

UNIVERSITE DE SAAD DAHLEB DE BLIDA

Faculté des Sciences de l'ingénieur
Département d'Electronique

MEMOIRE DE MAGISTER

Spécialité : Image et Parole

Contribution à la localisation dynamique du robot mobile
d'intérieur B21r en utilisant la plateforme multi sensorielle

Par

BOURAINÉ Sara

Devant le jury composé de

K. KARA	Maître de conférence, U. de Blida	Président
M. HAMERLAIN	Directeur de recherche au CDTA	Examineur
B. KAZED	Chargé de cours, U. de Blida	Examineur
K. ACHOUR	Directeur de recherche au CDTA	Rapporteur
O. DJEKOUNE	Chargé de recherche au CDTA	Co-rapporteur

Blida. Novembre 2007

RESUME

Le présent travail porte sur la localisation du robot mobile *B21r*. Ce dernier est muni de deux ceintures de 24 capteurs à ultrasons, une ceinture de 24 capteurs infrarouges, un capteur laser et une caméra CCD (N/B) dans un environnement d'intérieur. Le modèle d'environnement utilisé est les grilles d'occupation basé sur les données laser.

L'approche adoptée se base sur l'utilisation conjointe de la méthode de la mise en correspondance des grilles et le filtre de Kalman étendu (EKF).

Le Filtre de Kalman étendu fusionne les données odométriques et les données laser pour déterminer la position du robot. Il utilise la mise en correspondance des grilles pour déterminer la mesure (position du robot à partir des données laser), pour cela l'EKF lui fournit une estimation de la position et une estimation de la zone de recherche.

La mise en correspondance des grilles passe par trois étapes : L'acquisition des données laser à partir de la position courante du robot pour modéliser la grille locale, mise en correspondance de cette grille avec la grille globale définissant l'environnement et le calcul de la position et de l'orientation du robot mobile. La mise en correspondance est évaluée par des translations et des rotations d'une grille par rapport à l'autre autour d'une position estimée fournie par le filtre de Kalman.

ABSTRACT

A mobile robot self localisation method in a completely modeled indoor environment is presented. It is implemented on the *B21r* mobile robot equipped of two 24 ultrasonic sensors belts, a 24 infra-red sensors belt, a laser range finder sensor and a CCD (B/W) camera. An occupancy grid model of the environment is adopted using the laser sensor.

The adopted approach is based on the use of the matching method and the extended Kalman filter (EKF) together.

The extended Kalman filter merges odometric data and laser data to estimate the robot's position. The EKF use the matching method to determine the measure (robot's position based on laser data). This is why EKF provides a position estimation and research zone estimation.

The matching approach is composed of three steps: acquisition of information from robot current position and orientation thanks to the measures of the laser sensor to model the local grid, matching the local grid and the global grid (representing environment) and the determination of the robot's position and orientation in its environment. The correspondence is evaluated by translations and rotations between local grid and global grid surrounded of estimation position provided by the extended Kalman filter.

ملخص

هذا العمل يتعلق بتحديد المكان للجهاز اللالي المتحرك *B21r* المجهز بحزامين من 24 لاقط صوتي , حزام من 24 لاقط للاشعة تحت الحمراء , لاقط ليزر و آلة تصوير CCD (اسود/ابيض) في مكان مغلق. مكان اشغال الجهاز الالي ممثل بواسطة شبكة متكونة من خلايا مشغولة و اخرى فارغة باستعمال اللاقط الليزر. الطريقة المستعملة تتركز على استعمال طريقة التطابق بين الشبكات و المرشح كالمن الممدد معا. مرشح كالمن الممدد يدمج بين المعطيات الناجمة عن عداد المسافة و المعطيات الناجمة عن اللاقط الليزر لتحديد مكان الجهاز الالي. فهو يستعمل طريقة التطابق بين الشبكات لتحديد القياس (مكان الجهاز الالي باستعمال معطيات الليزر), لذا فمرشح كالمن الممدد يقدم لهذه الطريقة تقدير عن المكان و تقدير عن منطقة البحث. التطابق بين الشبكات يمر بثلاث مراحل : استقبال نظرة المحيط عن طريق اللاقط الليزر في الوضعية الحالية للجهاز الالي , التطابق بين شبكة مكان الاشتغال و الشبكة المكونة بواسطة اللاقط الليزر و حساب مكان و اتجاه الجهاز الالي. التطابق مقدر بانسحابات و دوران شبكة بالنسبة لشبكة اخرى حول الوضعية التقريبية المقدمة من مرشح كالمن الممدد.

