compte de connaissance sémantique ne sera pas abordée. Elle constitue toutefois la suite directe de nos travaux.

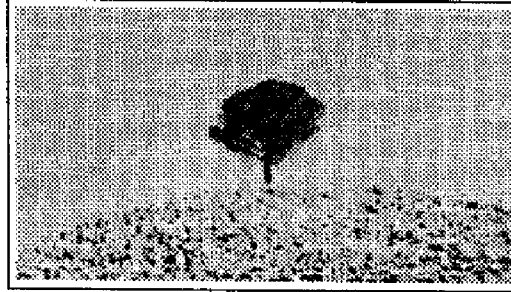


FIG. III.3 – Une scène naturelle dans laquelle le robot identifiera le sol (majoritaire et d'orientation uniforme) et un objet (partie connexe de matière).

Nous venons donc de décomposer l'environnement en sol et objets. Ainsi dans une image tridimensionnelle, nous chercherons à identifier les pixels correspondant au sol et ceux correspondant à chaque objet. Mais si l'on considère uniquement ces deux entités, il restera des pixels de l'image pour lesquels aucune étiquette n'aura été affectée: ceux correspondant à des parties trop éloignées pour que le capteur les perçoivent. Aussi, pour affecter une étiquette à toute partie d'une image, il faut définir la notion de fond d'une scène :

Définition 6 (Fond) *Etant donnée une scène de l'environnement, le fond correspond aux parties lointaines que le capteur du robot n'a pas pu percevoir.*

Une scène comprenant N objets peut alors se décomposer de manière complémentaire entre ces trois éléments caractéristiques :

$$\text{Scène} = \left(\bigoplus_{i=1}^N \text{Objet}_i \right) \oplus \text{Sol} \oplus \text{Fond} \quad (\text{III.1})$$

Enfin, notons que la notion d'objets a été définie par opposition à celle de sol. Aussi, cette définition d'objets couvre un grand nombre d'éléments de l'environnement. Concrètement, on dénombre parmi les objets des formes aussi variées que des buissons, des haies, des arbres, des rochers, des monticules de rochers, des bâtiments, etc... De surcroît ceux-ci n'ont pu être perçus que partiellement. Si l'on s'intéresse à la localisation du robot, il apparaît de manière évidente qu'il faut procéder à une deuxième sélection pour ne prendre en compte que l'information utile et sûre. Nous abordons alors la nécessité de la recherche d'amers. Cette notion est essentielle pour la robustesse de la navigation du robot.

III.3 Des objets particuliers : les amers

Nous avons vu dans la partie II.1.1 que le robot doit trouver dans l'environnement des points de repère particuliers qui lui permettront de se localiser. Ces points de repère sont désignés sous le terme d'**amers**. La question maintenant posée est de déterminer comment le robot les choisira? Nous donnons dans cette section quelques éléments de réponse.

Un amer est un élément remarquable ou une configuration de tels éléments (par exemple un alignement). Il doit avoir des propriétés qui permettent de localiser le robot dans l'environnement. On recense notamment les deux propriétés suivantes [Betgé-Brezetz 96] :

- *Distinction*: un amer doit être facile à distinguer de ce qui est autour de lui dans l'environnement.
- *Précision*: On doit pouvoir extraire d'un amer des attributs précis pour réduire l'incertitude sur la position du robot.

Notons que ces deux propriétés sont locales. Aussi, nous devons associer à chaque amer une région, appelée **région de validité**, dans laquelle ces propriétés sont vérifiées.

D'autre part, parmi les amers possibles, un sous-ensemble particulier pourrait être sélectionné sur la base des critères suivants :

- la précision demandée sur la position du robot ;
- la répartition des amers dans le but de couvrir l'ensemble de l'environnement (ce qui se réfère directement à la notion de région de validité).

A titre d'exemple, dans le cas d'une région où beaucoup d'amers ont été perçus, seul le meilleur sous-ensemble sera conservé. Dans le cas inverse d'une région possédant peu d'amers, la sélection de ceux-ci se fera plus tolérante. Cette seconde étape ne pourra être réalisée qu'une fois l'environnement suffisamment modélisé.

En ce qui concerne notre approche, les amers sont choisis parmi les objets que le robot aura détectés dans l'environnement. Nous montrerons comment ces amers seront sélectionnés, et comment nous en extrairons une caractéristique qui permettra de réestimer la position du robot.

III.4 Démarche de la construction du modèle par objets

Ayant introduit les différentes notions de sol, d'objets et d'amers; nous proposons maintenant la démarche suivante pour mettre en œuvre l'approche de modélisation par objets de l'environnement.

Dans un premier temps nous chercherons à identifier le sol et les objets dans une perception. Nous établirons pour cela une procédure de segmentation d'image tridimensionnelle qui sera bien évidemment basée sur les définitions précédentes.

Une fois les objets et le sol segmentés, nous chercherons à les représenter par un modèle approprié. Dans nos travaux, nous nous sommes plus particulièrement focalisés sur les objets pour lesquels nous avons développé deux types de modèles.

Le premier, appelé **modèle géométrique**, décrit grossièrement leur forme, et éventuellement des caractéristiques - précises - qui pourraient être extraites de leur surface. Ce modèle est calculé indépendamment pour chaque objet.

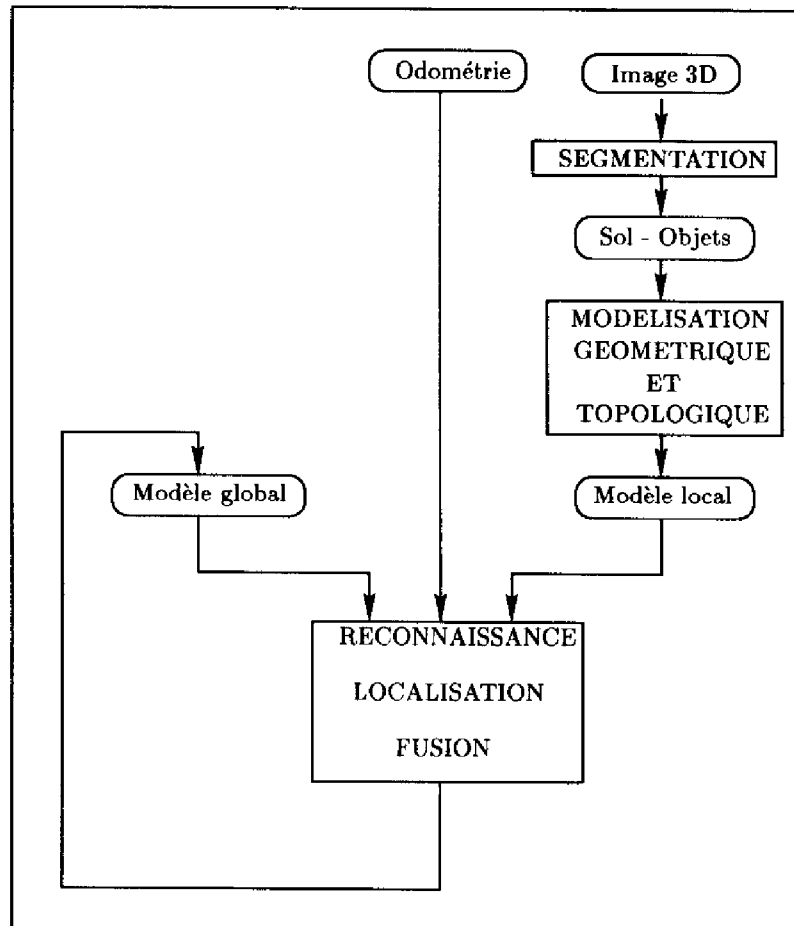
Le second modèle, appelé **modèle topologique**, décrit la disposition des objets dans la scène, et ce, sans prendre en compte leur forme. Il repose sur la notion de **relations topologiques** entre objets et permet de donner une description qualitative des objets dans la scène. Cette dernière sera alors représentée par un graphe dont les sommets sont des objets, et les arcs des relations topologiques.

Ces deux modèles sont ensuite regroupés pour former le **modèle local** (d'une scène). Les amers sont ensuite sélectionnés parmi les objets sur la base des critères établis précédemment, et à partir d'informations géométriques et topologiques.

Le robot établit un modèle local pour chaque perception. Le **modèle global** de l'environnement est construit par la fusion itérative des différents modèles locaux. Ce processus nécessite les trois fonctionnalités suivantes :

- la reconnaissance des objets du modèle local dans le modèle global déjà construit ;
- le calcul de la position du robot à partir de amers reconnus ;
- la mise à jour des modèles géométrique et topologique des objets.

La figure III.4 résume l'ensemble de ces opérations en présentant séquentiellement les différentes étapes de l'approche de modélisation de l'environnement par objets. Au cours de ce mémoire, nous reviendrons successivement sur chacune d'elles, en décrivant les différentes méthodes qui ont été élaborées et mises en œuvre.

FIG. III.4 – *Algorithme de modélisation incrémentale.*

Deuxième Partie

Segmentation et modélisation de scène



Chapitre IV

Segmentation d'une scène

Nous avons défini précédemment quelles étaient les différentes parties élémentaires d'une scène (*i.e.* le sol, les objets et le fond). Considérons maintenant une image tridimensionnelle d'une scène fournie par un capteur de type caméra laser ou stéréovision. Se pose alors le problème de la **segmentation** qui consiste à identifier ces différentes parties dans l'image. Cette opération est réalisée en deux temps : le robot extrait d'abord le sol, puis segmente ensuite les différents objets entre eux. Les méthodes que nous allons proposer pour l'extraction du sol et des objets font évidemment appel aux définitions établies au chapitre précédent.

IV.1 Les capteurs tridimensionnels

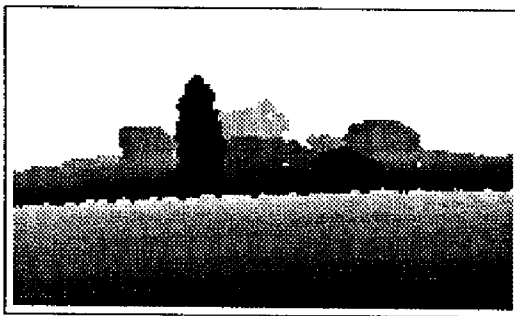
Avant tout, intéressons-nous aux données des capteurs. D'une manière générale, un capteur tridimensionnel peut être considéré comme un système fournissant un tableau de taille $N_l \times N_c$ de points 3D exprimés en coordonnées cartésiennes dans le **repère capteur** R_c . Un tel tableau est appelé **image tridimensionnelle** et on le représente par :

$$\mathcal{I} = \{P_{ij} = [x_{ij} \ y_{ij} \ z_{ij}]^T \text{ avec } i = 1..N_l \text{ et } j = 1..N_c\} \quad (\text{IV.1})$$

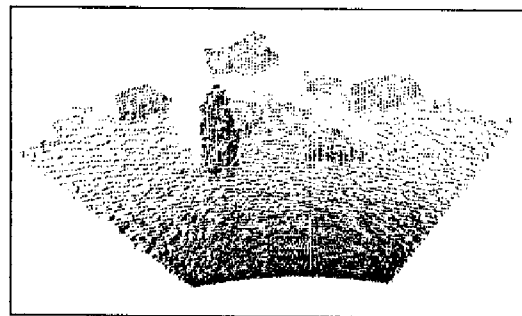
N_l et N_c désignent le nombre de lignes et de colonnes de l'image \mathcal{I} . i et j déterminent la position du pixel dans l'image (numéro de la ligne et de la colonne). De plus, on possède pour chaque point P_{ij} de l'image une matrice d'incertitude σ_{ij} de taille 3×3 (calculée à partir du modèle du capteur). Enfin, si l'on dispose d'une estimée fiable de la verticale (à l'aide de deux inclinomètres ou d'une centrale inertielle) on peut définir le **repère redressé** R_{red} dont l'origine est celle du repère capteur R_c et l'axe z_{red} est dirigé suivant la verticale. Les points 3D peuvent alors être exprimés dans l'un ou l'autre de ces deux repères.



(a)



(b)



(c)

FIG. IV.1 – Image vidéo (a), image polaire (b) et image tridimensionnelle de la scène "cailloux". Dans l'image polaire (b), la distance est codée en niveaux de gris (variant périodiquement de 0 à 255).

Les capteurs laser

Les capteurs laser ne délivrent pas directement ce type d'image, mais une image en coordonnées polaires. Celle-ci consiste en un ensemble de distances ρ_{ij} mesurées pour différentes valeurs d'azimut θ_i et de site φ_j du faisceau ($i=1, N_i$; $j=1, N_c$). L'image tridimensionnelle est calculée à partir de cette image polaire et d'un modèle du capteur [Nashashibi 93a]. L'incertitude des coordonnées 3D du point cartésien P_{ij} est calculée par propagation, à travers ce modèle, des incertitudes de ρ_{ij} , θ_i , φ_j et de celles des paramètres intrinsèques du capteur. La figure IV.1 représente une scène, l'image polaire et l'image tridimensionnelle correspondante.

Les capteurs stéréoscopiques

La technique de stéréovision par corrélation permet de calculer une image tridimensionnelle à partir d'une paire de caméras vidéo en appariant des pixels des images. La mise en correspondance est facilitée par une procédure de rectification [Ayache 88b] qui permet

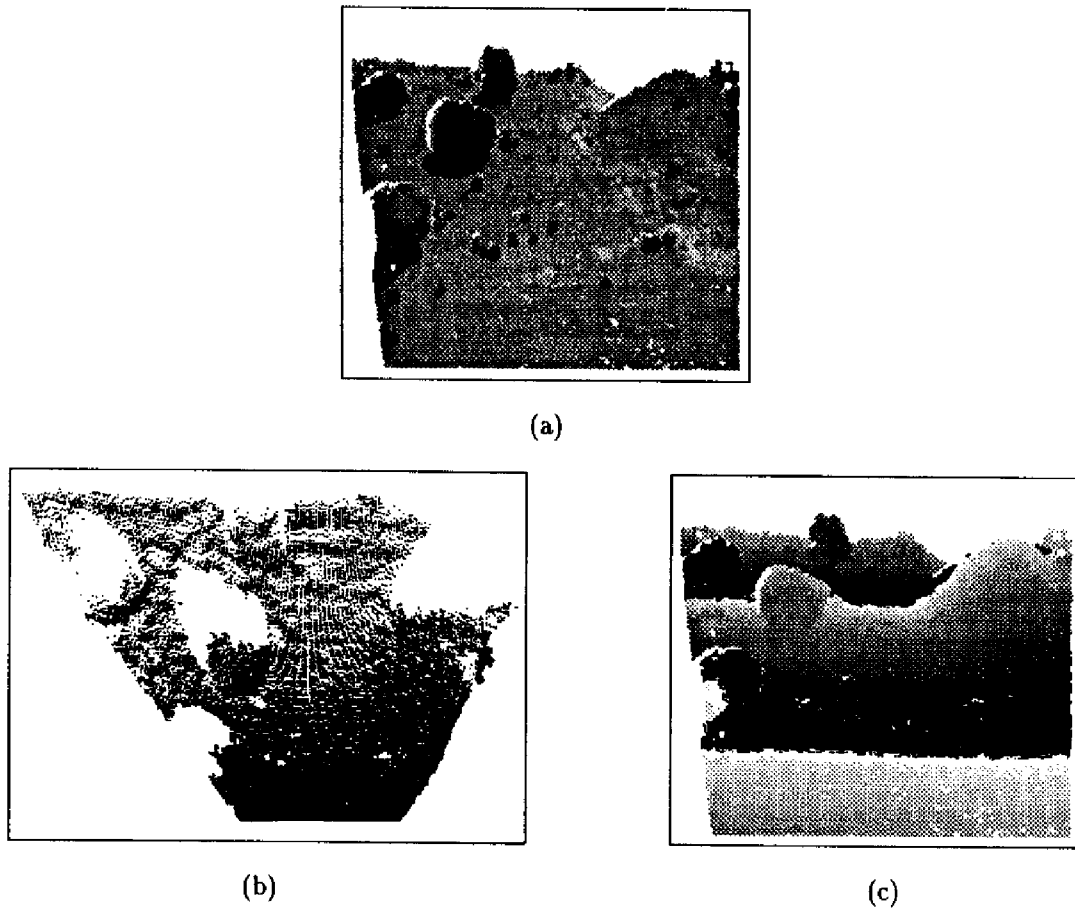


FIG. IV.2 – Image des pixels corrélés (a), image tridimensionnelle (b) et image de distance (c) obtenue par stéréovision. Dans l'image de distance (c), la distance est codée en niveaux de gris (variant périodiquement de 0 à 255).

de limiter la recherche d'appariement d'un pixel à une seule ligne de l'autre image. Pour chaque appariement, l'algorithme calcule une valeur de disparité qui représente la distance entre les deux pixels. Les coordonnées des points 3D sont alors calculées en fonction de cette distance et de la position en ligne et colonne des pixels. La figure IV.2 montre une image prise sur le site Geroms par la caméra gauche d'un système stéréoscopique et l'image de points 3D calculée au CNES [Hotz 91]. On peut aussi calculer une image des distances des points au capteur (IV.2 (c)) où la distance est codée en niveau de gris.

Intéressons-nous maintenant à l'extraction des objets et du sol à partir de l'image tridimensionnelle.

IV.2 Extraction du sol

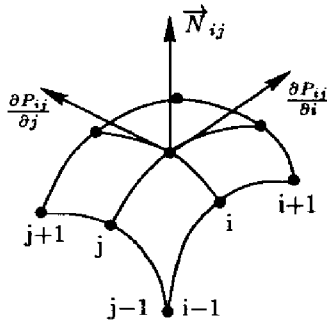


FIG. IV.3 – Principe de l'évaluation des normales

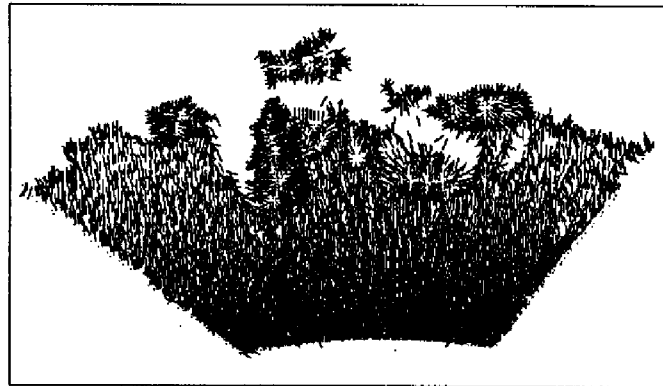


FIG. IV.4 – Résultat de l'estimation des normales sur l'image "cailloux".

D'après la définition donnée du sol (cf définition 4), nous devons chercher dans la scène une grande surface d'orientation uniforme. Celle-ci correspond alors à un ensemble de points ayant tous des directions de normale équivalentes. Nous allons donc calculer les normales en chaque point. Considérons pour cela l'image tridimensionnelle comme une surface définie par une équation paramétrique. Cette équation correspond à la forme générique d'une image 3D dont la formule IV.1 fait état et où i et j sont considérés comme les paramètres. On peut calculer la normale \vec{N}_{ij} au point P_{ij} de la manière suivante [Boumaza 94, Carmo 76]:

$$\vec{N}_{ij} = \frac{\partial P_{ij}/\partial i \wedge \partial P_{ij}/\partial j}{\|\partial P_{ij}/\partial i \wedge \partial P_{ij}/\partial j\|} \quad (\text{IV.2})$$

avec :

$$\frac{\partial P_{ij}}{\partial i} = \begin{bmatrix} \frac{\partial x_{ij}}{\partial i} \\ \frac{\partial y_{ij}}{\partial i} \\ \frac{\partial z_{ij}}{\partial i} \end{bmatrix} \quad \text{et} \quad \frac{\partial P_{ij}}{\partial j} = \begin{bmatrix} \frac{\partial x_{ij}}{\partial j} \\ \frac{\partial y_{ij}}{\partial j} \\ \frac{\partial z_{ij}}{\partial j} \end{bmatrix} \quad (\text{IV.3})$$

Le calcul de la dérivée de chaque coordonnée est simplement réalisé à l'aide d'un filtre de Sobel de taille 3×3 . Il existe d'autres méthodes plus précises d'approximation de normales [Yokoya 89] mais elles sont plus coûteuses en temps de calcul. De plus, la précision obtenue par la méthode que nous présentons se révèle tout à fait satisfaisante pour notre application. Le résultat de l'estimation des normales appliqué à l'image 3D "cailloux" est montré figure IV.4. Chaque normale y est représentée comme une aiguille. Leur orientation dépend du produit vectoriel et a été choisie de telle sorte que toutes les normales \vec{N}_{ij} soient dans la direction du capteur.

IV.2.1 Histogramme des normales

Notons plus généralement $\vec{N}_P = [N_x \ N_y \ N_z]^T$ la normale en un point P . Le sol correspondant à des points ayant des normales équivalentes, nous exprimons la direction de chacune d'elle dans un système de coordonnées sphériques, soit $\vec{N}_P \equiv (\alpha, \beta) \in [-\pi; \pi] \times [-\frac{\pi}{2}; \frac{\pi}{2}]^1$.

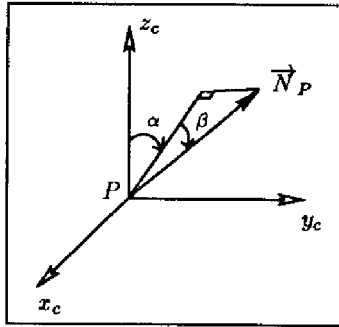


FIG. IV.5 - Système de coordonnées sphériques utilisé pour extraire le sol :

$$\begin{aligned} \alpha &= \text{atan2}(N_y, N_x) \\ \beta &= \arcsin N_z \end{aligned} \quad (\text{IV.4})$$

Le système de coordonnées sphériques utilisé est montré figure IV.5. Il est tel que les coordonnées $(\alpha, \beta) = (0, 0)$ correspondent à des normales dirigées suivant l'axe z_c du repère capteur.

Construction de l'histogramme

A la manière de la transformée de Hough ou de la méthode développée dans [Laurendeau 86], nous calculons l'histogramme bi-variable H des normales qui représente le nombre de points ayant une direction de normale donnée :

$$H(\alpha, \beta) = \text{Card} \{ P \mid \vec{N}_P \equiv (\alpha, \beta) \} \quad (\text{IV.5})$$

Cet histogramme des normales est construit sur l'ensemble des points P_{ij} ($i=1, N_i; j=1, N_c$) de l'image. Pour cela, nous discrétisons chacun des intervalles angulaires $[-\pi; \pi]$ et $[-\frac{\pi}{2}; \frac{\pi}{2}]$ respectivement en n_α et n_β sous-intervalles. Ces deux paramètres sont fixés de telle sorte qu'ils permettent de distinguer des normales dont la direction est au moins supérieure à 10 degrés, l'écart angulaire entre la direction des normales du sol et celle des objets étant d'au moins une quarantaine de degrés. L'extraction du sol dépend peu des paramètres n_α et n_β comme l'ont montré les nombreux essais que nous avons faits sur des sites différents.

Pour une scène composée d'un sol uniforme, la prédominance de ce dernier apparaît comme un pic dans l'histogramme H des normales. C'est le cas pour la scène "cailloux" dont la figure IV.6 montre l'histogramme H correspondant. Notons que les zones proches du capteur sont plus denses en points que les zones éloignées ce qui favorise dans ce cas la prédominance du sol.

1. En réalité, l'ensemble des valeurs angulaires doit être de la forme $[-\pi; \pi] \times [-\frac{\pi}{2}; \frac{\pi}{2}] \cup \{0, \frac{\pi}{2}\} \cup \{0, -\frac{\pi}{2}\}$ pour que α et β soient déterminés de manière unique ; mais cela ne change en rien la suite de notre exposé.

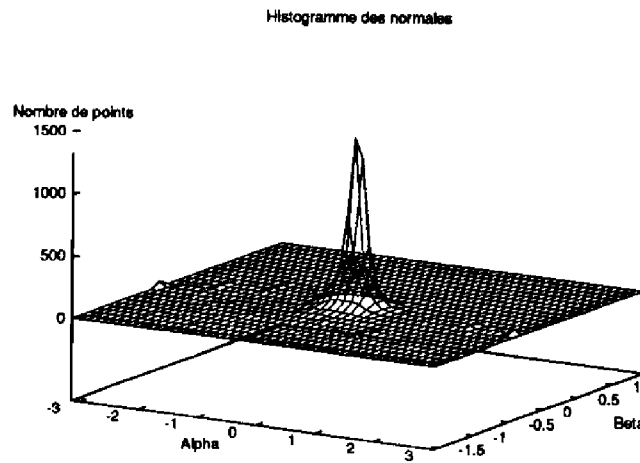


FIG. IV.6 – Histogramme bi-variable H des normales. Le sol étant majoritaire et uniforme, il apparaît comme un pic dans l'histogramme.

De plus, dans le cas où les points P_{ij} sont exprimés dans le repère redressé R_{red} la contrainte d'horizontalité du sol peut être prise en compte en vérifiant que le pic se situe à l'intérieur d'un rectangle $[-\alpha_{Horiz}, \alpha_{Horiz}] \times [-\beta_{Horiz}, \beta_{Horiz}]$ fixé.

Par ailleurs, le système de coordonnées sphérique montré figure IV.5 ne conviendrait pas pour détecter des surfaces perpendiculaires à la direction x_c . En effet, pour de telles surfaces, de petites variations des normales autour de x_c peuvent faire varier l'angle α de $-\pi$ à π (l'angle β restant quant à lui à des valeurs proches de $\frac{\pi}{2}$). Il n'apparaîtra dans ce cas pas de pic significatif dans l'histogramme H car les points seront repartis tout au long de l'axe α . Pour détecter ces surfaces (*i.e.* pour les faire apparaître comme un pic dans l'histogramme des normales) il convient d'utiliser le système de coordonnées sphériques montré figure IV.7. L'histogramme H dès lors construit sera tel qu'une surface perpendiculaire à l'axe x_c formera un pic en $(0, 0)$. Nous verrons par la suite à quelle fin ce système de coordonnées sera utilisé.

Sélection des points du pic

Une procédure récursive partant du sommet H_{Max} de l'histogramme et descendant progressivement va sélectionner les points 3D correspondant à ce pic. Appelons α_u et β_v les deux variables angulaires de l'histogramme ($1 \leq u \leq n_\alpha$; $1 \leq v \leq n_\beta$), et α_U et β_V celles du sommet (*i.e.* $H_{Max} = H(\alpha_U, \beta_V)$). La figure IV.8 détaille cette procédure qui est appelée par : **sélectionne_pic** (U, V, H_{Max}).

Quatre critères en fixent l'arrêt :

- Les limites de l'histogramme suivant l'axe β (*i.e.* $1 \leq v \leq n_\beta$). En revanche, il n'y a

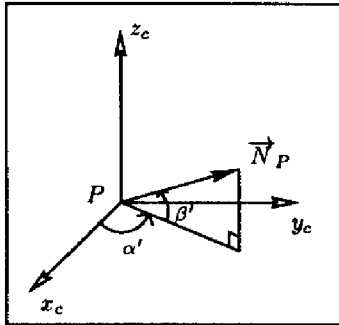


FIG. IV.7 – *Système de coordonnées sphériques utilisé pour extraire des surfaces perpendiculaires à la direction de visée :*

$$\begin{aligned}\alpha' &= \text{atan2}(N_y, N_x) \\ \beta' &= \arcsin N_z\end{aligned}\quad (\text{IV.6})$$

pas de limite suivant l'axe α car pour une valeur de β fixé, les normales ayant une valeur en α proche de π ont des orientations équivalentes à celles dont la valeur en α est proche de $-\pi$. Par conséquent, lorsque l'on est au bord $\alpha \simeq \pi$, il convient alors de poursuivre la recherche par le bord $\alpha \simeq -\pi$, et inversement (cf les deux tests de l'algorithme).

- Les cellules $H(\alpha_u, \beta_v)$ déjà étudiées.
- Lors de la "descente" du pic on ne tolère pas de pouvoir "remonter" (i.e. la valeur $H(\alpha_u, \beta_v)$ courante doit être inférieure à la valeur précédente $H_{P_{itéc}}$). Si cela n'était pas le cas, ce phénomène indiquerait la présence d'un autre pic (et donc peut être d'une autre partie du sol). Nous reviendrons sur ce point dans la section IV.2.3.
- Une limite inférieure sur la taille $H(\alpha_u, \beta_v)$. En mettant une limite sur $H(\alpha_u, \beta_v)$ et non sur α_u et β_v , on n'impose pas de borne sur les orientations possibles du sol. On s'assure ainsi que des sols très courbés seront aussi détectés.

IV.2.2 Histogramme des distances au sol

Tous les points sélectionnés dans le pic de l'histogramme H précédent ont des directions de normales équivalentes (le pic de H correspond en réalité à une direction vectorielle). La majeure partie d'entre eux appartient au sol, mais certains ont pu être pris sur des surfaces externes à ce sol mais de même orientation, le sommet des objets par exemple. Pour sélectionner seulement les points appartenant réellement au sol, nous procédons de la manière suivante :

Nous calculons d'abord l'équation d'un plan \mathcal{P} passant par l'origine du capteur et perpendiculaire à la direction vectorielle correspondant au pic. Nous construisons ensuite un histogramme mono-variable H' des distances au sol. Il représente le nombre de points sélectionnés précédemment et situés à une distance d du plan \mathcal{P} :

$$H'(d) = \text{Card} \{P \text{ sélectionnés} \mid \text{dist}(P, \mathcal{P}) = d\} \quad (\text{IV.7})$$

Pour le construire, nous discrétisons l'espace des distances d'un pas proportionnel à l'incertitude moyenne des points. La figure IV.9 représente l'histogramme H' calculé à

```

Procédure sélectionne_pic( $u, v, H_{Préc}$ )

  /* Conditions d'arrêt */
  Si ( $\beta_v$  est hors des limites de  $H$ ) OU ( $H(\alpha_u, \beta_v)$  a déjà été étudiée) OU
    ( $H(\alpha_u, \beta_v) \geq H_{Préc}$ ) OU ( $H(\alpha_u, \beta_v) \leq .05 \times H_{Max}$ ) alors
    Fin de la procédure

  Sinon
    /* Ajout des points de la cellule à l'ensemble des points du sol */
    Points du sol  $\leftarrow \{ P_{ij} \text{ appartenant à } H(\alpha_u, \beta_v) \}$ 

    /* Recherche dans les cellules voisines */
    Pour  $(\Delta_u, \Delta_v) \in \{-1, 0, 1\}^2 \setminus (0, 0)$ 

      /* Test pour détecter si on est sur le bord  $\alpha \simeq -\pi$  */
      Si  $(u + \Delta_u = 0)$ 
        sélectionne_pic( $n_\alpha, v + \Delta_v, H(\alpha_u, \beta_v)$ )

      /* Test pour détecter si on est sur le bord  $\alpha \simeq +\pi$  */
      Si  $(u + \Delta_u = n_\alpha + 1)$ 
        sélectionne_pic( $0, v + \Delta_v, H(\alpha_u, \beta_v)$ )

    sélectionne_pic( $u + \Delta_u, v + \Delta_v, H(\alpha_u, \beta_v)$ )

```

FIG. IV.8 – Procédure récursive de sélection des points d'un pic de l'histogramme.

partir de l'histogramme des normales précédent (cf figure IV.6). Les points appartenant au sol apparaîtront comme un pic important dans l'histogramme H' et une même procédure récursive est appliquée pour les sélectionner. De petits artefacts du sol peuvent éventuellement ne pas appartenir à ce pic, mais cela peut être résolu en intégrant au sol les points dont la distance à ce dernier est inférieure à une certaine valeur (fonction de l'erreur moyenne des points).

Le résultat final de l'extraction des points du sol de l'image "cailloux" est montré ainsi figure IV.10.

IV.2.3 Analyse des histogrammes

Cette méthode permet aussi de traiter des cas plus complexes où le sol serait composé de plusieurs régions (satisfaisant chacune la propriété d'orientation uniforme). Ces régions apparaîtront comme d'autres pics soit dans l'histogramme H (si les orientations moyennes de chacune sont différentes), soit dans l'histogramme H' (si ces orientations moyennes sont équivalentes). L'extraction du sol est réalisée de la même manière.

Nous illustrons cela par une image prise sur une région structurée composée de trois parties différentes (cf figure IV.11 (a)). L'histogramme H des normales est montré figure

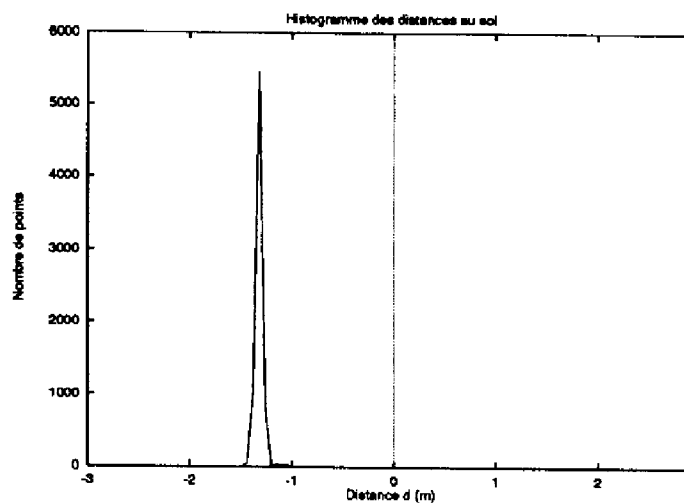


FIG. IV.9 – Histogramme mono-variable H' . Le sol majoritaire apparaît à nouveau comme un pic.



FIG. IV.10 – Résultat de l'extraction du sol de la scène "cailloux".

IV.11 (b) où l'on voit apparaître deux pics. Le plus haut correspond aux deux régions horizontales et le plus petit à la surface inclinée. La figure IV.11 (c) montre ensuite l'histogramme H' construit à partir des points 3D du premier pic. On y observe à nouveau deux pics qui indiquent la présence de deux régions d'orientations équivalentes mais situées à des niveaux différents. Le plus haut pic de l'histogramme H' correspond à la région qui est proche du capteur. Ce pic est plus haut et plus étroit car la résolution des points et leur précision est plus importante pour cette région.

Nous soulignons que le sol est déterminé à l'aide de pics construits à partir de points situés n'importe où dans l'image. Aussi cette méthode peut tout à fait traiter du cas où les différentes parties du sol sont disjointes.

Par ailleurs, l'extraction du sol permet de déterminer son contour avec les objets. Notons que des méthodes utilisant les discontinuités d'élévation n'auraient pas été suffisantes pour détecter ces contours. En effet, la frontière entre les objets et le sol peut être très

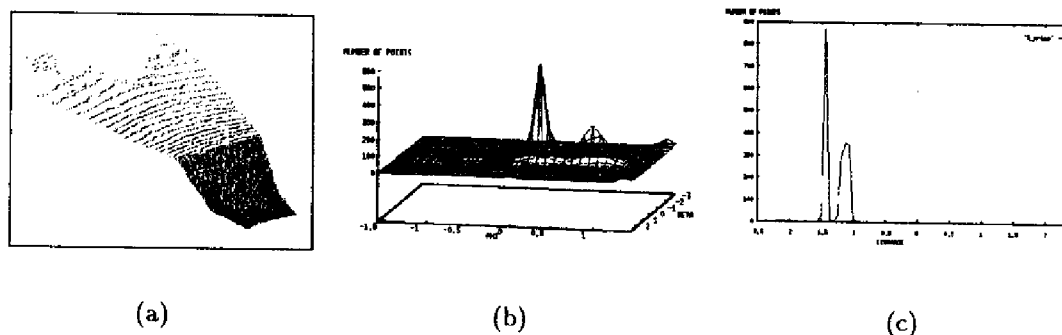


FIG. IV.11 – Histogramme d'une scène composée de différentes régions d'orientation uniforme. Image 3D (a), histogramme bi-variable H (b) et histogramme mono-variable H' calculé avec le plus grand pic de l'histogramme bi-variable (c).

lissée dans les environnements naturels.

Enfin nous pouvons vérifier que la scène se décrit bien en termes de sol et d'objets sur la base des critères suivants :

- *Nombre de pics*: la détection de plusieurs pics (dans H ou H') de hauteurs équivalentes indique que le sol est composé de plusieurs régions ; inversement, l'absence de pic significatif prévient qu'il sera difficile de différencier le sol des objets.
- *Taille de la base du pic*: plus le pic est fin, plus le sol est plan, autrement le sol est incurvé. Soulignons ici que le principe même de sélection du pic impose que le sol ait des normales de directions équivalentes, mais ne l'oblige **en aucun cas** à être plan.
- *L'erreur moyenne* estimée par l'approximation des points du sol avec une surface donnée (plan ou quadrique). Plus cette erreur est faible, plus le sol est uniforme.

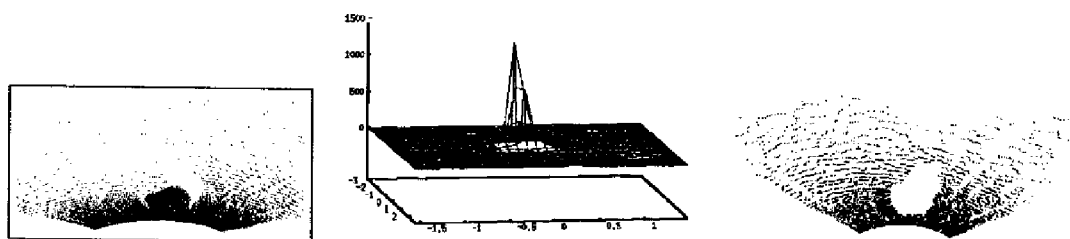


FIG. IV.12 – Extraction du sol pour une autre scène

Pour l'image "cailloux", nous avons obtenu un seul pic dans l'histogramme H avec une base de $0.65rd \times 0.35rd$. L'erreur moyenne de l'approximation de ses points à un plan est

de 1cm. Dans une autre scène naturelle montrée figure IV.12. nous avons aussi obtenu un seul pic dans l'histogramme H mais sa base est de $1.0rd \times 0.5rd$ et son erreur moyenne de 5cm. Dans ce cas le sol est aussi détecté, mais ces paramètres indiquent qu'il n'est pas aussi régulier que celui de l'image "cailloux".

IV.3 Extraction des objets

L'étape précédente nous a permis d'extraire le sol de la scène, et par conséquent d'identifier les contours entre le sol et les objets. Mais ce stade, les objets forment un tout indistinct et nous allons étudier dans cette section comment le robot peut les isoler les uns des autres.

IV.3.1 Identification des contours entre objets

Nous reprenons la définition des objets (définition 5) comme étant des parties connexes de matière. Une fois le sol identifié et si l'on ne considère que l'ensemble des objets, ces derniers apparaissent séparés les uns des autres par des espaces vides. Or, dans l'image polaire (ou image de distance), ces espaces se caractérisent par des discontinuités de profondeur qui forment le contour des objets. Nous allons ainsi calculer le gradient de l'image en lui appliquant un filtre dérivant. Les contours correspondront aux maxima du gradient. Le filtre de Deriche a fourni les meilleurs résultats pour détecter et localiser les contours des objets ; de plus, un filtrage par hystérésis permet d'éliminer les contours non significatifs [Monga 90]. Cette recherche de contours a été appliquée à l'image polaire IV.1 (b), et après avoir extrait le sol, le résultat est montré figure IV.13.

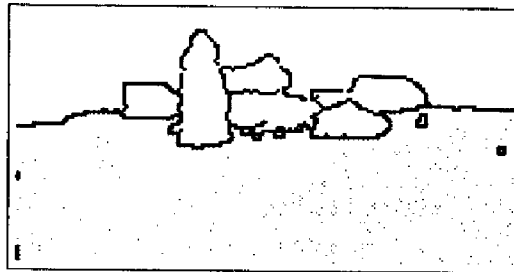


FIG. IV.13 – Image des contours. En gris : les pixels appartenant au sol.

Des problèmes de fermeture de contours se posent bien évidemment, mais peuvent être en partie résolus en utilisant un autre filtre opérant en quatre étapes.

▷ Tout d'abord les pixels correspondant à des Extrémités de Contour à Fermer, notés ECF, sont détectés dans l'image de contours par l'application d'un simple masque morphologique tournant de taille 3×3 (seulement cinq configurations sont caractérisées).

▷ Puis, pour chaque ECF, nous recherchons l'ensemble \mathcal{L} des autres pixels de contour qui sont à l'intérieur d'un disque centré sur lui et appelé **disque de recherche** (cf figure IV.14 (a)). Ces pixels sont les éventuels candidats qui nous permettront de fermer ce contour.

▷ Nous devons bien évidemment éviter de fermer le contour "sur lui même" (par exemple avec le pixel noté A dans la figure IV.14 (a)). Pour cela, nous éliminons de \mathcal{L} tous les pixels qui pourraient être connectés par un contour à l'ECF considéré, à condition que ce contour soit totalement inclus dans un deuxième disque appelé **disque de connexion**. Les candidats restant forment alors un ensemble \mathcal{L}' représenté en gris clair dans la figure IV.14 (b).

▷ Finalement, en utilisant les données 3D associées à chaque pixel, nous sélectionnons parmi les éléments de \mathcal{L}' le pixel dont la position 3D est la plus proche de celle de l'ECF. Nous traçons alors une ligne droite 2D dans l'image de distance pour fermer le contour. Cette droite 2D ne correspond à aucune information 3D, elle est simplement utilisée pour segmenter les objets dans l'image tridimensionnelle.

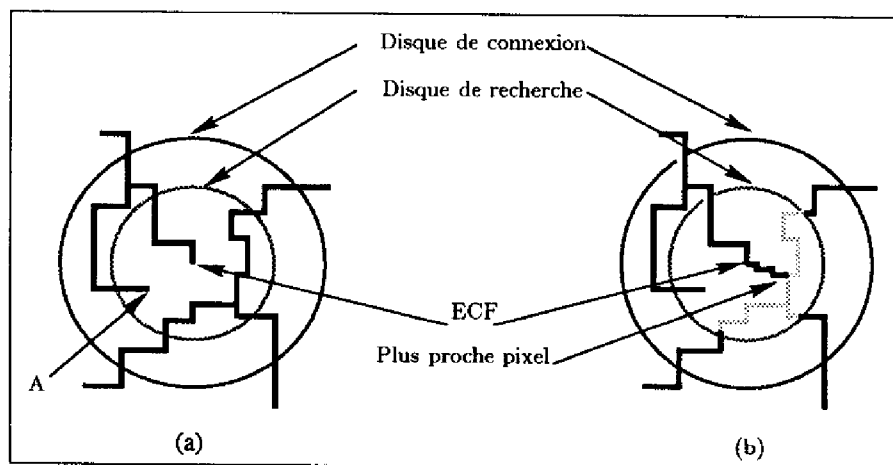


FIG. IV.14 – Régions considérées pour la fermeture de contours. En clair dans la figure (b): ensemble \mathcal{L}' des pixels candidats.