REMERCIEMENTS

Tout d'abord je voudrais exprimer ma reconnaissance à Monsieur K. Kara pour avoir accepté la présidence du jury de ma thèse.

Je remercie également l'ensemble des membres du jury : Monsieur B. Kazed et Monsieur M. Hamerlain pour avoir accepté d'examiner ce manuscrit et pour m'avoir fait l'honneur d'évaluer mon travail.

J'adresse mes remerciements à mon rapporteur de thèse Monsieur K. Achour pour m'avoir encadré et accueilli au sein de son équipe au CDTA.

J'exprime toute ma gratitude à mon co-rapporteur Monsieur O. Djekoune pour avoir orienté mon travail à travers ses remarques très pertinentes, pour m'avoir offert de nombreuses occasions d'approfondir mes connaissances en robotique et pour m'avoir fait confiance.

Je tient à remercier Ryadh pour son aide et ses encouragements, Zahia et Amina pour leur oreille attentive et Nacera qui ma aidé à y voir plus clair à travers sa documentation.

Enfin mes dernières pensées vont à l'ensemble des personnes de ma famille qui m'ont toujours encouragés à poursuivre cette voie et soutenus dans les moments difficiles, ainsi que tous mes amis.

TABLE DES MATIERES

RESUME	
REMERCIEMENTS	
TABLE DES MATIERES	
LISTES DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX	
INTRODUCTION.....	12
1. LES ROBOTS MOBILES.....	16
1.1 Introduction.....	16
1.2 Roulement sans glissement.....	17
1.3 Les grandes classes de robots mobiles et leurs modèles.....	19
1.3.1 Disposition des roues et centre instantané de rotation.....	19
1.3.2 Robots mobiles différentiels.....	20
1.3.3 Robots mobiles de type tricycle.....	23
1.3.4 Robots mobiles de type voiture.....	25
1.3.5 Robots mobiles omnidirectionnels.....	26
1.3.6 Robots mobiles à traction synchrone.....	28
1.3.6.1 Structure des roues du robot mobile <i>B21r</i>	29
1.3.6.2 Centre instantané de rotation.....	30
1.3.6.3 La cinématique du robot mobile <i>B21r</i>	31
1.4 Holonomie et non holonomie.....	33
1.5 Conclusion.....	35
2. PERCEPTION.....	36
2.1 Introduction.....	36
2.2 Capteurs de perception.....	36
2.2.1 Les capteurs proprioceptifs.....	37
2.2.1.1 Les capteurs de déplacement.....	37
2.2.1.1.1 Les odomètres.....	37
2.2.1.1.2 Les accéléromètres.....	39
2.2.1.1.3 le radar Doppler.....	39
2.2.1.2 Les capteurs d'attitude.....	40
2.2.1.2.1 Le gyroscope, gyromètre et gyrocompas.....	40
2.2.1.2.2 Les magnétomètres ou les compas magnétiques.....	41
2.2.2 Les capteurs extéroceptifs.....	42
2.2.2.1 Les capteurs télémétriques.....	42
2.2.2.1.1 Les capteurs à ultrasons.....	42
2.2.2.1.2 Les capteurs laser.....	44
2.2.2.1.3 Les capteurs infrarouges.....	45
2.2.2.1.4 Le GPS.....	46
2.2.2.2 Les systèmes de vision.....	47

2.2.2.2.1 Caméras simples.....	47
2.2.2.2.2 Caméras stéréoscopiques.....	48
2.2.2.2.3 Caméras panoramiques.....	48
2.2.2.3 Autres capteurs.....	50
2.2.2.3.1 Les capteurs tactiles.....	50
2.2.2.3.2 Les balises.....	50
2.3 Modélisation des mesures.....	50
2.3.1 Grille d'occupation.....	50
2.3.2 Modèle géométrique.....	54
2.3.2.1 Par télémètre.....	54
2.3.2.2 Par vision.....	55
2.4 Modélisation d'environnement.....	55
2.4.1 Les cartes topologiques.....	55
2.4.2 Les cartes métriques.....	57
2.4.2.3 Grille d'occupation.....	58
2.4.2.4 Modèle géométrique.....	61
2.5 Conclusion.....	63
3. LOCALISATION.....	64
3.1 Introduction.....	64
3.2 Les méthodes de localisation.....	65
3.2.1 Localisation relative.....	65
3.2.2 Localisation absolue.....	66
3.2.2.1 Localisations par balises.....	67
3.2.2.1.1 Différents types de balises.....	67
3.2.2.1.1.1 Les repères artificiels.....	67
3.2.2.1.1.2 Les repères naturels.....	69
3.2.2.1.2 Méthodes de calcul de la position du robot.....	70
3.2.2.1.2.1 La trilatération (ou multilatération).....	70
3.2.2.1.2.2 La triangulation.....	71
3.2.2.2 Mise en correspondance de modèle.....	72
3.2.2.2 .1 Les cartes géométriques.....	73
3.2.2.2 .2 Les grilles.....	73
3.2.3 Localisation hybride.....	74
3.2.3.1 Méthode de Suivi d'une hypothèse unique.....	76
3.2.3.1.1 Le filtre de Kalman.....	76
3.2.3.1.2 Le filtre de Kalman étendu (Extended Kalman Filter ou EKF).....	83
3.2.3.1.3 Le filtre de Kalman inodore (Unscented Kalman Filter ou UKF).....	85
3.2.3.1.3.1 La transformé inodore (unscented transform).....	85
3.2.3.1.3.2 Le filtre de Kalman avec la transformée inodore.....	87
3.2.3.2 Suivi de plusieurs hypothèses.....	92
3.3 Conclusion.....	96
4. ALGORITHME DE LOCALISATION DEVELOPPE.....	97
4.1 Introduction.....	97
4.2 Le robot mobile <i>B21r</i>	98
4.2.1 Architecture matérielle.....	98
4.2.2 Architecture logicielle.....	100

4.2.3 Comportement du <i>B21r</i>	101
4.3 Modélisation de l'environnement.....	102
4.3.1 Modélisation de l'environnement basée sur les capteurs à ultrasons.....	102
4.3.1.1 Le Capteur à ultrasons.....	102
4.3.1.2 Modélisation des mesures ultrasonores	104
4.3.1.3 Création de la grille.....	108
4.3.2 Le capteur infrarouge.....	111
4.3.3 Modélisation de l'environnement basée sur le capteur laser.....	112
4.3.3.1 Le capteur laser.....	112
4.3.3.2 Modélisation des mesures laser.....	112
4.3.3.3 Création de la grille.....	113
4.4 Méthodes de localisation appliquées au <i>B21r</i>	115
4.4.1 Localisation par la mise en correspondance des grilles.....	115
4.4.1.1 Définition de la grille locale.....	115
4.4.1.2 Définition de la grille globale.....	115
4.4.1.3 Principe.....	116
4.4.1.4 Critère de mise en correspondance.....	117
4.4.1.5 Zone de recherche.....	119
4.4.1.6 L'algorithme.....	120
4.4.1.7 Résultat.....	121
4.4.2 La localisation par le filtre de Kalman étendu.....	126
4.4.2.1 Formulation mathématique.....	126
4.4.2.2 Le modèle d'évolution.....	126
4.4.2.3 Le modèle de mesure.....	132
4.4.2.4 Le modèle d'incertitude.....	133
4.4.2.5 L'algorithme de localisation basé sur le filtre de Kalman étendu.....	133
4.4.2.5.1 Prédiction de la position du robot	134
4.4.2.5.2 La mesure.....	135
4.4.2.5.3 Estimation final de la position.....	138
4.4.2.6 Résultats obtenus.....	139
4.5 Conclusion.....	147
CONCLUSION.....	148
.	150
APPENDICE.....	153
REFERENCES.....	

LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX

Figure 1.1 :	Roulement sans glissement	18
Figure 1.2 :	Les principaux types de roues pour robots mobiles	19
Figure 1.3 :	Robot mobile différentiel	20
Figure 1.4 :	Exemple de robots mobiles différentiels	21
Figure 1.5 :	Centre instantané de rotation d'un robot mobile différentiel	22
Figure 1.6 :	Robot mobile de type tricycle	23
Figure 1.7 :	Exemple de robots mobiles de type tricycle	23
Figure 1.8 :	Centre instantané de rotation du robot mobile de type tricycle	24
Figure 1.9 :	Exemple de robots mobiles de type voiture	25
Figure 1.10 :	Robot mobile de type voiture et sont <i>CIR</i>	26
Figure 1.11 :	Robot mobile omnidirectionnel	27
Figure 1.12 :	Exemple de robots mobiles omnidirectionnels	27
Figure 1.13 :	Représentation 3D d'un essieu	28
Figure 1.14 :	Disposition des 3 essieux	28
Figure 1.15 :	L'architecture d'une plate forme pour robot mobile à traction synchrone	28
Figure 1.16 :	Robot mobile <i>B21r</i> à traction synchrone	29
Figure 1.17 :	Roue décentrée orientable	30
Figure 1.18 :	Position du centre instantané de rotation dans le cas d'un virage à gauche	31
Figure 1.19 :	Disposition des quatre roues du robot mobile <i>B21r</i> .	32
Figure 1.20 :	Le modèle cinématique utilisé pour le robot mobile <i>B21r</i>	33
Figure 2.1 :	Principe du capteur à effet Doppler	40
Figure 2.2 :	Exemples de Gyroscope et gyromètre	41
Figure 2.3 :	Exemples de télémètres ultrasonores	43
Figure 2.4 :	Principe du télémètre à ultrasons et un exemple de télémètre réel	44
Figure 2.5 :	Illustration d'un télémètre laser et un exemple de télémètre laser SICK à balayage fournissant 180 mesures réparties sur 180 degrés.	45
Figure 2.6 :	Télémètre infrarouge Sharp	46
Figure 2.7 :	Un exemple de banc stéréoscopique	48
Figure 2.8 :	Principe des caméras panoramiques	49
Figure 2.9 :	Construction d'une carte à fusion de données de plusieurs capteurs [d'après 27]	52
Figure 2.10 :	Création d'un RCD (Region of Constant Depth)	54
Figure 2.11 :	Un exemple de la modélisation topologique [d'après 34]	55
Figure 2.12 :	Un exemple de la modélisation par grille d'occupation [d'après 34]	59
Figure 2.13 :	Quatre étapes montrant comment l'environnement est graduellement exploré au cours du déplacement du robot.	60
Figure 2.14 :	Un exemple de la modélisation géométrique [d'après 34]	61
Figure 3.1 :	Localisation par mesure de distances de trois balises	71

Figure 3.2 :	Localisation par triangulation	72
Figure 3.3 :	Cycle de rétroaction du filtre de Kalman	77
Figure 3.4 :	Le cycle complet du filtre de Kalman avec le équations de prédiction et de mesure et correction du filtre de Kalman	83
Figure 3.5 :	L'algorithme du filtre de Kalman étendue	85
Figure 3.6 :	Cycle de prédiction-correction du filtre de Kalman inodore (UKF).	90
Figure 3.7 :	Illustration de la méthode de suivi de plusieurs hypothèses	93
Figure 3.8 :	L'idée basic de la localisation de Markov	94
Figure 3.9 :	Illustration du fonctionnement de l'algorithme de filtrage particulière	95
Figure 4.1 :	la géométrie du robot mobile B21r.	98
Figure 4.2 :	Console du Robot <i>B21r</i>	100
Figure 4.3 :	Quatre différents types de mouvement réalisé par le robot mobile B21r.	101
Figure 4.4 :	Deux obstacles renvoyant la même mesure.	102
Figure 4.5 :	Phénomène de spécularité.	103
Figure 4.6 :	Exemple de réflexions multiples.	104
Figure 4.7 :	Exemples de diaphonie.	104
Figure 4.8 :	Fonction de probabilité d'un capteur idéal.	105
Figure 4.9 :	Grille pour capteur idéal.	105
Figure 4.10 :	Grille pour un capteur idéal prenant en compte l'espace libre.	106
Figure 4.11 :	Exemple d'obstacle mal perçu.	106
Figure 4.12 :	Grille prenant en compte l'ouverture angulaire du capteur.	107
Figure 4.13 :	Modélisation du capteur à ultrasons.	108
Figure 4.14 :	Configuration du cône d'émission d'un capteur à ultrasons et de sa position par rapport au robot mobile.	109
Figure 4.15 :	Cellules correspondantes à une mesure d'un capteur à ultrasons.	110
Figure 4.16 :	Création de la carte de l'environnement utilisant les deux ceintures à ultrasons.	111
Figure 4.17 :	Modélisation du capteur laser	113
Figure 4.18 :	Configuration de la position d'un obstacle P par rapport au repère mobile R^R .	113
Figure 4.19 :	Création de la carte d'environnement utilisant le capteur laser.	114
Figure 4.20 :	Création d'une carte locale basée sur le capteur laser.	115
Figure 4.21 :	Reconstruction de la carte globale en utilisant le capteur laser.	116
Figure 4.22 :	Mise en correspondance des grilles.	117
Figure 4.23 :	Signe du produit des cellules.	117
Figure 4.24 :	Mise en correspondance des grilles.	118
Figure 4.25 :	Exemple de zone de recherche pour une position donnée.	119
Figure 4.26 :	Algorithme de mise en correspondance	120
Figure 4.27 :	Résultat de la localisation par mise en correspondance des grilles d'occupation pour une trajectoire rectiligne.	122
Figure 4.28 :	Résultat de la localisation par mise en correspondance des grilles d'occupation pour une trajectoire courbe.	123
Figure 4.29 :	Résultats de la localisation par la mise en correspondance des grilles pour trois dimensions différentes de la zone de recherche.	124