Cette méthode permet de fermer des contours sans privilégier toutefois de direction particulière, cela présente une certaine importance dans la mesure où les objets ont des formes quelconques. Elle demande de surcroît peu de temps de calcul car l'algorithme est linéaire en fonction du nombre de points dans l'image.

IV.3.2 Identification des objets

Une fois cette procédure achevée, les objets sont alors simplement définis par les régions fermées. Un simple algorithme de grossissement de régions partant d'un pixel situé à l'intérieur d'un objet permet d'en sélectionner les points.

Remarquons toutefois que les contours des objets ont été déterminés par une recherche de maxima de gradient sur l'image de distance (à l'aide du filtre de Deriche). Or l'échantillonnage des points varie avec la distance dans une image tridimensionnelle, et des discontinuités de profondeur peuvent apparaître artificiellement sur des objets situés loin

du robot. Pour remédier à ce problème, nous étudions tous les objets ayant un contour commun. Nous calculons pour ceux-là l'écart moyen entre les points situés de part et d'autre du contour et nous estimons alors, en fonction de leur distance, si les deux objets correspondant doivent être regroupés.

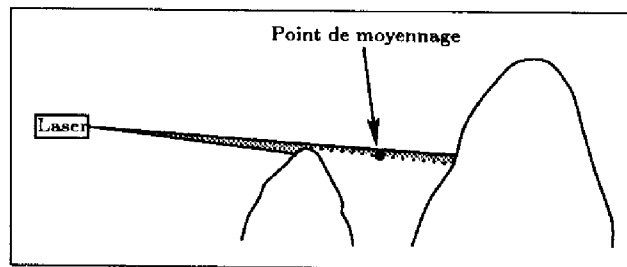


FIG. IV.15 – *Point de moyennage : point artificiel pris entre deux objets.*

Mentionnons aussi à ce niveau le problème des points de moyennage (qui ne concerne que les images tridimensionnelles issues d'un capteur laser). Ces points de moyennage sont obtenus artificiellement lorsque le faisceau est réfléchi en partie sur deux objets (cf figure IV.15). La valeur de distance, qui résulte d'un calcul de moyenne, ne correspond alors à aucune surface réelle. Pour éviter d'altérer les objets par ce phénomène, nous choisissons de ne pas affecter les points de contours aux objets.

La figure IV.16 montre les objets extraits avec un niveau de gris associé à chacun. A partir des pixels étiquetés de cette image, nous construisons la liste des points 3D appartenant à chaque objet.

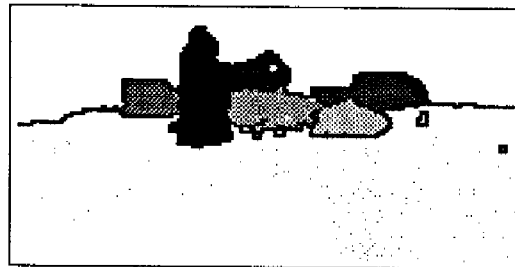


FIG. IV.16 – *Résultat de l'extraction des objets de la scène "cailloux".*

IV.4 Conclusion

Nous avons présenté dans ce chapitre une méthode de segmentation d'image tridimensionnelle pour les scènes naturelles. Cette technique est basée sur les notions de sol et

d'objets introduites au chapitre précédent. Nous l'avons appliquée avec différents types de capteurs et dans des scènes très diverses.

Nous montrons par exemple figure IV.18 (a) une image vidéo du terrain d'expérimentation EDEN du LAAS. Le robot Adam a pris sur ce terrain une image tridimensionnelle montrée IV.18 (b) dont le résultat de segmentation est donné figure IV.18 (c). La figure IV.19 (a) montre une autre partie de ce terrain où sont situés deux importants amas de rochers. La figure IV.19 (b) représente une image tridimensionnelle prise sur cette zone et la figure IV.19 (c) la segmentation correspondante. Les deux tas de rochers ont été perçus comme deux objets.

Comme nous le verrons par la suite, le robot Adam s'est aussi déplacé sur le site Geroms du CNES et des segmentations de ce terrain sont montrées figures VIII.6 (p. 119) et VIII.14 (p. 125).

En ce qui concerne les images 3D obtenues par stéréovision, nous montrons figure IV.17 la segmentation de l'image présentée figure IV.2. La figure IV.20 montre la segmentation d'une autre image tridimensionnelle stéréoscopique. Le robot a détecté le rocher mais bien évidemment aussi la faille et la fosse en arrière. Nous étudierons dans le prochain chapitre comment ces deux derniers "objets" seront identifiés en tant que "trous".

Enfin, nous verrons que cette méthode peut aussi s'étendre au cas des scènes d'intérieur composées d'objets (cf figure V.19, p. 70), un exemple de segmentation est montré figure V.21 (p. 70).

Ces différents résultats permettent d'apprécier la **robustesse** de l'algorithme de segmentation, propriété essentielle pour notre approche de modélisation par objets.

Le temps utilisé pour le calcul des normales, la construction des histogrammes, la recherche du sol, l'estimation des contours et l'identification des objets est de l'ordre de 2.5s sur une station Sparc 10. Le regroupement éventuel des objets ayant un contour commun (cf section IV.3.2) est en revanche plus coûteux (10s), mais cet algorithme pourrait être optimisé ou remplacé par des techniques de filtrage adaptatif.

Enfin cette méthode a l'avantage de ne faire aucune hypothèse sur la forme des objets. De plus, la technique de sélection du sol n'impose en aucun cas que celui-ci soit plan. Cette méthode de segmentation est ainsi tout à fait adaptée pour les environnements naturels.

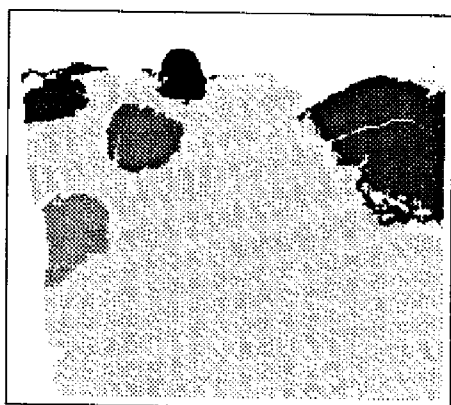


FIG. IV.17 – Segmentation de l'image tridimensionnelle obtenue par stéréovision et montrée figure IV.2 (p. 37).

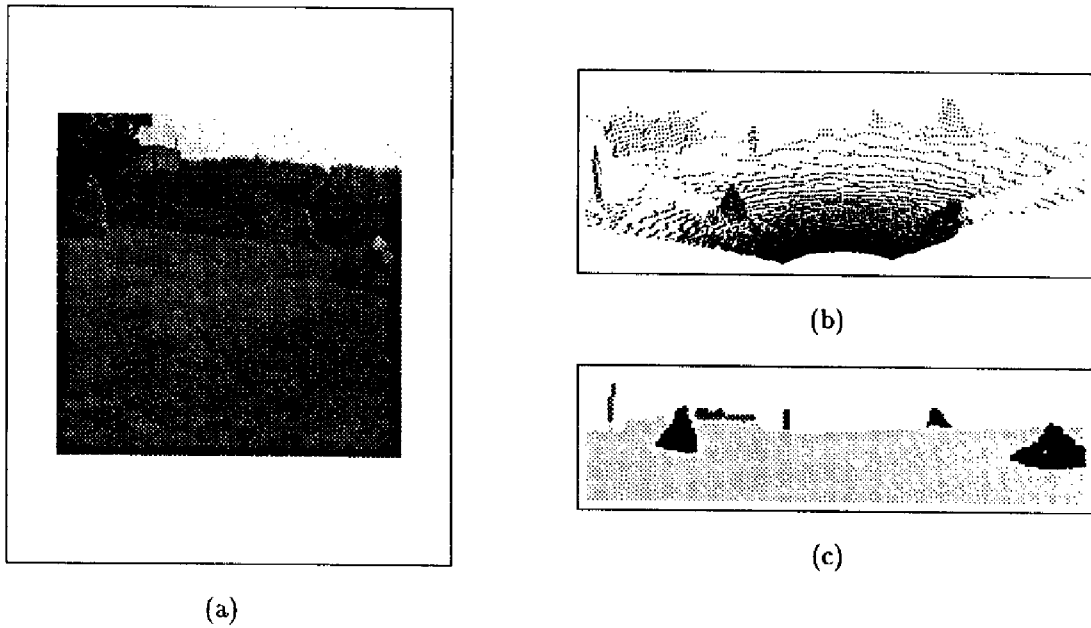


FIG. IV.18 – Image vidéo (a), image tridimensionnelle obtenue par le capteur laser d'Adam (b) et segmentation (c) dans le cas du terrain d'expérimentation EDEN du LAAS.

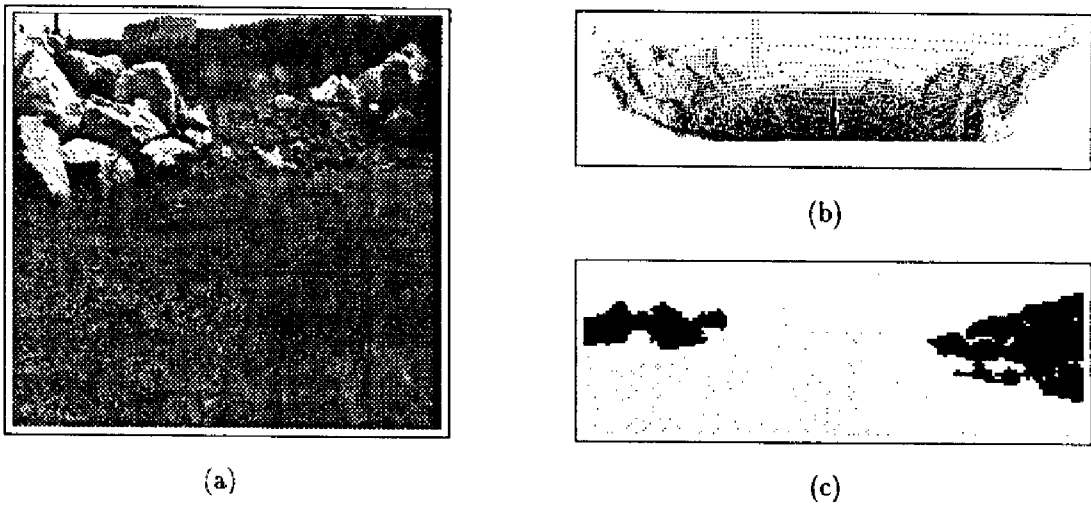


FIG. IV.19 – Image vidéo (a), image tridimensionnelle obtenue par le capteur laser d'Adam (b) et segmentation (c) dans le cas d'une partie du terrain EDEN composée de deux amas de rochers (le point de vue de l'image vidéo n'est pas exactement celui de l'image tridimensionnelle).

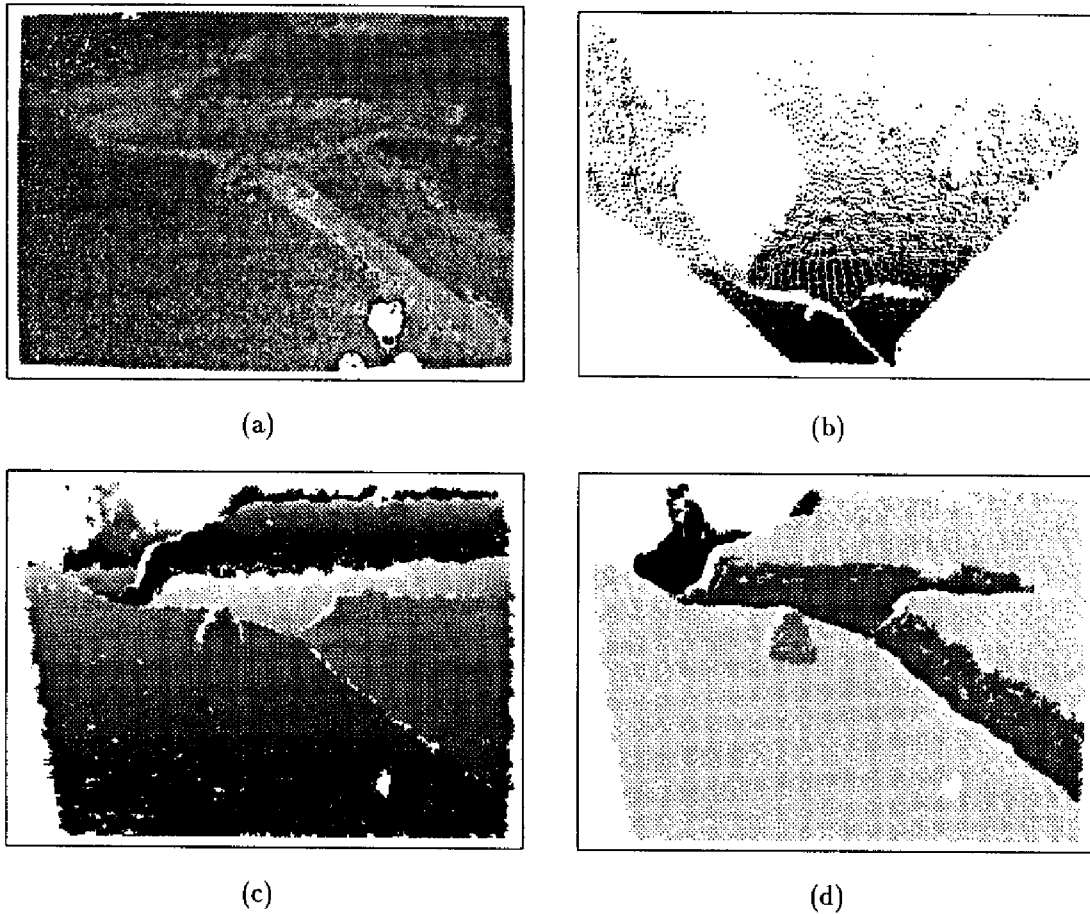


FIG. IV.20 – Image vidéo (a), image tridimensionnelle obtenue par stéréovision (b), image de distance (c) et segmentation (d) d'une partie du terrain d'expérimentation Geroms. Le rocher, mais aussi la faille et la fosse en arrière plan ont été détectés. Nous verrons au chapitre suivant que ces deux derniers "objets" particuliers seront appelés des "trous".

Chapitre V

Modélisation d'une scène

Les objets (et le sol) sont donc identifiés par la liste des points 3D qui ont été perçus à leur surface. Partant de ces données, nous allons maintenant établir un modèle de la scène comprenant deux types de représentation. La première consiste en un modèle géométrique qui caractérise la forme de chaque objet. Ce modèle géométrique se décompose lui-même en deux niveaux décrivant cette forme soit grossièrement, soit plus finement. La deuxième représentation, appelée modèle topologique, regroupe différentes informations relatives à la disposition des objets dans la scène. Il est basé sur la notion de relation topologique.

Nous reviendrons aussi dans ce chapitre sur la notion d'amers. Nous montrerons comment le robot les sélectionne dans la scène, et comment il extrait pour chacun d'eux une caractéristique (et son incertitude) qui lui permettra de réestimer sa position.

Nous présenterons aussi quelques résultats obtenus par extension dans des scènes intérieures.

V.1 Modélisation géométrique

V.1.1 Modèle géométrique grossier

Pour tous les objets, nous calculons un premier modèle simple qui caractérise de façon très sommaire leur forme. Ce modèle, qui intégrera ensuite les diverses perceptions de l'objet, permet de le décrire par :

- une estimée de sa position ;
- une estimée de son orientation ;
- une estimée de sa taille.

Plus précisément, le modèle grossier consiste en un ellipsoïde dont la position et l'orientation sont évaluées à partir du centre de gravité $G (X_G, Y_G, Z_G)$ et de la matrice d'inertie

M_{Inertie} du nuage de points $(x_i, y_i, z_i)_{i=1,n}$ de l'objet :

$$M_{\text{Inertie}} = \frac{1}{n} \sum_{i=1}^n [x_i - X_G, y_i - Y_G, z_i - Z_G]^T [x_i - X_G, y_i - Y_G, z_i - Z_G] \quad (\text{V.1})$$

Les axes de l'ellipsoïde correspondent aux directions des vecteurs propres de M_{Inertie} et s'intersectent au centre de gravité. Nous précisons que la matrice d'inertie sera conservée dans le modèle géométrique pour pouvoir réestimer ces axes lorsque l'objet sera à nouveau perçu.

La taille de l'ellipsoïde suivant chaque axe est estimée en fonction de la dispersion des points autour du centre de gravité. Elle est calculée indépendamment dans chaque direction de la manière suivante. Les points 3D sont d'abord projetés sur l'axe considéré. Ensuite nous construisons un histogramme représentant le pourcentage de points projetés dont la distance au centre est inférieure à une certaine valeur. La taille suivant cet axe est alors simplement calculée de telle sorte qu'elle inclut 90% des points (cf figure V.1). Cette méthode permet d'estimer simplement la dispersion des points suivant chaque axe séparément. La taille étant estimée sur un sous-ensemble des points, elle présente surtout l'avantage de ne pas être sensible à d'éventuels points erronés. La figure V.2 montre l'ellipsoïde correspondant au rocher le plus proche du capteur de la scène "cailloux".

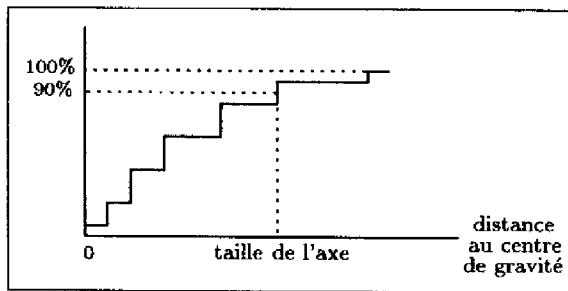


FIG. V.1 – Principe de l'estimation de la taille de l'ellipsoïde suivant un axe.

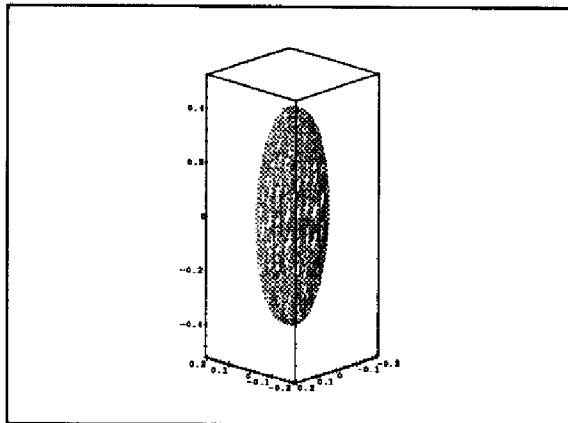


FIG. V.2 – Ellipsoïde (de taille 14.5cm x 33.5cm x 84cm) calculé pour le rocher le plus proche du capteur de la scène "cailloux".

Dans nos travaux, nous nous sommes focalisés sur la modélisation des objets de l'environnement. Nous avons vu que différentes représentations cohabitent à bord du robot, aussi la représentation du terrain revient plus spécifiquement à des modèles dédiés à la planification de trajectoires (bitmap, carte des régions, MNT ou B-splines). Toutefois, nous construisons un modèle très grossier du sol perçu, en le représentant simplement par un plan calculé par moindres carrés sur l'ensemble de ses points.

V.1.2 Modèle géométrique fin

L'ellipsoïde est une représentation très grossière d'un objet. Une représentation plus fine peut s'avérer nécessaire pour modéliser certains objets particuliers. Aussi, envisageons-nous un deuxième niveau dans le modèle géométrique qui comprendra une (ou des) représentation plus fine de la forme de l'objet. Cette représentation sera construite lorsqu'il y aura assez de points perçus sur l'objet et que ceux-ci seront suffisamment distribués sur sa surface.

Nous avons étudié la représentation superquadrique [Jouvencel 91, Pouzyreff 91, Solina 90, Barr 91] qui décrit les surfaces, dans leur repère propre, par l'équation implicite V.2 ou explicite V.3.

$$F(x, y, z) = \left[\left[\left| \frac{x}{a_1} \right|^{\frac{2}{\varepsilon_2}} + \left| \frac{y}{a_2} \right|^{\frac{2}{\varepsilon_2}} \right]^{\frac{\varepsilon_2}{\varepsilon_1}} + \left| \frac{z}{a_3} \right|^{\frac{2}{\varepsilon_1}} \right]^{\varepsilon_1} = 1 \quad (\text{V.2})$$

$$S(\eta, \omega) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_1 (\cos \eta)^{\varepsilon_1} (\cos \omega)^{\varepsilon_2} \\ a_2 (\cos \eta)^{\varepsilon_1} (\sin \omega)^{\varepsilon_2} \\ a_3 (\sin \eta)^{\varepsilon_1} \end{bmatrix} \quad \text{avec} \quad \begin{cases} \eta \in [-\frac{\pi}{2}, \frac{\pi}{2}] \\ \omega \in [-\pi, \pi] \end{cases} \quad (\text{V.3})$$

Les paramètres a_1 , a_2 et a_3 caractérisent la taille de la superquadrique et peuvent prendre des valeurs quelconques, tandis que les paramètres ε_1 et ε_2 expriment sa "carré-tude" respectivement dans le plan (x, y) et suivant l'axe z . Ils peuvent prendre des valeurs comprises entre 0 et 1 (0 correspond à une forme carré, 1 à une forme circulaire). D'autres paramètres comme la torsion, l'effilage ou l'inclinaison peuvent être ajoutés afin d'enrichir l'ensemble des formes représentables. Ces primitives, avec un nombre minimum de onze paramètres (six de position-orientation et cinq de forme) offrent une base intéressante pour la modélisation des nombreux objets naturels (rochers, arbres, ...). De plus ces paramètres ont un sens explicite caractérisant chacun un attribut géométrique particulier de l'objet. Enfin à plus long terme, des objets plus complexes pourraient être modélisés par une combinaison de superquadriques. D'autres primitives comme les cylindres généralisés semblent offrir *a priori* une variété plus limitée et ne sont pas représentables par une seule équation.

Dans notre application, nous avons pris en compte les onze premiers paramètres et deux d'effilage linéaire: K_x, K_y , $-1 \leq K_x, K_y \leq 1$. Les paramètres de torsion ou d'inclinaison n'ont pas été utilisés car ils sont moins explicites pour des objets naturels et sont plus difficiles à estimer. La formule V.4 donne alors l'équation de la surface.

$$\begin{aligned}
 X &= f_x(z)x & \text{où } f_x(z) &= \frac{K_x}{a_3}z + 1 \\
 Y &= f_y(z)y & \text{où } f_y(z) &= \frac{K_y}{a_3}z + 1 \\
 Z &= z
 \end{aligned}
 \tag{V.4}$$

Les différents paramètres sont calculés à l'aide d'une méthode de descente de gradient de Levenberg-Marquardt. Les valeurs initiales de position, d'orientation et de taille sont celles de l'ellipsoïde, les paramètres de carrétude et d'effilage sont respectivement fixés à 1 et 0. La fonction à minimiser est donnée par :

$$Q = \sqrt{a_1 a_2 a_3} |F' - 1| \tag{V.5}$$

F' est construite à partir de F et prend en compte l'effilage linéaire. Comme nous ne disposons pas nécessairement de points tout autour de la surface de l'objet, nous ponctuons le critère par le terme $\sqrt{a_1 a_2 a_3}$ qui assure une forme de volume minimale [Gupta 89].

Nous montrons figure V.3 la superquadrique obtenue à partir des 443 points 3D du premier objet de la scène "cailloux". Le temps de calcul est de 20s sur une station de travail Sparc 10.

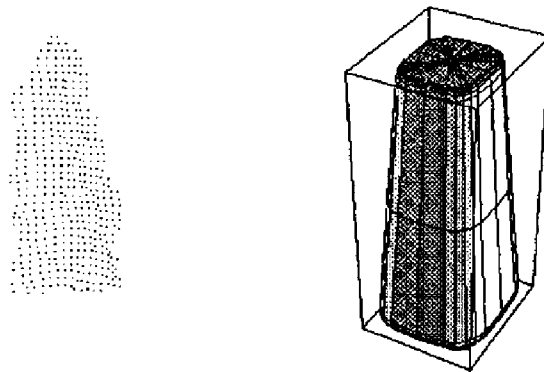


FIG. V.3 – Points 3D et superquadrique d'un objet de la scène "cailloux".

Les superquadriques sont des primitives volumiques qui permettent donc de représenter des *volumes fermés*. Or nous ne disposons que de vues partielles. Il est alors essentiel que ces vues apportent un maximum d'informations sur le pourtour de l'objet, et ce, pour d'éviter des problèmes de convergence de l'algorithme d'optimisation (malgré l'utilisation d'un critère de volume minimum).

On pourrait aussi attendre que suffisamment de points aient été perçus autour de l'objet pour calculer un modèle superquadrique. Nous proposons un moyen permettant d'apprécier sommairement la proportion perçue du pourtour de sa surface : les directions d'observation.

Directions d'observation

Pour chaque perception d'un objet, nous cherchons la direction selon laquelle il a été perçu. Pour cela nous calculons la droite (du plan horizontal) qui passe par le centre du capteur et par le centre de gravité des points de la perception. La direction d'observation correspond à l'angle que forme cette droite avec l'axe x_c du capteur. Nous exprimerons ensuite cet angle par rapport à l'axe x_G du repère global \mathcal{R}_G . Nous retiendrons les angles correspondant à chaque perception en les stockant dans un vecteur qui représente de manière discrète (dans notre cas tous les 10 deg) l'ensemble de ces directions (cf figure V.4 pour quatre observations). Ce vecteur permet d'estimer grossièrement sous quels points de vue a été perçu l'objet et quelle est la proportion perçue autour. Un exemple réel de ce vecteur sera donné par la suite.

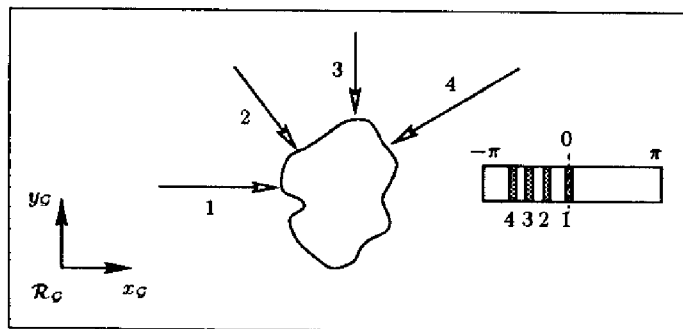


FIG. V.4 – Principe de l'estimation des directions d'observation. L'objet est perçu selon quatre points de vue différents. Les parties grisées du vecteurs (à droite) caractérisent chacune une direction d'observation.

Autres modèles envisageables

Le modèle géométrique fin se limite pour l'instant à la primitive superquadrique (ce qui peut être insuffisant notamment pour des objets non convexes). Mais un prolongement direct consisterait à opérer une deuxième segmentation sur chaque objet afin d'en identifier les différentes parties. On calculerait alors une superquadrique pour chacune d'elles, et le modèle de l'objet serait composé de leur agencement ([Solina 94, Gupta 93, Pentland 86]).

Toutefois les superquadriques ne constituent pas un modèle très précis de la surface de l'objet et l'on pourrait envisager l'utilisation d'autres types de représentations :

- Les surfaces déformables. Ces modèles nécessitent un maillage initial de la surface (obtenu par exemple à l'aide d'ellipsoïdes [Hebert 95a], de superquadriques [Terzopoulos 91] ou d'autres volumes triangulés [Chen 94, DeCarlo 94]). On définit ensuite des forces internes (*i.e.* entre les nœuds du maillage) qui garantissent la cohésion de la surface, et des forces externes appliquées sur chaque nœud et qui dépendent d'éléments extérieurs à la surface (par exemple les points 3D initiaux). Le maillage est alors considéré comme une structure déformable qui évolue par application des lois de la dynamique. Après un certain nombre d'itérations (symbolisant

le temps) le système se stabilise sur le modèle final. Ces représentations, souvent utilisées dans le domaine de l'imagerie médicale [Cohen 92], sont bien adaptées pour réestimer le modèle de l'objet au fur et à mesure des perceptions.

- Les maillages 3D. Ils sont constitués d'un réseau de points calculés à partir des points de la perception. Les triangulations de Delaunay en sont un exemple classique [Soucy 95, Boissonnat 87]. De plus, le problème de la fusion des maillages 3D a l'avantage de bénéficier d'une littérature abondante [Hel-Or 95, Zhang 94, Higuchi 94, Besl 92, Chen 91].

V.2 Modélisation topologique

Le modèle topologique caractérise, lui, des propriétés relatives à la disposition des objets dans la scène ce qui permet de la décrire à un niveau d'abstraction plus élevé. Il est basé sur la notion de relation topologique qui, à l'inverse des représentations géométriques, ne dépend pas de la forme des objets. En revanche, nous rappelons que ce chapitre traite de la modélisation d'une scène, par conséquent ces relations (et donc le modèle topologique de la scène) sont dépendantes du point de vue du robot.

V.2.1 Relations topologiques

Nous proposons tout d'abord une définition générale d'une relation topologique (cf définition 7) :

Définition 7 (Relation topologique) *Etant donné un point de vue, une relation topologique caractérise une configuration qualitative particulière entre un objet et d'autres objets ou le sol.*

Plus précisément, nous avons défini cinq relations topologiques binaires caractérisant la disposition relative de deux objets, et deux relations unaires équivalentes à un attribut topologique particulier d'un objet. Pour les représenter, nous adoptons la notation suivante : considérant deux objets o_p et o_q et une relation topologique \mathcal{R} , nous notons $\mathcal{R}(o_p, o_q)$ la relation entre o_p et o_q (dans le cas unaire nous la notons $\mathcal{R}(o_p)$). Ces relations sont ou non vérifiées. Nous considérons les relations topologiques suivantes :

- *à-gauche-de* (o_p, o_q) (respectivement *à-droite-de* (o_q, o_p)) ;
- *au-dessus-de* (o_p, o_q) (respectivement *au-dessous-de* (o_q, o_p)) ;
- *devant* (o_p, o_q) (respectivement *derrière* (o_q, o_p)) ;
- *contour-commun* (o_p, o_q) ;
- *contact-avec-le-sol* (o_p) ;
- *trou* (o_p).

Tout d'abord nous précisons que ces relations n'auront de sens que dans la mesure où la verticale est dirigée vers le haut de l'image (ce qui est bien évidemment vrai dans le cas général).

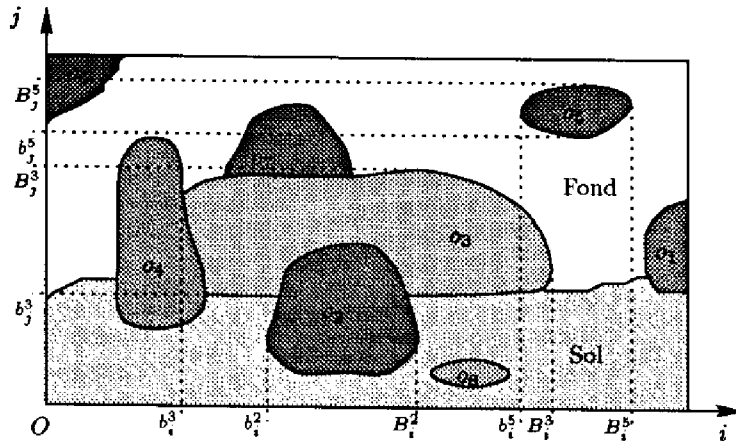


FIG. V.5 – Segmentation d'une scène générique.

Nous décrivons alors ces relations à l'aide de la figure V.5 représentant une scène générique où huit objets $(o_p)_{p=1,8}$ ont été segmentés. Chacun des n_p points d'un objet o_p est identifié par ses coordonnées $(i_k^p, j_k^p)_{k=1, n_p}$ dans le repère (O, i, j) de l'image segmentée. Nous définissons les bornes minimale b_i^p et maximale B_i^p d'un objet suivant l'axe i par :

$$b_i^p = \text{Min}_{k=1, n_p} i_k^p, \quad B_i^p = \text{Max}_{k=1, n_p} i_k^p \quad (\text{V.6})$$

• La relation *à-gauche-de* (o_p, o_q) est alors vérifiée si et seulement si :

$$(b_i^p < b_i^q) \text{ ET } (B_i^p < B_i^q) \quad (\text{V.7})$$

On peut aussi définir la relation plus stricte *entièrement-à-gauche-de* dans le cas où :

$$B_i^p < b_i^q \quad (\text{V.8})$$

A titre d'exemple, dans la figure V.5, la relation *à-gauche-de* n'est pas vérifiée pour le couple (o_2, o_3) ; en revanche, elle l'est pour les couples (o_4, o_3) et (o_3, o_5) , et "entièrement" pour le couple (o_4, o_2) .

La relation *à-droite-de* (respectivement *entièrement-à-droite-de*) est obtenue en inversant les indices p et q dans l'équation V.7 (respectivement V.8). Ces deux relations vérifient la propriété suivante :

$$(\text{à-gauche-de } (o_p, o_q) = \text{VRAI}) \Leftrightarrow (\text{à-droite-de } (o_q, o_p) = \text{VRAI}) \quad (\text{V.9})$$

D'autre part, chacune de ces relations est transitive (mais bien sûr non symétrique). On obtient ainsi la relation :

$$\begin{aligned} & \{ (\text{\grave{a-droite-de}}(o_p, o_q) = \text{VRAI}) \text{ ET } (\text{\grave{a-droite-de}}(o_q, o_l) = \text{VRAI}) \} \\ & \quad \downarrow \\ & (\text{\grave{a-droite-de}}(o_p, o_l) = \text{VRAI}) \end{aligned} \quad (\text{V.10})$$

- Les relations *au-dessus-de* et *au-dessous-de* sont établies de manière équivalente en considérant l'axe j de l'image. Les relations *devant* et *derrière* utilisent aussi le même principe, mais en considérant cette fois la distance des points 3D au capteur comme paramètre. De la même façon, une relation plus stricte peut se définir à partir de chacune d'entre elles (nous avons par exemple *entièrement-au-dessus-de* (o_5, o_3) qui est vérifiée dans la figure V.5). Enfin toutes ces relations sont transitives ; et, de part leur définition, nous pouvons aussi établir des propriétés équivalentes à celle décrite par l'équation V.9 :

$$(\text{au-dessus-de}(o_p, o_q) = \text{VRAI}) \Leftrightarrow (\text{au-dessous-de}(o_q, o_p) = \text{VRAI}) \quad (\text{V.11})$$

$$(\text{devant}(o_p, o_q) = \text{VRAI}) \Leftrightarrow (\text{derrière}(o_q, o_p) = \text{VRAI}) \quad (\text{V.12})$$

- La relation *contour-commun* (o_p, o_q) caractérise le fait que les deux objets o_p et o_q ont un contour en commun (par exemple les objets o_4 et o_3). C'est une relation symétrique mais non transitive.

- La relation *contact-avec-le-sol* (o_p) caractérise le fait que o_p possède un contour en commun avec le sol. C'est le cas pour les objets o_1, o_2, o_3, o_4 et o_8 . Elle peut aussi se définir à partir de la relation précédente :

$$\text{contact-avec-le-sol}(o_p) = \text{contour-commun}(o_p, \text{Sol}) \quad (\text{V.13})$$

Dans le cas où deux objets ont un contour en commun, nous pouvons définir deux autres relations *localement-devant* et *localement-derrière* qui caractérisent que, localement autour de ce contour, les points 3D d'un objet sont situés devant ou derrière les points 3D de l'autre. Ces deux relations vont avoir un intérêt tout particulier dans l'étude des occultations.

- Enfin si l'objet est en contact avec le sol, nous pouvons calculer, par moindre carrés, un plan passant par les points du sol qui sont autour de l'objet. La relation *trou* est alors vérifiée pour un objet si ses points sont situés au dessous de ce plan. C'est le cas pour la faille et la fosse montrées figure IV.20 (p. 50), et c'est aussi le cas de l'objet o_8 de la figure V.5 (en supposant que les points 3D correspondant sont au dessous du sol).

V.2.2 Relations d'occultation

La notion d'occultation dépend du point de vue. Un objet d'une scène peut être occulté de deux manières :

- l'objet s'occulte par lui-même ;
- l'objet est occulté par un autre objet, par le bord de l'image, voire par le sol dans certains types particuliers d'environnements.

Le premier type d'occultation a toujours lieu (sauf si l'objet est entièrement vu de dessus, ce qui est une configuration peu fréquente). En revanche, le second type d'occultation dépend du point de vue et des modalités de la perception. Nous allons ainsi définir deux relations d'occultation : l'une entre deux objets, et l'autre entre un objet et le bord de l'image.

La relation *occulte* caractérisant le fait que l'objet o_p occulte l'objet o_q est définie par :

$$\{ (\text{contour-commun}(o_p, o_q) = \text{VRAI}) \text{ ET } (\text{localement-devant}(o_p, o_q) = \text{VRAI}) \} \\ \Updownarrow \\ (\text{occulte}(o_p, o_q) = \text{VRAI}) \quad (\text{V.14})$$

C'est le cas pour le couple (o_2, o_3) dans la figure V.5 (en admettant qu'autour de la frontière commune entre ces deux objets, o_2 se trouve devant o_3).

La relation *occulté-par-le-bord* indique quant à elle que o_p est occulté par le bord de l'image (cas des objets o_1 et o_6 dans la figure V.5).

On remarque que le déplacement du robot est nécessaire pour lever l'occultation d'un objet par un autre, tandis que l'élargissement du champ de vue peut suffire dans le cas de l'occultation d'un objet par le bord de l'image.

Enfin, nous précisons que d'autres relations topologiques peuvent bien évidemment être construites à partir des toutes les relations précédentes.

V.2.3 Modèle topologique

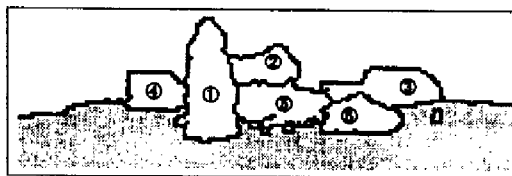


FIG. V.6 - Objets identifiés dans l'image "cailloux".

Le modèle topologique d'une scène est constitué de l'ensemble des relations topologiques et des relations d'occultation de chaque objet. Reprenons l'exemple de la scène

"cailloux" dont la figure V.6 donne l'image segmentée avec un numéro d'identification pour chaque objet. Nous donnons à titre d'exemple les modèles topologiques des objets 1 et 2 dans les tableaux des figures V.7 et V.8.

Objet no 1	
<i>contact-avec-le-sol</i>	VRAI
<i>obj. devant</i>	∅
<i>obj. derrière</i>	2, 3, 4, 5, 6
<i>obj. à droite</i>	2, 3, 5, 6
<i>obj. à gauche</i>	4
<i>obj. avec une front. com.</i>	2, 4, 5
<i>obj. occulté par cet obj.</i>	2, 4, 5
<i>obj. occultant cet obj.</i>	∅

FIG. V.7 – Modèle topologique de l'objet no 1 de la scène "cailloux".

Objet no 2	
<i>contact-avec-le-sol</i>	FAUX
<i>obj. devant</i>	1, 3, 4, 5, 6
<i>obj. derrière</i>	∅
<i>obj. à droite</i>	3, 5, 6
<i>obj. à gauche</i>	1, 4
<i>obj. avec une front. com.</i>	1, 5
<i>obj. occulté par cet obj.</i>	∅
<i>obj. occultant cet obj.</i>	1, 5

FIG. V.8 – Modèle topologique de l'objet no 2 de la scène "cailloux".

Ces relations topologiques permettent quelques interprétations simples. La règle V.15 (cf figure V.9) en est un exemple qui pourrait s'appliquer aux objets numéros ①, ④ et ⑤ de la figure V.6 et qui indiquerait que les objets ④ et ⑤ ne seraient qu'un seul et même objet.

Mais elles pourraient aussi permettre des raisonnements qualitatifs pour choisir une position suivante d'acquisition (planification de perception). La règle V.16 (cf figure V.10) en donne un exemple et peut s'appliquer aux objets numéros ① et ④ de la figure V.6. Cette règle indique que pour observer l'objet ④, il serait préférable de contourner l'objet ① par la gauche.

Ces relations sont des représentations abstraites de la scène. Aussi on pourrait envisager de s'en servir dans l'interaction entre l'homme et la machine en définissant des commandes telles que: "passer à droite de tel objet et à gauche de tel autre", "avancer jusqu'à ce que tel objet ne soit plus occulté"...

$$\begin{aligned}
 & (\text{\grave{a-gauche}}(o_p, o_q) = \text{VRAI}) \text{ ET } (\text{\grave{a-droite}}(o_p, o_l) = \text{VRAI}) \\
 & \text{ ET } (\text{devant}(o_p, o_q) = \text{VRAI}) \text{ ET } (\text{devant}(o_p, o_l) = \text{VRAI})
 \end{aligned} \tag{V.15}$$

↓

o_q et o_l forment peut être un seul et même objet qui est occulté par o_p .

$$\begin{aligned}
 & (\text{\grave{a-gauche}}(o_p, o_q) = \text{VRAI}) \text{ ET } (\text{occulte}(o_q, o_p) = \text{VRAI}) \\
 & \text{ ↓ }
 \end{aligned} \tag{V.16}$$

pour observer o_p se positionner à gauche de o_q .

Dans notre travail, ces relations n'ont pas été utilisées dans cet esprit, mais uniquement pour la sélection d'amers parmi les objets perçus.

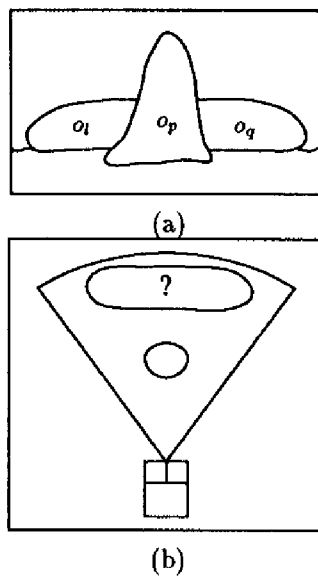


FIG. V.9 - *Interprétation de scène à l'aide du modèle topologique. La règle V.15 indique que o_l et o_q forment peut-être le même objet.*

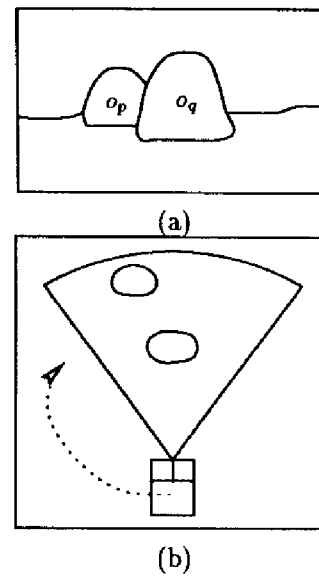


FIG. V.10 - *Planification qualitative de perception à l'aide du modèle topologique. La règle V.16 indique que pour observer o_p il est préférable de contourner o_q par la gauche.*

V.3 Sélection et modélisation des amers

Ayant présenté les deux modèles (géométriques et topologique) qui sont calculés pour l'ensemble des objets de la scène, nous allons maintenant étudier comment le robot en sélectionne certains comme amers en s'appuyant sur des critères de précision et de détection (cf section III.3).