Figure 4.30 :	Le filtre de Kalman étendu représenter comme une configuration de rétroaction	126
Figure 4.31 :	Position du robot dans le repère univers R^u en fonction des données odométriques.	127
Figure 4.32 :	Déplacement circulaire du robot entre deux instants d'échantillonnage	129
Figure 4.33 :	Algorithme de localisation.	134
Figure 4.34 :	Algorithme de mise en correspondance des grilles pour déterminer la mesure.	137
Figure 4.35 :	Résultat de la localisation par le filtre de Kalman étendu (EKF) pour une trajectoire rectiligne.	140
Figure 4.36 :	Résultat de la localisation par le filtre de Kalman étendu (EKF) pour une trajectoire courbe.	140
Figure 4.37 :	Résultat de la localisation par le filtre de Kalman étendu (EKF) pour différentes trajectoires.	142
Figure 4.38 :	Comparaison entre les positions fournies par les odomètres, l'EKF et la mesure (trajectoire a)	143
Figure 4.39 :	Comparaison entre les positions fournies par les odomètres, l'EKF et la mesure (trajectoire b)	144
Figure 4.40 :	comparaison entre EKF et la mise en correspondance avec des zones de recherche différentes.	146

INTRODUCTION

Les robots mobiles constituent aujourd'hui un formidable potentiel technologique, où on pourrait même parler d'une prochaine révolution électronique, qui verrait les robots mobiles déferler dans notre quotidien. Ils sont déjà arrivés dans les foyers via les applications de divertissements ou pour soulager les hommes des diverses tâches domestiques, ils commencent même à investir les espaces publics et les lieux de travail. Ils permettent d'atteindre des lieux inaccessibles pour l'homme, dans tous les milieux : les zones confinées, les constructions élevées, les fonds sous-marins, les environnements dangereux ou les autres planètes. On parle alors de façon générale de robotique en milieu d'intérieur ou en milieu d'extérieur.

Un tel cadre d'utilisation requiert que le système robotisé dispose d'un niveau minimum d'autonomie et de facilités de navigation. Pour ce faire, le système doit accomplir trois tâches de base qui sont la localisation, la planification et la navigation. Parmi ces tâches, la localisation relativement à l'environnement occupe une place de choix puisqu'elle détermine le bon déroulement des deux autres. Elle consiste à calculer et à maintenir à jour la connaissance de la position et de l'orientation du robot dans un repère absolu lié à l'environnement de travail. On se restreint ici au cas de système navigant sur le plan. Le robot est alors complètement localisé par deux paramètres de position et un paramètre d'orientation.

Pour se localiser, un système robotisé dispose de capteurs qui peuvent être utilisés pour calculer sa position ou pour percevoir l'environnement dans lequel il évolue, cela permet au robot de cartographier par lui-même son environnement au lieu d'utiliser une carte construite manuellement, ce qui offre des avantages considérables : possibilité de mises à jour fréquentes, adaptation de la représentation employée aux algorithmes du robot, modélisation de zones difficiles à atteindre pour l'homme voire dangereuses ou hostiles, etc.

Le modèle de l'environnement peut se présenter sous diverses formes comme les grilles d'occupation, les modèles à primitives géométriques ou encore les modèles topologiques. Ces modèles peuvent être utilisés pour la localisation du robot.

Il existe principalement trois méthodes de localisation ; la localisation relative, la localisation absolue et la localisation hybride qui combine les méthodes relatives et absolues.

La localisation relative (généralement odométrique) permet de maintenir à jour la connaissance de la position et de l'orientation avec un très haut niveau d'échantillonnage mais sa précision se détériore au fil du déplacement à cause de l'accumulation d'erreurs. Contrairement à ces méthodes, les méthodes de localisation absolue sont connues pour leur précision. Parmi les différentes approches existantes, une technique très répandue est celle basée sur un modèle (localisation par la mise en correspondance). En effet la localisation dans un environnement intérieur utilise le plus souvent une carte, ou modèle de l'environnement. Le principe général est que le robot perçoit en ligne et grâce à ses capteurs extéroceptifs un modèle local de son espace de travail qu'il met en rapport avec le modèle global puis estime la position et l'orientation courantes à partir desquelles il a acquis ces données. La plupart des systèmes adoptent une approche qui consiste à combiner cette méthode avec une localisation relative, ce qui permet de lui fournir une estimation de la position et de l'orientation pour réduire la complexité et le temps de calcul. On parle alors de localisation incrémentale.

C'est dans ce contexte que se situe notre travail, notre objectif est de développer un algorithme de localisation fiable qui permet de déterminer la position réelle du robot dans son environnement et donc de gérer les erreurs de perception dues aux capteurs odométriques.

Ce travail est réalisé au sein du *Centre de Développement des Techniques Avancées* (CDTA), et plus particulièrement au sein de l'équipe Vision Artificielle et ANalyse d'IMage (VAANIM). Où notre algorithme sera implémenté sur le robot mobile *B21r* de type à traction synchrone de la famille *iRobot*, muni de deux ceintures de télémètres à ultrasons, une ceinture de télémètres infrarouges, un télémètre laser et une caméra CCD (N/B).

La localisation avec la construction d'une représentation de l'espace sous forme d'une grille d'occupation à l'aide des capteurs embarqués sur le robot mobile apparaît comme un choix idéal pour atteindre notre objectif de modélisation en temps réel d'un environnement dynamique. Où l'on va adopter la méthode de la mise en correspondance des grilles [1] qui consiste à modéliser les informations acquises par le système de perception du robot. Ce modèle va être mis en œuvre pour la

reconstruction de l'environnement dans sa totalité en premier lieu puis la partie perçue par le robot pendant son déplacement. Par correspondance du modèle perçu et l'environnement, on détermine la position et l'orientation qui correspond au meilleur appariement des deux modèles.

La problématique de la méthode de la mise en correspondance des grilles est qu'elle souffre fortement du grand coût calculatoire malgré sa précision, une précision qui nécessite l'emploi de capteur assez précis, ce qui rend contraignant l'utilisation de ce système pour des applications en temps réel et dans de grands espaces, ce qui est dû en partie à la dimension de sa zone de recherche.

L'originalité de notre travail réside dans l'utilisation conjointe du filtre de Kalman étendu [2-5] et la mise en correspondance des grilles pour la localisation du robot mobile *B21r* dans un environnement d'intérieur et cela pour remédier aux problèmes de la mise en correspondance d'une part et les erreurs cumulatives du système odométrique d'autre part et puis pour bénéficier des avantages de chacune d'entre elles. L'EKF optimise la méthode de la mise en correspondance en réduisant sa zone de recherche (autour d'une position qu'il a estimé), et utilise la position calculée à partir des mesures laser avec celles fournies par le système odométrique pour estimer la position du robot mobile.

Ce mémoire se compose de quatre chapitres organisés comme suit :

Le chapitre 1 est une introduction à la robotique mobile, où on a passé en revue les différents types de robots mobiles à roues et leurs contraintes cinématiques, y compris notre robot mobile *B21r*.

Dans le chapitre 2 nous introduisons les différents capteurs utilisés dans la localisation des robots mobiles, et les approches qui traitent de la modélisation des mesures issues de ses capteurs.