V.3.1 Caractéristique utilisée : le sommet

Pour se localiser, le robot doit extraire de ses amers une (ou plusieurs) caractéristique précise.

Dans la littérature, des caractéristiques comme les maxima de courbure [Goldgoff 89], les maxima d'élévation [Fillatreau 94, Sutherland 94] ont été étudiées. S. Li utilise des caractéristiques situées sur la ligne d'horizon de scène extérieure d'immeubles [Li 92].

Dans notre cas, les amers, sélectionnés parmi les objets, peuvent avoir des formes quelconques. Aussi, nous avons choisi d'étudier une caractéristique générale: leur **sommet**. Si nous désirions construire un repère pour chaque amer, ce sommet en formerait l'origine. En revanche, il est beaucoup plus difficile d'orienter ce repère, c'est à dire d'attacher à cet amer des axes réobservables. On pourrait imaginer l'utilisation des axes d'inerties, mais

ne disposant que de vues partielles, ceux-ci s'avéreraient bien trop instables. On pourrait envisager une analyse plus poussée de la surface de l'amer pour y détecter des caractéristiques particulières [Thirion 94, Betgé-Brezetz 93, Monga 92, Shirai 86, Brady 85]. Mais on est alors confronté au problème de l'échantillonnage des points qui varie suivant la distance de perception et dans notre cas au nombre relativement faible de points disponibles sur les objets. Aussi, face à ces contraintes et dans un souci de robustesse, nous nous limiterons à l'utilisation des sommets et donc à l'origine des repères pour réestimer la position du robot et des amers. Chaque sommet sera alors représenté par un vecteur x_i de dimension 3 (correspondant à une translation). En revanche, la position du robot (i.e. son repère) pourra s'exprimer par un vecteur x_r de dimension 6 correspondant à une transformation 3D complète.

V.3.2 Sélection des amers

Nous avons identifié trois critères pour sélectionner certains objets en tant qu'amers :

1. *Visibilité*. L'objet ne doit pas être occulté par un autre objet ou par le contour de l'image. Dans le cas où cet objet est occulté, il sera à la fois difficile de le reconnaître et d'avoir une bonne estimée de son sommet, qui pourrait être sur la partie non visible.
2. *Non ambiguïté*. L'objet doit être en contact avec le sol. Cette condition n'est pas essentielle, mais elle réduit les ambiguïtés de reconnaissance.
3. *Précision*. Le sommet de l'objet doit être précis. Cette précision sera fonction du bruit du capteur, de la résolution et de la forme de l'objet.

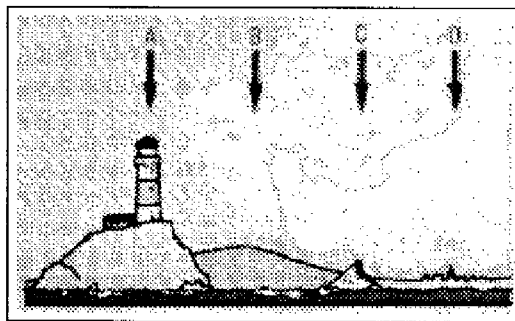


FIG. V.11 – “Les amers A, C, D sont bons, ils présentent une pointe verticale marquée, B n'est pas un bon amer avec son dos tout rond.” (Extrait du cours de navigation des Glénants [Glénants 85], p. 534)

La caractéristique retenue étant le sommet, la précision d'un amer dépendra donc de l'étroitesse de son sommet. Nous pouvons faire l'analogie avec la navigation côtière où un navigateur choisira aussi les amers étroits pour se localiser (cf figure V.11). En fonction de la largeur de l'amer et de la graduation de son compas, il cherchera la meilleure estimée

de l'angle selon lequel il voit l'amer. Nous allons voir comment, par un procédé équivalent, nous estimerons le sommet d'un objet et sa précision.

V.3.3 Calcul de l'incertitude du sommet d'un objet

Appelons $\mathbf{P} = \{P_i = (x_i, y_i, z_i)_{i=1,n}\}$ l'ensemble des points 3D segmentés d'une image tridimensionnelle pour un objet. Pour chaque point, nous avons une estimée \hat{P}_i (la mesure) et une matrice de covariance σ_i donnée par le modèle d'incertitude du capteur. Ces points 3D seront exprimés soit dans le repère capteur R_c , soit dans le repère redressé R_{red} (positionné sur l'origine du capteur et obtenu après correction de l'attitude du robot). Nous cherchons l'estimée \hat{S} du vrai point sommet S de l'objet et son incertitude σ_S .

Soit $P_{Max} \in \mathbf{P}$ le point 3D ayant l'élévation maximale (i.e. \hat{z}_{Max}) par rapport au repère redressé R_{red} . Nous remarquons que S n'est pas nécessairement P_{Max} en raison de la précision du capteur et de la résolution de l'image. Nous décomposons alors l'incertitude σ_S de S en deux termes.

Le premier, appelé σ_{forme} exprime la précision du sommet par rapport à sa forme et à l'incertitude des mesures de distance du capteur. Ce terme rend compte de la possibilité de confondre P_{Max} avec d'autres points dans son voisinage.

Le second terme $\sigma_{rés}$ dépend de la résolution du capteur et exprime le fait que le point mesuré d'élévation maximale P_{Max} n'est pas nécessairement le sommet S recherché en raison de l'échantillonnage du capteur.

Incertitude de forme

Calculons tout d'abord σ_{forme} . P_{Max} peut être confondu avec tous les points P_i d'élévation z_i dans R_{red} , tels que la distance de Mahalanobis¹ appliquée à la relation $\delta_i = z_{Max} - z_i$ est plus petite que la valeur χ_i^2 d'une distribution de χ^2 à un degrés de liberté pour un pourcentage fixé (pour 90% χ_i^2 vaut 2.71). Nous appelons $Dm(\delta_i)$ cette distance et \mathcal{S}_{Max} cet ensemble de points :

$$Dm(\delta_i) = \frac{(\hat{z}_{Max} - \hat{z}_i)^2}{E[z_{Max}^2] + E[z_i^2]} \quad (V.17)$$

$$\mathcal{S}_{Max} = \{P_i \mid i \in \{1..n\} \mid Dm(\delta_i) \leq \chi_i^2\} \quad (V.18)$$

$E[z_{Max}^2]$ et $E[z_i^2]$ désignent respectivement les "termes" de variance en z extraits des matrices de covariance σ_{Max} et σ_i . \mathcal{S}_{Max} représente l'ensemble des points du sommet de l'objet.

L'incertitude de forme σ_{forme} rend compte de la distribution des points du sommet (i.e. de \mathcal{S}_{Max}) autour de \hat{P}_{Max} . Pour en donner une estimation nous recherchons le plus petit cylindre \mathcal{C}_{Max} (dans R_{red}) centré sur \hat{P}_{Max} qui inclut les points de \mathcal{S}_{Max} et leur incertitude. La forme cylindrique permet de ne pas privilégier de direction particulière pour l'incertitude dans le plan (O, x_{red}, y_{red}) . La figure V.12 illustre ce principe pour un simple contour 2D qui serait peu bruité (a) ou très bruité (b), \mathcal{C}_{Max} y est représenté en pointillés. La matrice

1. nous supposons pour cela des bruits gaussiens sur les points.

de covariance σ_{forme} est alors définie par le second moment d'une loi ayant une distribution uniforme dans C_{Max} (ce qui correspond au pire cas). En utilisant les équations données à l'annexe A.3, et si r et h représentent le rayon et la hauteur de C_{Max} , alors σ_{forme} a pour valeur :

$$\sigma_{\text{forme}} = \begin{pmatrix} \frac{r^2}{4} & 0 & 0 \\ 0 & \frac{r^2}{4} & 0 \\ 0 & 0 & \frac{h^2}{12} \end{pmatrix} \quad (\text{V.19})$$

Cette incertitude est ensuite exprimée dans R_c .

Par ailleurs, nous rappelons qu'un des critères de sélection d'un amer est qu'il ne soit pas occulté. On pourrait toutefois nuancer ici ce critère en ne considérant que l'occultation de son sommet, *i.e.* de la surface formée par les points de S_{Max} . Cela permettrait notamment de profiter des objets élevés (dont le domaine de visibilité est important), mais qui seraient occultés à leur base par d'autres objets.

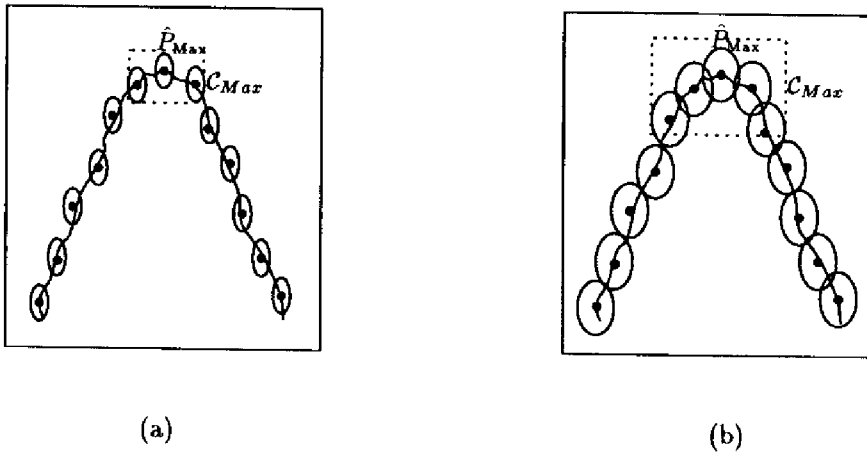


FIG. V.12 - Représentation schématique, vue du capteur, de l'incertitude de forme pour un même objet 2D qui serait peu bruité (a) et très bruité (b). C_{Max} (en pointillés) contient \hat{P}_{Max} et l'ensemble des points qui peuvent se confondre avec lui.

Incertitude de résolution

Pour illustrer la nécessité de prendre en compte la résolution du capteur, imaginons la situation suivante. Le robot Adam perçoit à 10 m un objet de 50 cm de haut. Supposons que la partie visible de cet objet soit verticale (la répartition des points dépendant de l'inclinaison de sa surface) et que le pas en site du capteur soit de 1 deg. Alors l'échantillonnage des points au niveau de son sommet est de l'ordre de 18 cm. En revanche, le même objet vu à 2 m donnerait un pas d'échantillonnage de 5 cm.

Dans cette partie nous allons étudier ce problème de résolution en le transformant en un calcul d'incertitude. Cette nouvelle incertitude sera ajoutée à celle du sommet.

Appelons R'_c le repère centré sur l'origine du capteur et obtenu par une rotation du repère capteur R_c autour de son axe z_c , de telle sorte que \hat{P}_{Max} appartienne au plan

(O, x'_c, z'_c) . La partie haute de la figure V.13 montre ainsi un objet coupé par ce plan. Nous précisons que dans notre cas le capteur est un télémètre laser, mais que l'approche que nous allons proposer reste aussi valide pour la stéréovision si O représente l'œil cyclopatique, les pas en site et en azimuth pouvant être évalués à partir des paramètres intrinsèques des caméras du système stéréoscopique.

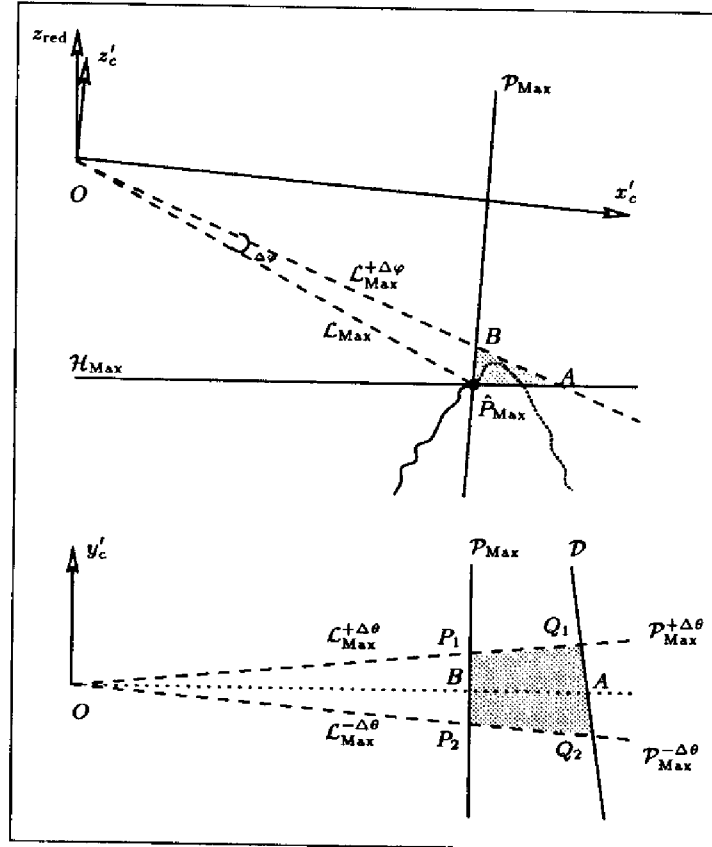


FIG. V.13 – Incertitude de résolution illustrée dans les deux directions.

Appelons \mathcal{L}_{Max} le rayon (faisceau laser, visée) correspondant au point \hat{P}_{Max} , le plus haut point. $\mathcal{L}_{Max}^{+\Delta\varphi}$ est le rayon suivant sur la même colonne dans l'image polaire. Ce rayon est indiqué dans la figure V.13. Il correspond, dans l'image polaire représentée par la figure V.14, à la croix placée au dessus de \hat{P}_{Max} . Ce dernier étant le plus haut point de l'objet (dans R_{red}), le faisceau $\mathcal{L}_{Max}^{+\Delta\varphi}$ n'intersecte pas l'objet². Appelons ensuite \mathcal{H}_{Max} le plan horizontal (i.e. perpendiculaire à la vraie verticale (O, z_{red})) et passant par \hat{P}_{Max} . Le vrai sommet S de l'objet est alors situé au dessus de ce plan. Aussi, en ne regardant que la coupe (O, x'_c, z'_c) de l'objet, S peut être à l'intérieur d'un triangle formé par \mathcal{L}_{Max} , $\mathcal{L}_{Max}^{+\Delta\varphi}$

2. En réalité, cela n'est vrai que si l'attitude du robot ne fait pas un angle de plus de 90 deg. avec la verticale, hypothèse au demeurant raisonnable...

et \mathcal{H}_{Max} . Toutefois, nous ferons l'hypothèse que la vraie valeur du sommet n'est pas plus près du capteur que le plan \mathcal{P}_{Max} passant par \hat{P}_{Max} et perpendiculaire à (O, x'_c) . Si nous notons $A = \mathcal{H}_{\text{Max}} \cap \mathcal{L}_{\text{Max}}^{+\Delta\varphi}$ et $B = \mathcal{P}_{\text{Max}} \cap \mathcal{L}_{\text{Max}}^{+\Delta\varphi}$, les valeurs possibles du vrai sommet sont donc à l'intérieur du triangle $(A, B, \hat{P}_{\text{Max}})$ (région ombrée du haut de la figure V.13).

Considérons maintenant l'erreur de résolution dans la direction d'azimut. La partie basse de la figure V.13 est une coupe de l'objet par le plan $(O, (OA), y'_c)$. $\mathcal{L}_{\text{Max}}^{-\Delta\theta}$ et $\mathcal{L}_{\text{Max}}^{+\Delta\theta}$ seraient les deux rayons situés, dans l'image polaire, sur la même ligne et de chaque côté de $\mathcal{L}_{\text{Max}}^{+\Delta\varphi}$. Nous notons P_1, Q_1, P_2 et Q_2 respectivement l'intersection de ces deux droites avec \mathcal{P}_{Max} et \mathcal{H}_{Max} . Les points P_1 et P_2 sont montrés figure V.14. $\mathcal{P}_{\text{Max}}^{-\Delta\theta}$ et $\mathcal{P}_{\text{Max}}^{+\Delta\theta}$ sont les deux plans passant par (O, z'_c) et respectivement par $\mathcal{L}_{\text{Max}}^{-\Delta\theta}$ et $\mathcal{L}_{\text{Max}}^{+\Delta\theta}$. On appelle enfin \mathcal{D} la droite (Q_1, Q_2) . Dans cette coupe, les positions possibles du sommet de l'objet sont alors définies par le quadrilatère (P_1, Q_1, Q_2, P_2) (ie la zone ombrée de bas de la figure V.13). Notons encore P_3 et P_4 les deux points définis par les équations V.20 et représentés figure V.14.

$$\begin{aligned} P_3 &= \mathcal{P}_{\text{Max}}^{-\Delta\theta} \cap \mathcal{P}_{\text{Max}} \cap \mathcal{H}_{\text{Max}} \\ P_4 &= \mathcal{P}_{\text{Max}}^{+\Delta\theta} \cap \mathcal{P}_{\text{Max}} \cap \mathcal{H}_{\text{Max}} \end{aligned} \quad (\text{V.20})$$

En résumé, le faisceau qui pourrait intersecter le vrai sommet S décrit l'angle solide (O, P_1, P_2, P_3, P_4) (partie grisée de l'image polaire V.14). De plus S est au dessus de \mathcal{H}_{Max} et à l'arrière de \mathcal{P}_{Max} . Aussi S appartient au polyèdre $\mathcal{P} = (P_1, P_2, P_3, P_4, Q_1, Q_2)$ qui est montré figure V.15.

En revanche, nous n'avons aucune information qui nous permettrait de savoir où, à l'intérieur de cette forme, ce sommet aurait plus de chance de se trouver. Autrement dit, il a une probabilité identique de se situer à n'importe quelle position à l'intérieur de \mathcal{P} . S suit donc une loi de probabilité uniforme dans \mathcal{P} . L'estimée finale du point sommet \hat{S} et la matrice de covariance $\sigma_{\hat{S}}$, sont les deux premiers moments de cette distribution. On les calcule à l'aide des équations données dans l'annexe A.3 (en intégrant \mathcal{P} dans le repère (O, x'_c, y'_c, z'_c)). Conclusion :

L'estimée finale du sommet est alors \hat{S} ; son incertitude σ_S est simplement calculée en tant que somme des incertitudes σ_{forme} et $\sigma_{\text{rés}}$, négligeant ainsi les corrélations entre ces deux bruits. C'est donc en fonction de cette incertitude σ_S que le deuxième critère de caractérisation d'un amer se vérifie.

Remarques :

1. Dans le cas particulier où le capteur du robot est vertical (i.e. les repères R_c et R_{red} sont confondus) (P_1, P_2, P_3, P_4) est un rectangle et (Q_1, Q_2, P_3, P_4) un trapèze. Si maintenant on simplifie cette forme en la majorant par un coin de taille (a, b, c) (cf figure V.16), alors l'estimée du sommet est $(\frac{a}{3}, 0, \frac{c}{3})$ (le centre de gravité), sa variance de résolution est donnée par l'équation V.21.

2. Si nous considérons un objet apparaissant comme étroit dans la perception mais dont le sommet serait situé plus haut que le capteur (i.e. au dessus du demi plan (O, x_c, y_c)),

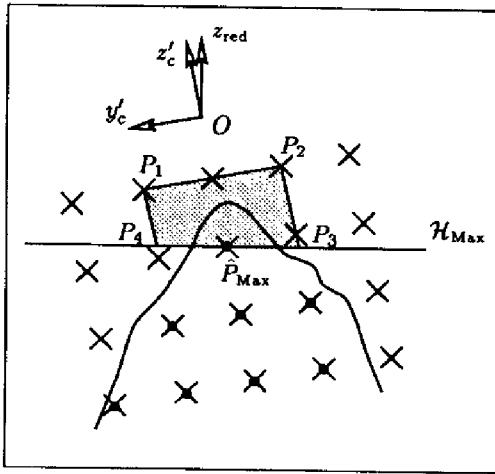


FIG. V.14 - Image polaire. Le faisceau pouvant intersecter le vrai sommet S décrit l'angle solide (O, P_1, P_2, P_3, P_4) .

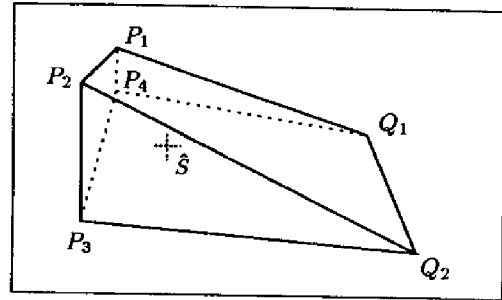


FIG. V.15 - Polyèdre \mathcal{P} à l'intérieur duquel est situé le vrai sommet S .

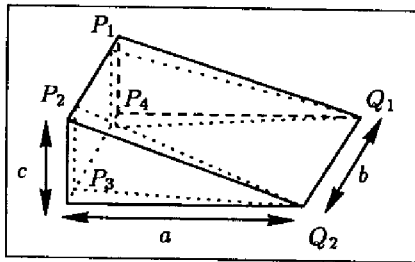


FIG. V.16 - Polyèdre \mathcal{P} (en pointillés) dans le cas d'un robot vertical.

$$\sigma_{rés} = \begin{pmatrix} a^2/36 & 0 & -ac/72 \\ 0 & b^2/24 & 0 \\ -ac/72 & 0 & c^2/36 \end{pmatrix} \quad (V.21)$$

Variance de résolution si \mathcal{P} est approximé par un coin de taille (a, b, c) (dessiné en trait continu dans la figure V.16).

alors le domaine des positions possibles de son vrai sommet n'est pas borné (cf figure V.17). Ce sommet pourrait être soit dans l'angle solide défini précédemment et résultant de l'échantillonnage, soit occulté par l'objet lui-même. On pourrait ne pas tenir compte de cet objet comme amer. Mais en réalité, il serait regrettable de ne pas profiter d'un objet que le robot pourrait percevoir depuis de nombreuses positions différents (du fait de sa hauteur).

Aussi une solution consisterait à borner artificiellement le domaine des positions possibles de S . Le sommet \hat{S} pourrait être estimé au point d'élévation maximale \hat{P}_{Max} et son incertitude serait principalement dirigée dans la direction de visée. La réobservation de ce même objet, selon un point de vue différent, permettrait de réduire cette incertitude.

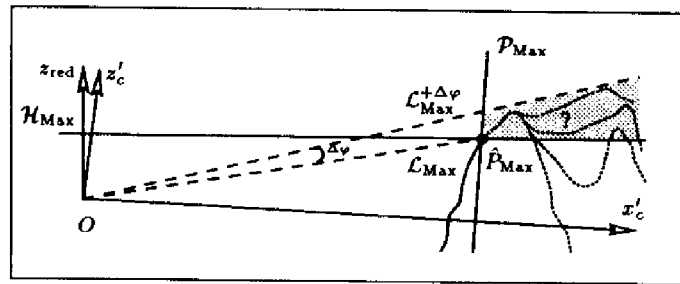


FIG. V.17 – Cas d'un objet dont le sommet est plus haut que le capteur. Le domaine des positions possibles du vrai sommet n'est pas borné.

V.3.4 Un exemple de sélection d'amers

Le robot Adam est placé sur le terrain d'expérimentation Geroms (dont nous montrons une photo figure VIII.13 p. 125). De sa position, il perçoit l'environnement et va chercher à identifier les amers.

Il détecte tout d'abord dix objets comme l'indique la segmentation montrée en haut de la figure V.18. À l'aide des critères définis auparavant, il sélectionne six de ces objets comme amers (les objets numérotés de 1 à 6). Plus précisément, l'objet numéro 7 n'a pas été sélectionné parce qu'il est occulté par le bord de l'image. C'est aussi le cas de l'objet 9 (la colline) qui de surcroît a un sommet qui n'est pas assez précis. L'objet 8 n'est pas occulté, mais il n'est pas assez précis. Enfin, l'objet 10 qui est sur la colline, n'a pas été sélectionné car il n'est pas en contact avec le sol.

La partie basse de la figure V.18 montre la carte locale de l'environnement obtenue par cette perception avec les six amers retenus et leur incertitude correspondante. Les ellipses correspondent aux projections de distributions gaussiennes dessinées à 99% (cf annexe A.1). Nous remarquons tout d'abord que l'incertitude décroît globalement avec la distance. Si nous regardons la figure dans plus de détail, nous pouvons voir que l'amer 1 est près du capteur et que son sommet est étroit : il est donc très précis. En revanche, l'amer 6 est loin du robot et son sommet est large, donc d'une incertitude importante. Les amers 3 et 4 sont près l'un de l'autre, mais comme le sommet de l'amer 4 est plus effilé que celui de l'amer 3, il a une incertitude plus petite. Enfin, l'amer 5 a un sommet étroit mais il est loin du capteur, son incertitude provient plus de la résolution que de sa forme ; son ellipse d'incertitude est ainsi dans la direction du capteur.

V.4 Un détour par un environnement intérieur

Nous allons provisoirement rentrer à l'intérieur du laboratoire pour examiner comment la modélisation par objet peut être mise à profit dans ce type d'environnement. Nous avons ainsi disposés dans notre salle expérimentale différentes boîtes de carton comme le montre la figure V.19. Le robot Adam est bien entendu resté à l'extérieur, et c'est le robot Hilare II qui a pris le relais. Hilare II est aussi équipé d'un capteur laser qui lui permet de prendre des images tridimensionnelles comme celle montrée figure V.20.

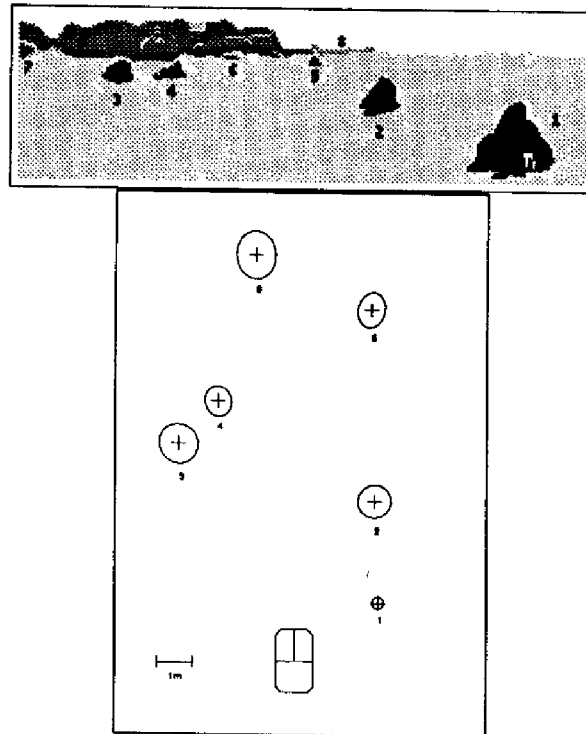


FIG. V.18 – **En haut** : Les dix objets segmentés de la scène. **En bas** : Les amers sélectionnés et l'incertitude de leur sommet qui est fonction du bruit du capteur, de sa résolution et de la forme du haut de l'objet.

V.4.1 Segmentation d'une scène structurée

Dans les environnements intérieurs la décomposition en objets et sol (qui est dans ce cas plan) est toujours valide, la particularité de ces environnements réside principalement dans le fait que la plupart des objets sont polyédriques. Notre algorithme de segmentation appliqué à l'image V.20 donne le résultat montré figure V.21 où l'on voit les objets détectés. On y remarque les boîtes et les murs qui sont bien entendu aussi considérés comme des objets. Etant donné l'importance de ces murs, l'hypothèse d'horizontalité du sol pourrait être utilisée pour bien détecter ce dernier dans l'histogramme H . En réalité, la répartition des points dans l'image diminuant avec la distance, il est encore perçu un nombre majoritaire de points sur le sol (et notamment sur la partie proche du capteur). Aussi, seul un pic a été détecté dans l'histogramme des normales.

Nous précisons qu'à partir d'une certaine distance, les faisceaux laser qui se réfléchissent sur le sol deviennent trop spéculaires et le capteur n'en a alors pas d'écho. Aussi seule une partie du sol peut être perçue, comme le montre la figure V.21.

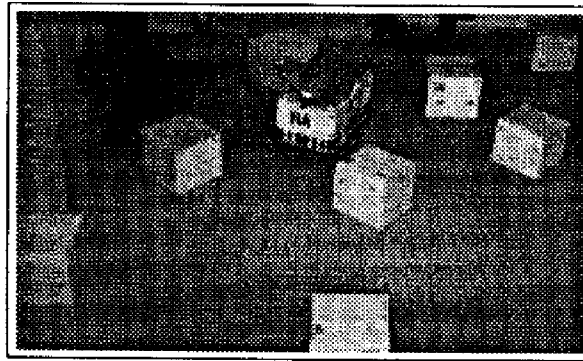


FIG. V.19 – Photo d'un environnement intérieur dont les amers sont des boîtes qui sont perçues par le robot Hilare II.

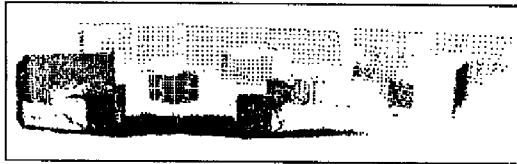


FIG. V.20 – Image 3D d'une scène structurée.

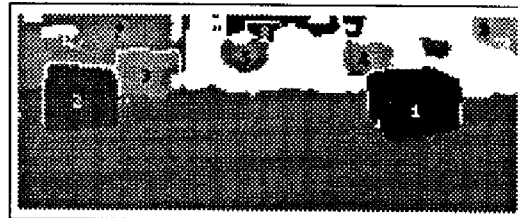


FIG. V.21 – Segmentation d'une scène structurée. Les objets sont les murs et les boîtes.

V.4.2 Modélisation d'une scène structurée

Le modèle topologique de cette scène se calcule comme il a été montré dans la partie V.2.

En revanche, pour établir le modèle géométrique des objets, nous pouvons prendre en compte la connaissance *a priori* que nous avons sur leur forme polyédrique. Nous représenterons ainsi chaque objet par ses différentes faces planes. Pour les trouver, il suffit d'appliquer, pour *chaque objet séparément*, la méthode des histogrammes bi-variable et mono-variable que nous avons décrite à la section IV.2. Dans ce cas, il faut bien évidemment utiliser le système de coordonnées sphériques (α', β') (cf figure IV.7 p. 41) bien adapté à l'extraction de surfaces verticales. Les faces verticales observables d'une boîte selon un point de vue ne pouvant pas être parallèles, elles apparaîtront comme des pics différents dans l'histogramme bi-variable des normales. La figure V.22 montre par exemple l'histogramme H construit à partir des normales des points de la boîte numéro 1 de la segmentation V.21. Deux pics apparaissent qui correspondent à ses deux faces verticales, la face horizontale est constituée par les restants des points. Cette dernière n'apparaît toutefois pas comme un pic car cet histogramme est calculé dans le système de coordonnées (α', β') (et par conséquent les points de la face horizontale sont répartis le long de l'axe α').

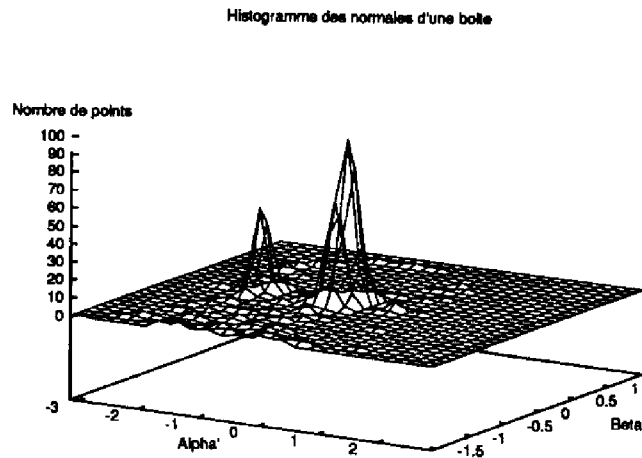


FIG. V.22 – Histogramme des normales de la boîte numéro 1 de l'image V.21. Les deux faces verticales apparaissent comme deux pics dans cet histogramme $H(\alpha', \beta')$.

V.4.3 Extraction de repères

Nous pouvons construire un repère à partir des faces de chaque boîte. Ces repères sont tout à fait pertinents pour estimer la position du robot et les boîtes constituent ainsi les amers qu'il sera intéressant de choisir. Ces objets vérifient en effet les propriétés d'un amer : distinction et précision. Pour les identifier, nous en donnons alors la description : un parallélépipède posé sur le sol. Autrement dit, le robot peut en observer au plus deux faces verticales et au plus une face horizontale dans une même perception. Si cela n'est pas le cas pour un objet, alors celui-ci n'est pas une boîte. D'autres informations peuvent être fournies au robot sur les boîtes pour les sélectionner comme leur hauteur ou leur largeur et profondeur maximales.

Le haut de la figure V.24 montre ainsi les différentes boîtes que le robot a identifiées dans la segmentation V.21 (et correspondant aux objets no 1 à 6). Les faces de chaque boîte y sont indiquées par des niveaux de gris différents.

Le robot va donc construire un repère à partir des faces de chaque boîte. Le sol étant plan et les boîtes parallélépipédiques, nous nous limiterons à une représentation 2D de ce repère. Nous positionnerons son origine à un coin de la boîte et nous l'orienterons en utilisant ses faces (ce qui n'a pas pu être fait dans le cas des environnements extérieurs). Les amers et le robot seront alors représentés dans le système par un vecteur de dimension 3 : (x, y, θ) .

Par convention, nous choisissons de poser le repère de chaque boîte de telle sorte que celle-ci soit située dans le quart de plan $(x > 0, y > 0)$ de son repère.

Nous expliquons brièvement comment nous calculons ce repère et son incertitude. Deux cas sont à traiter séparément suivant que deux ou une seule face verticale est perçue.

▷ **Deux faces verticales sont perçues.** Nous notons $(P_i)_{i=1,n}$ et $(P'_i)_{i=1,n'}$ les deux ensembles de points détectés respectivement sur chaque face. Ce sont bien évidemment des "points aléatoires" caractérisés par leurs deux premiers moments \hat{P}_i , \hat{P}'_i , σ_i et σ'_i . Un simple calcul de moindres carrés nous permet d'obtenir une première estimation des paramètres de l'équation du plan de chaque face (ces paramètres sont représentés par les vecteurs \mathbf{p} et \mathbf{p}' ³). Pour obtenir une première matrice de variance sur ces vecteurs, nous appliquons pour chaque face le principe décrit dans [Hébert 94] (p. 71) ou [Bard 74] (p. 157), $(x_i)_{i=1,N}$ sont les données (pour nous P_i et P'_i), \mathbf{y} le vecteur des paramètres du modèle (\mathbf{p} ou \mathbf{p}') et $g(x_i, \mathbf{y}) = 0$ l'équation du modèle (équation de plan). La variance $\sigma_{\mathbf{y}}$ (i.e. $\sigma_{\mathbf{p}}$ et $\sigma_{\mathbf{p}'}$) est alors obtenue par le calcul suivant :

$$A_i = \frac{\partial g}{\partial x_i}(\hat{x}_i, \hat{\mathbf{y}}), \quad B_i = \frac{\partial g}{\partial \mathbf{y}}(\hat{x}_i, \hat{\mathbf{y}}) \quad (\text{V.22})$$

$$S_i = A_i \sigma_i A_i^T, \quad W = \sum_{i=1}^N B_i^T S_i^{-1} B_i, \quad \sigma_{\mathbf{y}} = W^{-1} \quad (\text{V.23})$$

Ensuite, pour chaque face, et à partir de ces premières estimées et variance, nous appliquons itérativement un filtre de Kalman en utilisant tous les points 3D (l'équation de mesure étant bien entendu celle du plan). Le filtre de Kalman sera présenté en détails dans le chapitre suivant. A la fin de cette boucle, nous obtenons les paramètres finaux de la face et leur variance finale. Nous pouvons alors estimer le repère (x, y, θ) et son incertitude : son origine (x, y) est l'intersection des deux faces (et du sol), son orientation θ est calculée à partir de l'une des faces (différents cas sont à considérer suivant le format de chacune des équations de face). La variance de ce repère est obtenue simplement par propagation de l'incertitude à travers les équations qui ont permis de calculer son estimation. La contrainte de perpendicularité des deux faces peut être intégrée à la matrice de variance du repère en appliquant à nouveau un filtre de Kalman (l'équation de mesure caractérisant dans ce cas la perpendicularité des faces).

▷ **Une seule face verticale est perçue.** Nous pouvons calculer les paramètres de l'équation de cette face ainsi que la variance sur ces paramètres à l'aide de la technique décrite précédemment. Cette face permet d'orienter le repère et il faut fixer l'origine sur l'un de ses deux bords. Mais l'estimation de ce bord dépend grandement de l'échantillonnage des points (ce qui n'était pas le cas lorsque nous avions deux faces, car le bord était calculé par leur intersection). Notons que cela correspond **exactement** au problème de l'estimation du vrai sommet d'un objet compte tenu de son échantillonnage (cf section V.3.3). Toutefois, le problème est ici simplifié car l'on possède un modèle de la surface de l'objet (i.e. un plan).

Etudions le calcul de l'incertitude de résolution du bord en considérant le problème en 2D. La figure V.23 montre le repère capteur (O, x_c, y_c) supposé parallèle au sol, et une

3. Les faces des boîtes sont des plans verticaux qui peuvent être représentés par l'un des deux formats suivant [Ayache 88a]: $x = ay + bz + p$ ou $y = ax + bz + p$. \mathbf{p} et \mathbf{p}' sont les vecteurs $[a \ b \ p]^T$, mais il convient aussi de retenir le format correspondant.

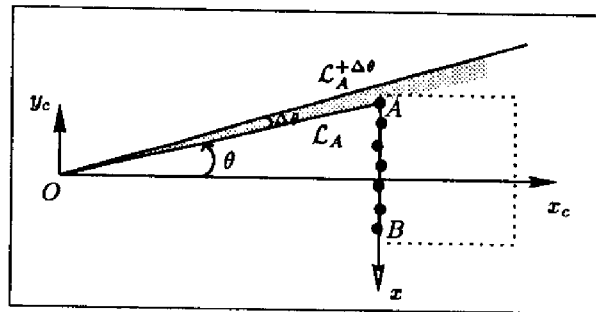


FIG. V.23 – Incertitude de résolution d'une face. Les contours de la vraie boîte sont représentés en pointillés et les points de mesure par des symboles: "•". x est l'axe des ordonnées du repère de la boîte. \mathcal{L}_A est le rayon du dernier point (A) perçu sur le bord gauche de la face, $\mathcal{L}_A^{+\Delta\theta}$ est le rayon suivant dont le point correspondant n'appartient pas à la face. Le rayon \mathcal{L} qui intersecterait le vrai bord décrit de manière uniforme la zone ombrée.

coupe 2D de la face dont les deux extrémités sont données par les deux derniers points de mesure A et B. Pour que la boîte soit dans la partie ($x > 0, y > 0$) du repère objet, il faut placer son origine du côté de A et l'axe des ordonnées x dans la direction de B. Appelons \mathcal{L}_A le rayon qui a donné la mesure A (avec $\theta = (x_c, \mathcal{L}_A)$). $\mathcal{L}_A^{+\Delta\theta}$ est le rayon suivant, il n'intersecte donc pas l'objet ($\Delta\theta$ étant le pas du laser dans cette direction). Le rayon \mathcal{L} qui intersecterait le vrai bord fait un angle compris entre 0 et $\Delta\theta$ par rapport à \mathcal{L}_A . Aucune information ne permet de dire, à l'intérieur de cet intervalle angulaire, où la direction de ce rayon a le plus de chance de se trouver; autrement dit sa direction suit une loi de probabilité uniforme dans $[0, \Delta\theta]$. Si α est l'angle (aléatoire) entre \mathcal{L} et \mathcal{L}_A , nous pouvons calculer ses deux premiers moments (cf annexe A.3), soit :

$$\hat{\alpha} = \frac{\Delta\theta}{2}, \quad \sigma_\alpha = \frac{\Delta\theta^2}{12} \quad (\text{V.24})$$

Si $y = a.x$ est l'équation de droite correspondant au rayon, alors le paramètre a vaut $\tan \alpha$ et ses deux premiers moments sont :

$$\hat{a} = \tan \hat{\alpha}, \quad \sigma_a = \frac{\sigma_\alpha}{\cos^4 \hat{\alpha}} \quad (\text{V.25})$$

Pour être rigoureux, il faut ensuite rajouter à cette incertitude celle de \mathcal{L}_A .

Nous montrons dans le bas de la figure V.24 les différents repères calculés pour chaque boîte et l'ellipse d'incertitude de leur origine (tracée à 99%). On observe que l'incertitude augmente globalement avec la distance. Mais regardons plus précisément la boîte située au fond à droite de la scène (objet no 6) et dont le robot n'a observé qu'une seule face. Cette boîte est loin du capteur, aussi l'incertitude de l'origine de son repère est principalement due à l'échantillonnage des points. Cette incertitude doit être plus importante suivant l'axe correspondant à la face perçue, ce que l'on observe sur son ellipse représentée dans la figure V.24.

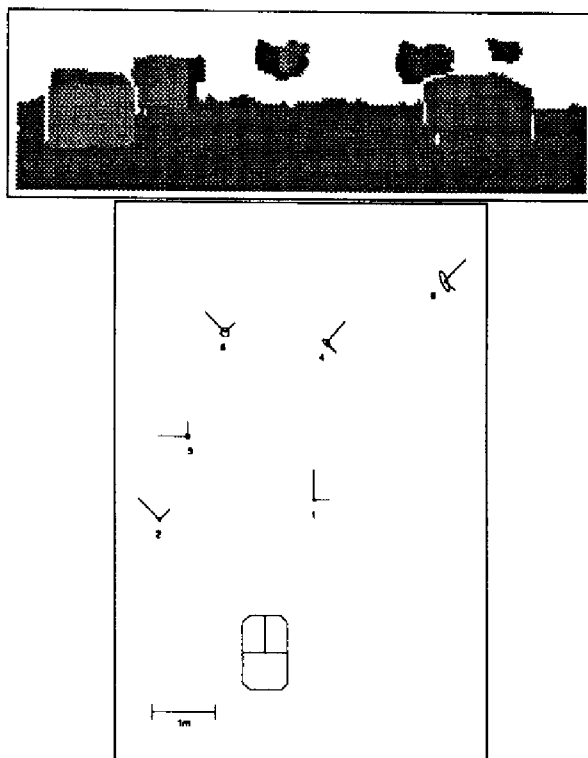


FIG. V.24 – Sélection des amers d'une scène structurée (cf figure V.21). En haut : Boîtes détectées (leurs faces sont représentées par des niveaux de gris différents). En bas : Les repères de chaque boîte et l'ellipse d'incertitude de leur origine.

V.5 Structure du modèle local

A partir de la segmentation d'une scène, nous avons extrait pour chaque objet différentes informations de natures géométrique et topologique. Nous regroupons ces informations dans une même structure de données à laquelle est ajouté un indicateur précisant que l'objet a été classé comme amer et un identificateur propre à chaque objet de la perception. L'ensemble constitue le **modèle local d'un objet** comme le montre la figure V.25.

Le **modèle local de la scène** (ou modèle local) regroupe la liste des modèles locaux des objets ainsi que diverses informations sur la perception. Le sol (ou plus exactement ses éventuelles différentes parties) y est aussi représenté. Mais, comme nous l'avons vu, la représentation de ce dernier reste toutefois très rudimentaire. La structure de données du modèle local d'une scène est montrée figure V.26.

Modèle local d'un objet	
Identificateur	
Amer (Oui / Non)	
Modèle géométrique	
	- matrice d'inertie, centre de gravité
	- nombre de points perçus
	- ellipsoïde
	- superquadrique*
	- repère et incertitude
	Eventuellement simplement l'origine (<i>i.e.</i> le sommet)
Modèle topologique	
	- relations topologiques avec les autres objets
	- contact avec le sol
	- trou
Occultations	
	- par d'autres objets
	- par le bord de l'image
Directions d'observation	
Liste des points 3D*	

FIG. V.25 – *Modèle local d'un objet. Les parties optionnelles sont indiquées par une étoile.*

Modèle local d'une scène	
Information générale sur la perception	
	- nombre de lignes et de colonnes de l'image 3D
	- pour le télémètre: champ de vue, pas en site et azimut.
Liste des modèles locaux des objets	
Liste des modèles locaux des parties du sol	

FIG. V.26 – *Modèle local d'une scène.*

V.6 Conclusion

Nous avons construit différentes représentations pour décrire les objets et leur disposition dans la scène. Mais il est bien évident que d'autres informations pourraient encore être extraites si la tâche du robot le nécessiterait. Le modèle géométrique pourrait par exemple être enrichi de représentations plus fines (surfaces déformables, maillage 3D, ...). L'élaboration d'autres relations topologiques pourrait aussi aider à la planification de perception.

Mais il serait surtout très intéressant de bénéficier d'autres types d'informations provenant de capteurs de natures différentes. La couleur en est un exemple typique. Dans le cas d'un système de stéréovision couleur, elle pourrait directement s'extraire des images des caméras. A chaque objet seraient alors associées des informations photométriques qui permettraient de mieux le caractériser et donc de mieux le différencier dans l'environnement.

Ces informations pourraient s'avérer tout à fait pertinentes pour la sélection des amers. Des travaux sont actuellement menés dans notre groupe par P. Lasserre et R. Murrieta pour le traitement des images couleurs en environnement naturel. Une extension de notre travail consisterait à fusionner les deux types d'informations (3D et couleur).

Aussi nous avons construit les structures d'objets d'une manière générique, de telle sorte que d'autres modèles (géométriques ou topologiques) et d'autres types d'information puissent y être facilement intégrés.

Le modèle local constitue une des trois entrées du module principal de modélisation incrémentale dont nous avons donné le schéma général à la figure III.4 (p. 32). Ce module de reconnaissance, de localisation et de fusion va composer les informations du modèle local avec celles du modèle de l'environnement (modèle global) déjà établi. Il mettra ainsi à jour la position du robot, la représentation qu'il a de chaque objet et intégrera dans le modèle global les nouveaux objets perçus.

Dans la partie suivante, nous allons détailler les différentes étapes de la réestimation du modèle global. Un point central est la gestion des incertitudes à travers l'ensemble du processus de modélisation incrémentale. Nous nous appuyerons pour cela sur l'outil puissant de la carte stochastique et du filtre de Kalman auxquels nous consacrons le chapitre suivant.

Troisième Partie

**Modélisation par objets de
l'environnement**

Chapitre VI

Modèle par objets et incertitudes

Nous traitons dans cette partie de la boucle de modélisation incrémentale à proprement parler. Celle-ci va construire itérativement le modèle de l'ensemble de l'environnement à partir des différents modèles locaux. Un souci majeur de notre travail a été de prendre en compte de manière rigoureuse les incertitudes des capteurs (odométrie, capteur 3D) en les propageant à travers toute la chaîne de modélisation. Nous montrerons comment cela permet par ailleurs d'assurer la cohérence globale du modèle.

Dans ce chapitre nous présentons l'approche de la carte probabiliste sur laquelle nous sommes appuyés. Nous développerons le formalisme utilisé ainsi que les deux outils probabilistes nécessaires pour la construire (propagation de l'incertitude et filtre de Kalman). Nous étudierons alors les différentes mises en œuvre qui en ont été réalisées. Enfin certains points particuliers (corrélations, biais) seront analysés et discutés afin de mieux évaluer cette approche et de mieux en percevoir l'intérêt pour les environnements extérieurs.

VI.1 La carte stochastique

La carte stochastique a été proposée comme modèle pour représenter l'ensemble des informations relatives à l'organisation des objets dans l'environnement et à la position du robot tout au long de son déplacement [Moutarlier 91b, Smith 90, Leonard 90]. Cette carte est constituée par un ensemble de relations incertaines représentant les positions et orientations (configurations) des objets et du robot ("objet" particulier observateur de l'environnement et pouvant se déplacer). Ces relations sont des transformations dans l'espace qui auraient pu être exprimées dans des repères différents, mais nous supposons qu'il existe un repère global \mathcal{R}_G dans lequel nous les représenterons toutes. Elles sont caractérisées par des vecteurs aléatoires x_u :

$$x_u = [x \ y \ z \ \theta \ \varphi \ \psi]^T (\in \mathbb{R}^6) \quad (\text{VI.1})$$

$[x \ y \ z]^T$ correspond à une translation et $[\theta \ \varphi \ \psi]^T$ à une représentation minimale d'une orientation. Dans le cas particulier d'un robot se déplaçant dans un plan, ces relations sont des transformations planes. Le vecteur x_u sera alors simplement caractérisé par :

$$x_u = [x \ y \ \theta]^T (\in \mathbb{R}^3) \quad (\text{VI.2})$$

Chaque vecteur aléatoire x_u sera représenté par ses deux premiers moments statistiques \hat{x}_u et σ_{uu} :

$$\hat{x}_u = E[x_u] \quad (\text{VI.3})$$

$$\varepsilon_{x_u} = x_u - \hat{x}_u \quad \sigma_{uu} = E[\varepsilon_{x_u} \varepsilon_{x_u}^T] \quad (\text{VI.4})$$

Il est important de souligner que cette représentation n'impose pas de distribution particulière sur x_u ; toutefois, quand la distribution est gaussienne, ces deux moments suffisent pour la définir complètement.

Nous appelons x_i et x_r les relations représentant respectivement la configuration d'un objet O_i ($i \in \{1, \dots, N\}$) et du robot dans \mathcal{R}_g . Ces relations sont regroupées dans un même vecteur qui constitue la carte stochastique. A l'instant k , nous représentons ainsi cette carte par le vecteur aléatoire \mathbf{X}_k :

$$\mathbf{X}_k = \begin{bmatrix} x_r \\ x_1 \\ \vdots \\ x_N \end{bmatrix}_k \quad (\text{VI.5})$$

La carte stochastique inclut l'espérance mathématique $\hat{\mathbf{X}}_k$ de \mathbf{X}_k et sa matrice de covariance $\mathbf{\Lambda}_k$. Cette carte regroupe l'ensemble de l'information connue sur l'environnement depuis le début et jusqu'à l'instant k . Si $\varepsilon_{\mathbf{X}_k}$ représente l'erreur sur \mathbf{X}_k , nous pouvons écrire le système de la manière suivante :

$$\hat{\mathbf{X}}_k = E[\mathbf{X}_k] \quad (\text{VI.6})$$

$$\mathbf{X}_k = \hat{\mathbf{X}}_k + \varepsilon_{\mathbf{X}_k} \quad \mathbf{\Lambda}_k = E[\varepsilon_{\mathbf{X}_k} \varepsilon_{\mathbf{X}_k}^T] \quad (\text{VI.7})$$

$$\hat{\mathbf{X}}_k = \begin{bmatrix} \hat{x}_r \\ \hat{x}_1 \\ \vdots \\ \hat{x}_N \end{bmatrix}_k \quad \mathbf{\Lambda}_k = \begin{bmatrix} \sigma_{rr} & \sigma_{r1} & \dots & \sigma_{rN} \\ \sigma_{1r} & \sigma_{11} & \dots & \sigma_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{Nr} & \sigma_{N1} & \dots & \sigma_{NN} \end{bmatrix}_k \quad (\text{VI.8})$$

où :

$$\sigma_{uv} = E[\varepsilon_{x_u} \varepsilon_{x_v}^T] \quad (u, v) \in \{r, 1, \dots, N\}^2. \quad (\text{VI.9})$$

Les matrices de variance σ_{rr} et σ_{ii} décrivent l'imprécision¹ des configurations du robot et des objets, tandis que celles de covariance σ_{uv} ($u \neq v$) caractérisent leurs corrélations.

1. par abus de langage, nous l'appellerons aussi incertitude

Ces matrices de covariance permettront, par un mécanisme que nous allons décrire, de mettre à jour l'ensemble du système lorsqu'une relation aura changé. Par ailleurs, Λ_k étant une matrice symétrique, nous avons : $\sigma_{uu} = \sigma_{uu}^T$ et $\sigma_{uv} = \sigma_{vu}^T$.

Pour alléger l'écriture nous omettrons par la suite l'indice du temps k lorsque celui-ci ne s'avérera pas nécessaire.

VI.2 Construction de la carte

VI.2.1 Les outils

Les déplacements du robot et ses perceptions vont modifier la carte stochastique de l'environnement. Deux outils seront alors nécessaires pour réestimer le système $(\hat{\mathbf{X}}, \Lambda)$: la propagation d'incertitude et le filtre de Kalman.

a. Propagation de l'incertitude

Considérons un vecteur \mathbf{x} composé de n transformations aléatoires $(x_i)_{i=1..n}$ et caractérisé par ses deux premiers moments $(\hat{\mathbf{x}}, \sigma_{\mathbf{xx}})$:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad \hat{\mathbf{x}} = E[\mathbf{x}] \quad (\text{VI.10})$$

$$\mathbf{x} = \hat{\mathbf{x}} + \varepsilon_{\mathbf{x}} \quad \text{avec} \quad x_i = \hat{x}_i + \varepsilon_{x_i} \quad (1 \leq i \leq n) \quad (\text{VI.11})$$

$$\sigma_{\mathbf{xx}} = E[\varepsilon_{\mathbf{x}} \varepsilon_{\mathbf{x}}^T] \quad \text{avec} \quad \sigma_{ij} = E[\varepsilon_{x_i} \varepsilon_{x_j}^T] \quad (1 \leq i, j \leq n) \quad (\text{VI.12})$$

Soit \mathbf{f} une fonction vectorielle applicable sur \mathbf{x} . Appelons \mathbf{y} l'image de \mathbf{x} par cette fonction :

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) = \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_p(\mathbf{x}) \end{bmatrix} \quad (\text{VI.13})$$

Le problème de propagation de l'incertitude est celui de l'estimation des deux premiers moments du vecteur aléatoire \mathbf{y} . Dans le cas général d'une fonction non linéaire, l'approche habituelle consiste à utiliser un développement de \mathbf{f} en séries de Taylor :

$$\mathbf{y} = \mathbf{f}(\hat{\mathbf{x}}) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\hat{\mathbf{x}}) \varepsilon_{\mathbf{x}} + \dots \quad (\text{VI.14})$$

où :

$$y_l = f_l(\hat{\mathbf{x}}) + \sum_{i=1}^n \frac{\partial f_l}{\partial x_i}(\hat{\mathbf{x}}) \varepsilon_{x_i} + \dots \quad (1 \leq l \leq p) \quad (\text{VI.15})$$

Soit $\mathbf{F}_{\mathbf{x}}$ la matrice jacobienne de \mathbf{f} par rapport à \mathbf{x} :

$$\mathbf{F}_{\mathbf{x}} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\hat{\mathbf{x}}), \quad \mathbf{F}_{\mathbf{x}} = [\mathbf{F}_1 \dots \mathbf{F}_n] \quad \text{avec} \quad \mathbf{F}_i = \begin{bmatrix} \frac{\partial f_1}{\partial x_i}(\hat{\mathbf{x}}) \\ \vdots \\ \frac{\partial f_p}{\partial x_i}(\hat{\mathbf{x}}) \end{bmatrix} \quad (\text{VI.16})$$

Les approximations des deux premiers moments s'écrivent alors :

$$\hat{y} \simeq f(\hat{x}) \quad (\text{VI.17})$$

$$\sigma_{yy} \simeq \mathbf{F}_x \sigma_{xx} \mathbf{F}_x^T = \sum_{i=1}^n \mathbf{F}_i \sigma_{ii} \mathbf{F}_i^T + \sum_{i \neq j} \mathbf{F}_i \sigma_{ij} \mathbf{F}_j^T \quad (\text{VI.18})$$

Dans l'expression décomposée de σ_{yy} , le dernier terme de droite exprime l'influence des corrélations entre les variables x_i et x_j . Si f est linéaire, les deux premiers moments obtenus sont exactement ceux de y .

Si de plus x est corrélé avec un autre vecteur aléatoire w par une matrice de corrélation $\sigma_{xw} = [\sigma_{1w}^T \cdots \sigma_{nw}^T]^T$, une approximation de la corrélation entre y et w est donnée par :

$$\sigma_{yw} \simeq \mathbf{F}_x \sigma_{xw} = \sum_{i=1}^n \mathbf{F}_i \sigma_{iw} \quad (\text{VI.19})$$

Nous avons notamment le cas particulier $\sigma_{yx} \simeq \mathbf{F}_x \sigma_{xx}$. Dans le cas linéaire, l'équation VI.19 devient à nouveau exacte. Pour alléger la notation, nous écrivons par la suite cette équation ainsi que celles VI.17 et VI.18 avec un symbole égalité.

b. Filtre de Kalman généralisé

Quand une relation (ou un ensemble de relations) est réobservée, un outil de réestimation est nécessaire. Le filtre de Kalman a souvent été utilisé comme estimateur. Initialement, le filtre de Kalman a été développé pour estimer itérativement un vecteur d'état à partir d'un ensemble de données indépendantes et reliées à ce vecteur par une équation de mesure linéaire. Il a été étendu aux fonctions non linéaires en linéarisant autour des estimées de l'état et de la mesure [Bar-Shalom 88, Maybeck 82, Jazwinski 70]. Le filtre de Kalman généralisé est une reformulation établie par R. Chatila et P. Moutarlier qui peut être utilisée dans des cas non linéaires et en présence de bruit corrélés et colorés¹ [Moutarlier 91a]. Ce filtre permet de réestimer tout ou une partie du vecteur d'état en garantissant sa cohérence par le maintien des matrices de corrélation.

Formulation du filtre

Considérons un système représenté par un vecteur d'état $x = [x_1^T \cdots x_n^T]^T$ et ses deux premiers moments (\hat{x}, σ_{xx}) . Une mesure est une relation aléatoire connue que l'on sait vérifiée entre les vraies valeurs des objets x_i de l'état. On la représente par une équation de mesure :

$$z = \mathbf{h}(x_1, \dots, x_n) \quad (\text{VI.20})$$

1. Dans le cas particulier de bruit blanc et non corrélé, les expressions classiques du filtre de Kalman peuvent être retrouvées.

On prédit alors la valeur de la mesure :

$$\hat{\mathbf{z}} = \mathbf{h}(\hat{x}_1, \dots, \hat{x}_n) \quad (\text{VI.21})$$

L'ancienne estimée du vecteur $\hat{\mathbf{x}}$ est alors corrigée d'une valeur proportionnelle à l'écart de mesure δ entre la prédiction $\hat{\mathbf{z}}$ et la valeur mesurée \mathbf{z} . Si nous appelons $\mathbf{H}_x = [\mathbf{H}_1 \dots \mathbf{H}_n]$ la jacobienne de \mathbf{h} par rapport à \mathbf{x} , nous calculons :

la nouvelle estimée :

$$\hat{\mathbf{x}}' = \hat{\mathbf{x}} + \Gamma_{\mathbf{xz}} \Gamma_{\mathbf{zz}}^{-1} \delta \quad (\text{VI.22})$$

avec

$$\delta = \mathbf{z} - \hat{\mathbf{z}}$$

$$\Gamma_{\mathbf{xz}} = E[\varepsilon_{\mathbf{x}} \cdot \delta^T] \quad \text{avec} \quad \Gamma_{x_i z} = \sum_{j=1}^n \sigma_{ij} \mathbf{H}_j^T$$

$$\Gamma_{\mathbf{zz}} = E[\delta \cdot \delta^T] = \sum_{i=1}^n \sum_{j=1}^n \mathbf{H}_i \sigma_{ij} \mathbf{H}_j^T$$

la variance de la nouvelle estimée :

$$\sigma_{\mathbf{x}'\mathbf{x}'} = \sigma_{\mathbf{xx}} - \Gamma_{\mathbf{xz}} \Gamma_{\mathbf{zz}}^{-1} \Gamma_{\mathbf{zx}}^T \quad (\text{VI.23})$$

et sa corrélation avec une autre relation x_j :

$$\sigma_{\mathbf{x}'x_j} = \sigma_{\mathbf{xx}_j} - \Gamma_{\mathbf{xz}} \Gamma_{\mathbf{zz}}^{-1} \Gamma_{x_j z}^T \quad (\text{VI.24})$$

avec

$$\Gamma_{x_j z} = E[\varepsilon_{x_j} \cdot \delta^T]$$

Dans le cas d'équations de mesures non-linéaires, nous pouvons affiner l'estimée $\hat{\mathbf{x}}$, autour de laquelle on effectue la linéarisation, par un processus itératif [Smith 90] :

$$\hat{\mathbf{x}}^{i+1} = \hat{\mathbf{x}} + \Gamma_{\mathbf{xz}}^i \Gamma_{\mathbf{zz}}^{i-1} \delta^i, \quad \hat{\mathbf{x}}^0 = \hat{\mathbf{x}} \quad (\text{VI.25})$$

$$\delta^i = \mathbf{z} - (\mathbf{h}(\hat{\mathbf{x}}^i) + \mathbf{H}_x^i (\hat{\mathbf{x}} - \hat{\mathbf{x}}^i)) \quad (\text{VI.26})$$

Les matrices $\Gamma_{\mathbf{xz}}^i$, $\Gamma_{\mathbf{zz}}^{i-1}$ et \mathbf{H}_x^i sont calculées en $\hat{\mathbf{x}}^i$. La dernière itération f étant fixée par une limite sur le terme $\frac{\|\delta^{i+1} - \delta^i\|}{\|\delta^i\|}$ (et sur le nombre d'itérations), nous obtenons l'estimée finale et nous pouvons alors calculer directement les matrices de variance et de corrélation :

$$\hat{\mathbf{x}}' = \hat{\mathbf{x}}^f \quad (\text{VI.27})$$

$$\sigma_{\mathbf{x}'\mathbf{x}'} = \sigma_{\mathbf{xx}} - \Gamma_{\mathbf{xz}}^f \Gamma_{\mathbf{zz}}^{f-1} \Gamma_{\mathbf{zx}}^{fT} \quad (\text{VI.28})$$

$$\sigma_{x'x_j} = \sigma_{xx_j} - \Gamma_{x\mathbf{z}}^f \Gamma_{\mathbf{z}\mathbf{z}}^{f-1} \Gamma_{\mathbf{z},x}^{fT} \quad (\text{VI.29})$$

La figure VI.1 montre le principe du filtre de Kalman itératif dans le cas unidimensionnel. C_h est la courbe définie par la fonction de mesure h . A chaque étape i , on calcule la tangente T_i de C_h en $(\hat{x}^i, h(\hat{x}^i))$. δ^i correspond à l'écart entre la valeur z et l'intersection de T_i avec la droite verticale d'équation $x = \hat{x}$. La formule VI.25 permet alors de calculer \hat{x}^{i+1} , et ainsi de suite. On tend vers la meilleure estimée \hat{x}^* .

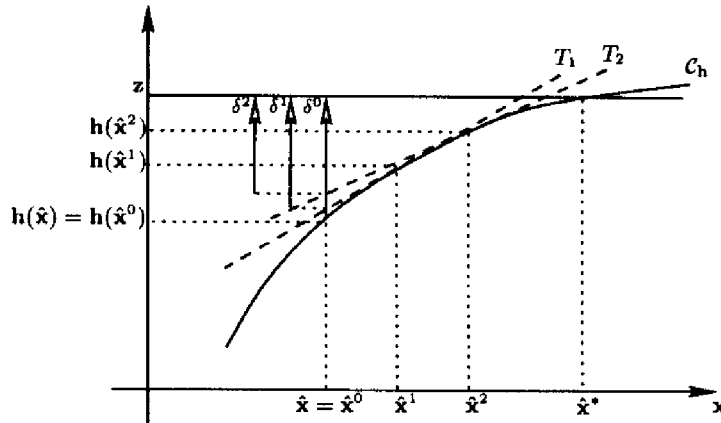


FIG. VI.1 – Estimation itérative du point de linéarisation. Cas unidimensionnel.

Propriétés du filtre

Si les distributions sont gaussiennes, le filtre est l'estimateur bayésien du maximum de vraisemblance *a posteriori* (i.e. une fois donnée la première statistique \mathbf{x} et la statistique de la mesure \mathbf{z}). Aucun filtre, linéaire ou non linéaire, ne peut donner une estimée avec une plus petite variance.

Si les distributions ne sont pas gaussiennes, alors le filtre n'est pas optimal, mais il est de *tous* les estimateurs linéaires, celui qui donne l'estimation non biaisée de variance minimale.

La formulation ci-dessus montre que pour réestimer une relation x_j , il suffit qu'elle soit corrélée à au moins une relation de la mesure (i.e. $\Gamma_{x_j, \mathbf{z}} \neq 0$), quand bien même celle-ci ne ferait pas partie de la mesure. Par ailleurs, après application du filtre, toutes les relations impliquées dans la mesure deviennent corrélées (ou leurs corrélations évoluent si elles l'étaient déjà).

Ce filtre a fait l'objet de nombreuses applications robotiques dans notre équipe, pour la modélisation incrémentale bien entendu [Moutarlier 91a, Nashashibi 93a, Fillatreau 94], mais aussi pour la stéréovision dynamique [Herrb 91], le calibrage inter-capteur, la fusion multisensorielle et la localisation d'objets [Grandjean 91a, Boumaza 94].

VI.2.2 Les trois événements élémentaires

La carte stochastique (\hat{X}, Λ) est modifiée à trois occasions :

- a - quand le robot change d'emplacement ;
- b - quand le robot perçoit un nouvel objet ;
- c - quand le robot reperçoit un objet de la carte.

Pour ces trois actions, nous utiliserons la fonction de composition de deux transformations incertaines x_i et x_j . Nous appelons γ cette fonction :

$$\gamma : (x_i, x_j) \mapsto x_k = \gamma(x_i, x_j) = x_i \circ x_j \quad (\text{VI.30})$$

Nous notons Υ la matrice jacobienne de γ par rapport à (x_i, x_j) . Elle sera parfois décomposée en deux sous matrices Υ_i et Υ_j telles que $\Upsilon = [\Upsilon_i \Upsilon_j]$ (cf équation VI.16). Les expressions de ces matrices sont données en annexe B.1 dans les cas 2D et 3D. Cette fonction est non linéaire pour les deux raisons suivantes :

1. la présence de fonctions trigonométriques ;
2. le produit de fonctions.

En revanche, nous pouvons remarquer que dans le cas où les erreurs angulaires sont nulles (*i.e.* les paramètres de rotation ne sont pas des variables aléatoires), γ est une fonction linéaire.

Examinons maintenant comment la carte stochastique évolue pour chaque action élémentaire.

a. Le robot change d'emplacement

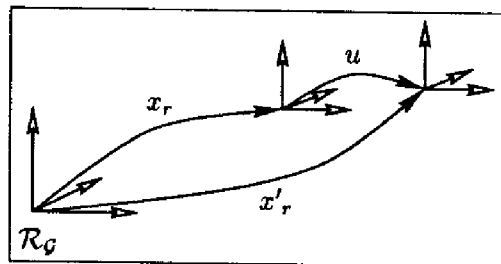


FIG. VI.2 – Le robot se déplace. u est le déplacement relatif, x_r l'ancienne position et x'_r la nouvelle position.

Le robot, initialement à la position x_r dans le repère global \mathcal{R}_G , effectue un déplacement relatif u pour arriver à la nouvelle position x'_r (cf figure VI.2) :

$$x'_r = \gamma(x_r, u) \quad (\text{VI.31})$$

Si \hat{u} est l'estimée du déplacement (i.e. la valeur de son odométrie), l'outil de propagation appliqué à γ (cf équations VI.17 et VI.18) permet de calculer la nouvelle estimée de la position du robot et son incertitude :

$$\hat{x}'_r = \gamma(\hat{x}_r, \hat{u}) \quad \sigma'_{rr} = \Upsilon \begin{bmatrix} \sigma_{rr} & \sigma_{ru} \\ \sigma_{ur} & \sigma_{uu} \end{bmatrix} \Upsilon^T \quad (\text{VI.32})$$

Le déplacement u étant relatif, il ne dépend pas de l'ancienne position (ni des objets de la carte), soit :

$$\sigma_{ru} = \sigma_{ur} = 0 \quad (\text{VI.33})$$

En décomposant Υ en Υ_r et Υ_u , nous obtenons :

$$\sigma'_{rr} = \Upsilon_r \sigma_{rr} \Upsilon_r^T + \Upsilon_u \sigma_{uu} \Upsilon_u^T \quad (\text{VI.34})$$

La corrélation entre la nouvelle position du robot et la position des objets devient (cf équation VI.19) :

$$\sigma'_{ri} = \Upsilon_r \sigma_{ri} \quad (\text{VI.35})$$

La matrice Λ étant symétrique, il y a $N + 1$ matrices à réestimer.

b. Le robot perçoit un nouvel objet

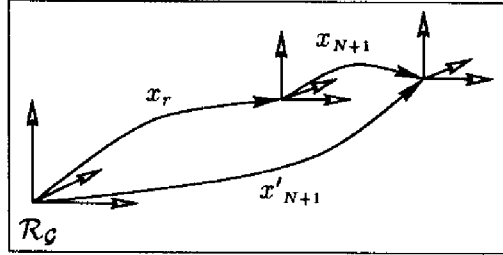


FIG. VI.3 – Le robot observe un nouvel objet. x_r est la position courante du robot, x_{N+1} la position de l'objet exprimée dans le repère robot, et x'_{N+1} sa position exprimée dans le repère global \mathcal{R}_G .

Si la position d'un nouvel objet est estimée dans le repère capteur, nous la réestimons dans le repère robot (en la composant avec la position calibrée du capteur). C'est aussi par un calcul de composition que nous transformons la position de cet objet x_{N+1} dans le repère robot, en sa position x'_{N+1} dans le repère global \mathcal{R}_G (cf figure VI.3) :

$$x'_{N+1} = \gamma(x_r, x_{N+1}) \quad (\text{VI.36})$$

soit :

$$\hat{x}'_{N+1} = \gamma(\hat{x}_r, \hat{x}_{N+1}) \quad \sigma'_{N+1,N+1} = \Upsilon \begin{bmatrix} \sigma_{rr} & \sigma_{r,N+1} \\ \sigma_{N+1,r} & \sigma_{N+1,N+1} \end{bmatrix} \Upsilon^T \quad (\text{VI.37})$$

La position de l'objet dans le repère robot étant indépendante de la position du robot, nous écrivons :

$$\sigma_{N+1,r} = \sigma_{r,N+1} = 0 \quad (\text{VI.38})$$

$$\sigma'_{N+1,N+1} = \Upsilon_r \sigma_{rr} \Upsilon_r^T + \Upsilon_{N+1} \sigma_{N+1,N+1} \Upsilon_{N+1}^T \quad (\text{VI.39})$$

Les corrélations de ce nouvel objet avec le robot et les autres objets deviennent :

$$\sigma'_{N+1,r} = \Upsilon_r \sigma_{rr} \quad \sigma'_{N+1,i} = \Upsilon_r \sigma_{ri} \quad (\text{VI.40})$$

Ces corrélations sont bien sûr nulles si l'objet a été perçu dans le repère global \mathcal{R}_G . C'est notamment le cas si le robot est à sa première position et que celle-ci est prise comme repère global.

L'estimée de l'objet ainsi que ses termes de variance et de corrélation (soit $N + 1$ matrices différentes) sont ensuite ajoutés au système (\hat{X}, Λ) .

c. Le robot repère un objet

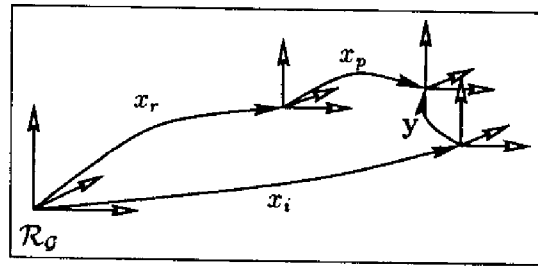


FIG. VI.4 – Le robot repère un objet. x_r est la position courante du robot, x_p celle de l'objet tel qu'il vient d'être perçu (exprimée bien sûr dans le repère robot), x_i sa position déjà connue dans le repère global \mathcal{R}_G et z la mesure.

Le robot perçoit un objet x_p (exprimé dans son repère) qui correspond à l'objet x_i du modèle (cf figure VI.4). Nous traiterons au chapitre VII la méthode permettant d'établir cette correspondance. Nous utilisons ici le filtre de Kalman pour réestimer non seulement les positions et orientations du robot et de cet objet, mais aussi celles de tous les objets qui leur sont corrélés. Nous devons donc avoir la relation :

$$\mathbf{y} = \gamma(x_r, x_p) - x_i = 0 \quad (\text{VI.41})$$

Appelons $\mathbf{z} = \mathbf{y} - \hat{\mathbf{y}} = -\hat{\mathbf{y}}$ la mesure et $\hat{\mathbf{z}} = 0$ la prédiction, on applique alors les équations du filtre de Kalman pour réestimer la position du robot et de l'objet (mais bien entendu aussi celle des objets qui leurs sont corrélés).

VI.3 Réalisations expérimentales

R. Chatila et J. P. Laumond [Chatila 85] ont introduit la carte stochastique. R. Smith, M. Self et P. Cheeseman [Smith 90] ont développés le formalisme en considérant un bruit blanc sur les données des capteurs et ils ont montré comment des contraintes incertaines entre les objets pouvaient être prises en compte. Toutefois, des problèmes de biais sur la position du robot peuvent amener des instabilités dans la construction de la carte si l'on traite directement l'ensemble du vecteur d'état.

Pour résoudre ce problème, P. Moutarlier et R. Chatila [Moutarlier 89b] ont utilisé une procédure plus robuste de recalage-fusion. Dans un premier temps, ils réestiment la position du robot sur la base de tous les objets revus puis, dans un second temps, celle de ces mêmes objets et de ceux qui leur sont corrélés. Cela permet de réduire le biais éventuel de la position du robot et de ne pas le propager à l'ensemble du vecteur d'état lors des fusions. Cette procédure semble tout à fait pertinente dans le cas d'environnements naturels où les biais notamment causés par le glissement des roues peuvent être très importants. Cette stratégie a aussi été adoptée par N. Ayache [Ayache 89] avec la différence que le modèle n'est pas reconstruit dans un repère global, mais uniquement dans le repère courant du robot.

L'approche recalage-fusion est une solution sous-optimale au problème de construction de carte, mais elle accroît cependant la stabilité du processus. P. Moutarlier et R. Chatila ont observé expérimentalement une meilleure qualité de la carte construite par rapport à l'approche directe [Moutarlier 91a]. Ils ont montré que dans le cas linéaire les mesures peuvent être incorporées dans un ordre quelconque ; en revanche, dans le cas non linéaire, cet ordre n'est pas neutre. Dans ce cas, il convient de considérer en premier lieu les mesures les plus précises afin de réduire au mieux ce biais.

Le maintien des corrélations augmente la charge de calcul, aussi certains auteurs tentent d'éviter leur traitement. J. Leonard et H. Durrant-Whyte [Leonard 92] proposent une procédure sous-optimale selon laquelle ces corrélations seront négligées. Ils évaluent la précision obtenue à partir d'une mesure. Quand la position du robot est suffisamment précise, ils mettent la carte à jour. Si la précision de la localisation des objets est suffisante, ils recalent le robot. Dans les autres cas, cette mesure est ignorée.

Dans la section suivante, nous étudierons plus précisément l'apport et le risque de la prise en compte des corrélations ainsi que différents problèmes liés à la construction de carte.

VI.4 Discussion

P. Hébert a développé un simulateur 2D de construction de cartes stochastiques pour vérifier les résultats de ses analyses en simulation. Une partie du travail a été présentée dans la communication [Hébert 95b] à laquelle nous avons participé. Nous reprenons dans cette section ces résultats afin d'illustrer les différentes méthodes évoquées de construction de carte.

Ce simulateur nécessite :

1. Une carte idéale des objets de l'environnement (i.e. un ensemble de triplets (x_i, y_i, θ_i)).
2. Un modèle d'erreur de capteur (sous forme d'une matrice de covariance). Au cours de la simulation, chaque mesure est perturbée par un bruit gaussien de moyenne nulle et fonction de cette matrice.
3. Un modèle de l'odométrie. L'erreur en translation est proportionnelle à la distance parcourue, celle en rotation étant constante.
4. Une trajectoire (i.e. un ensemble de commandes appliquées au robot).

Nécessité du maintien des corrélations

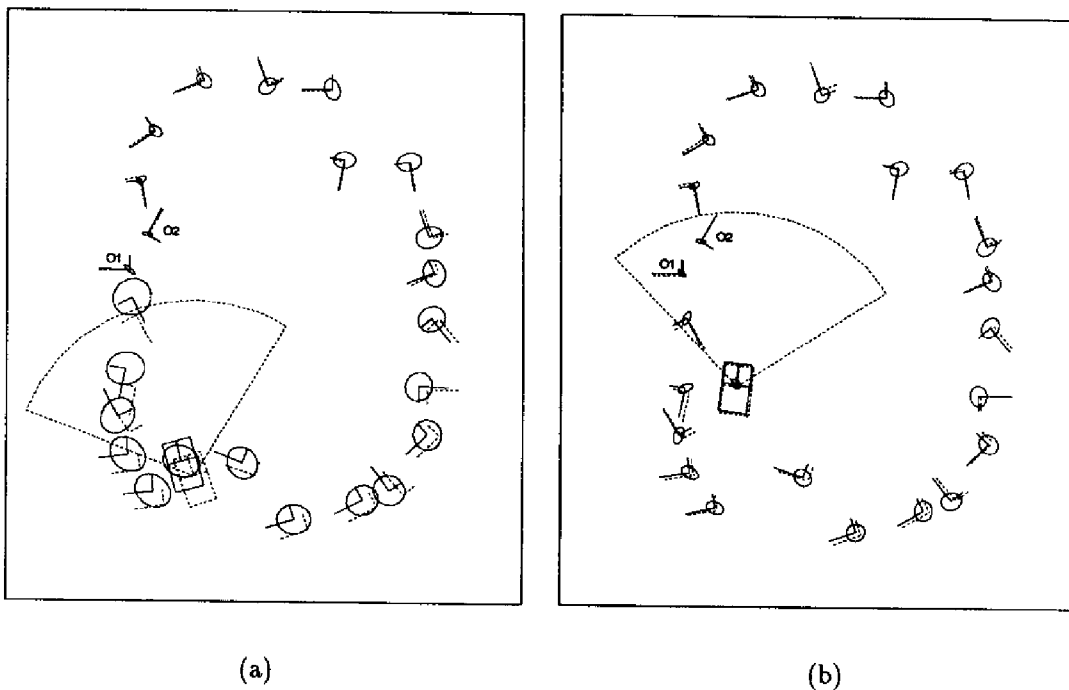


FIG. VI.5 – Apport des corrélations pour la construction de la carte. Le robot effectue une boucle (dans le sens horaire) en commençant par observer les objets $O1$ et $O2$. En pointillés : position réelle du robot et des objets. En traits continus : position estimée et incertitude du robot et des objets. **Figure (a)** : avant réobservation des objets $O1$ et $O2$. **Figure (b)** : après réobservation de ces objets. Le robot et l'ensemble de la carte ont bien été corrigés, et leurs incertitudes ont diminué.

En premier lieu, nous allons montrer, à travers l'exemple suivant, l'intérêt de la prise en compte des corrélations pour la construction d'une carte fiable de l'environnement. Nous avons créé un monde d'objets dans lequel le robot effectue une trajectoire en forme de

boucle. L'erreur odométrique est fixée à 5% de la distance parcourue pour la translation et à 7 degrés pour l'orientation. La figure VI.5 représente :

- les repères des objets (par des formes en "L"). Les objets x_i de la carte stochastique construite par le robot sont dessinés en trait continu avec une ellipse caractérisant l'incertitude sur leur position. Cette incertitude est calculée à partir de σ_{ii} comme précisé dans l'annexe A.2 et en considérant une distribution gaussienne dessinée à 99%. La position et l'orientation réelle des objets est représentée par un repère en pointillés (et bien entendu sans ellipse d'incertitude).
- Le robot (par une forme rectangulaire). La position où il estime se trouver (*i.e.* le repère x_r de la carte stochastique) est dessinée en traits continus, l'ellipse correspond à l'incertitude de sa position (calculée à partir de σ_{rr} et en considérant de même une distribution gaussienne à 99%). La position réelle du robot est représentée en pointillés avec le champ de vue de la perception.

Le robot effectue une boucle (dans le sens horaire) en commençant par observer les objets O1 et O2 et en revenant vers ceux-ci à la fin de sa trajectoire. La figure VI.5 (a) représente l'état du système avant la réobservation de ces deux objets. On y remarque une déformation significative de la carte due aux erreurs accumulées sur la position du robot au cours de ses déplacements (et ce malgré les recalages). Le robot avance à nouveau et réobserve les premiers objets O1 et O2. Ceux-ci ayant une grande précision dans le repère global (car perçus au tout début) ils vont apporter une correction importante sur la position du robot et, via les corrélations, sur l'ensemble de la carte. Les résultats du recalage et des fusions sont montrés figure VI.5 (b). On remarque comment la position du robot a été bien corrigée mais aussi, à travers les corrélations, celle de *tous* les objets de la carte. De même, on constate que l'incertitude du robot et celle de *tous* les objets a aussi diminué.

Cet exemple illustre l'intérêt de prendre en compte les corrélations pour la modélisation incrémentale de l'environnement. Revenant sur le problème du trajet en boucle mentionné à la partie III.1, nous voyons ici que *l'utilisation des corrélations y apporte une solution et permet d'assurer la cohérence globale du modèle.*

Nous allons aussi montrer comment la négligence des corrélations peut conduire à des incohérences au niveau des incertitudes. A chaque observation, le robot met à jour l'ensemble des matrices de covariance. Lorsque les objets sont considérés comme non corrélés, leur incertitude tendent à être sous-estimées. Une simple simulation permet de l'illustrer. A une position donnée dans l'environnement, le robot observe un objet. L'incertitude de cet objet est obtenue par l'équation VI.39. Le robot réobserve ensuite cet objet un certain nombre de fois et met à jour, à chaque fois, la position de l'objet et son incertitude ainsi que sa propre position et incertitude. Il utilise pour cela le filtre de Kalman. Les figures VI.6 (a), (b) et (c) illustrent l'évolution du système après la première, la cinquième et la dixième perception lorsque les corrélations sont prises en compte. Les ellipses dessinées correspondent à des distributions gaussiennes à 99%. Lorsque les corrélations sont considérées, l'incertitude sur la position de l'objet converge vers l'incertitude sur la position du robot à la première observation. Dans l'équation de mesure VI.41, \mathbf{Y}_r est la jacobienne de

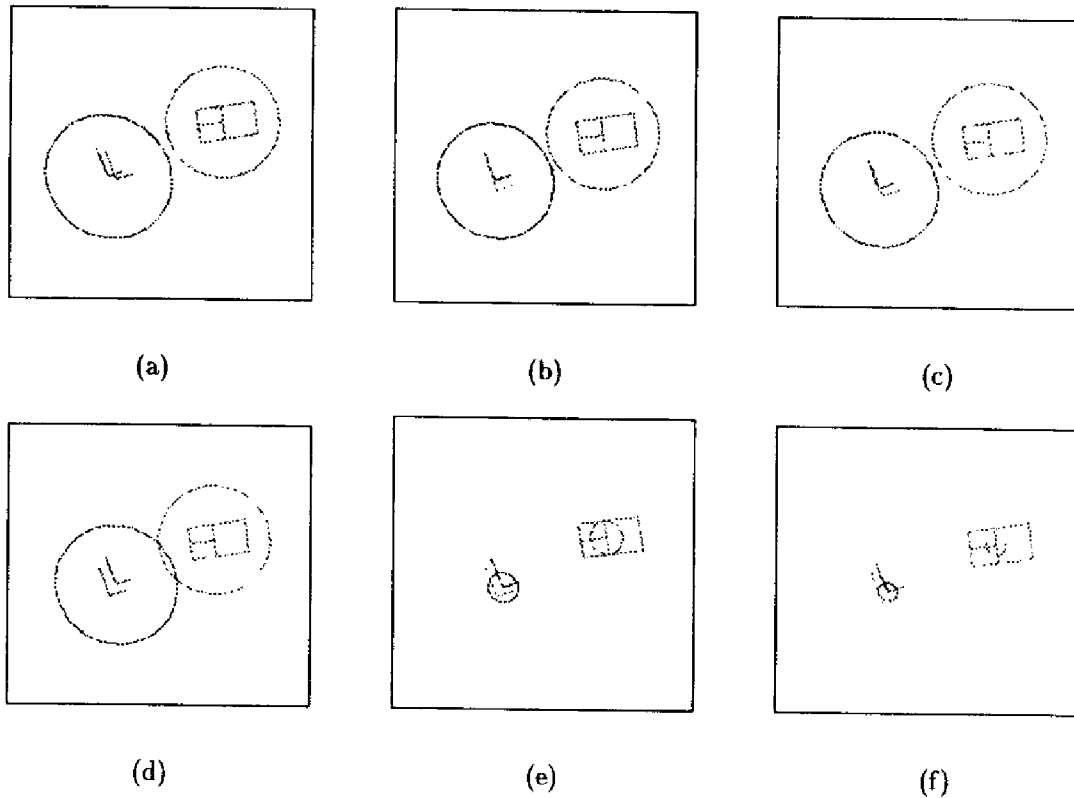


FIG. VI.6 - Evolution de l'incertitude avec et sans corrélations. Les figures (a) à (c) sont obtenues en prenant en compte les corrélations, tandis que les figures (d) à (f) sont obtenues avec la même procédure mais en négligeant ces corrélations.

γ par rapport à x_r et $-\mathbf{I}$ celle de γ par rapport à x_i (\mathbf{I} étant la matrice identité). Nous avons donc :

$$\Gamma_{x_i z} = \sigma_{rr} \Upsilon_r^T - \sigma_{ii} = 0 \quad (\text{VI.42})$$

Comme $\Gamma_{x_i z}$ est nulle, les matrices de covariance de la première ligne et de la première colonne de Λ ne changent pas. La mesure z dépend du robot mais ne lui est pas corrélée.

Lorsque les corrélations sont négligées, l'ensemble des objets inclus dans la mesure verront leurs incertitudes se réduire après chaque application du filtre. Ainsi, à la fois l'incertitude des objets et celle du robot tendent vers 0. C'est bien ce que montrent les figures VI.6 (d), (e) et (f).

Difficultés du maintien des corrélations

Nous avons vu pourquoi la prise en compte des corrélations est essentielle à la construction de la carte. Mais il est bien évident que leurs estimations doivent être satisfaisantes

afin de ne pas propager d'erreur à l'ensemble du modèle.

Ces corrélations sont calculées à partir des linéarisations faites à travers l'outil de propagation d'une part et le filtre de Kalman d'autre part. Il est essentiel que ces approximations soient suffisantes pour estimer les deux premiers moments des positions des objets et du robot. En réalité, ces linéarisations sont valides lorsque les erreurs angulaires sont faibles (cf VI.30). Une erreur dans l'estimation des seconds moments induit un biais sur la position du robot et par conséquent sur la carte (voir [Moutarlier 91a] p.47 pour la démonstration). De plus les linéarisations peuvent être calculées autour de mauvais points ce qui fausserait aussi l'estimation des corrélations. Dans ce cas, l'information serait mal propagée à l'ensemble de l'état. Une solution classique permettant de limiter les effets néfastes des linéarisations réside, comme nous l'avons vu, dans l'utilisation d'un filtre de Kalman itératif. D'autre part, citons S. Julier et J. Uhlmann [Julier 95] qui proposent une autre solution consistant à estimer le second moment par intégration de la transformation d'un ensemble de points.

Un autre problème persiste toutefois et concerne l'estimation de la position du robot fournie par l'odométrie. Notamment en raison de la nature imprévisible des interactions des roues avec le sol, nous avons soulevé la difficulté à développer un modèle probabiliste du déplacement du robot. Puisque les roues peuvent glisser, la distribution n'a pas alors nécessairement une moyenne nulle. Une solution consiste à surestimer l'incertitude sur l'odométrie, mais cela pourrait poser des problèmes dans le cas des orientations.

Dans notre cas, nous bénéficions pour nos expérimentations avec le robot Adam de l'avantage d'un système de centrale inertielle. Grâce à celui-ci, les erreurs d'orientation du robot restent relativement faibles (dérive de 1 à 3 degrés par heure). Nous pouvons alors surestimer l'erreur au niveau de la position du robot (soit 2% de la distance parcourue) tout en conservant la validité des linéarisations. De surcroît, l'utilisation permanente de la méthode de recalage-fusion permettra de minimiser les effets des biais sur la position.

VI.5 Utilisation de la carte stochastique en environnement naturel

Nous allons ainsi utiliser la carte stochastique pour représenter l'environnement et la position du robot. Nous prendrons en compte les corrélations entre les objets et le robot afin d'assurer la cohérence globale de notre modèle. Mais pour éviter les problèmes liés aux non linéarités des fonctions et au biais sur l'odométrie induit par le glissement des roues, nous utiliserons respectivement la version itérative du filtre de Kalman et la technique de recalage-fusion.

Par ailleurs, il est essentiel pour la navigation du robot et pour sa sécurité que sa position soit bien estimée. Nous avons vu l'influence réciproque de tous les objets présents dans la carte (via les liens de corrélation). Cela permet notamment, à partir d'un objet, de remettre à jour l'ensemble de la carte. Mais il y a un risque majeur : celui de corrompre toute la carte (et la position du robot) si un objet est mal estimé. Il faut donc s'assurer de la fiabilité des objets qui seront pris en compte. Aussi **nous n'intégrerons que les amers**

dans la carte stochastique. Ceux-ci présentent en effet une caractéristique précise et réobservable, ils sont de plus remarquables dans l'environnement. Ils permettent ainsi de garantir la robustesse de notre modèle stochastique.

Chapitre VII

Construction du modèle de l'environnement

Nous avons étudié dans la partie précédente comment le robot élaborait le modèle local d'une scène et comment il en sélectionnait les amers. A partir de ces modèles locaux et au fur et à mesure de ses déplacements, il va construire le modèle global de l'environnement (notamment avec une représentation de type carte stochastique). Le principe de la boucle de modélisation basée sur des objets a déjà été brièvement introduit au chapitre III (la figure III.4, p. 32, en montre les principaux éléments). Trois fonctionnalités forment le noyau de cette boucle : la reconnaissance, la localisation (ou le recalage) et la fusion. Au cours de ce chapitre, nous allons décrire successivement chacune de ces fonctionnalités. Nous présenterons alors le modèle par objets de l'ensemble de l'environnement (*i.e.* le modèle global) ainsi que sa structure.

Mais avant tout, revenons sur les trois fonctionnalités du processus de modélisation incrémentale. Le schéma de la figure III.4 les représentait en un seul bloc que nous tentons d'explicitier ici.

En premier lieu, le robot cherche à reconnaître *uniquement* les amers, ce qui lui permet de réestimer sa position. Mais, depuis cette nouvelle position, il est possible qu'il puisse à nouveau reconnaître d'autres amers. Il convient alors d'effectuer une boucle de reconnaissance d'amers et de recalage comme le montre la figure VII.1.

Le robot, bénéficiant de la meilleure estimée de sa position, cherchera ensuite à reconnaître les objets qui ne sont pas des amers.

Enfin, il procédera à la fusion des amers et des objets revus, et à l'ajout dans le modèle global des nouveaux objets s'il y en a.

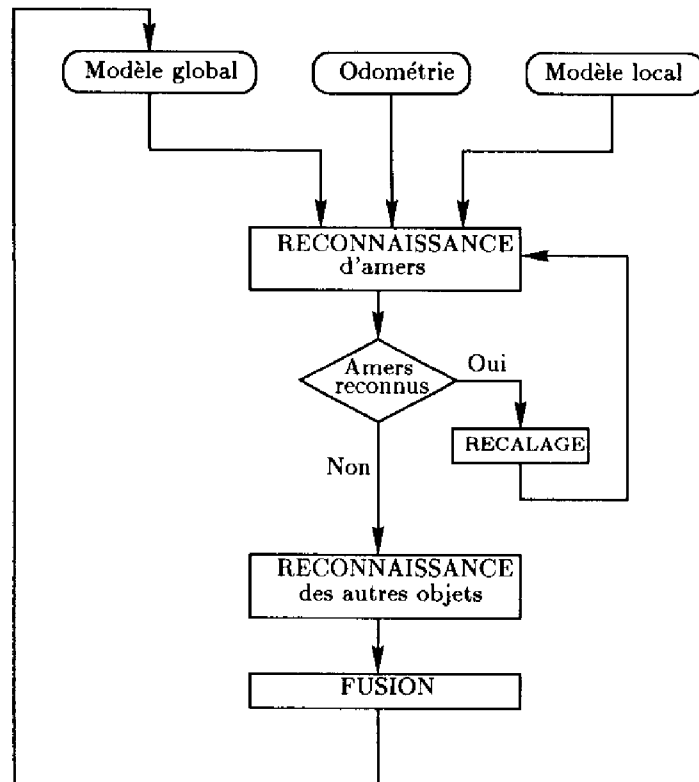


FIG. VII.1 – *Algorithme de modélisation incrémentale. Le robot réestime sa position tant qu'il reconnaît des amers (i.e. qu'il apparie des amers du modèle local avec ceux du modèle global). Le principe de recalage-fusion permet alors de ne corriger que la position du robot (et pas celle des amers). Ce n'est qu'une fois tous les amers reconnus, que le robot cherche à reconnaître les autres objets (utilisant la meilleure estimée de sa position). Enfin, à l'étape de fusion, il réestime les modèles des amers reconnus (fusionnant notamment leur sommet) et celui des autres objets.*

VII.1 Reconnaissance des amers et des objets

La reconnaissance des objets (et particulièrement celle des amers) est un problème essentiel pour la construction du modèle de l'environnement. C'est notamment par la mise en correspondance des amers du modèle local avec ceux du modèle global que sera réestimée la position du robot. Il existe de nombreuses méthodes d'appariement (cf [Moutarlier 91a, Fillatreau 94, Grandjean 91a, Grimson 85] pour une présentation des principales méthodes utilisées). Elles se basent principalement sur les deux critères complémentaires suivants :

- la comparaison de la forme globale des objets (*i.e.* leurs paramètres intrinsèques) ;
- la comparaison de la position relative des objets dans le but d'avoir un ensemble cohérent d'appariements.

VII.1.1 Approches connues

Nous présentons ici succinctement les méthodes les plus intéressantes dans le cadre de notre problématique. Nous les classons en trois catégories.

Méthodes exhaustives

Ces méthodes testent *toutes* les contraintes entre *toutes* les primitives.

Citons en premier lieu la méthode des *cliques maximales* qui consiste tout d'abord à établir un graphe de compatibilité des appariements (les nœuds sont les appariements possibles et les arcs expriment la compatibilité entre deux appariements). On extrait ensuite de ce graphe des sous graphes complets maximaux¹ (appelés cliques maximales) qui correspondent à des hypothèses d'appariements localement cohérentes. Cette technique a été utilisée dans notre équipe par [Colly 91, Pampagnin 90] pour la reconnaissance d'objets polyédriques.

Les techniques de *relaxation* permettent de trouver par un processus itératif les appariements les plus cohérents. A chaque primitive du modèle sont associées plusieurs primitives de la perception ainsi que des poids reflétant leur degré de cohérence. L'algorithme de recherche réévalue itérativement ces poids et doit converger vers la solution où les appariements sont les plus cohérents.

Citons enfin la *transformée de Hough*. Basée sur la construction d'histogrammes dans l'espace des paramètres, elle a été appliquée avec succès en 2D dans notre équipe [Moutarlier 91a].

En revanche, les méthodes exhaustives ont l'inconvénient de présenter une complexité exponentielle et elles ne sont en réalité envisageables que dans la mesure où le nombre de primitives à appairer est réduit.

Méthodes de recherche arborescente

Ces méthodes consistent en une recherche dans un arbre où chaque nœud correspond à un essai d'appariement entre un objet de la perception et un objet du modèle. La recherche peut se faire en profondeur. Dans ce cas l'algorithme évalue pour chaque nœud la conformité des paramètres intrinsèques des deux objets et s'assure que les contraintes

1. au sens de l'inclusion

relatives aux autres appariements déjà rassemblés sont valides. Si ces deux conditions sont remplies, la recherche dans cette branche se poursuit. L'arbre peut être développé *par les données perceptuelles* ("search-tree") ou *par le modèle* ("interpretation tree"). Ces méthodes arborescentes présentent l'avantage de garantir à tout moment la cohérence globale des hypothèses et de pouvoir focaliser la recherche en abandonnant certaines branches ("tree pruning").

Méthodes de génération et vérification d'hypothèses

Ce paradigme est très utilisé en Intelligence Artificielle pour la reconnaissance d'objet par vision monoculaire [Grandjean 91a, Bolles 82, Faugeras 83]. Dans une première phase de *génération*, on n'étudie qu'un sous-ensemble minimal d'appariements, ce qui permet d'émettre des hypothèses de *localisation*. Ces différentes hypothèses sont ensuite testées séquentiellement dans une seconde phase de *vérification* ce qui permet d'en garder les meilleures.

VII.1.2 Approche choisie

Dans notre cas le nombre d'objets réobservés est relativement faible. Rappelons qu'au niveau de leur représentation, nous ne conservons que très peu d'informations (modèle grossier et éventuellement superquadrique) et que ceux-ci ont des formes assez voisines.

Dans un premier temps nous nous sommes intéressés à la comparaison des ellipsoïdes des objets du modèle global avec ceux du modèle local [Betgé-Brezetz 95]. Pour cela nous avons défini différentes mesures estimant la distance des ellipsoïdes entre eux ou leur éventuelle intersection. En fonction de l'occultation des objets, seules certaines de ces mesures étaient retenues. L'appariement ou non des objets était décidé en fonction de leurs valeurs. Cette méthode présentait l'avantage de tenir compte de l'ensemble de la forme de l'objet dans la reconnaissance. Mais elle a toutefois montré certaine limite dans le cas d'environnements très encombré ou lorsque l'estimée de la position du robot est peu fiable.

Pour tenir compte de ces éléments, nous nous sommes orientés vers une deuxième solution, basée sur l'idée simple selon laquelle les objets ayant des formes voisines, leur disposition relative est un critère plus discriminant. Nous choisissons alors la stratégie suivante : prenant en compte la position du robot, son incertitude et son champ de vue, *nous cherchons dans la perception la plus grande structure d'objets que l'on peut retrouver dans le modèle*. Nous appelons ici "structure" un ensemble d'objets dont leur position relative est déterminée. Etudions maintenant quelle caractéristique est utilisée pour estimer la position d'un objet.

Dans le cas des amers, nous savons que leur sommet est précis, ce dernier est donc un bon estimateur de position.

En ce qui concerne les objets qui ne sont pas des amers nous opérons une deuxième distinction pour différencier parmi eux, ceux qui seraient particulièrement étendus (*i.e.* les collines, les grands monticules de rochers, les larges haies...) de ceux qui n'auraient

pas été sélectionnés amers parce qu'occultés ou vus de trop loin (et donc trop imprécis). L'identification des objets de très grande étendue se fait simplement sur la base de la dimension du modèle grossier. Ces objets (qui pourraient éventuellement être assimilables à du terrain) ne sont pas pris en compte actuellement dans le processus de reconnaissance. Nous reviendrons sur le traitement de ces derniers à la fin de ce chapitre. En ce qui concerne les autres objets, nous utilisons aussi leur sommet comme estimateur de position, mais ils ne sont bien évidemment pas utilisés pour le recalage du robot. Nous dirons toutefois de ces objets qu'ils "possèdent un sommet".

Finalement, la reconnaissance des objets (amers ou non) est ramenée à une même procédure d'appariement de deux ensembles de points (pour lesquels on dispose d'une matrice d'incertitude). Le principe de cette procédure est de rechercher la plus grande structure de points commune à ces deux ensembles. Elle est réalisée par une méthode à deux étapes.

Première étape : construction de la table de correspondance

Dans un premier temps, nous déterminons quel sous-ensemble d'objets (du modèle global) peut être perçu en fonction de la position du robot, de son incertitude et de son champ de vue (cette procédure n'est pas obligatoire, mais elle permettra de gagner du temps dans la reconnaissance). Nous regardons ensuite indépendamment tous les appariements possibles entre ces objets et ceux de la perception pour construire une table de correspondance. Cette dernière est un tableau à deux entrées (l'un pour les objets du modèle global, l'autre pour ceux du modèle local) qui indique tous les couples d'appariements possibles. Nous calculons pour cela une distance de Mahalanobis entre chaque couple de sommets. Soit x_p la position du sommet d'un objet du modèle local (et exprimée dans le repère robot), x_m celle du sommet d'un objet du modèle global (exprimé dans le repère global) et x_r la position courante du robot (*i.e.* l'odométrie si on n'a pas encore fait de recalage, sinon la dernière position recalée). Appelons δ la variable aléatoire caractérisant la distance entre deux sommets des modèles global et local :

$$\delta = \gamma'(x_r, x_p) - x_m \quad (\text{VII.1})$$

γ' désigne la composition d'une transformation (translation et rotation) avec une translation. Son expression et sa jacobienne $\mathbf{Y}' = (\mathbf{Y}'_1 \ \mathbf{Y}'_2)$ sont données en annexe B.1 dans le cas 3D qui est le nôtre.

Par propagation de l'incertitude, nous calculons alors les deux premiers moments de δ (la variable aléatoire x_p étant indépendante de x_r et x_m , mais en revanche pas x_r et x_m) :

$$\hat{\delta} = E[\delta] = \gamma'(\hat{x}_r, \hat{x}_p) - \hat{x}_m \quad (\text{VII.2})$$

$$\begin{aligned} \sigma_{\delta\delta} = E[\varepsilon_\delta \varepsilon_\delta^T] &= [\mathbf{Y}'_1 \ \mathbf{Y}'_2 \ -I_{3 \times 3}] \begin{bmatrix} \sigma_{rr} & 0 & \sigma_{rm} \\ 0 & \sigma_{pp} & 0 \\ \sigma_{mr} & 0 & \sigma_{mm} \end{bmatrix} \begin{bmatrix} \mathbf{Y}'_1{}^T \\ \mathbf{Y}'_2{}^T \\ -I_{3 \times 3} \end{bmatrix} = \\ &= \sigma_{rr} \sigma_{rr} \mathbf{Y}'_1{}^T + \mathbf{Y}'_2 \sigma_{pp} \mathbf{Y}'_2{}^T + \sigma_{mm} - \sigma_{mr} \mathbf{Y}'_1{}^T - \mathbf{Y}'_1 \sigma_{rm} \end{aligned} \quad (\text{VII.3})$$

	x_m^1	x_m^2	x_m^3	x_m^4	x_m^5
x_p^1	X				
x_p^2			X		
x_p^3					
x_p^4	X			X	

FIG. VII.2 - Exemple d'une table de correspondance. $(x_p^i)_{i=1,4}$ et $(x_m^j)_{j=1,5}$ représentent respectivement les sommets des objets du modèle local et ceux des objets du modèle global. L'objet numéro 3 du modèle local (i.e. issu de la perception) sera un nouvel objet. Les objets numéro 2 et 5 du modèle global ne sont pas reconnus. Quand aux autres objets, différentes hypothèses peuvent être formulées et il convient de choisir la "meilleure".

Toutefois, si l'objet du modèle n'est pas un amer, x_m n'est pas dans l'état X. Nous ne disposons pas dans ce cas de sa corrélation σ_{rm} avec le robot, celle-ci est alors négligée.

En supposant maintenant des distributions gaussiennes sur x_p , x_m et x_r , nous calculons la distance de Mahalanobis $Dm(\delta)$ de la variable aléatoire δ par :

$$Dm(\delta) = \hat{\delta} \sigma_{\hat{\delta}\hat{\delta}}^{-1} \hat{\delta}^T \quad (\text{VII.4})$$

Nous comparons cette distance à un seuil fixé par une loi de probabilité de χ^2 à trois degrés de liberté et concluons sur la possibilité de l'appariement des deux objets. Le résultat est indiqué dans la table de correspondance. Cette étape correspond à une approche exhaustive qui sera toutefois rapide vu le faible nombre d'objets segmentés dans chaque perception.

La figure VII.2 donne l'allure d'une table de correspondance. $(x_p^i)_{i=1,P}$ et $(x_m^j)_{j=1,M}$ désignent respectivement les sommets des objets extraits des modèles local et global (il y a dans ce cas quatre objets dans le modèle local et cinq dans le modèle global). Les "x" indiquent les appariements possibles. On peut alors définir une hypothèse d'appariement de la manière suivante : une hypothèse H_u est un ensemble d'appariements possibles donné par une bijection g entre $I_u \subset \{1, \dots, P\}$ et $J_u \subset \{1, \dots, M\}$:

$$H_u = \{(x_p^i, x_m^{g(i)}) \mid i \in I_u\}$$

A partir de la table de correspondance différentes hypothèses peuvent être construites (et un nombre d'autant plus important que la position du robot serait très imprécise ou l'environnement très encombré). De la table montrée figure VII.2 on retire par exemple l'hypothèse $\{(x_p^1, x_m^1), (x_p^2, x_m^3), (x_p^4, x_m^4)\}$ ou tout simplement l'hypothèse $\{(x_p^2, x_m^3)\}$. Le choix de la "meilleure" hypothèse fait l'objet de l'étape suivante.

Deuxième étape : recherche de la meilleure hypothèse d'appariements

Pour choisir une hypothèse, nous prenons en compte les contraintes entre les objets en choisissant parmi les hypothèses établies précédemment, celle qui donne le maximum d'appariements compatibles [Grimson 85]. Cela se formule comme suit :

Nous notons n_u le nombre d'appariements de H_u ($n_u = \text{Card}(H_u)$) et T_u la transformation minimisant la distance euclidienne entre les deux ensembles de sommets de H_u

[Haralick 89]. Nous nous limitons à une transformation plane pour T_u (*i.e.* calculée seulement sur les deux coordonnées x et y des sommets x_p^i et $x_m^{j(i)}$). Cela permet de traiter des hypothèses comprenant simplement deux couples d'amers (dans le cas d'une hypothèse composée d'un seul appariement, T_u est donnée à la transformation identité). Nous notons ε_u la qualité d'une hypothèse :

$$\varepsilon_u = \text{Max}_{i \in I_u} \| \hat{x}_m^{j(i)} - T_u(\hat{x}_p^i) \|$$

ε_u est l'erreur maximale entre les sommets après transformation. La figure VII.3 montre le principe du calcul de ε_u . L'erreur moyenne aurait pu être utilisée pour ε_u , mais celle-ci est moins sélective (par exemple dans le cas montré figure VII.3 où le nombre d'objets s'appariant correctement est important). Une hypothèse est alors dite compatible si ε_u est plus petit qu'une valeur fixée. Si C_H est l'ensemble des hypothèses compatibles, nous pouvons trouver n^* le nombre maximal d'appariements ;

$$n^* = \text{Max}_{H_u \in C_H} n_u$$

L'hypothèse sélectionnée est alors telle que :

1. Elle appartient à C_H .
2. Elle a n^* appariements.
3. Elle a l'erreur minimale ε^* :

$$\varepsilon^* = \text{Min}_{\substack{H_u \in C_H \\ n_u = n^*}} \varepsilon_u$$

Notons que cette étape *ne nécessite pas l'estimée de la position du robot* (puisque T_u est calculée directement à partir des appariements). Elle est réalisée par un algorithme récursif de recherche dans un arbre. Celui-ci est très rapide (de 50 à 100ms sur une station Sparc 10 et suivant le nombre d'objets étudiés), et ce pour les raisons suivantes :

- il y a peu d'objets à reconnaître entre le modèle et la perception ;
- à chaque étape, si H_u est la meilleure hypothèse trouvée, toutes les solutions qui donneraient un nombre d'appariements plus petit que n_u ne sont pas étudiées ;
- le recherche est faite en profondeur et les objets sont classés de telle sorte que les meilleures hypothèses sont trouvées en premier.

Cette étape correspond à une méthode arborescente, mais qui n'est réalisée que sur un sous-ensemble des objets du modèle local.

VII.2 Recalage du robot

Rappelons que le robot cherche tout d'abord à reconnaître les amers, *i.e.* des appariements entre les amers du modèle local avec ceux du modèle global (cf figure VII.1). Il



FIG. VII.3 – Principe d'appariement de structure d'objets. Les sommets des objets du modèle global sont représentés par des "x" et ceux des objets du modèle local par des "+". On étudie l'hypothèse qui associe les sommets les plus proches. **Figure (a)** : avant application de la transformation T_u . **Figure (b)** : après son application. On remarque que l'erreur maximale ε_{Max} est plus grande (donc plus sélective) que l'erreur moyenne ε_{Moy} . En effet, si l'on se réfère à ε_{Max} et à la valeur limite ε_{Lim} , cette hypothèse serait rejetée; en revanche pas celle associant les six sommets de gauche.

utilisera pour cela la procédure de reconnaissance présentée précédemment. Le recalage du robot sera effectué *uniquement* à partir de ces appariements d'amers. De surcroît, le robot peut reconnaître à nouveau d'autres amers depuis cette nouvelle position. Il réitérera alors cette opération de reconnaissance d'amers et de recalage (cf figure VII.1).

Une fois tous les couples d'amers reconnus, le robot cherchera les autres appariements, *i.e.* soit des appariements d'objets possédant un sommet, mais n'ayant pas été classés amers, soit des appariements d'amers avec de tels objets. Il utilisera pour cela la *même* procédure de reconnaissance. En revanche, les appariements trouvés ne serviront pas au recalage (les objets n'étant pas suffisamment sûr pour cela).

Pour chaque *couple d'amers appariés* le robot réestime donc sa position. Il applique pour cela le filtre de Kalman sur les sommets x_p et x_m respectivement des amers du modèle local et global. L'équation de mesure est la suivante :

$$\mathbf{y} = \gamma'(x_r, x_p) - x_m = 0 \quad (\text{VII.5})$$

Cette équation est non linéaire (γ' est non linéaire). Aussi il convient d'ordonner les couples d'amers par incertitude croissante (par exemple en utilisant le produit des valeurs propres de chaque σ_{pp} et σ_{mm}) pour exploiter en premier lieu les amers les plus précis.

Nous utilisons ici le principe du recalage-fusion qui permet de réestimer la position du robot indépendamment de celle des sommets des amers. Cette technique vise à minimiser l'effet de biais (notamment dû au glissement des roues) en évitant de le propager au reste de l'état.

Par ailleurs nous utilisons une version itérative du filtre de Kalman dans le but de réduire les effets de linéarisation (*i.e.* l'introduction d'un biais artificiel). Cela est nécessaire

dans la mesure où le robot évolue dans un environnement tridimensionnel impliquant l'utilisation de la composition de transformations 3D qui est fortement non linéaire (cf expression de γ' en annexe B.1). Les équations de ce type de filtre ont été présentées au chapitre précédent (section VI.2.1).

Nous présenterons à la fin de ce chapitre quelques résultats de recalage et dans le chapitre suivant une analyse plus complète de différentes expérimentations de modélisation incrémentales.

VII.3 Fusion et construction du modèle global de l'environnement

La fusion a trois fonctions :

- construire le modèle global des objets nouvellement perçus ;
- mettre à jour celui des objets qui ont été reconnus ;
- mettre à jour la carte stochastique.

Lorsqu'un objet est nouvellement perçu (ce qui est le cas de tous les objets à première perception), le robot construit pour cet objet une structure qui regroupe l'ensemble des informations extraites de ses différentes perceptions (ou plus exactement de ses différents modèles locaux). Cette structure constitue le modèle global d'un objet [Betgé-Brezetz 95] et nous donnons sa composition figure VII.4.

Lorsqu'un objet a déjà été perçu (et donc modélisé), cette structure est mise à jour à l'aide des informations issues du modèle local.

Nous allons décrire le modèle global d'un objet et comment le modèle local y est fusionné.

Modèle global d'un objet

- Le modèle global d'un objet comprend tout d'abord un modèle géométrique qui se décompose comme le modèle local en un modèle géométrique grossier (ellipsoïde) et éventuellement un modèle géométrique plus fin (superquadrique).

Si un objet est réobservé sa matrice d'inertie est mise à jour par la formule suivante :

$$M_{k+1} = \frac{N_k}{N_k + N_p} M_k + \frac{N_p}{N_k + N_p} M_p + \frac{N_k N_p}{(N_k + N_p)^2} [X_{G_{k+1}} - X_{G_p}, Y_{G_{k+1}} - Y_{G_p}, Z_{G_{k+1}} - Z_{G_p}]^T [X_{G_{k+1}} - X_{G_p}, Y_{G_{k+1}} - Y_{G_p}, Z_{G_{k+1}} - Z_{G_p}] \quad (\text{VII.6})$$

où M_{k+1} , M_k , M_p désignent respectivement la nouvelle matrice d'inertie, l'ancienne et celle de la perception (exprimée dans le repère global après recalage). $(X_{G_{k+1}}, Y_{G_{k+1}}, Z_{G_{k+1}})$ et $(X_{G_p}, Y_{G_p}, Z_{G_p})$ sont le nouveau centre d'inertie et celui de la perception, N_k le nombre de points perçus sur l'objet avant la perception, N_p celui de la perception et $N_{k+1} = N_k + N_p$. Les axes de l'ellipsoïde (*i.e.* son repère propre) sont calculés à partir des vecteurs propres de la matrice M_{k+1} . Nous exprimons ensuite l'ancien ellipsoïde du modèle global et celui du

Modèle global d'un objet	
Identificateur	
Amer (Oui / Non)	
Si amer : indice i dans l'état X : (\hat{X}_k, Λ_k)	
Nombre de perceptions	
Modèle géométrique	
- Matrice d'inertie, centre de gravité	
- Nombre de points perçus	
- Ellipsoïde	
- Superquadrique*	
- Repère et incertitude.	
	Eventuellement simplement l'origine (<i>i.e.</i> le sommet)
Modèle topologique	
- Objets voisins	
- Contact avec le sol	
- Trou	
Occultations	
- par d'autres objets	
- par le bord de l'image	
Directions de perception	
Liste des points 3D*	

FIG. VII.4 – *Modèle global d'un objet. Les parties optionnelles sont indiquées par une étoile.*

modèle local dans ce nouveau repère, leurs projections sur chacun de ces axes déterminent la taille du nouvel ellipsoïde.

Un modèle superquadrique peut éventuellement être calculé à partir des points 3D retenus sur l'objet. Les directions de perception (cf section V.1) sont mises à jour et donnent une estimation du pourcentage perçu autour de l'objet.

- Le modèle topologique du modèle global est plus simple que celui du modèle local. En effet, les relations topologiques que nous avons définies étaient dépendantes du point de vue et ne sont pas conservées dans le modèle global (on pourrait toutefois les mémoriser si certaines règles topologiques le nécessitent). En revanche nous retenons dans le modèle global d'un objet l'indicateur précisant que l'objet est un trou, ou qu'il n'est pas en contact avec le sol (*i.e.* qu'il n'a été perçu en contact avec le sol dans *aucune* de ses perceptions).

De la même manière un objet du modèle global est dit "occulté" par d'autres objets (resp. par le bord de l'image) s'il a toujours été perçu occulté par d'autres objets (resp. par le bord de l'image).

On définit aussi une relation de proximité entre deux objets. Celle-ci est vérifiée si la distance entre leur centre de gravité est inférieure à un seuil (défini à partir de la portée maximale du capteur).

• Enfin, concernant les amers, quatre situations différentes se présentent suivant que les objets des modèles global et local sont, ou non, des amers. Nous rappelons que si un objet du modèle global est un amer, alors son sommet figure dans la carte stochastique. Les différentes possibilités sont les suivantes :

- Les deux objets des modèles global et local sont des amers. Ils ont donc servi au recalage du robot (c'est le seul cas). L'objet du modèle global reste un amer et son sommet est réévalué en appliquant le filtre de Kalman sur la carte stochastique à l'aide l'équation VII.5. Le robot n'est bien évidemment plus recalé puisque l'on est à l'étape de fusion du procédé de recalage-fusion.
- L'objet du modèle global est un amer, mais pas celui du modèle local avec qui il a pourtant été identifié (ce dernier est par exemple occulté). La carte stochastique n'est alors pas réévaluée (pour ne pas être dégradée par une information peu sûre) et la valeur du sommet de l'objet du modèle global reste inchangée.
- L'objet du modèle global n'est pas un amer, mais celui du modèle local en est un. C'est par exemple le cas d'un objet qui jusque là avait été vu de trop loin et était donc trop imprécis. L'objet du modèle global devient un amer et a pour sommet celui du *modèle local* (qui est rajouté dans la carte stochastique). C'est en effet dans le sommet de l'objet issu de la perception que nous avons le plus de confiance.
- Aucun des deux objets des modèles global et local ne sont des amers. Les deux estimées du sommet pourraient être simplement fusionnées, mais cela risquerait de réduire artificiellement l'incertitude (par exemple dans le cas où les sommets seraient occultés). On préfère alors retenir l'estimée du sommet qui a la valeur plus précise.

Un objet qui n'est pas classé comme amer, à sa (ou ses) première perception, peut donc le devenir par la suite ; en revanche un objet qui a été classé amer le restera toujours (mais n'est pris en compte pour le recalage que dans la mesure où son modèle local est aussi classé amer). Toutefois, on pourrait imaginer à plus long terme une deuxième sélection parmi les amers. Prenons l'exemple d'une région explorée dans son ensemble et en possédant un grand nombre. On pourrait en sélectionner le sous-ensemble minimal (sur la base de critères à définir) qui permettrait de localiser le robot avec une précision suffisante en tout point de cette région. Il serait pour cela tout à fait judicieux de faire ressortir des configurations singulières de l'ensemble des amers (comme les alignements). Cela constitue une perspective intéressante, notamment pour minimiser l'information mémorisée.

Modèle global de l'environnement

Le modèle global de l'environnement est alors constitué de la liste des modèles globaux des objets et de la carte stochastique comme l'indique la figure VII.5. Remarquons que dans le cas où un objet est un amer, nous devons retenir dans son modèle quel est son indice i dans le vecteur d'état X_k .

De plus nous établissons un modèle grossier pour représenter le sol. Mais l'analyse des parties navigables est du ressort d'autres modèles présents à bord du robot (modèle bitmap, carte de régions), aussi nous sommes contents d'un modèle simple : un plan (mis à jour à l'aide d'un simple filtre de Kalman appliqué sur ses paramètres). Toutefois

Modèle global de l'environnement
Etat $X_k : (\bar{X}_k, \Lambda_k)$
Liste des modèles globaux des objets
Liste des modèles globaux des parties du sol

FIG. VII.5 – *Modèle global de l'environnement.*

l'analyse du sol (et de ses éventuelles différentes parties) serait un point intéressant à traiter et notamment par rapport aux objets de très grandes étendues que nous avons évoqués précédemment. Certains de ces objets, comme les collines vues de loin, pourraient éventuellement devenir des parties navigables lorsque le robot s'en rapprochera. D'autres objets, comme les amas de cailloux (cf figure IV.19, p. 49), seraient intéressants à modéliser plus finement (notamment par une surface B-spline). Ces zones sont donc à traiter en collaboration avec les autres modèles construits par le robot. Mais cet interfaçage devra être supervisé par un gestionnaire de modèle que nous avons déjà évoqué et qui permettra d'assurer la cohérence des différents modèles. La réalisation de ces interfaces et de ce superviseur est une des perspectives de recherche dans notre groupe.

VII.4 Une première illustration de la modélisation incrémentale

Nous illustrons le principe de la modélisation incrémentale basée sur des objets à travers le premier résultat suivant. Le robot Adam est placé sur un terrain composé d'un sol relativement régulier avec quelques rochers épars. Pour éviter de compliquer l'exemple par un nombre trop important d'objets, nous avons limité dans cette expérimentation la profondeur du champ de vue à 10m.

▷ La segmentation de la première perception est donnée dans le haut figure VII.7. Six objets ont été détectés et tous répondent aux critères d'un amer (l'incertitude de leur sommet est montrée dans le bas de la figure VII.7). Ils sont donc intégrés à la carte stochastique comme le montre la figure VII.8, l'incertitude sur la première position du robot étant nulle.

▷ Le robot se déplace (cf figure VII.9) et perçoit une nouvelle scène dont la segmentation est montrée figure VII.10. Quatre objets sont détectés mais seul trois sont identifiés comme amers, la colline n'ayant pas un sommet précis et étant de surcroît occultée. L'hypothèse associant les deux sommets de gauche aux deux amers numéros 1 et 4 de la carte stochastique est retenue et permet au robot de se recalculer. Le recalage a pour valeur 4 cm suivant x_g , 1 cm suivant y_g et .5 deg en cap (les corrections en altitude et attitude sont négligeables). Cette nouvelle position (plus précise) est montrée figure VII.11. Il n'y a pas d'autre amer à reconnaître, ni d'autre objet qui ne soit pas un amer mais qui posséderait un sommet (ce qui n'est pas le cas de la colline). La reconnaissance est donc achevée. Les modèles des deux amers reconnus sont alors fusionnés, l'incertitude de leur sommet est montrée figure VII.11 (notons la précision gagnée).

▷ Le robot se déplace à nouveau (cf figure VII.12), observe trois objets (cf figure VII.13)

Objet no 1
Amer : Oui
Indice dans l'état : 5
Nombre de perceptions : 3
Modèle géométrique
- Centre de gravité (11.41 m, 22.18 m, -0.14 m)
- Nombre de points perçus : 357
- Taille ellipsoïde dans son repère propre (29 cm, 35 cm, 69 cm)
- Position du sommet : (11.66 m, 22.5 m, 0.13 m)
- Incertitude (seulement la variance suivant chaque axe) : (11 cm, 7 cm, 2 cm)
Modèle topologique
- Objets proches : 2, 4, 5, 7, 8
- Contact avec le sol : oui
- Trou : non
Occultations
- par d'autres objets : non
- par le bord de l'image : non
Directions de perception (par rapport à x_g) : 50 deg, 60 deg et 110 deg

FIG. VII.6 – *Modèle global d'un objet de l'objet no 1 (cf figure VII.17 et VII.18).*

qu'il identifie tous comme amers. L'hypothèse appariant les deux plus proches aux deux amers 4 et 7 de la carte est retenue, cela permet de recalcr le robot. La correction de sa position est de 4 cm suivant x_g , 2 cm suivant y_g et 2. deg en cap (elle est négligeable pour les trois autres paramètres). La nouvelle position du robot (et incertitude) est montrée figure VII.14. Aucun objet non amer n'est détecté, le robot fusionne alors les deux amers qu'il a reconnu et la nouvelle carte stochastique est donnée figure VII.14 (on observe à nouveau le gain en précision des amers revus).

▷ Enfin, le robot arrive à sa dernière position après un grand déplacement (cf figure VII.15) d'où il observe deux amers (cf figure VII.16). Tous les deux ont déjà été vus et sont reconnus (ce sont les amers 1 et 2), le déplacement étant plus important l'erreur et donc le recalage le sont aussi (valeur du recalage : -32 cm suivant x_g , 15 cm suivant y_g et -5 deg en cap). Aucun autre objet ou amer n'est à reconnaître, le robot fusionne alors les modèles des amers 1 et 2 et la carte stochastique est donnée figure VII.17.

La figure VII.18 représente les ellipsoïdes calculés pour chacun des objets. Ils caractérisent sommairement leur dimension et leur orientation. On remarque la taille imposante de la colline. Par ailleurs, la figure VII.6 donne le modèle global de l'objet no 1 qui a été perçu trois fois.

Cet simple exemple illustre les mécanismes de base de la modélisation basée sur des objets, mais il convient d'analyser ses résultats en plus de détail. En effet, nous pouvons nous interroger sur la bonne correction des recalages effectués et sur la validité de la carte construite. Cela fait précisément l'objet du prochain chapitre qui regroupe différentes

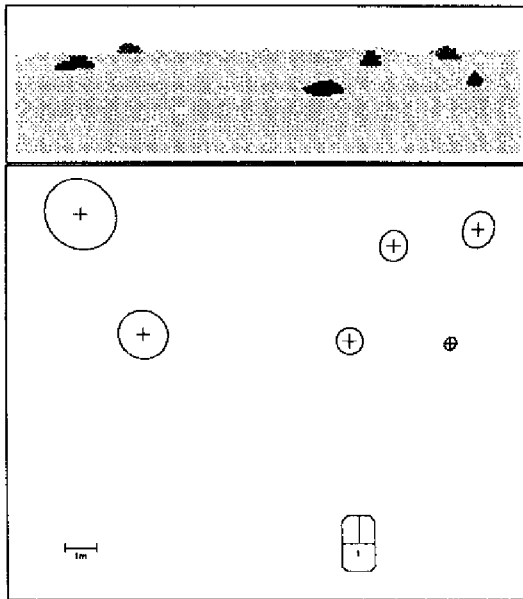


FIG. VII.7 - Première perception.
En haut : Image segmentée.
En bas : Les amers identifiés (dans ce cas tous les objets), leur sommet et leur incertitude.

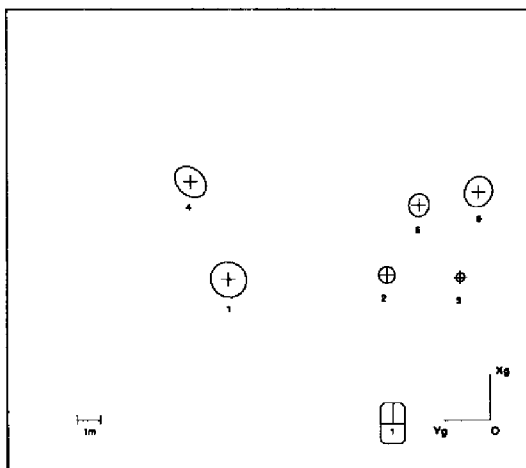


FIG. VII.8 - Carte stochastique, exprimée dans le repère global, à l'issue de la première perception. Le robot est à sa première position (1), son incertitude est nulle.

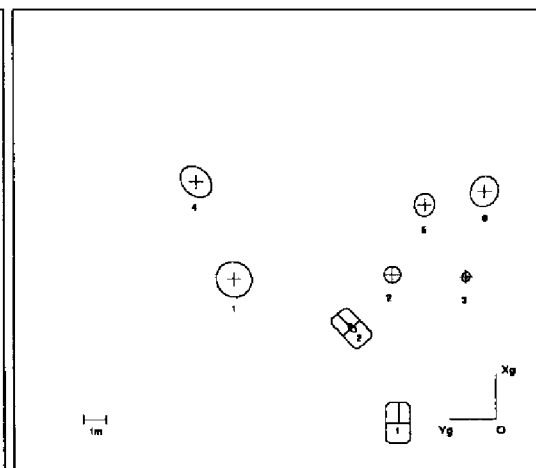


FIG. VII.9 - Le robot se déplace. L'estimée de sa nouvelle position 2 est donnée par l'odométrie et la centrale inertielle, son incertitude est représentée par une ellipse.

expérimentations de modélisation incrémentale.

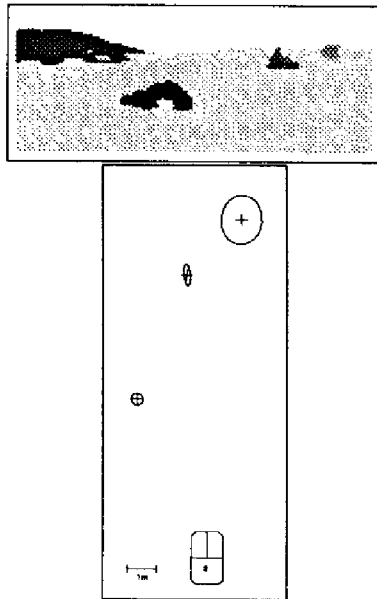


FIG. VII.10 – Deuxième perception.

En haut : Image segmentée.

En bas : Les amers identifiés (i.e. les trois rochers uniquement et non la colline), leur sommet et leur incertitude. Les deux amers de gauche sont reconnus dans le modèle (ce sont les amers no 1 et 4, cf figure VII.9), le troisième est un nouvel amer de la carte.

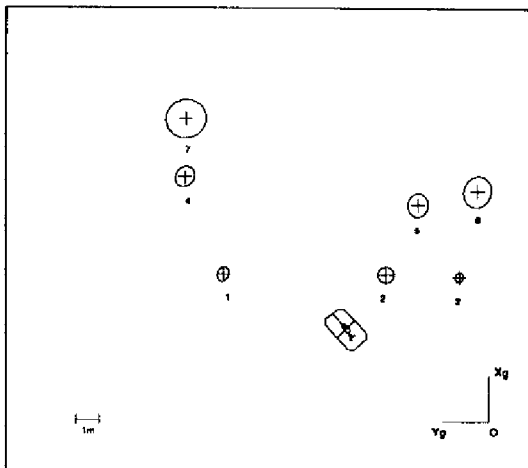


FIG. VII.11 – Carte stochastique à l'issue de la deuxième perception. Le robot s'est recalé (position 2') grâce aux deux amers revus et a réévalué la carte (réestimation du sommet des amers et ajout du troisième). L'incertitude des amers revus (et dans une moindre mesure celle du robot) ont diminué (cf figure VII.9).

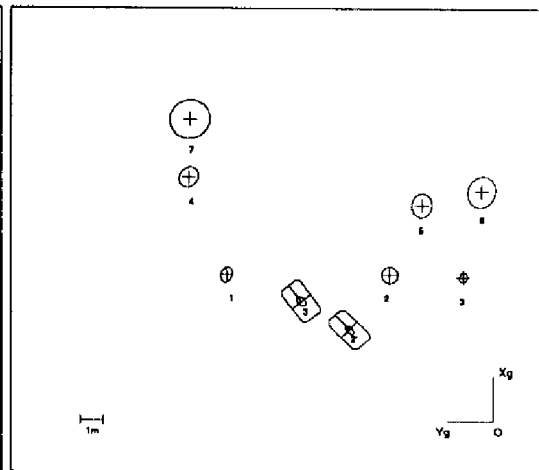


FIG. VII.12 – Le robot se déplace jusqu'à la position 3.

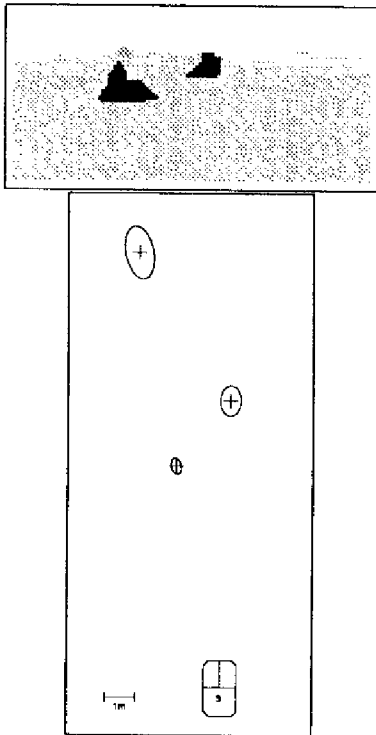


FIG. VII.13 - *Troisième perception.*
Trois objets (tous classés amers) sont détectés.
Les deux plus proches du robot sont reconnus (ce
sont les amers numéro 4 et 7).

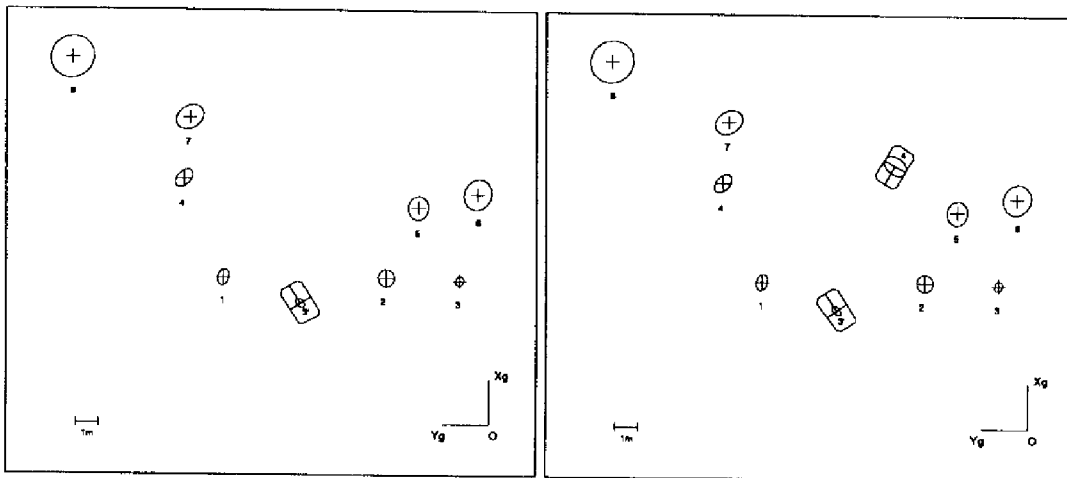


FIG. VII.14 - *Carte stochastique à l'issue de la troisième perception.* Le robot est recalé (position 3'), les deux sommets des amers revus sont fusionnés et le nouvel amer est ajouté à la carte. On observe à nouveau une diminution des incertitudes du robot et des amers reconnus (cf figure VII.12)

FIG. VII.15 - *Le robot effectue un grand déplacement jusqu'à la position 4, l'incertitude de l'estimée de sa position augmente de manière significative.*

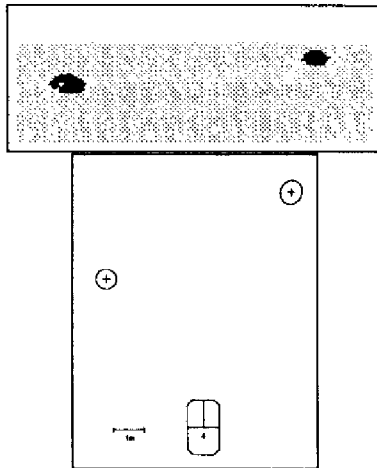


FIG. VII.16 - Quatrième (et dernière) perception. Deux amers sont perçus et reconnus (ce sont les numéros 1 et 2).

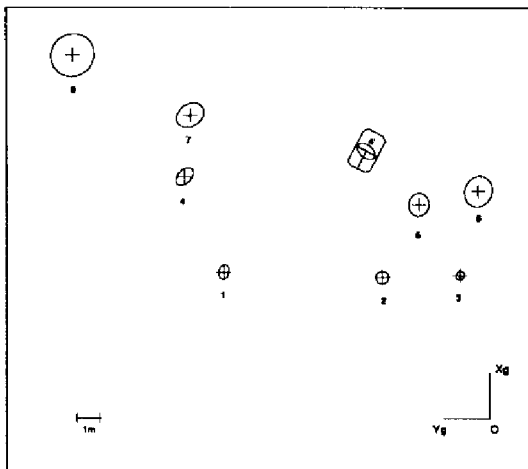


FIG. VII.17 - Carte stochastique l'issue de la dernière perception. L'incertitude du robot a réduit grâce à la précision dans le repère global des amers reconnus.

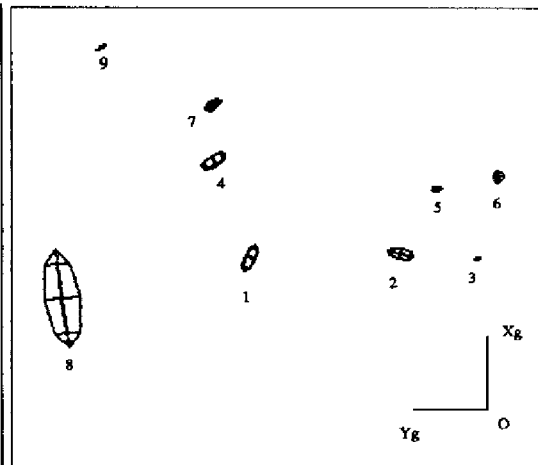


FIG. VII.18 - Modèle grossier (ellipsoïde). On observe la taille imposante de la colline (qui n'est pas un amer).



Chapitre VIII

Trois exemples de modélisation incrémentale

Nous avons décrit dans les chapitres précédents les principes et les algorithmes qui permettent au robot de construire incrémentalement un modèle à base d'objets. Nous avons commencé à apprécier l'intérêt de cette approche (traitement de l'incertitude, cohérence du modèle) à travers des simulations (cf section VI.4). Dans ce chapitre nous montrons les résultats qu'offre sa mise en œuvre complète dans des environnements réels. Nous évaluons l'approche en analysant en détail l'évolution de la position du robot et la carte finale construite. Nous examinons ensuite l'influence des incertitudes associées aux déplacements du robot et aux amers. Enfin, nous étudions la robustesse de l'approche aux erreurs de position non modélisées (notamment dues au glissement des roues).

VIII.1 Description des expérimentations

Nous présentons tout d'abord le cadre dans lequel les expérimentations ont été réalisées. Les principaux objectifs recherchés sont décrits ainsi que les moyens mis en œuvre. Enfin, dans un souci de rigueur, nous précisons la démarche expérimentale qui a été suivie.

VIII.1.1 Objectifs recherchés

Deux objectifs principaux sont visés :

▷ d'une part une certaine généralité des expériences et nous avons pour cela isoler deux scénarios différents :

- dans le premier, le robot traverse une région en avançant toujours dans la même direction. Il ne réobserve alors pas les amers qu'il laisse derrière lui ;
- dans le second, le robot réalise un trajet en boucle et revoit à la fin de sa trajectoire les premiers objets perçus.

Ces deux modes seront étudiés séparément à travers deux expérimentations différentes.

▷ Le second objectif de nos expérimentations sera de tester la **robustesse** de l'approche à une mauvaise estimation non modélisée de la position du robot (dont le glissement des roues en est une des raisons). Cela se traduit par un biais sur l'estimée de cette position.

VIII.1.2 Moyens nécessaires

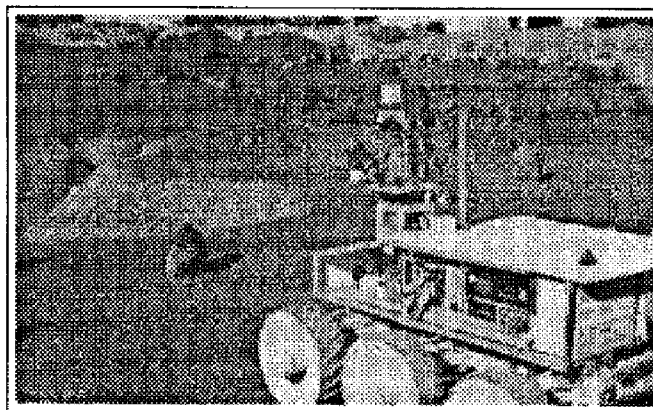


FIG. VIII.1 – Le robot mobile Adam dans le site d'essais de Geroms. La balise donnant une mesure externe de référence de sa position est la boîte blanche située au dessus de la plateforme multisensorielle.

Ces expérimentations ont été menées avec le robot mobile tout terrain Adam (cf figure VIII.1) déjà présenté au premier chapitre. Nous rappelons toutefois que ses principaux capteurs proprioceptifs et extéroceptifs sont les suivants : un télémètre laser, quatre caméras vidéo, un module odométrique et une centrale inertielle. A l'aide de ses capteurs, Adam va donc construire le modèle par objets de l'environnement comme nous l'avons décrit dans le mémoire.

D'autre part, nous avons pu bénéficier du site Geroms¹ du CNES qui est un terrain idéal pour des expérimentations de robotique mobile en extérieur. La partie que nous avons utilisée représente un environnement de type lunaire. Elle couvre une zone d'environ 100m×60m, principalement composée de crevasses, de collines et de régions plus ou moins accidentées avec des rochers épars. Deux photos de cette partie sont montrées en haut des figures VIII.5 et VIII.13 (pages 119 et 125).

1. Geroms (Groupement pour les Essais en RObotique Mobile Spatiale) est un groupement qui comprend le CNES, le CERT et le LAAS, dont l'objectif est de proposer une base d'expérimentation pour la robotique mobile en environnement naturel ou planétaire. Dans ce cadre, un site d'expérimentation de plus de 6000 m² a été construit au CNES.

Mais à partir de là, une question essentielle se pose pour toutes nos expérimentations :

Comment vérifier la validité du modèle construit ?

Deux éléments, liés l'un à l'autre, sont à considérer pour répondre à cette question : la **position du robot** et **celle des objets**.

L'estimation de la position du robot est centrale dans le processus de modélisation incrémentale. Cette position fait d'ailleurs partie intégrante de la carte stochastique. D'autre part, les objets étant perçus par rapport au robot, leur position y est très étroitement liée. Ainsi, l'exactitude de la **position du robot** apparaît comme une première mesure du bon fonctionnement de la modélisation incrémentale. Mais pour cela, il nous faudra connaître cette position avec une grande précision et le système qui fournira cette valeur devra être totalement extérieur au processus de modélisation incrémentale.

D'autre part, les **positions des objets** (et leur incertitude) pourront être comparées à une carte de référence (qui devra aussi être obtenue avec grande précision). Cette carte permettra de mesurer la validité du modèle construit.

Deux moyens s'avèrent ainsi nécessaires : un **appareil de mesure de la position du robot** et une **carte de référence** de l'environnement.

Le site d'essais Geroms dispose de tels moyens. Il est équipé d'un théodolite laser fournissant des mesures tridimensionnelles exprimées dans un repère de référence (appelé "repère Geroms"). Celui-ci est constitué d'un émetteur laser monté sur une plateforme orientable en site et en azimut. Si l'on place une balise dans l'environnement, le faisceau laser s'y réfléchit et délivre une information de distance. Les données 3D cartésiennes sont ensuite calculées à partir de cette information et des valeurs angulaires de l'orientation du faisceau. La précision atteinte sur les points 3D finaux est de l'ordre du centimètre. Ce système est également muni d'un logiciel de poursuite tel que le faisceau soit toujours maintenu sur la balise.

A l'aide de ce théodolite, nous avons mesuré la position des objets et celle du robot de la manière suivante.

▷ Avant toute expérimentation, nous construisons une **carte de référence** de l'environnement. La balise est montée sur une tige de longueur connue et munie d'un niveau à bulle permettant de garantir sa verticalité lors de la mesure. Nous avons ainsi estimé la position du sommet de chaque objet, pour constituer la carte de référence de l'environnement. Par ailleurs, nous avons relevé sur la surface du terrain un grand nombre de points pour visualiser sa forme. Ces points ont ensuite été transformés en carte d'élévation, à l'aide d'un algorithme de triangulation de Delaunay. L'algorithme de calcul de cette carte d'élévation fait aussi partie des moyens disponibles sur Geroms. Deux cartes d'élévation de référence qui correspondent aux photos sont montrées en bas des figures VIII.5 et VIII.13 (pages 119 et 125).

▷ D'autre part, nous avons équipé le robot Adam d'une balise pour mesurer sa position pendant l'expérimentation. Celle-ci est montrée figure VIII.1 et correspond à la boîte blanche située en haut de la plateforme multisensorielle. Le théodolite fournira alors sa position réelle à chacun des arrêts du robot. Nous l'appellerons **position de référence**

1. Construction de la carte de référence de la zone où va s'effectuer l'expérimentation (cette carte est exprimée dans le repère Geroms).
2. Mise en place de la balise sur le robot.
3. Mise du robot à sa première position et initialisation de son odométrie.
4. Tant que l'expérimentation n'est pas terminée, faire 5 et 6.
5. Exécution des trois opérations suivantes :
 - lecture de la valeur de l'odométrie;
 - lecture de la position de référence du robot (*i.e.* la position de la balise qui est exprimée dans le repère Geroms) ;
 - acquisition d'une image tridimensionnelle.
6. Déplacement du robot.

FIG. VIII.2 – Démarche suivie pour chaque expérimentation.

de la balise. Nous comparerons les positions de référence avec celles calculées à partir des positions du robot. Bien évidemment, il faudra pour cela avoir estimé la position de la balise dans le repère robot.

Mais cette balise ne fournit qu'un point dans l'espace et elle ne suffit pas à évaluer à la fois la position et l'orientation du robot (un nombre minimum de trois balises seraient nécessaires). De plus, elle n'est pas positionnée sur l'axe z_r du repère robot. Aussi l'erreur qu'elle permet d'estimer est une composition de l'erreur de position et d'orientation du robot (erreurs que nous ne saurions distinguer).

Enfin nous tenons à remercier P. Hébert pour l'aide qu'il nous a apportée dans la réalisation des expérimentations et notamment pour les longues heures que nous avons passées à mesurer le terrain au théodolite.

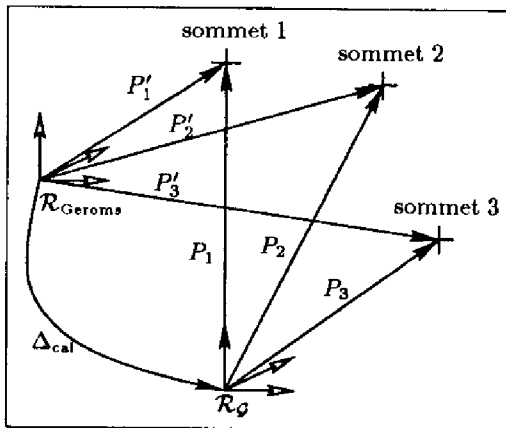
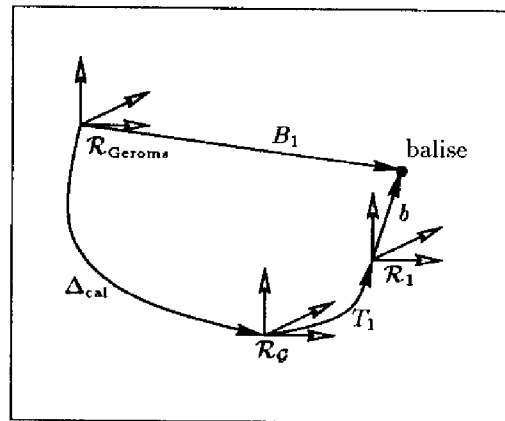
VIII.1.3 Démarche expérimentale

Pour chaque expérimentation, nous avons rigoureusement suivi la même démarche. Les différentes opérations à effectuer sont récapitulées figure VIII.2.

Pour pouvoir mener à bien ces expérimentations, il nous faut avant tout régler un problème essentiel : son **calibrage**. Nous précisons alors les différents repères et points de repères qui sont utilisés :

- le repère Geroms (noté $\mathcal{R}_{\text{Geroms}}$) dans lequel sont exprimées toutes les données issues du théodolite (*i.e.* la carte de référence des sommets des objets et les positions de référence de la balise).
- Le repère global (noté \mathcal{R}_G) dans lequel sont exprimées toutes les odométries.

- Les repères \mathcal{R}_i du robot correspondant à ses positions de perception et estimés par l'odométrie. Ils seront tous exprimés par rapport à \mathcal{R}_G (notons T_i leurs transformations correspondantes). Nous précisons que le repère robot de la première perception (i.e. \mathcal{R}_1) n'est pas forcément le repère global \mathcal{R}_G . Toutefois sa transformation T_1 étant fixée à l'initialisation du robot, elle est connue sans incertitude. En revanche, les autres transformations T_i des repères \mathcal{R}_i ($i \geq 2$) sont bien évidemment connues, mais de manière incertaine.
- Les repères \mathcal{R}'_i du robot correspondant aux positions de perception du robot réestimées par les recalages successifs. Ils sont aussi exprimés par rapport à \mathcal{R}_G et nous notons la transformation correspondante T'_i .
- La position de la balise dans le repère robot. C'est une translation que l'on note b et que nous devons estimer.
- La position de référence de la balise pour chaque position d'arrêt \mathcal{R}_i du robot. Ce sont des translations exprimées dans \mathcal{R}_{Geroms} et que nous noterons B_i .

FIG. VIII.3 - Calibrage de Δ_{cal} .FIG. VIII.4 - Calibrage de b .

Nous connaissons donc la transformation certaine exprimant \mathcal{R}_1 relativement à \mathcal{R}_G (i.e. T_1), mais en revanche pas celle de \mathcal{R}_G par rapport à \mathcal{R}_{Geroms} (et qui est propre à chaque expérimentation). Notons Δ_{cal} cette transformation.

Nous appelons alors "calibrage de l'expérimentation" le calcul de Δ_{cal} et b . Il est indispensable d'en avoir la meilleure estimée, car ils seront utilisés pour la validation des résultats. Nous procédons de la manière suivante.

- Pour calculer Δ_{cal} , nous utilisons le sommet des objets perçus lors de la première perception et nous les exprimons dans \mathcal{R}_G . Soient $(P_j)_{j=1,n}$ ces points dont nous connaissons la valeur $(P'_j)_{j=1,n}$ dans \mathcal{R}_{Geroms} à l'aide du théodolite (cf figure VIII.3). La transformation Δ_{cal} est alors simplement calculée comme étant celle qui minimise la somme des distances euclidiennes entre ces deux ensembles de points (cf [Haralick 89]). Un minimum de trois

points non coplanaires est nécessaire. Il est évident que plus le nombre de points est grand, meilleure sera l'estimation de Δ_{cal} .

En revanche, l'algorithme d'Haralick ne fournit que l'estimée de Δ_{cal} et pas son incertitude. Nous sommes ainsi confrontés à la difficulté d'estimer la précision de cette transformation.

• La figure VIII.4 donne ensuite le principe du calcul de b . Nous connaissons la position de référence B_1 de la balise lorsque le robot est à sa position initiale \mathcal{R}_1 . La position de la balise dans le repère robot se calcule alors simplement par la formule suivante :

$$b = (\Delta_{\text{cal}}.T_1)^{-1}.B_1 \quad (\text{VIII.1})$$

Les résultats du calibrage pourraient être améliorés si l'on disposait provisoirement des objets structurés dans l'environnement (par exemple des boîtes). On pourrait assurer leur horizontalité avec un niveau à bulle et les faces verticales permettraient de calculer un repère pour chaque boîte (cf section V.4). Le repère le plus précis serait utilisé pour obtenir une première estimée de Δ_{cal} (et de son incertitude). Cette valeur serait ensuite affinée par l'application d'un filtre de Kalman avec les autres repères. Cette méthode permettrait alors d'estimer Δ_{cal} (puis b) ainsi que leur incertitude.

Enfin, pour simplifier la lecture des sections suivantes, nous précisons ici la terminologie employée :

- positions réelles de la balise : positions de référence (B_i) ;
- positions odométriques de la balise : positions de la balise calculées à partir de l'odométrie ($\Delta_{\text{cal}}.T_i.b$) ;
- positions recalées de la balise : positions de la balise calculées à partir des positions du robot réestimées par les recalages ($\Delta_{\text{cal}}.T'_i.b$) ;
- erreurs de l'odométrie : distances euclidiennes entre les positions odométriques et réelles de la balise ($\| \Delta_{\text{cal}}.T_i.b - B_i \|$) ;
- erreurs de recalage : distances euclidiennes entre les positions recalées et réelles de la balise ($\| \Delta_{\text{cal}}.T'_i.b - B_i \|$).

VIII.2 Première expérimentation

Contexte de l'expérimentation

Nous avons voulu tester la modélisation incrémentale et le recalage lorsque le robot traverse une région en avançant toujours dans la même direction. La figure VIII.5 montre une partie de la région traversée (appelée "A") et sa carte d'élévation de référence. Nous remarquons que le terrain est dans ce cas incurvé, la trajectoire du robot suivant le fond de cette concavité. Elle est représentée approximativement par une courbe noire allant de D jusque A.

Le robot a parcouru une vingtaine de mètres en prenant sept perceptions. Figure VIII.6 nous donnons simplement les deux segmentations obtenues aux troisième et cinquième perceptions.

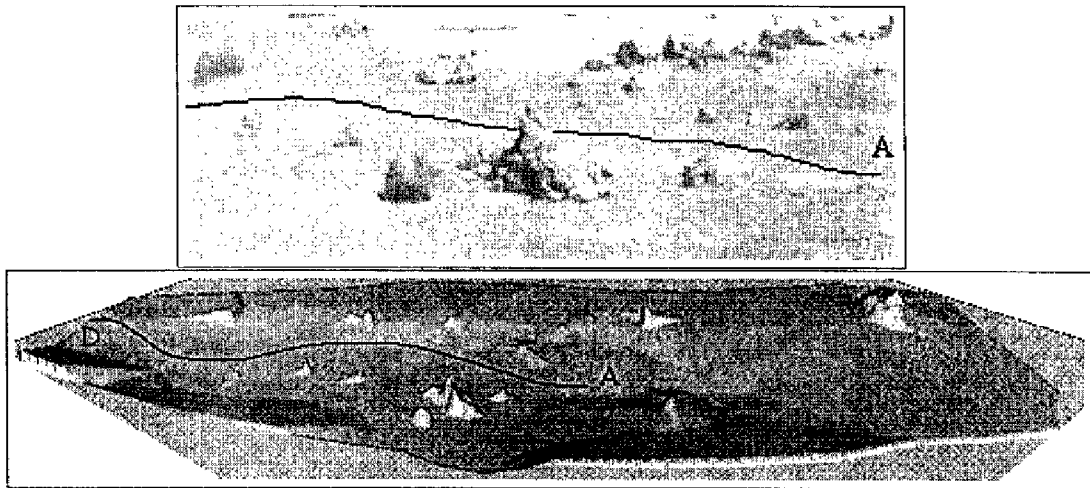


FIG. VIII.5 – Photo et carte d'élévation de référence de la partie "A" du site Geroms. Dimension de la région: 20m×35m. La photo correspond en partie à la carte d'élévation et son point de vue est proche de celui de la carte.



FIG. VIII.6 – Segmentations de la partie "A" de Geroms obtenues à la troisième (a) et à la cinquième (b) perception (cf figure VIII.10). On y remarque des rochers et des collines (ces dernières ne constituant bien évidemment pas des amers).

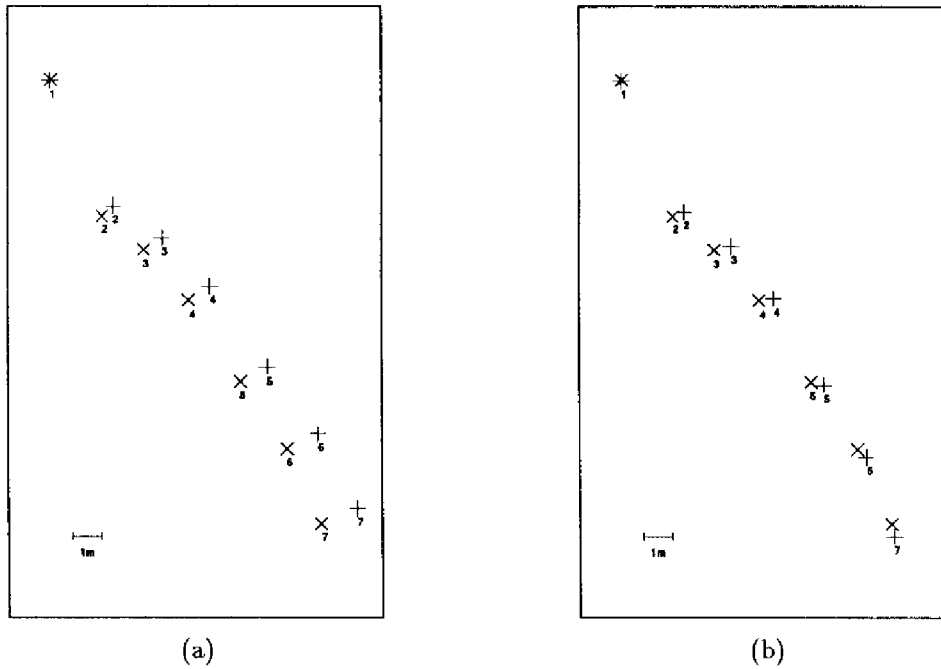


FIG. VIII.7 – Comparaison des positions de la balise. Les positions de référence de la balise sont représentées dans (a) et (b) par des "x", tandis que les "+" représentent les positions de la balise sans recalage dans (a) et avec recalage dans (b). Cette dernière montre la bonne correction du recalage.

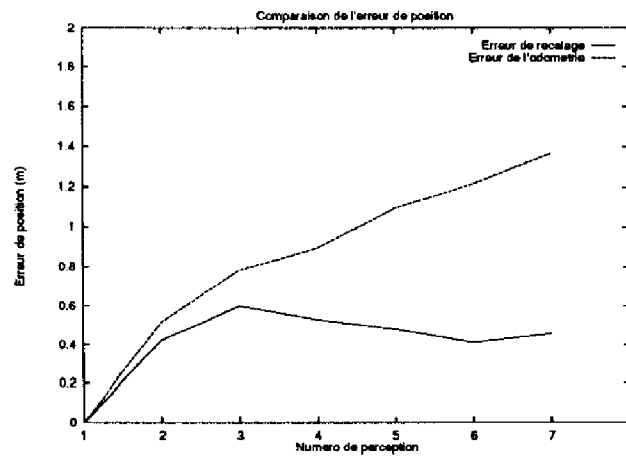


FIG. VIII.8 – Comparaison de l'erreur de position de la balise. L'erreur odométrique augmente tandis que l'erreur de recalage reste bornée : la position du robot est bien corrigée.

Analyse de l'évolution de la position du robot

Regardons maintenant figure VIII.7 les différentes positions réelles, odométriques et recalées du robot (ou plus exactement de la balise). On y observe une dérive progressive de l'odométrie vers la gauche² (cf figure VIII.7 (a)) que les recalages permettent de bien corriger (cf figure VIII.7 (b)). Ce résultat s'apprécie davantage figure VIII.8 qui donne les courbes des erreurs de l'odométrie et de recalage pour les différentes positions de perception. Nous constatons que l'erreur de l'odométrie augmente progressivement (jusqu'à 1.4 m), tandis que l'erreur de recalage est bien entendu *inférieure* et reste même bornée (à environ 50 cm). Cela montre clairement comment le robot corrige bien sa position en utilisant les amers qu'il a détectés.

Si nous étudions maintenant ces deux courbes de plus près, nous pouvons les diviser en deux parties :

- de la perception 1 à 3 : l'erreur de recalage "suit" l'erreur de l'odométrie. La précision de l'odométrie est importante, aussi son influence dans le filtre de Kalman est grande par rapport à celle des amers.
- De la perception 4 à 7 : l'erreur de recalage a tendance à diminuer alors que celle de l'odométrie continue de croître. La précision de l'odométrie devenant beaucoup plus faible, le poids des amers est alors plus important.

Dans les figures VIII.9 et VIII.10 nous pouvons observer respectivement l'évolution de l'incertitude de l'odométrie et celle des positions recalées du robot. La première augmente rapidement, tandis la seconde croît aussi, mais plus lentement. Le robot avançant droit devant lui, il ne reperçoit pas les premiers amers modélisés (et précis) et ne réduit alors que partiellement l'incertitude accumulée au cours du déplacement.

Les valeurs du recalage et le nombre d'amers reconnus à chaque perception sont donnés dans le tableau de la figure VIII.11. On y observe que la dérive de l'odométrie vers la gauche est principalement corrigée par un recalage de l'orientation θ du robot dans le sens horaire.

Analyse de la carte construite

La carte de référence de la zone est montrée figure VIII.10 où les sommets des rochers mesurés au théodolite sont représentés par le symbole "x". La position estimée par le robot de ces sommets est représentée par un symbole "+" et leur incertitude par une ellipse dessinée à 99%. D'autres amers ont été détectés, mais seul sont représentés ceux pour lesquels nous disposons d'une position de référence. Ces sommets, qui constituent la carte stochastique, sont tous exprimés dans le même repère global \mathcal{R}_G , leur incertitude (i.e. σ_{ii}) intègre donc celle du robot. Cela explique la plus grande imprécision (dans \mathcal{R}_G) des derniers amers perçus. Nous rappelons toutefois que chacun de ces sommets est lié aux autres (et au robot) par des liens de corrélation (qui ne sont pas représentés dans la figure).

2. relativement au robot

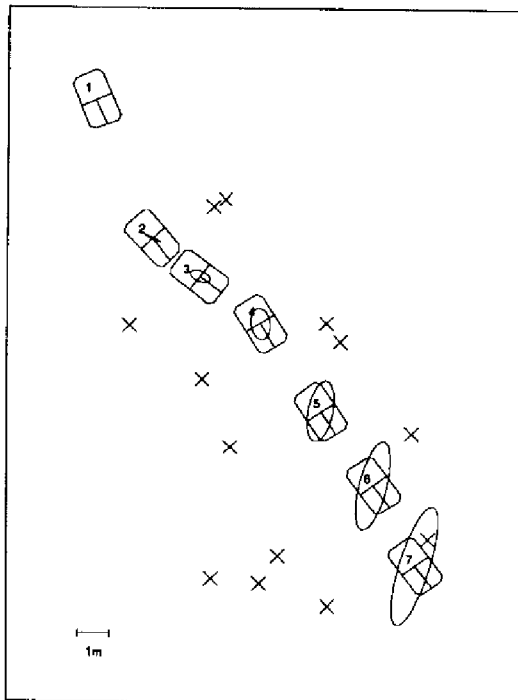


FIG. VIII.9 - Positions et incertitudes odométriques du robot.

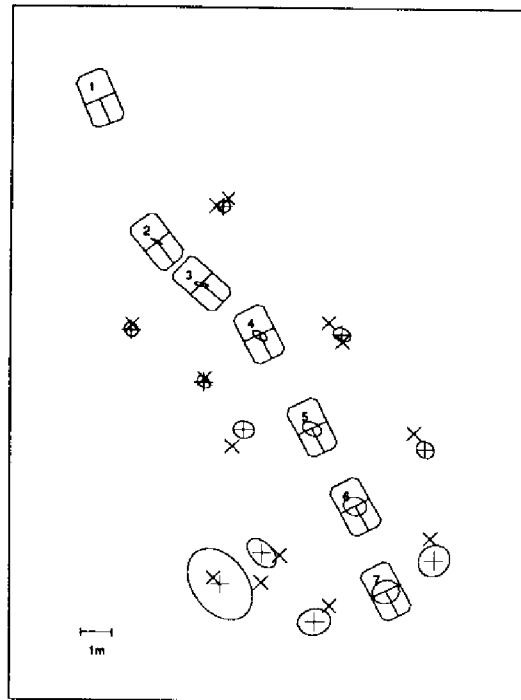


FIG. VIII.10 - Positions et incertitudes recalées du robot. Amers de référence ("x") et leurs correspondants ("+") dans la carte construite (avec leur incertitude).

L'erreur moyenne sur l'ensemble des sommets de la carte est de 57 cm. Toutefois, l'erreur des premiers sommets perçus (et précis dans \mathcal{R}_g) est de l'ordre de 15 cm, celle des derniers objets (imprécis) est d'environ 70 cm. La figure VIII.10 nous permet ainsi de comparer les sommets identifiés par le robot (et leur incertitude) avec ceux de la carte de référence. Nous apprécions, pour la plupart des sommets, la validité de la carte construite. Toutefois quelques sommets de la carte de référence sont situés à l'extérieur des ellipses d'incertitude des sommets de la carte stochastique. Cela pourrait être imputable à une sous estimation de l'erreur de l'odométrie ou à ses biais, mais aussi à l'imprécision du calibrage de l'expérimentation.

Robustesse à une estimation biaisée de la position

Enfin nous avons testé la robustesse de l'approche à une mauvaise estimation de la position. Pour cela nous avons ajouté artificiellement un biais sur l'odométrie de *chaque* déplacement (simulant un glissement ou une dérive constante). Nous nous sommes placés dans la pire situation en dirigeant ce biais vers l'avant et la gauche du robot. La valeur du biais est de 25cm dans les deux directions. Les résultats des recalages sont alors montrés à la figure VIII.12. Comme précédemment, on observe l'influence de la précision de l'odométrie

Perception	Dép. estimés		Recalages		Nombre d'amers reconnus
	(x, y, z) m.	(θ, φ, ψ) deg.	(x, y, z) m.	(θ, φ, ψ) deg.	
2	5.02	19.9	.15	-3.5	4
	-0.04	0.6	0.00	0.	
	-0.35	3.0	0.00	-0.1	
3	1.98	10.3	0.03	-0.2	3
	0.33	-2.1	0.00	0.0	
	0.07	0.1	0.01	0.0	
4	2.50	-19.9	0.00	-1.0	3
	0.00	-1.4	0.00	0.1	
	0.00	-0.1	0.00	0.0	
5	3.50	1.8	0.02	-2.0	3
	0.12	1.7	0.00	0.1	
	-0.05	-0.2	0.00	0.0	
6	3.01	1.8	0.00	-0.6	1
	0.03	-0.6	0.00	0.0	
	0.02	-2.1	0.00	0.0	
7	2.97	1.2	-0.02	-2.2	2
	-0.47	0.5	0.00	-0.1	
	0.02	-0.5	0.00	0.0	

FIG. VIII.11 – Déplacements relatifs estimés, recalages et nombre d'amers reconnus pour chaque perception.

sur les premières positions. Ensuite, grâce aux amers que le robot a réussi à reconnaître, ce dernier arrive à corriger une partie de l'erreur de l'odométrie. L'erreur de recalage reste alors inférieure à 1 m tandis que celle de l'odométrie croît jusqu'à 2 m.

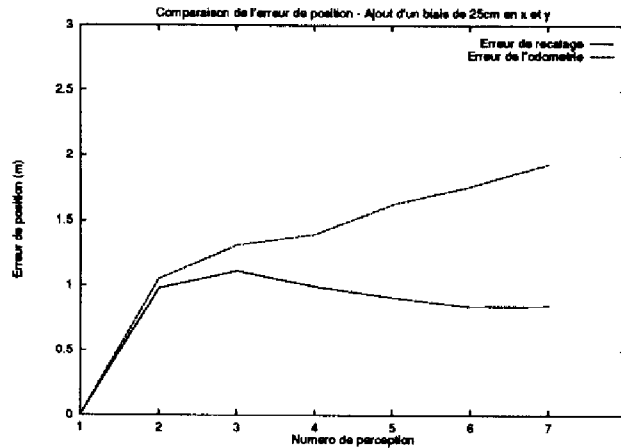


FIG. VIII.12 – Comparaison de l'erreur de position de la balise pour l'ajout d'un biais constant de 25cm en x et y .

VIII.3 Deuxième expérimentation

Contexte de l'expérimentation

Le robot effectue dans ce cas une trajectoire en boucle d'une quarantaine de mètres. On peut voir figure VIII.13 une photo et la carte d'élévation de référence de la partie appelée "B" du terrain Geroms parcourue par le robot (le trait noir indique grossièrement la trajectoire suivie). A la fin de cette trajectoire, le robot réobserve les quatre rochers situés près du point A. On donne d'ailleurs figure VIII.14 deux segmentations de cette zone obtenues pour deux points de vue à peu près opposés.

Analyse de l'évolution de la position du robot

Nous montrons figure VIII.15 la comparaison entre les positions de référence de la balise et ses positions odométriques (cf figure VIII.15 (a)) et recalées (cf figure VIII.15 (b)). La figure VIII.15 (b) montre que les recalages ont bien rectifié la position du robot. La figure VIII.16 donne l'évolution des courbes d'erreur de l'odométrie et du recalage. Intéressons-nous au cas où les corrélations sont prises en compte et faisons quelques remarques :

- à la position 7, le robot a corrigé une grande part de son erreur. Il a bénéficié d'une part des amers (ceux notamment vus depuis la position 5), mais certainement aussi, d'autre part, d'un déplacement qui, par chance, a comblé une partie cette erreur (cf la décroissance de l'erreur de l'odométrie pour cette position).

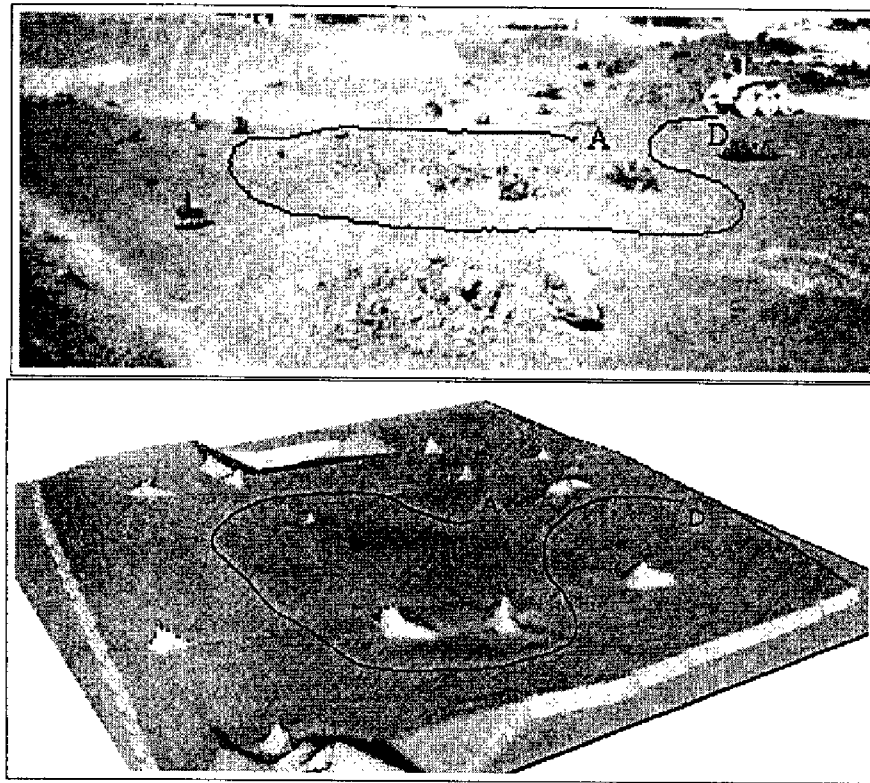


FIG. VIII.13 – Photo et carte d'élévation de référence de la partie "B" du site Geroms. Dimension de la région: $25m \times 25m$. La photo correspond en partie à la carte d'élévation et son point de vue est proche de celui de la carte.

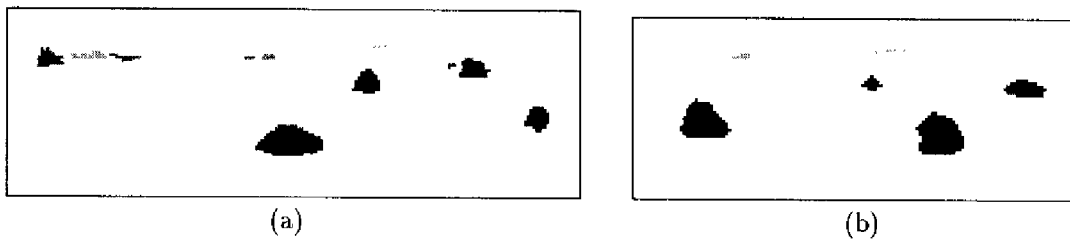


FIG. VIII.14 – Segmentations de la partie "B" de Geroms obtenues à la deuxième (a) et à la onzième (b) perception (i.e. respectivement aux environs des points D et A de la figure VIII.13). Les quatre rochers de droite dans l'image (a) correspondent donc aux quatre rochers de l'image (b), mais sous un point de vue à peu près opposé.

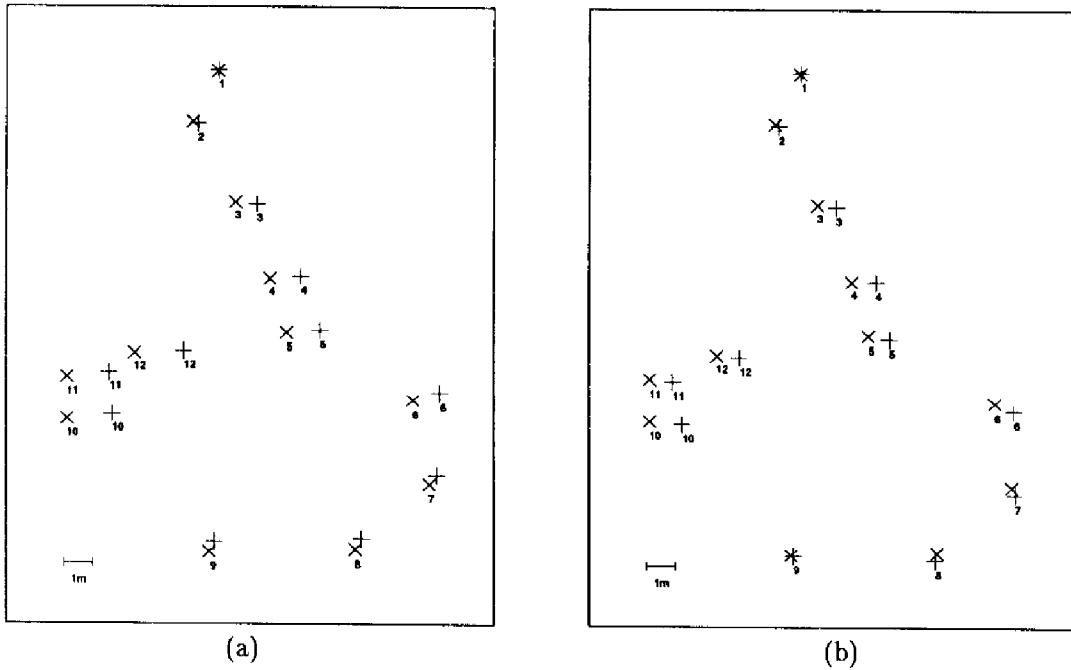


FIG. VIII.15 – Comparaison des positions de la balise. Les positions de référence de la balise sont représentées dans (a) et (b) par des "x", tandis que les "+" représentent les positions de la balise sans recalage dans (a) et avec recalage dans (b).

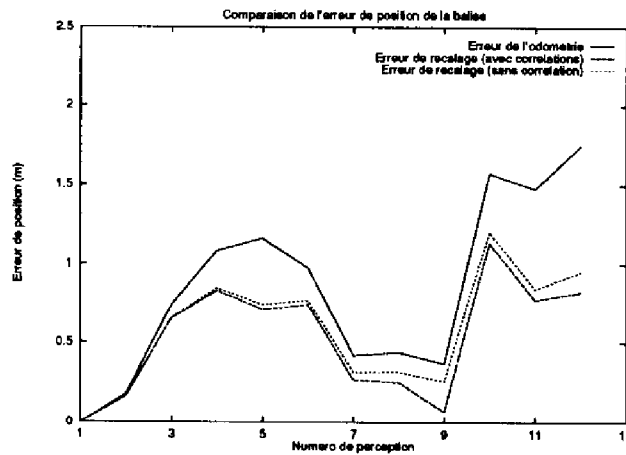


FIG. VIII.16 – Comparaison de l'erreur de position de la balise.

- Après la position 9, le robot a pratiquement corrigé toute son erreur, mais ensuite nous lui avons fait effectuer une grande trajectoire de telle sorte qu'il arrive à la position 10 avec beaucoup d'erreur.
- Les positions 10, 11 et 12 permettent de bien illustrer l'effet du recalage. Le robot réobserve depuis ces positions les quatre premiers amers perçus. Il reconnaît ce groupement, et ce, malgré son importante erreur de position. Cela lui permet de corriger progressivement cette erreur, bénéficiant de la précision de ces amers dans le repère global (son erreur décroît à .8 m tandis que celle de l'odométrie est de 1.7 m).

Par ailleurs, la courbe d'erreur de recalage sans prise en compte des corrélations montre que ces dernières semblent bien donner des résultats sensiblement meilleurs.

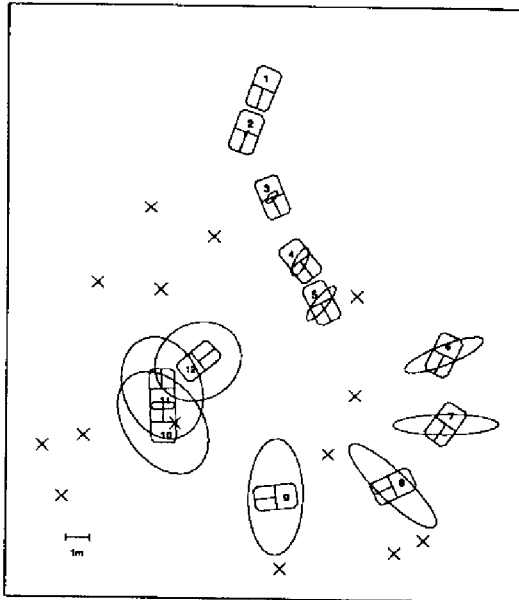


FIG. VIII.17 - Positions et incertitudes odométriques du robot.

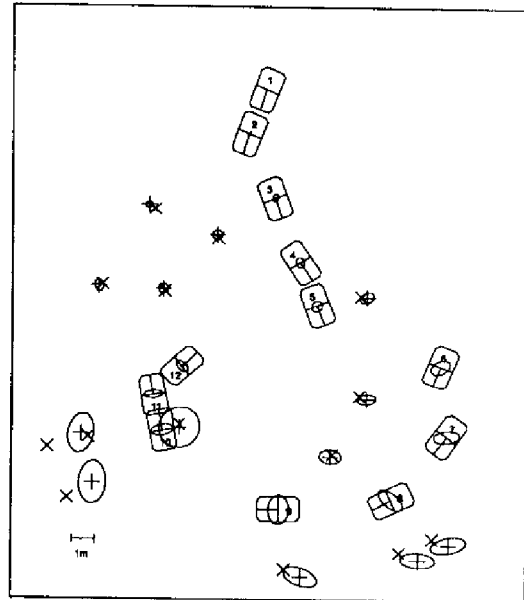


FIG. VIII.18 - Positions et incertitudes recalées du robot. Amers de référence ("x") et leurs correspondants ("+") dans la carte construite (avec leur incertitude).

Les figures VIII.17 et VIII.18 montrent ensuite l'évolution respective de l'incertitude de la position odométrique et de la position recalée du robot. La première augmente bien évidemment tout au long de son déplacement du robot. En revanche la seconde évolue de la manière suivante :

- de la position 1 à 9, l'incertitude augmente (tout en restant inférieure à celle de l'odométrie grâce aux recalages) ;

- de la position 10 à 12, l'incertitude diminue. Cela provient de la réobservation des quatre premiers amers dont la précision est importante dans le repère global.

Enfin nous donnons dans le tableau de la figure VIII.19 les différentes valeurs de recalage et le nombre d'amers reconnus pour chaque perception. Deux recalages sont plus importants :

- celui de la position 6. En effet le déplacement du robot de la position 5 à 6 a demandé plusieurs manœuvres pour pouvoir passer entre les deux rochers (ce qui a faussé l'odométrie) ;
- celui de la position 10. La trajectoire suivie pour aller en 10 a été volontairement compliquée ce qui a aussi entraîné une erreur importante de l'odométrie (erreur en partie bien corrigée par la réobservation des premiers amers).

Analyse de la carte construite

La figure VIII.18 représente par des symboles "x" les positions des sommets des rochers issues de la carte de référence. Leurs positions estimées par le robot sont indiquées par des symboles "+" avec leurs ellipses d'incertitude tracées à 99% (les amers pour lesquels nous ne disposons pas de position de référence ne sont pas représentés). L'erreur moyenne de la carte construite est dans ce cas 43 cm (et 46 cm si nous ne prenons pas en compte les corrélations). L'erreur des sommets des premiers amers perçus est de l'ordre d'une dizaine de centimètres.

Nous observons donc que la carte construite dans cette expérimentation est plus précise que celle de l'expérimentation précédente. Ceci s'explique par le fait que le robot a réobservé plusieurs fois les amers et notamment les premiers à la fin de sa trajectoire.

Robustesse à une estimation biaisée de la position

Dans l'expérimentation précédente nous avons montré le comportement de l'approche dans le cas d'une dérive constante de l'odométrie. Nous désirons maintenant tester cette approche à une erreur très importante dans l'estimation d'une position. Nous avons alors augmenté artificiellement l'erreur sur l'estimée de la position 10 (en ajoutant un biais d'une quarantaine de centimètres). La figure VIII.20 donne les courbes d'erreur de l'odométrie et du recalage. Le robot ayant à nouveau bien reconnu les quatre amers, nous observons encore les bonnes corrections du recalage qui réduit son erreur (la décroissance de l'erreur se faisant bien évidemment "plus lentement" dans ce cas).

VIII.4 Troisième expérimentation : cas d'un environnement structuré

Contexte de l'expérimentation

Dans les expérimentations précédentes, nous avons évalué l'efficacité des recalages à l'aide d'une balise montée sur le robot. Mais celle-ci ne délivre qu'une information incomplète sur sa position et son orientation. D'autre part, le processus de localisation, tel que

Perception	Dép. estimés		Recalages		Nombre d'amers reconnus
	(x, y, z) m.	(θ, φ, ψ) deg.	(x, y, z) m.	(θ, φ, ψ) deg.	
2	2.00	0.3	0.01	0.6	6
	-0.1	0.4	0.00	0.0	
	-0.01	0.1	0.00	0.0	
3	2.24	42.6	0.00	-1.1	3
	2.01	1.6	-0.02	0.0	
	-0.01	1.1	0.00	0.1	
4	3.00	17.0	-0.01	-3.3	2
	0.06	0.6	-0.03	0.1	
	0.01	-1.1	-0.02	0.0	
5	1.96	-12.0	0.01	-1.2	4
	-0.44	-1.0	0.00	0.0	
	0.04	-0.7	-0.08	0.0	
6	4.37	-52.6	-0.22	9.8	4
	3.63	-0.8	0.19	-0.1	
	0.02	-0.6	-0.04	0.0	
7	2.74	-10.5	0.04	-3.2	1
	1.42	0.4	-0.01	0.0	
	-0.01	-0.1	0.00	0.0	
8	3.50	-28.1	0.09	-0.6	2
	-0.21	0.3	-0.08	0.0	
	0.03	-0.1	0.01	0.0	
9	4.71	-19.6	-0.18	-6.3	1
	-1.70	-0.4	0.14	0.0	
	0.02	0.4	-0.01	0.0	
10	4.44	-94.6	.50	14.7	6
	-3.70	-1.1	-.42	-0.1	
	0.08	1.2	0.07	0.2	
11	1.50	1.0	-0.04	-1.8	4
	0.02	0.4	-0.12	0.0	
	0.00	0.3	0.00	0.0	
12	1.18	-53.4	0.16	-4.3	4
	-1.63	-1.7	-0.14	0.0	
	0.02	-0.9	0.02	-0.1	

FIG. VIII.19 – Déplacements relatifs estimés, recalages et nombre d'amers reconnus pour chaque perception.

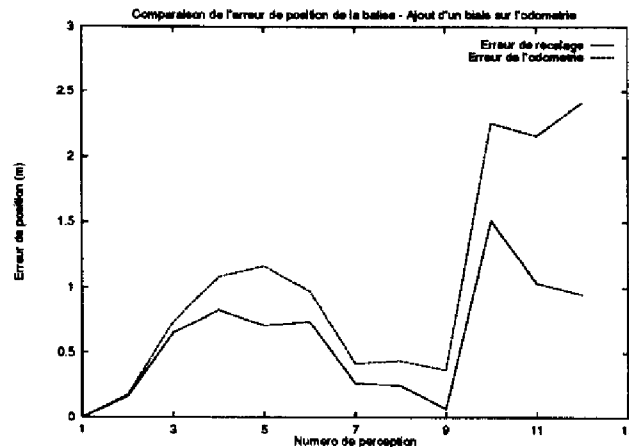


FIG. VIII.20 – Comparaison de l'erreur de position de la balise pour l'ajout d'un biais de 50 cm sur l'odométrie de la position 10.

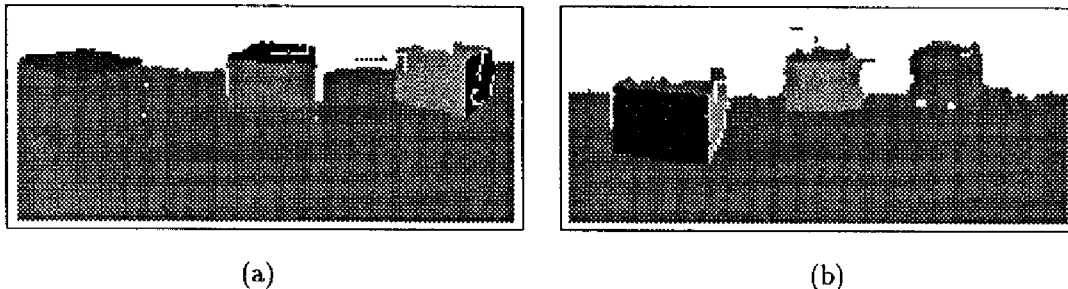


FIG. VIII.21 – Trois boîtes percées de deux points de vues différents à la quatrième (a) et à la onzième (b) perception (cf figure VIII.24). Les faces de chaque boîtes ont des niveaux de gris différents.

nous l'avons utilisé pour les deux expériences précédentes, ne prend en compte que l'origine des repères (en raison de la difficulté de fixer des axes sur des objets de forme quelconque). Aussi, et dans le but d'étudier de manière plus complète le comportement de l'approche, nous présentons quelques résultats de modélisation par objets en environnement intérieur.

Dans ce type d'environnement (cf figure V.19 p. 70), les objets sont structurés et l'on peut alors fixer une orientation pour chaque repère (cf section V.4). De plus, l'environnement où nos expérimentations se sont déroulées comporte un marquage régulier sur le sol qui permet de localiser un point avec une précision centimétrique. Le robot Hilare II étant équipé de 3 LED émettant un faisceau lumineux sur le sol, l'utilisation de ce marquage nous permet d'estimer à la fois sa position et son orientation.

Dans cette expérimentation nous avons uniquement voulu étudier la construction de la carte et la localisation du robot. Aussi avons nous découplé les problèmes de reconnaissance en effectuant celle-ci sans erreur de manière interactive. Par ailleurs, le robot pouvant percevoir un objet sous différents points de vue, il n'attachera pas un repère toujours sur

le même coin (il y a quatre possibilités). Un modèle de la boîte (taille de ses faces) s'avère donc nécessaire. Pour éviter d'introduire des problèmes liés à l'estimation de ce modèle, nous donnons au robot la connaissance *a priori* de la taille exacte de chacune des boîtes. Tout se passe alors comme si le robot "posait" à chaque fois le repère d'une boîte sur le même coin.

Le robot effectue ainsi une trajectoire en boucle dans l'environnement. Les segmentations de la figure VIII.21 donnent à titre d'exemple les trois mêmes boîtes segmentées à la quatrième et à la onzième perception. Nous avons suivi la procédure décrite à la figure VIII.2 et avons testé l'utilisation ou non des corrélations. Nous analysons ensuite la carte obtenue et les positions du robot.

Analyse de l'évolution de la position et de l'orientation du robot

Les courbes de la figure VIII.22 expriment les erreurs de position et d'orientation de l'origine du repère robot. Notons que la décroissance de l'erreur odométrique pour les positions 6, 9 et 11 est un simple hasard. Les différentes positions du robot sont représentées figure VIII.24 et numérotées de 1 à 13.

Nous pouvons distinguer deux étapes :

- De la position 1 à 10. Le robot découvre son environnement. Il réobserve donc simplement quelques boîtes qu'il avait perçues précédemment. Le recalage ne lui permet de corriger qu'une part de son erreur. A ce stade l'utilisation ou non des corrélations donnent des résultats équivalents.
- De la position 11 à 13. Le robot revient "sur ses pas" et observe les premières boîtes perçues. En raison de leur précision, il corrige fortement sa position et son orientation. La méthode utilisant les corrélations semblent donner de meilleurs résultats.

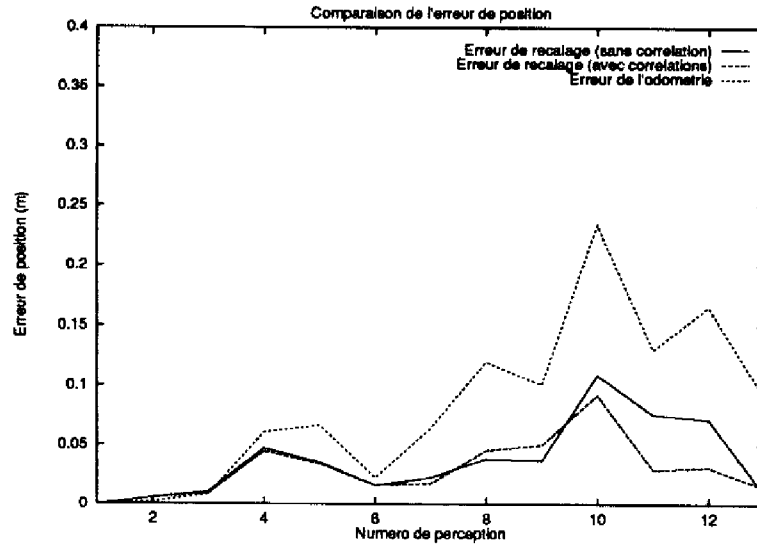
Analyse de la carte construite

Le tableau de la figure VIII.23 permet de comparer la carte obtenue avec chaque méthode. Il donne les erreurs moyennes et maximales des origines des repères objets et de leurs orientations. La figure VIII.24 montre la carte construite dans les deux cas. Nous observons bien évidemment des résultats similaires, à savoir que l'utilisation des corrélations améliore la carte de l'environnement.

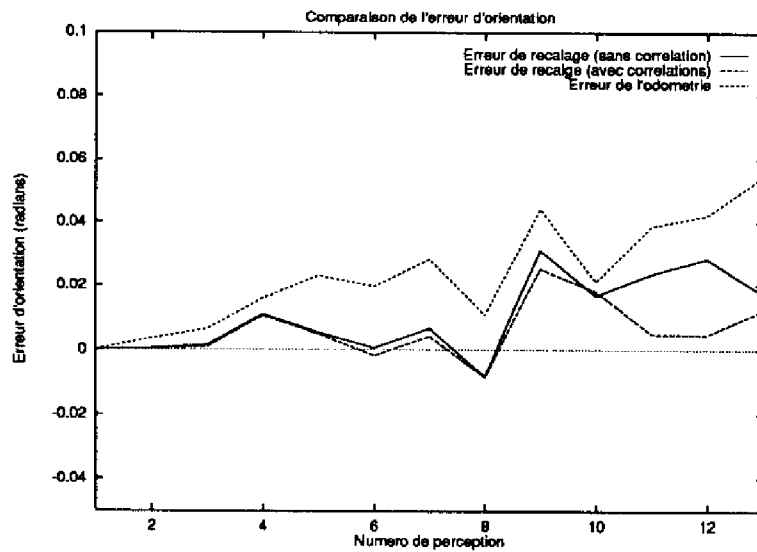
En conclusion, vu les meilleures incertitudes de perception et de l'odométrie, mais aussi grâce à l'utilisation de l'orientation des repères, nous notons ainsi une bien meilleure qualité de la localisation (quelques centimètres) et de la carte construite en milieux structurés.

VIII.5 Conclusion

Ces expérimentations permettent d'apprécier le comportement de la modélisation par objets ainsi que sa robustesse. Elles ont montré comment cette approche permettait de corriger efficacement une partie de l'erreur de position du robot et de construire une carte des amers de l'environnement. Nous avons vu qu'un avantage important est la possibilité



(a)



(b)

FIG. VIII.22 – Erreur de translation (a) et de rotation (b) sans recalage et avec recalage (en utilisant ou non les corrélations).

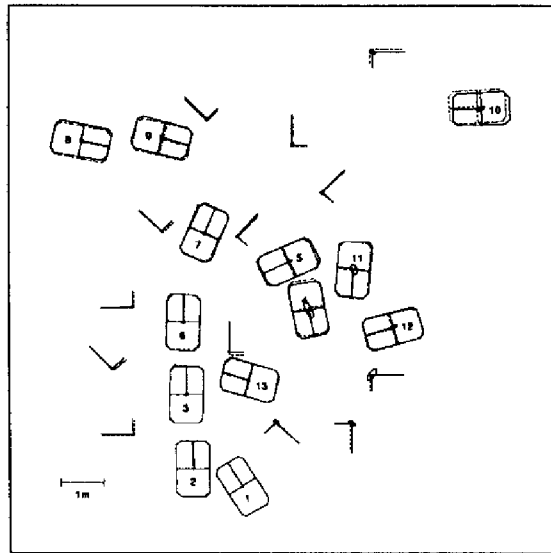
	Erreur Trans. (cm)		Erreur Rot. (deg)	
	Moy.	Max	Moy.	Max.
Avec corrélations	4.	8.	.6	1.
Sans corrélation	6.	10.	1.	1.7

FIG. VIII.23 – *Erreur en translation et en rotation de la carte construite*

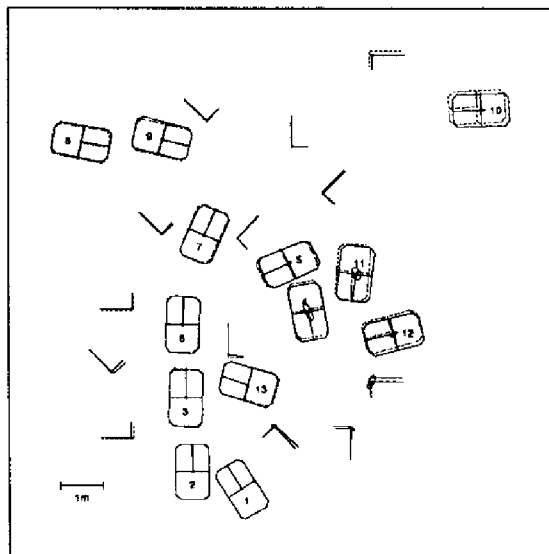
de pouvoir maintenir un modèle cohérent de l'environnement, et ce grâce à une gestion rigoureuse des incertitudes.

Afin de bien valider cette approche, il faudrait maintenant l'évaluer sur des trajectoires de plus longue distance, mais aussi avec des données issues de la stéréo. Aussi, dans le cadre du projet I-ARES, nous sommes en train de la tester sur une longue série d'images stéréo (prises aussi sur le terrain Geroms). Il sera intéressant de voir quelle précision on peut atteindre à la fois sur la position du robot et sur la carte construite avec ce type de données. De plus, pour accélérer le calcul de l'image 3D, il est possible de ne considérer qu'un pixel sur deux (puis un sur quatre...); il faudra alors étudier de quelle manière cela affecte la carte et le recalage.

Enfin, un point très important qui reste à régler consiste en l'intégration du modèle par objets dans l'ensemble des modèles de l'environnement présents à bord du robot. La structure simple de représentation des objets permet de faciliter grandement les interfaçages à réaliser. Des travaux ont ainsi débuté dans le groupe pour l'intégration de ce modèle avec la carte des régions.



(a)



(b)

FIG. VIII.24 – Modélisation incrémentale et recalage du robot. En pointillés : position réelle du robot et des boîtes. En noir : position du robot et des boîtes estimées en utilisant les corrélations (a) et sans les utiliser (b).

Conclusion Générale

Dans ce travail de thèse, nous avons abordé la modélisation incrémentale d'un environnement naturel par un robot mobile autonome.

Après une analyse des différentes caractéristiques que doivent présenter les modèles, nous avons tiré profit d'une certaine structure inhérente à un grand nombre d'environnements naturels pour proposer une représentation particulière basée sur la notion d'objets. Celle-ci s'oppose aux approches plus classiques où le terrain est représenté par une surface continue. Mais ces deux types de représentation sont en réalité complémentaires et seront construites en fonction de la nature du terrain pour s'intégrer dans un modèle multi-représentations qui sera présent à bord du robot.

Tout au long de ce mémoire, nous avons alors décrit les méthodes et les algorithmes permettant de construire le modèle par objets, puis leurs résultats dans différents environnements extérieurs ont été présentés. Les contributions de ce travail se situent principalement au niveau des deux fonctions suivantes :

- Modélisation de l'environnement

Le modèle par objets permet une description plus abstraite de l'environnement (notamment par la définition de relations topologiques). Il permet certaines interprétations et raisonnements qualitatifs.

C'est une représentation peu volumineuse car focalisée sur la modélisation de zones d'intérêts (*i.e.* les objets).

Mais c'est aussi un modèle précis de l'environnement par les caractéristiques extraites sur les objets sélectionnés en tant qu'amers. L'utilisation de la carte stochastique permet de gérer les incertitudes de manière rigoureuse ; de plus, faisant partie des représentations discrètes, le modèle par objets conserve sa cohérence malgré les erreurs de localisation.

- Localisation

Le robot sélectionne les amers parmi les objets sur la base de critères de distinction et de précision. Les caractéristiques extraites (et leur incertitude) ont permis de construire la carte stochastique. La localisation du robot (ainsi que la mise à jour de la carte) est réalisée grâce au filtre de Kalman. Le principe de recalage-fusion et une version itérative du filtre ont été utilisés pour résoudre des problèmes de biais (fréquents en environnement naturel) et de non linéarité des équations de mesure.

Enfin, nous nous sommes attachés à évaluer l'approche par objets pour ces deux fonctions en comparant à des valeurs de référence la carte construite et les positions recalées

du robot. Mais toute approche de modélisation incrémentale demande à être validée par de nombreuses expérimentations. Il convient de multiplier les essais dans des sites et avec des capteurs différents. D'autres tests sont ainsi actuellement menés pour apprécier le modèle construit à partir de données stéréoscopiques.

Par ailleurs différentes directions de recherche sont envisageables. Nous avons déjà évoqué au cours du mémoire la possibilité d'enrichir le modèle des objets par des représentations plus fines (surfaces déformables, maillages 3D, ...) ou de les décomposer en différentes parties. La reconnaissance des objets et du but bénéficierait bien évidemment de cette information supplémentaire.

Mais on pourrait aussi exploiter la notion d'objets pour d'autres fonctions de la navigation du robot. Nous donnons deux exemples :

– Perception dynamique

Dans ce travail, nous avons considéré le robot dans un cycle de perception à l'arrêt puis de déplacement. Le temps d'acquisition d'une image tridimensionnelle par notre capteur laser en est la principale raison (1 à 2 mn). Mais la technologie des capteurs 3D évolue et cette contrainte de temps pourraient grandement être réduite (utilisation par exemple d'algorithmes rapides de stéréovision). Il conviendrait alors d'étudier les modifications que cela entraînerait dans la construction du modèle par objets.

On pourrait par exemple imaginer le scénario suivant : le robot est tout d'abord à l'arrêt et prend une vue panoramique de l'environnement dont il détecte et modélise les objets. En fonction de la direction qu'il doit emprunter, il sélectionne quelques amers. Il se déplace alors dans cette direction tout en réobservant ces amers et en les suivant d'une image à l'autre (la segmentation devant être suffisamment rapide et focalisée). La perception sera dans ce cas réalisée pendant le déplacement et prendra en compte les équations du mouvement [Zhang 92].

– Perception active

Une autre direction particulièrement intéressante consisterait à exploiter les objets dans la planification de perception.

Nous avons évoqué comment le modèle topologique d'une scène permettait de planifier qualitativement un point de vue pour lever des contraintes d'occultation entre objets. Ces relations topologiques pourraient être mises à profit dans d'autres situations (notamment la relation de proximité entre les objets du modèle global qui permettrait de déterminer les prochains objets à percevoir).

D'autre part, les amers et leur incertitude pourraient guider le choix d'un point de vue. Cet aspect a été déjà commencé à être exploré et réalisé par S. Lacroix et il serait intéressant d'y intégrer la notion d'objet. Le robot prédit (par exemple à l'aide du filtre de Kalman) comment la perception de certains amers réduit l'incertitude sur sa position (gain de localisation). Il serait dans ce cas tout à fait intéressant d'en identifier des configurations spécifiques qui maximiseraient ce gain. Par ailleurs, le robot estime l'intérêt de modéliser certaines parties encore inconnues ou d'affiner le modèle d'autres déjà perçues (gain de modélisation). Le choix de la prochaine

perception est un compromis entre ces deux gains et peut être traité par la théorie de la décision [Lacroix 95a, Hu 93].

Enfin une des priorités reste l'intégration de ce modèle avec les autres représentations. Cela permettrait de montrer comment la multiplicité des représentations permet d'adapter la navigation du robot à son environnement et par ce moyen d'en augmenter l'autonomie.



Annexe A

Représentation de l'incertitude

A.1 Représentation graphique de l'incertitude d'un point

Considérons un point 3D représenté par un vecteur aléatoire $P = [x \ y \ z]^T$ et ses deux premiers moments \hat{P} et σ_P . Pour représenter graphiquement l'incertitude de ce point, il convient d'abord de choisir une distribution particulière (ce choix n'impliquant bien entendu que la représentation graphique). Les distributions gaussiennes sont habituellement utilisées car elles ont l'avantage d'être entièrement définies par leur deux premiers moments. La forme générale de ce type de distribution pour un vecteur \mathbf{x} de dimension n et dont la moyenne et la variance sont respectivement $\hat{\mathbf{x}}$ et $\sigma_{\mathbf{x}}$ est donnée par l'équation A.1 [Hébert 94, Smith 86].

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det \sigma_{\mathbf{x}}}} e^{-1/2[(\mathbf{x}-\hat{\mathbf{x}})^T \sigma_{\mathbf{x}}^{-1} (\mathbf{x}-\hat{\mathbf{x}})]} \quad (\text{A.1})$$

Nous pouvons représenter les surfaces d'isoprobabilité de \mathbf{x} par l'équation A.2.

$$(\mathbf{x} - \hat{\mathbf{x}})^T \sigma_{\mathbf{x}}^{-1} (\mathbf{x} - \hat{\mathbf{x}}) = k^2 \quad (\text{A.2})$$

Ces surfaces sont des hyper-ellipsoïdes de \mathfrak{R}^n ; k est une constante qui définit le pourcentage de chance que le vrai point a de se trouver à l'intérieur de l'hyper-ellipsoïde. Pour un pourcentage fixé, cette constante est déterminée par une loi de χ^2 à n degrés de liberté.

Dans notre cas, n a la valeur 3 et les surfaces d'isoprobabilité sont donc des ellipsoïdes représentables en perspective dans l'espace 3D. Pour cela il suffit de calculer les vecteurs propres $(\vec{V}_i)_{i=1,3}$ et les valeurs propres $(v_i)_{i=1,3}$ de σ_P . On calcule alors l'ellipsoïde dans son repère propre par l'équation A.3. χ_3^2 représente la valeur d'un χ^2 à 3 degrés de liberté, nous donnons dans le tableau de la figure A.1 ses valeurs pour différents pourcentages. On établit ensuite son équation dans le repère global en appliquant une rotation fixée par les vecteurs propres calculés.

$$\begin{bmatrix} \chi_3^2 \sqrt{v_1} \cos \theta \cos \varphi \\ \chi_3^2 \sqrt{v_2} \sin \theta \cos \varphi \\ \chi_3^2 \sqrt{v_3} \sin \varphi \end{bmatrix} \text{ avec } \begin{cases} \theta \in [-\pi, \pi] \\ \varphi \in [-\frac{\pi}{2}, \frac{\pi}{2}] \end{cases} \quad (\text{A.3})$$

L'ellipsoïde est une représentation complète de l'incertitude d'un point, mais elle n'est pas toujours souhaitable. Il est parfois préférable de ne représenter le point que par ses deux premières coordonnées (*i.e.* dans le plan (x, y)). Nous chercherons alors à représenter la partie du plan (x, y) dans laquelle le point a un certain pourcentage de chance de se trouver; et cela, **quelque soit sa valeur en z** . Cela revient à définir les courbes d'isoprobabilité (ellipses) de la loi suivante :

$$p'(x, y) = \int_{-\infty}^{+\infty} p(x, y, z) dz \quad (\text{A.4})$$

On montre que le second moment du vecteur $[x \ y]^T$ est alors égal à la sous matrice 2×2 de σ_P en x et y . On calcule les vecteurs propres $(\vec{V}'_i)_{i=1,2}$ et les valeurs propres $(v'_i)_{i=1,2}$ de cette sous matrice et on exprime l'équation de l'ellipse dans son repère propre (en utilisant cette fois une valeur de χ^2 à 2 degrés de liberté) :

$$\begin{bmatrix} \chi_2^2 \sqrt{v'_1} \cos \theta \\ \chi_2^2 \sqrt{v'_2} \sin \theta \end{bmatrix} \text{ avec } \theta \in [-\frac{\pi}{2}, \frac{\pi}{2}] \quad (\text{A.5})$$

Pour cette représentation, on parle de projection de distribution gaussienne dans le plan (x, y) . Toutefois, nous soulignons que cette ellipse ne correspond pas à la projection géométrique de l'ellipsoïde dans le plan (x, y) , mais à la région de ce plan où le point a un certain pourcentage de chance de se projeter.

Aussi nous soulignons que la surface de l'ellipse ne représente qu'une partie de l'incertitude du repère. A titre d'exemple, observons la figure VIII.17 (p. 127) qui montre différentes positions odométriques du robot Adam et leurs incertitudes. Ces dernières ne peuvent bien évidemment que croître au cours du déplacement. Or nous observons qu'entre les positions 11 et 12 la surface de l'ellipse a diminué. L'incertitude de l'ensemble de la configuration du robot a pourtant augmenté, mais en raison du mouvement de rotation entre ces deux positions, la matrice de variance a été modifiée de telle sorte que sa projection dans le plan (x, y) donne une ellipse plus petite. On peut visualiser cette situation en considérant l'ellipsoïde d'incertitude dans son repère propre. Le mouvement de rotation du robot l'ayant fait tourner dans son repère, sa visualisation dans le plan (x, y) est modifiée.

Le tableau de la figure A.1 donne la valeur du χ^2 à un, deux ou trois degrés de liberté et pour les pourcentages 90, 95 et 99 ([Spiegel 83]).

A.2 Représentation graphique de l'incertitude d'un repère

On peut représenter l'incertitude d'un repère 2D $[x \ y \ \theta]^T$ dans un espace 3D comme nous l'avons décrit pour le point, en remplaçant z par θ . Mais cette représentation n'a

	99%	95%	90%
χ_1^2	6.63	3.84	2.71
χ_2^2	9.21	5.99	4.61
χ_3^2	11.3	7.81	6.25

FIG. A.1 – Différentes valeurs de la distribution de χ^2 .

pas de sens physique. Aussi on préfère ne représenter graphiquement que l'ellipse correspondant à sa translation (x, y) , en appliquant la technique précédente. En revanche, si ce repère correspond à la position et à l'orientation d'un robot, on pourra tracer sur celui-ci un secteur angulaire fonction de la variance de son cap θ (de valeur $\chi_1^2 \sqrt{\sigma_{\theta\theta}}$).

L'incertitude complète d'un repère 3D est impossible à représenter dans l'espace 3D (six variables sont nécessaires). On peut représenter la translation 3D (x, y, z) par un ellipsoïde, mais ceci est plus difficile pour les rotations. Une représentation 2D est plus souvent utilisée. On se contente alors de représenter le sous système (x, y, θ) avec toujours le même procédé.

Si maintenant on veut représenter un ensemble de n repères $(x_i)_{i=1,n}$ corrélés les uns aux autres, la seule solution est de les représenter indépendamment les uns des autres. Il est important de remarquer que les corrélations n'apparaîtront pas dans cette visualisation et donc que cette dernière est très incomplète.

A.3 Calcul des deux premiers moments d'une distribution uniforme

Soit $\mathbf{x} = (x_1, \dots, x_n)$ un vecteur aléatoire à valeur dans \mathbb{R}^n et p une loi de probabilité sur \mathbf{x} qui soit définie uniformément à l'intérieur d'un ensemble $\Omega \subset \mathbb{R}^n$. Si V_Ω désigne le volume de Ω (donné par l'équation A.6), alors la distribution uniforme p est définie par la formule A.7.

$$V_\Omega = \int_\Omega d\mathbf{x} \quad (\text{A.6})$$

$$p(\mathbf{x}) = \begin{cases} \frac{1}{V_\Omega} & \text{si } \mathbf{x} \in \Omega \\ 0 & \text{sinon} \end{cases} \quad (\text{A.7})$$

Les équations A.8 et A.9 expriment alors les deux premiers moments statistiques de \mathbf{x} .

$$\hat{\mathbf{x}} = E[\mathbf{x}] = \frac{1}{V_\Omega} \int_\Omega \mathbf{x} d\mathbf{x} \quad (\text{A.8})$$

$$\sigma_{\mathbf{x}} = E[\varepsilon_{\mathbf{x}} \varepsilon_{\mathbf{x}}^T] = \frac{1}{V_\Omega} \int_\Omega (\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T d\mathbf{x} \quad (\text{A.9})$$

avec

$$\sigma_{ij} = \frac{1}{V_\Omega} \int_\Omega (x_i - \hat{x}_i)(x_j - \hat{x}_j)^T d\mathbf{x}$$

Exemple de distribution uniforme dans un cas unidimensionnel

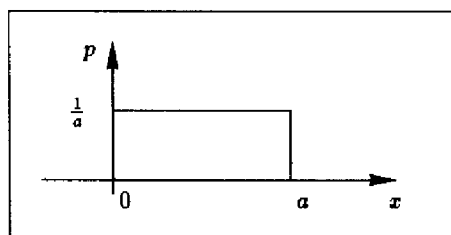


FIG. A.2 – Distribution de probabilité uniforme et unidimensionnelle.

Considérons une variable aléatoire x à valeur dans \Re telle que sa loi de probabilité p soit uniformément définie dans $[0, a]$. Dans ce cas $V_{[0,a]} = a$, la figure A.2 donne alors l'allure de p .

Le premier moment de x vaut :

$$\hat{x} = \frac{1}{a} \int_0^a x \, dx = \frac{a}{2} \quad (\text{A.10})$$

et son second moment :

$$\sigma_x = \frac{1}{a} \int_0^a \left(x - \frac{a}{2}\right)^2 \, dx = \frac{a^2}{12} \quad (\text{A.11})$$

Ce résultat est par exemple utilisé pour tenir compte de l'échantillonnage d'un capteur dans l'estimation du rayon intersectant le bord réel d'une boîte (cf section V.4).

Annexe B

Combinaison de transformations incertaines

Nous rappelons dans cette annexe les différentes formules qui permettent de combiner des transformations incertaines dans le plan ou dans l'espace. Ces formules seront applicables dans tous les cas de figure à la condition que si deux transformations sont corrélées, elles doivent alors (i) se suivre et (ii) n'être corrélées avec aucune des autres transformations. Pour plus de détails, consulter la référence [Smith 90].

Reprenons tout d'abord les expressions d'une transformation τ_i dans le plan :

$$\tau_i = [x_i \ y_i \ \theta_i]^T \quad (\text{B.1})$$

et dans l'espace (en utilisant les angles de Briant) :

$$\tau_i = [x_i \ y_i \ z_i \ \theta_i \ \varphi_i \ \psi_i]^T \quad (\text{B.2})$$

B.1 Composition de deux transformations incertaines

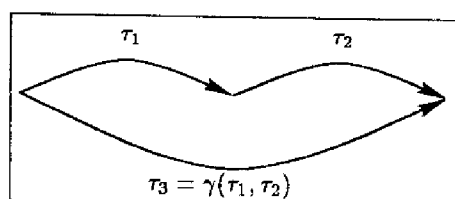


FIG. B.1 – Composition de deux transformations incertaines.

Considérons deux transformations incertaines τ_1 et τ_2 . Ce sont deux vecteurs aléatoires dont nous possédons pour chacun les deux premiers moments statistiques : $(\hat{\tau}_1, \sigma_{11})$ et

$(\hat{\tau}_2, \sigma_{22})$. Ces transformations peuvent être corrélées, auquel cas nous devons connaître leur matrice de corrélation σ_{12} . Appelons maintenant τ_3 la transformation obtenue par composition de τ_1 et τ_2 (cf fig. B.1). Rappelant que γ est la fonction de composition de deux transformations (cf VI.30), nous pouvons alors écrire :

$$\tau_3 = \gamma(\tau_1, \tau_2) \quad (\text{B.3})$$

Nous cherchons à estimer les deux premiers moments de τ_3 , $(\hat{\tau}_3, \sigma_{33})$, ce qui est simplement un problème de propagation d'incertitude comme il a été présenté dans la partie VI.2.1. Désignons par Υ la jacobienne de γ par rapport à (x_1, x_2) , nous pouvons appliquer directement les formules VI.17 et VI.18 sur γ :

$$\hat{\tau}_3 = \gamma(\hat{\tau}_1, \hat{\tau}_2) \quad (\text{B.4})$$

$$\sigma_{33} = \Upsilon \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \Upsilon^T \quad \text{avec} \quad \Upsilon = (\Upsilon_1 \Upsilon_2) \quad (\text{B.5})$$

Les expressions de γ , Υ_1 et Υ_2 seront données dans les cas 2D ou 3D. Nous supprimerons l'indicateur de l'estimée (" $\hat{\cdot}$ ") pour alléger l'écriture.

B.1.1 Composition : cas 2D

$$\tau_3 = \begin{bmatrix} x_2 \cos \theta_1 - y_2 \sin \theta_1 \\ x_2 \sin \theta_1 + y_2 \cos \theta_1 \\ \theta_1 + \theta_2 \end{bmatrix} \quad (\text{B.6})$$

$$\Upsilon_1 = \begin{bmatrix} 1 & 0 & -(y_3 - y_1) \\ 0 & 1 & x_3 - x_1 \\ 0 & 0 & 1 \end{bmatrix}, \quad \Upsilon_2 = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 \\ \sin \theta_1 & -\cos \theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.7})$$

B.1.2 Composition : cas 3D

Appelons \mathbf{R}_i la matrice de rotation de la transformation τ_i :

$$\begin{aligned} \mathbf{R}_i &= \begin{bmatrix} \cos \theta_i \cos \varphi_i & \cos \theta_i \sin \varphi_i \sin \psi_i - \sin \theta_i \cos \psi_i & \cos \theta_i \sin \varphi_i \cos \psi_i + \sin \theta_i \sin \psi_i \\ \sin \theta_i \cos \varphi_i & \sin \theta_i \sin \varphi_i \sin \psi_i + \cos \theta_i \cos \psi_i & \sin \theta_i \sin \varphi_i \cos \psi_i - \cos \theta_i \sin \psi_i \\ -\sin \varphi_i & \cos \varphi_i \sin \psi_i & \cos \varphi_i \cos \psi_i \end{bmatrix} \\ &= (r_{uv}^i)_{u,v=1..3} \end{aligned}$$

$$\mathbf{T} = \mathbf{R}_1 \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}, \quad \mathbf{R}_3 = \mathbf{R}_1 \mathbf{R}_2$$

$$\mathbf{A} = \begin{bmatrix} \text{atan2}(r_{21}^3, r_{11}^3) \\ \text{atan2}(-r_{11}^3, r_{11}^3 \cos \theta_3 + r_{21}^3 \sin \theta_3) \\ \text{atan2}(r_{13}^3 \sin \theta_3 - r_{23}^3 \cos \theta_3, r_{12}^3 \sin \theta_3 + r_{22}^3 \cos \theta_3) \end{bmatrix}$$

$$\tau_3 = \begin{bmatrix} \mathbf{T} \\ \mathbf{A} \end{bmatrix} \quad (\text{B.8})$$

$$\mathbf{M} = \begin{bmatrix} -(y_3 - y_1) & (z_3 - z_1) \cos \theta_1 & r_{13}y_2 - r_{12}z_2 \\ x_3 - x_1 & (z_3 - z_1) \sin \theta_1 & r_{23}y_2 - r_{22}z_2 \\ 0 & -x_2 \cos \varphi_1 - y_2 \sin \varphi_1 \sin \psi_1 - z_2 \sin \varphi_1 \cos \psi_1 & r_{33}y_2 - r_{32}z_2 \end{bmatrix}$$

$$\mathbf{K}_1 = \begin{bmatrix} 1 & [\sin \varphi_3 \sin(\theta_3 - \theta_1)] / \cos \varphi_3 & [r_{12}^2 \sin \psi_3 + r_{13}^2 \cos \psi_3] / \cos \varphi_3 \\ 0 & \cos(\theta_3 - \theta_1) & -\cos \varphi_1 \sin(\theta_3 - \theta_1) \\ 0 & [\sin(\theta_3 - \theta_1)] / \cos \varphi_3 & [\cos \varphi_1 \cos(\theta_3 - \theta_1)] / \cos \varphi_3 \end{bmatrix}$$

$$\mathbf{K}_2 = \begin{bmatrix} [\cos \varphi_2 \cos(\psi_3 - \psi_2)] / \cos \varphi_3 & [\sin(\psi_3 - \psi_2)] / \cos \varphi_3 & 0 \\ -\cos \varphi_2 \sin(\psi_3 - \psi_2) & \cos(\psi_3 - \psi_2) & 0 \\ [r_{13}^1 \cos \theta_3 + r_{23}^1 \sin \theta_3] / \cos \varphi_3 & [\sin \varphi_3 \sin(\psi_3 - \psi_2)] / \cos \varphi_3 & 1 \end{bmatrix}$$

$$\mathbf{\Upsilon}_1 = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{M} \\ \mathbf{O}_{3 \times 3} & \mathbf{K}_1 \end{bmatrix}, \quad \mathbf{\Upsilon}'_2 = \begin{bmatrix} \mathbf{R}_1 & \mathbf{O}_{3 \times 3} \\ \mathbf{O}_{3 \times 3} & \mathbf{K}_2 \end{bmatrix} \quad (\text{B.9})$$

Remarque:

Si τ'_2 ne représente qu'une translation incertaine (i.e. $\tau'_2 = [x_2 \ y_2]^T$ en 2D, ou $\tau'_2 = [x_2 \ y_2 \ z_2]^T$ en 3D), alors appelons γ' la fonction permettant de calculer la translation τ'_3 composée de la transformation τ_1 avec cette translation τ'_2 :

$$\tau'_3 = \gamma'(\tau_1, \tau'_2) \quad (\text{B.10})$$

γ' se déduit simplement de γ en supprimant les termes de rotation liés à la deuxième variable. La matrice jacobienne $\mathbf{\Upsilon}'$ de γ' se décompose de même en deux sous matrices $\mathbf{\Upsilon}'_1$ et $\mathbf{\Upsilon}'_2$ qui valent en 2D:

$$\mathbf{\Upsilon}'_1 = \begin{bmatrix} 1 & 0 & -(y_3 - y_1) \\ 0 & 1 & x_3 - x_1 \end{bmatrix}, \quad \mathbf{\Upsilon}'_2 = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 \\ \sin \theta_1 & -\cos \theta_1 \end{bmatrix} \quad (\text{B.11})$$

et en 3D:

$$\mathbf{\Upsilon}'_1 = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{M} \end{bmatrix}, \quad \mathbf{\Upsilon}'_2 = \mathbf{R}_1 \quad (\text{B.12})$$

B.2 Inversion d'une transformation incertaine

Considérons la transformation τ_1 précédente, et appelons π la fonction qui inverse une transformation. Si τ_4 désigne l'inverse de τ_1 (cf fig. B.2), nous pouvons écrire:

$$\tau_4 = \pi(\tau_1) \quad (\text{B.13})$$

Si de plus $\mathbf{\Pi}$ est la jacobienne de π par rapport à τ_1 , nous pouvons calculer les deux premiers moments de τ_4 par propagation de l'incertitude à travers π :

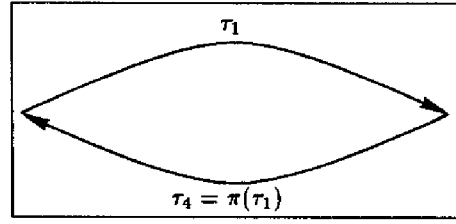


FIG. B.2 – Inversion d'une transformation incertaine.

$$\hat{\tau}_4 = \pi(\hat{\tau}_1) \quad (\text{B.14})$$

$$\sigma_{44} = \mathbf{\Pi} \sigma_{11} \mathbf{\Pi}^T \quad (\text{B.15})$$

Nous donnons les expressions de π et $\mathbf{\Pi}$ dans les cas 2D et 3D.

B.2.1 Inversion : cas 2D

$$\tau_4 = \begin{bmatrix} -x_1 \cos \theta_1 - y_1 \sin \theta_1 \\ x_1 \sin \theta_1 - y_1 \cos \theta_1 \\ -\theta_1 \end{bmatrix} \quad (\text{B.16})$$

$$\mathbf{\Pi} = \begin{bmatrix} -\cos \theta_1 & -\sin \theta_1 & y_4 \\ \sin \theta_1 & \cos \theta_1 & -x_4 \\ 0 & 0 & -1 \end{bmatrix} \quad (\text{B.17})$$

B.2.2 Inversion : cas 3D

$$\tau_4 = \begin{bmatrix} -(r_{11}^1 x_1 + r_{21}^1 y_1 + r_{31}^1 z_1) \\ -(r_{12}^1 x_1 + r_{22}^1 y_1 + r_{32}^1 z_1) \\ -(r_{13}^1 x_1 + r_{23}^1 y_1 + r_{33}^1 z_1) \\ \text{atan2}(r_{12}^1, r_{11}^1) \\ \text{atan2}(-r_{13}^1, r_{11}^1 \cos \theta_4 + r_{12}^1 \sin \theta_4) \\ \text{atan2}(r_{31}^1 \sin \theta_4 - r_{32}^1 \cos \theta_4, -r_{21}^1 \sin \theta_4 - r_{22}^1 \cos \theta_4) \end{bmatrix} \quad (\text{B.18})$$

$$\mathbf{N} = \begin{bmatrix} r_{21}^1 x_1 - r_{11}^1 y_1 & -r_{31}^1 x_1 \cos \theta_1 - r_{31}^1 y_1 \sin \theta_1 + z_1 \cos \varphi_1 & 0 \\ r_{22}^1 x_1 - r_{12}^1 y_1 & -r_{32}^1 x_1 \cos \theta_1 - r_{32}^1 y_1 \sin \theta_1 + z_1 \sin \theta_1 \sin \psi_1 & z_4 \\ r_{23}^1 x_1 - r_{13}^1 y_1 & -r_{33}^1 x_1 \cos \theta_1 - r_{33}^1 y_1 \sin \theta_1 + z_1 \sin \theta_1 \cos \psi & -y_4 \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} -r_{33}^1/(1 - r_{13}^1 r_{13}^1) & -r_{23}^1 \cos \theta_1/(1 - r_{13}^1 r_{13}^1) & r_{11}^1 r_{13}^1/(1 - r_{13}^1 r_{13}^1) \\ r_{23}^1/(1 - r_{13}^1 r_{13}^1)^{1/2} & -r_{33}^1 \cos \theta_1/(1 - r_{13}^1 r_{13}^1)^{1/2} & r_{12}^1/(1 - r_{13}^1 r_{13}^1)^{1/2} \\ r_{33}^1 r_{13}^1/(1 - r_{13}^1 r_{13}^1) & -r_{12}^1 \cos \psi_1/(1 - r_{13}^1 r_{13}^1) & -r_{11}^1/(1 - r_{13}^1 r_{13}^1) \end{bmatrix}$$

$$\mathbf{\Pi} = \begin{bmatrix} -\mathbf{R}^T & \mathbf{N} \\ \mathbf{O}_{3 \times 3} & \mathbf{Q} \end{bmatrix} \quad (\text{B.19})$$

Remarque :

Les matrices de variance σ_{11} et σ_{44} ont les mêmes valeurs propres. En effet, si D_1 est la matrice diagonale des valeurs propres de σ_{11} et P_1 celle de ses vecteurs propres, nous avons :

$$\sigma_{11} = P_1 D_1 P_1^T \quad (\text{B.20})$$

La matrice de variance σ_{44} s'obtient en appliquant la formule B.15 :

$$\sigma_{44} = \mathbf{\Pi} \sigma_{11} \mathbf{\Pi}^T = \mathbf{\Pi} P_1 D_1 P_1^T \mathbf{\Pi}^T = (\mathbf{\Pi} P_1) D_1 (\mathbf{\Pi} P_1)^T = P_2 D_1 P_2^T \quad (\text{B.21})$$

D_1 est donc bien la matrice des valeurs propres de σ_{44} , la matrice des vecteurs propres P_2 étant le produit de la jacobienne $\mathbf{\Pi}$ avec la matrice des vecteurs propres P_1 de σ_{11} . Ceci se comprend dans la mesure où l'inversion d'une transformation n'ajoute bien évidemment pas d'incertitude sur la transformation résultante.

Si nous supposons maintenant que τ_1 et τ_4 suivent des lois de distributions gaussiennes, alors les ellipses (ou ellipsoïdes) d'incertitude les représentant (cf A.2) sont identiques. Elles sont simplement orientées différemment.

B.3 Inversion et composition de deux transformations incertaines



FIG. B.3 – Inversion et composition de deux transformations incertaines.

Appelons δ la fonction d'inversion-composition de deux transformations incertaines (cf fig. B.3). En reprenant les notations précédentes nous pouvons calculer τ_5 (inverse-composée respectivement de τ_1 et τ_2) par :

$$\tau_5 = \delta(\tau_1, \tau_2) = \gamma(\pi(\tau_1), \tau_2) \quad (\text{B.22})$$

Appelons Δ la jacobienne de δ par rapport à (τ_1, τ_2) , nous obtenons :

$$\hat{\tau}_5 = \delta(\hat{\tau}_1, \hat{\tau}_2) \quad (\text{B.23})$$

$$\sigma_{55} = \Delta \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \Delta^T \quad \text{avec} \quad \Delta = (\Delta_1 \Delta_2) \quad (\text{B.24})$$

Dans les cas 2D et 3D, nous pouvons exprimer de la même manière les matrices Δ_1 et Δ_2 en fonction des matrices Υ_1 , Υ_2 et Π calculée auparavant :

$$\Delta_1 = \Upsilon_1 \Pi, \quad \Delta_2 = \Upsilon_2 \quad (\text{B.25})$$

Références bibliographiques

- [Asada 88] M. Asada. *Building a 3-D World Model for a Mobile Robot from Sensory Data*. In IEEE International Conference on Robotics and Automation, Philadelphia (USA), 1988.
- [Asada 90] M. Asada. *Map Building for a Mobile Robot from Sensory Data*. IEEE Transactions on Systems, Man and Cybernetics, vol. 20, no. 6, 1990.
- [Ayache 88a] N. Ayache. *Construction et Fusion de Représentations Visuelles 3D. Applications à la Robotique Mobile*. Thèse d'état, Université de Paris-sud, 1988.
- [Ayache 88b] N. Ayache & C. Hansen. *Rectification of images for binocular and trinocular stereovision*. In 9th IEEE International Conference on Pattern Recognition, Rome (Italy), pages 11–16, November 1988.
- [Ayache 89] N. Ayache. *Vision stéréoscopique et perception multisensorielle. Applications à la robotique mobile*. InterEditions, Paris, France, 1989.
- [Ayache 90] N. Ayache & O. D. Faugeras. *Maintaining Representations of the Environment of a Mobile Robot*. Autonomous Robot Vehicules, pages 205–220, 1990.
- [Bar-Shalom 88] Y. Bar-Shalom & T. E. Fortmann. Tracking and data association, volume 179 of *Mathematics in Science and Engineering*. Academic Press, 1988.
- [Bard 74] Y. Bard. *Non linear parameter estimation*. Academic Press, New York, 1974.
- [Barr 91] A. Barr. *Superquadrics and Angle-Preserving Transformations*. In IEEE Computer Graphics and Applications, 1991.
- [Besl 92] P. J. Besl & N. McKay. *A Method for Registration of 3-D Shapes*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, 1992.
- [Betgé-Brezetz 93] S. Betgé-Brezetz. *Fusion d'images tridimensionnelles par extraction de caractéristiques*. Tech. Report LAAS-93527, 1993.

- [Betgé-Brezetz 94a] S. Betgé-Brezetz, R. Chatila & M. Devy. *Natural Scene Understanding for Mobile Robot Navigation*. In Proc. IEEE International Conference on Robotics and Automation, San Diego, California, USA, May 1994.
- [Betgé-Brezetz 94b] S. Betgé-Brezetz, R. Chatila, M. Devy, P. Fillatreau & F. Nashashibi. *Adaptive Localization of an Autonomous Mobile Robot in Natural Environments*. In Proc. of the IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI'94), Las Vegas, USA, October 1994.
- [Betgé-Brezetz 95] S. Betgé-Brezetz, R. Chatila & M. Devy. *Object-based Modelling and Localization in Natural Environment*. In Proc. IEEE International Conference on Robotics and Automation, Nagoya, Japan, May 1995.
- [Betgé-Brezetz 96] S. Betgé-Brezetz, P. Hébert, R. Chatila & M. Devy. *Uncertain Map Making in Natural Environment*. In submitted to IEEE International Conference on Robotics and Automation, West Lafayette, USA, April 1996.
- [Biederman 87] I. Biederman. *Recognition-by-components: A theory of human image understanding*. *Psychological Rev.*, vol. 94, no. 2, pages 115-147, 1987.
- [Bobick 92] A. F. Bobick & R. Bolles. *The Representation Space Paradigm of Concurrent Evolving Object Description*. *IEEE Transactions on Robotics and Automation*, vol. 14, no. 2, 1992.
- [Boissonnat 87] J.D. Boissonnat, O.D. Faugeras & E. Le Bras-Mehlan. *Représentation de données stéréo par la triangulation de Delaunay*. In 6ème Congrès AFCET Reconnaissance des Formes et Intelligence Artificielle, Antibes (France), pages 687-701, November 1987.
- [Bolles 82] R.C. Bolles & R.A. Cain. *Recognising and locating partially visible objects: the local-feature-focus method*. *International Journal of Robotics Research*, vol. 1, no. 3, pages 57-82, 1982.
- [Boumaza 94] R. Boumaza. *Perception Multisensorielle pour la Reconnaissance d'objets tridimensionnels*. Thèse de l'Université Paul Sabatier, Toulouse (France) 94353, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), July 1994.
- [Brady 85] M. Brady, J. Ponce, A. Yuille & H. Asada. *Describing Surfaces*. *Computer Vision, Graphics, and Image Processing*, vol. 32, pages 1-28, 1985.
- [Bulata 96] H. Bulata & M. Devy. *Incremental Construction of a Landmark-based and Topological Model of Indoor Environments by a Mobile Robot*. In submitted to IEEE International Conference on Robotics and Automation, West Lafayette, USA, April 1996.

- [Carmo 76] M.P. Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, 1976.
- [Chatila 85] R. Chatila & J.P. Laumond. *Position referencing and consistent world modeling for mobile robots*. In IEEE International Conference on Robotics and Automation, St Louis (USA), April 1985.
- [Chatila 93a] R. Chatila. *Representation + Reason + Reaction → Robot Intelligence*. In 6th International Symposium on Robotics Research. ISSR, Hidden Valley (Pennsylvania, USA), October 1993.
- [Chatila 93b] R. Chatila, S. Fleury, M. Herrb, S. Lacroix & C. Proust. *Autonomous navigation in natural environment*. In 3rd International Symposium on Experimental Robotics. ISER, Kyoto (Japan), October 1993.
- [Chatila 93c] R. Chatila, B. Dacre Wright, M. Devy, P. Fillatreau, S. Lacroix, F. Nashashibi, P. Pignon & T. Siméon. *Génération de déplacements autonomes*. Contrat RISP, Projet I.ARES, Juillet 1993, 108p. 93272, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), Toulouse (France), July 1993.
- [Chen 91] Y. Chen & G. Medioni. *Object Modeling by Registration of Multiple Range Images*. In IEEE International Conference on Robotics and Automation, Sacramento, (USA), pages 2724 - 2729, April 1991. Fusion et Intégration d'Images de Profondeur.
- [Chen 94] Y. Chen & G. Medioni. *Surface Description of Complex Objects from Multiple Ranges Images*. In IEEE Conference on Computer Vision and Pattern Recognition, Seattle, Washington, 1994.
- [Cohen 92] L. D. Cohen & I. Cohen. *Deformable Models for 3D Medical Images using Finite Elements and Ballons*. In IEEE Conference on Computer Vision and Pattern Recognition, Champaign, Illinois (USA), pages 592-598, 1992.
- [Colly 91] J. Colly. *Reconnaissance d'Objets Articulés Tridimensionnels à Partir d'Informations Multisensorielles pour des Applications Robotiques*. Thèse de l'Université Paul Sabatier, Toulouse (France), Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), October 1991.
- [Connell 87] J.H. Connell & M. Brady. *Generating and Generalizing Models of Visual Objects*. *Artificial Intelligence*, vol. 31, no. 2, pages 159-183, 1987.
- [Dai 93] D. Dai & D. T. Lawton. *Range-Free Qualitative Navigation*. In IEEE International Conference on Robotics and Automation, Atlanta, (USA), 1993.

- [Daily 88] M. Daily, J. Harris, D. Keirse, K. Olin, D. Payton, K. Reiser, J. Rosenblatt, D. Tseng & V. Wong. *Autonomous Cross-country Navigation with the ALV*. In IEEE International Conference on Robotics and Automation, Philadelphia (USA), pages 718-726, 1988.
- [DeCarlo 94] D. DeCarlo & D. Metaxas. *Blended Deformable Models*. In IEEE International Conference on Computer Vision and Pattern Recognition, Seattle, pages 566-572, 1994.
- [Devy 95a] M. Devy & H. Buiata. *Landmark-based vs Feature-based Localization of a mobile robot in a structured environment*. In Proceedings of the 7th International Conference on Advanced Robotics (ICAR'95), Sant Feliu de Guixols (Spain), September 1995.
- [Devy 95b] M. Devy, R. Chatila, P. Fillatreau, S. Lacroix & F. Nashashibi. *On Autonomous Navigation in a Natural Environment*. Robotics and Autonomous Systems, vol. 16, no. 10, 1995.
- [Durrant-Whyte 87] H.F. Durrant-Whyte. *Integration, Coordination and Control of Multi-Sensor Robot Systems*. Kluwer Academic Publ., Boston, MA, 1987.
- [Elfes 91] A. Elfes. *Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception*. Autonomous Mobile Robot Perception, Mapping and Navigation. S. S. Iyengar and A. Elfes Editors. IEEE Computer Society Press. Los Alamitos, vol. 1, 1991.
- [Elfes 92] A. Elfes. *Dynamic Control of Robot Perception Using Multi-Property Inference Grids*. In IEEE International Conference on Robotics and Automation, Nice, (France), 1992.
- [Faugeras 83] O.D. Faugeras & M. Hebert. *A 3-D recognition and positioning algorithm using geometrical matching between primitive surfaces*. In 8th International Joint Conference on Artificial Intelligence (IJCAI), Karlsruhe (RFA), pages 996-1002, 1983.
- [Fillatreau 94] P. Fillatreau. *Localisation et Modélisation tridimensionnelles pour un robot mobile autonome tout-terrain*. Thèse de l'Université Paul Sabatier, Toulouse (France) 94108, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), January 1994.
- [Gat 92] E. Gat. *Integrating Planning and Reacting in a Heterogeneous Asynchronous Architecture for Controlling Real-World Mobile Robots*. In 10th National Conference on Artificial Intelligence (AAAI '88), San Jose, Ca (USA), 1992.
- [Gat 94] E. Gat, R. Desai, R. Ivlev, J. Loch & D.P. Miller. *Behavior Control for Robotic Exploration of Planetary Surfaces*. IEEE Transactions on Robotics and Automation, vol. 10, no. 4, August 1994.

- [Giralt 91] G. Giralt & J.L. Lacombe. *France national developments in mobile robots for hostile or difficult work environments*. International Symposium on Advanced Robot Technology ('91 ISART), Tokyo (Japon), March 1991.
- [Giralt 92a] G. Giralt & L. Boissier. *THE FRENCH PLANETARY ROVER VAP: Concept and Current Developments*. In IEEE International Workshop on Intelligent Robots and Systems (IROS '92), Raleigh (North Carolina, USA), pages 1391–1398, July 1992.
- [Giralt 92b] G. Giralt, R. Chatila & R. Alami. *Task-level programmable intervention robots: the remote operated autonomous robot concept N.2*. Robot, vol. 4, pages pp.1–14, March 1992.
- [Giralt 93] G. Giralt, R. Chatila & R. Alami. *Remote Intervention, Robot Autonomy, And Teleprogramming: Generic Concepts And Real-World Application Cases*. In IEEE International Workshop on Intelligent Robots and Systems (IROS '93), Yokohama, (Japan), pages 314–320, July 1993.
- [Glénants 85] Les Glénants. Cours de navigation des glénants. Seuil, 1985.
- [Goldgoff 89] D. Goldgoff, T. Huang & H. Lee. *A Curvature Approach to Terrain Recognition*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 11, 1989.
- [Grandjean 91a] P. Grandjean. *Perception Multisensorielle et Interpretation de Scènes*. Thèse de l'Université Paul Sabatier, Toulouse (France) 91305, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), October 1991.
- [Grandjean 91b] P. Grandjean, M. Ghallab & E. Dekneuevel. *Multisensor Scene Interpretation: Model-Based Object Recognition*. In IEEE International Conference on Robotics and Automation, Sacramento, (USA), April 1991.
- [Grimson 85] W.E.L. Grimson & T. Lozano-Perez. *Recognition and Localization of overlapping parts from sparse data in two and three dimensions*. In IEEE International Conference on Robotics and Automation, St Louis (USA), pages 61–66, April 1985.
- [Gupta 89] A. Gupta, L. Bogoni & R. Bajcsy. *Quantitative and Qualitative Measures for the Evaluation of the Superquadric Models*. In IEEE Workshop on Interpretation of 3D Scenes, Austin (USA), pages 162–176, 1989.
- [Gupta 93] A. Gupta & R. Bajcsy. *Volumetric Segmentation of range Images of 3D Objects Using Superquadrics Models*. CVGIP: Image Understanding, vol. 58, no. 3, pages 302–326, Nov. 1993.

- [Haralick 89] R. M. Haralick & al. *Pose Estimation from Corresponding Point Data*. Machine Vision for Inspection and Measurement (H. Freeman Ed.). New York: Academic, 1989.
- [Hebert 86] M. Hebert & Al. *Outdoor Scene Analysis Using Range Data*. In IEEE International Conference on Robotics and Automation, San Francisco (USA), pages 1426,1432, April 1986.
- [Hebert 92] M. Hebert. *3-D Landmark Recognition from Range Images*. In IEEE Conference on Computer Vision and Pattern Recognition, Champaign, Illinois (USA), pages 360–365, June 1992.
- [Hébert 94] P. Hébert. *From Points to Shape Recovery: Reliable Geometric Primitive Extraction*. Thèse de doctorat, Laboratoire Vision et système numérique, Dpt. de Génie Electrique, Université Laval, Quebec, 1994.
- [Hebert 95a] M. Hebert, Ikeuchi K & H. Delinguette. *A Spherical Representation for Recognition of Free-Form Surfaces*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 10, pages 681–689, July 1995.
- [Hébert 95b] P. Hébert, S. Betgé-Brezetz & R. Chatila. *Probabilistic Map Learning: Necessity and Difficulties*. In Proc. Workshop on Reasoning on Uncertainty in Robotics, Amsterdam, 1995.
- [Hel-Or 95] Y. Hel-Or & M. Werman. *Pose Estimation by Fusing data of Different Dimensions*. IEEE Transactions on Robotics and Automation, vol. 17, no. 2, 1995.
- [Herrb 91] M. Herrb. *Vision en mouvement pour la robotique mobile*. Thèse de l'Université Paul Sabatier, Toulouse (France) 91048, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), February 1991.
- [Higuchi 94] R. Higuchi, M. Hebert & K. Ikeuchi. *Building 3-D Models from Unregistered Range Images*. In IEEE International Conference on Robotics and Automation, San Diego California, (USA), pages 2248–2253, 1994.
- [Hirata 92] S. Hirata, Y. Shirai & M. Asada. *Scene Interpretation Using 3-D Information Extracted from Monocular Color Image*. In Proc. of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems, July 1992.
- [Hotz 91] B. Hotz. *Etude de techniques de stéréovision par corrélation - Application au programme Véhicule Autonome Planétaire (V.A.P.)*. Rapport de DEA en Robotique et Vision Artificielle, Centre National d'Etudes Spatiales (Toulouse), September 1991.

- [Hu 93] H. Hu, M. Brady & P. Probert. *A Decision Theoretic Approach to Real-time Obstacle Avoidance for a Mobile Robot*. In IEEE International Workshop on Intelligent Robots and Systems (IROS '93), Yokohama, (Japan), pages 1457–1464, July 1993.
- [Jazwinski 70] A. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.
- [Jouvencel 91] B. Jouvencel & J.E. Symphor. *The variable modelling of mobile robot environments*. In IEEE International Workshop on Intelligent Robots and Systems (IROS '91), Osaka (Japan), pages 1044–1051, 1991.
- [Julier 95] S. Julier & J. Uhlmann. *A General Method for Approximating Nonlinear Transformations of Probability Distributions*. Rapport technique, Robotics Research Group, Dept. of Engineering Science, Oxford University, University of Oxford, Oxford, OX1 3PJ, United Kingdom, 1995.
- [Kheng 91] J.Y. Kheng, M. Barth & S. Tsuji. *Autonomous Landmark Selection for Route Recognition by A Mobile Robot*. In IEEE International Conference on Robotics and Automation, Sacramento, (USA), pages 2004–2009, 1991.
- [Krotkov 91] E. Krotkov, J. Bares, T. Kanade, T. Mitchell, R. Simmons & R. Whittaker. *Ambler: A Six-Legged Planetary Rover*. In '91 International Conference on Advanced Robotics (ICAR), Pisa (Italy), pages 717–722, June 1991.
- [Krotkov 94] E. Krotkov, E. Bares, J. Katragadda, L. Simmons & R. Whittaker. *Lunar Rover Technology Demonstrations with Dante and Ratler*. In Proc. Intl. Symp. Artificial Intelligence, Robotics, and Automation for Space, Jet Propulsion Laboratory, Pasadena, California, 1994.
- [Krotkov 96] E. Krotkov & R. Simmons. *Perception, Planning and Control for Autonomous Walking With the Ambler Planetary Rover*. International Journal of Robotics Research, vol. 15, no. 2, 1996.
- [Kuipers 88] B. J. Kuipers & Y. T. Byun. *A Robust, Qualitative Method for Robot Spatial Learning*. In 7th National Conference on Artificial Intelligence (AAAI '88), Saint Paul (USA), 1988.
- [Kuipers 91] B. Kuipers & Y. T. Byun. *A Robot Exploration and Mapping Strategy Based on a Semantic Hierarchy of Spatial Representations*. Journal of Robotics and Autonomous Systems, vol. 8, no. 1-2, 1991.

- [Kweon 91] I. S. Kweon. *Modeling Rugged Terrain by Mobile Robots with Multiple Sensors*. PhD thesis, Robotics Institute, Carnegie Mellon University, February 1991. Modelisation Multisensorielle d'un Terrain Accidenté.
- [Kweon 92] I. S. Kweon & T. Kanade. *High-Resolution Terrain Map from Multiple Sensor Data*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, 1992.
- [Lacombe 91] J. L. Lacombe & T. Blais. *PROGRAMME AMR DE ROBOTS MOBILES AVANCES. Solutions retenues pour le développement des modules de perception et d'interface homme-machine du démonstrateur ADAM*. In 4th International Symposium on Offshore, Robotics and Artificial Intelligence (ORIA), Marseille (France), pages 223–230. Institut International de Robotique et d'Intelligence Artificielle de Marseille, December 1991.
- [Lacroix 94] S. Lacroix, R. Chatila, S. Fleury, M. Herrb & T. Simeon. *Autonomous Navigation in Outdoor Environment: Adaptive Approach and Experiment*. In IEEE International Conference on Robotics and Automation, San Diego, California, 1994.
- [Lacroix 95a] S. Lacroix. *Stratégies de perception et de déplacement pour la navigation d'un robot mobile autonome en environnement naturel*. Thèse de l'Université Paul Sabatier, Toulouse (France), Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), avril 1995.
- [Lacroix 95b] S. Lacroix, T. Siméon & R. Chatila. *Stratégies de Déplacement et de Perception pour la Navigation*. Projets iares - sous système gda. rapport laas n. 94512, LAAS - CNRS, 1995.
- [Laurendeau 86] D. Laurendeau & D. Poussart. *3D Model Building using a Fast Range Finder*. In IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, Florida (USA), pages 424–426, 1986.
- [Lebegue 92] X. Lebegue & J. K. Aggarwal. *Extraction and Interpretation of Semantically Significant Line Segments for a Mobile Robot*. In IEEE International Conference on Robotics and Automation, Nice, (France), May 1992.
- [Leonard 90] J. Leonard, H. Durrant-Whyte & I. J. Cox. *Dynamic Map Building for an Autonomous Mobile Robot*. In IEEE International Workshop on Intelligent Robots and Systems (IROS '90), Tsuchiura (Japan), pages 89–96, 1990. Construction de Cartes d'Élévation et Estimation de la Localisation.
- [Leonard 92] J. Leonard & H. Durrant White. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academy Pub., 1992.

- [Levitt 90] T. S. Levitt & D. T. Lawton. *Qualitative Navigation for Mobile Robots*. Artificial Intelligence, vol. 44, no. 3, 1990.
- [Li 92] S. Li & S. Tsuji. *Selecting Distinctive Scene Features for Landmarks*. In IEEE International Conference on Robotics and Automation, Nice, (France), pages 53–59, 1992.
- [Matthies 88] L. Matthies & A. Elfes. *Integration of Sonar and Stereo Range Data Using a Grid-Based Representation*. In IEEE International Conference on Robotics and Automation, Philadelphia (USA), May 1988.
- [Matthies 92] L. Matthies. *Stereo vision for Planetary Rovers: Stochastic Modelling to near-real time implementation*. International Journal of Computer Vision, vol. 8, no. 1, 1992.
- [Maybeck 82] P. S. Maybeck. *Stochastic models, estimation and control*. vol. 2. Academic Press, 1982.
- [Miller 92] D. P. Miller, R. S. Desai, E. Gat, R. Ivlev & J. Loch. *Experiments With a Small Behaviour Controlled Planetary Rover*. In Misions, Technologies and Design of Planetary Mobile Vehicles, CNES, Toulouse, France, Sept. 1992.
- [Monga 90] O. Monga, R. Deriche, G. Malandin & J.P. Coquerez. *Recursive Filtering and Edge Closing: two primary tools for 3D edge detection*. In 1st European Conference on Computer Vision, Antibes (France), pages 56–65, 1990.
- [Monga 92] O. Monga, S. Benayoun & O. D. Faugeras. *From Partial Derivative of 3D Density Images to Ridge Lines*. In IEEE Conference on Computer Vision and Pattern Recognition, Champaign, Illinois (USA), pages 354–359, 1992.
- [Moutarlier 89a] P. Moutarlier & R. G. Chatila. *An Experimental System for Incremental Environment Modelling by an Autonomous Mobile Robot*. In Proc. ISER, Montreal, June 1989.
- [Moutarlier 89b] P. Moutarlier & R. G. Chatila. *Stochastic Multisensory Data Fusion for Mobile Robot Location and Environment Modelling*. In Proc. International Symposium on Robotics Research, Tokyo, 1989.
- [Moutarlier 91a] P. Moutarlier. *Modélisation Autonome de l' Environnement par un Robot Mobile*. Thèse de l'Université Paul Sabatier, Toulouse (France) 91381, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), October 1991.
- [Moutarlier 91b] P. Moutarlier & R. Chatila. *Incremental free-space modelling from uncertain data by an autonomous mobile robot*. IEEE International

- Workshop on Intelligent Robots and Systems (IROS'91), Osaka (Japon), November 1991.
- [Nashashibi 93a] F. Nashashibi. *Perception et Modélisation d'un Environnement Tridimensionnel pour la Navigation Autonome d'un Robot Mobile*. Thèse de l'Université Paul Sabatier, Toulouse (France) 93117, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), January 1993.
- [Nashashibi 93b] F. Nashashibi & M. Devy. *3D Incremental Modeling and Robot Localization in a Structured Environment using a Laser Range Finder*. In IEEE Transactions on Robotics and Automation, Atlanta (USA), May 1993.
- [Nashashibi 94] F. Nashashibi, Ph. Fillatreau, B. Dacre-Wright & T. SimÉOn. *3D Autonomous Navigation in a Natural Environment*. In Submitted to IEEE International Conference on Robotics and Automation, San Diego, California, USA, May 1994.
- [Pampagnin 90] L.H. Pampagnin. *Reconnaissance d'objets tridimensionnels en perception monoculaire et multisensorielle — Application à la robotique spatiale* —. Thèse de l'Université Paul Sabatier, Toulouse (France) 779, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), October 1990.
- [Pentland 86] A.P. Pentland. *Perceptual organization and the representation of natural form*. Artificial Intelligence, no. 28, pages 293–331, 1986.
- [Pentland 91] A. Pentland & S. Sclaroff. *Closed-Form Solutions for Physically Based Shape Modeling and Recognition*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 7, pages 715–729, July 1991.
- [Pouzyreff 91] S. Pouzyreff. *Description de données de télémétrie laser à l'aide de primitives volumiques*. Rapport de D.E.A. d'informatique, Université Paul Sabatier de Toulouse, 1991.
- [Shirai 86] Y. Shirai. *Three-dimensional computer vision*. Springer-Verlag, 1986.
- [Smith 86] R. C. Smith & P. Cheeseman. *On the Representation of Spatial Uncertainty*. International Journal of Robotics Research, vol. 5, no. 4, 1986.
- [Smith 90] R. C. Smith, M. Self & P. Cheeseman. *Estimating Uncertain Spatial Relationships in Robotics*. Autonomous Robot Vehicules, pages 167–193, 1990.
- [Solina 90] F. Solina & R. Bajcsy. *Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations*.

- IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 2, pages 131–147, 1990.
- [Solina 94] F. S. Solina, A. Leonardis & A. Macer. *A Direct Part-Level Segmentation of Range Images Using Volumetric Models*. In IEEE International Conference on Robotics and Automation, San Diego, California, pages 2254–2259, 1994.
- [Soucy 95] M. Soucy & D. Laurendeau. *A General Surface Approach to the Integration of a Set of range Views*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 4, 1995.
- [Spiegel 83] M. R. Spiegel. *Formules et tables mathématiques*. Schaum, 1983.
- [Sutherland 94] K.T. Sutherland & B. Thompson. *Localizing in Unstructured Environments: Dealing with the Errors*. IEEE Transactions on Robotics and Automation, vol. 10, no. 6, 1994.
- [Terzopoulos 91] D. Terzopoulos & D. Metaxas. *Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 7, pages 703–714, July 1991.
- [Thirion 94] J. P. Thirion. *Extremal Points: definition and application to 3D image registration*. In IEEE International Conference on Computer Vision and Pattern Recognition, Seattle, pages 587–592, 1994.
- [Thorpe 88] C. Thorpe, M. Hebert, T. Kanade & S. Shafer. *Vision and Navigation for the Carnegie-Mellon Navlab*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 3, pages 362–373, May 1988.
- [Yokoya 89] N. Yokoya & M.D. Levine. *Range Image Segmentation Based on Differential Geometry: A Hybrid Approach*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 6, pages 643–649, June 1989.
- [Zhang 92] Z. Zhang & O. Faugeras. *Three-Dimensional Motion Computation and Object Segmentation in a Long Sequence of Stereo Frames*. International Journal of Computer Vision, vol. 7, no. 3, 1992.
- [Zhang 94] Z. Zhang. *Iterative Point Matching for Registration of Free-Form Curves and Surfaces*. International Journal of Computer Vision, vol. 13, no. 2, pages 119–152, 1994.

Modélisation incrémentale et localisation par amers pour la navigation d'un robot mobile autonome en environnement naturel

Cette thèse porte sur la modélisation d'environnements naturels et la localisation d'un robot mobile autonome. L'environnement, inconnu ou partiellement connu au départ, est perçu incrémentalement par le robot au moyen d'un capteur 3D (télémètre laser ou vision stéréoscopique). Les domaines d'application de ce travail se situent dans le cadre de la robotique mobile d'intervention et de l'exploration planétaire.

Une première partie établit sous la forme d'un cahier des charges les caractéristiques que doit présenter le modèle de l'environnement et fait une analyse critique des différentes représentations étudiées dans la littérature. L'approche proposée est alors introduite et se base sur une décomposition de l'environnement en deux entités élémentaires: le sol et les objets.

La deuxième partie porte sur la modélisation d'une perception. Elle précise comment le robot segmente le sol et les objets à partir des images 3D et l'illustre par de nombreux exemples sur des données issues de scènes et de capteurs différents. Un modèle géométrique est calculé indépendamment pour chacun des objets et des relations topologiques (caractérisant leurs dispositions relatives) sont établies entre eux. Des objets particuliers, appelés amers, sont sélectionnés sur la base de critères de précision et de distinction. Le robot identifie pour chacun d'eux leur sommet qui est une caractéristique reconnaissable et en calcule la position et l'incertitude en fonction de la forme de l'amer et de la résolution du capteur.

La troisième et dernière partie traite de la modélisation incrémentale de l'ensemble de l'environnement. Après chaque perception, le robot cherche à reconnaître les amers ou une configuration d'amers dans le modèle. Les appariements trouvés lui permettent de réestimer à la fois sa position et son incertitude ainsi que celles des amers. Cette opération de localisation et de construction de la carte des amers est réalisée à l'aide du filtre de Kalman étendu. Le maintien des liens de corrélation entre les amers et le robot permet de conserver la cohérence globale de la carte.

Plusieurs expérimentations de modélisation incrémentale sont présentées et analysées méthodiquement. Les positions estimées du robot et celles des amers sont comparées à des valeurs de références, notamment obtenues grâce à un théodolite laser, et permettent de conclure sur la validité des localisations et de la carte construite.

Mots clés : Robots mobiles, Perception, Segmentation d'objets, Modélisation 3D, Localisation, Fusion de données.

Incremental Modelling and Localisation with Landmarks for Autonomous Mobile Robot Navigation in Natural Environment

This thesis deals with natural environment modelling and localisation for an autonomous mobile robot. The environment, initially unknown or partially known, is perceived incrementally by the robot with a 3D sensor (laser range finder or stereo vision). Applications of this work are intervention mobile robots and planetary exploration.

The first part establishes the specifications of the environment model and analyses the different representations of the litterature. The proposed approach is then introduced and is based on a decomposition of the environment in two entities: the ground and the objects.

The second part deals with the modelling of one perception. It precises how the robot segments the ground and the objects from range images and illustates this method with numerous examples of different scenes and sensors. A geometrical model is computed independently for each object and topological relations (characterizing their relative layout) are established between them. Specific objects, called landmarks, are selected on the basis of precision and distinction. For each one, the robot identifies its top wich is a recognisable feature and computes its position and precision according to landmark shape and sensor scanning.

The third and last part deals with incremental modelling of the whole environment. After each perception, the robot recognizes a configuration of landmarks in the model. The matches are used to compute both robot position and uncertainty and the landmark' one. These relocation and map building stages are done with an Extended Kalman Filter. The correlations between robot and objects are maintained in order to keep the coherence of the global map.

Several experiments of incremental modelling are shown and analysed methodically. The estimated robot positions and landmark' one are compared to reference values, notably obtained with a laser theodolite, and are used to conclude the validity of the relocations and of the build map.

Keywords: Mobile robots, Perception, Objet segmentation, 3D Modelling, Localisation, Data fusion.