Le chapitre 3 présente un état de l'art des méthodes de localisation existantes et tire profit des avantages et inconvénients de chaque méthode pour déterminer celle la plus appropriée pour notre cas. Selon les conclusions déduites dans le chapitre 2 et le chapitre 3, nous déterminons le modèle à utiliser et les méthodes de localisation à employer pour déterminer la position et l'orientation du robot.

Au début du chapitre 4, nous décrivons l'architecture matérielle et logicielle du robot mobile *B21r* qui va nous servir de plateforme expérimentale pour l'implémentation de nos algorithmes. Puis nous décrivons les capteurs embarqués sur ce robot et leurs modèles. Dans un deuxième temps, nous nous focalisons sur la

localisation par la mise en correspondance des grilles, et nous fournissons les résultats de l'implémentation sur le robot *B21r*. Vu les contraintes de cette méthode, on propose d'utiliser le filtre de Kalman étendu (EKF). Une étude détaillée et les résultats expérimentaux de cette méthode y sont présentés.

Nous terminons notre travail par une conclusion générale récapitulant ce qui a été fait et exposant les perspectives de notre travail.

CHAPITRE 1 LES ROBOTS MOBILES

1.1 Introduction

En réalité un robot est une machine capable d'agir sur son environnement, de réaliser des tâches diverses, et qui est douée de la capacité de s'auto adapter d'elle-même à une situation nouvelle. Il est équipé de capteurs lui permettant de percevoir l'environnement dans lequel il évolue, et d'effecteurs (roue, bras, pince, jambe, etc.) lui permettant d'agir sur cet environnement. On distingue deux groupes de robots, les robots fixes désignant les robots manipulateurs, et les robots mobiles ayant la tendance de se déplacer, ces derniers peuvent être équipés ou non de manipulateurs selon l'utilisation.

De manière générale, on regroupe sous l'appellation robots mobiles l'ensemble des robots à base mobile, par opposition notamment aux robots manipulateurs. L'usage veut néanmoins que l'on désigne le plus souvent par ce terme les robots mobiles à roues. Les autres robots mobiles sont en effet le plus souvent désignés par leur type de locomotion, qu'ils soient marcheurs, sous-marins ou aériens.

On peut estimer que les robots mobiles à roues constituent la grande partie des robots mobiles. Historiquement, leur étude est venue assez tôt, suivant celle des robots manipulateurs. Leur faible complexité en a fait de bons premiers sujets d'étude pour les roboticiens intéressés par les systèmes autonomes. Cependant, malgré leur simplicité apparente, ces systèmes ont soulevés un grand nombre de problèmes difficiles, nombreux de ceux-ci ne sont d'ailleurs toujours pas résolus. Néanmoins, l'intérêt indéniable de la robotique mobile est d'avoir permis d'augmenter considérablement nos connaissances sur la localisation et la navigation de systèmes autonomes.

Ce chapitre a d'abord pour objectif de définir le principe de roulement sans glissement, puis de présenter les différents types de robots mobiles à roues à savoir les plus utilisés en robotique mobile, ainsi que leurs caractéristiques cinématiques. Comme cas particulier, l'étude cinématique de notre plateforme le robot mobile *B21r*

est accomplie. Enfin, ce chapitre introduit aussi la notion de l'holonomie et la non holonomie.

1.2 Roulement sans glissement

La locomotion à l'aide de roues exploite la friction au contact entre roue et sol. Pour cela, la nature du contact (régularité, matériaux en contact) a une forte influence sur les propriétés du mouvement relatif de la roue par rapport au sol. Dans de bonnes conditions, il y a roulement sans glissement (r.s.g.) de la roue sur le sol, c'est-à-dire que la vitesse relative de la roue par rapport au sol au point de contact est nulle. Théoriquement, pour vérifier cette condition, il faut réunir les hypothèses suivantes :

- le contact entre la roue et le sol est ponctuel.
- les roues sont indéformables, de rayon r .

En pratique le contact se fait sur une surface, ce qui engendre bien évidemment de légers glissements. De même, alors qu'il est raisonnable de dire que des roues pleines sont indéformables, cette hypothèse est largement fautive avec des roues équipées de pneus. Malgré cela, on supposera toujours qu'il y a r.s.g. et, par ailleurs, que le sol est parfaitement plan.

Mathématiquement, pour exprimer la condition de r.s.g sur une roue. On considère une roue verticale qui roule sans glisser sur un sol plat (figure 1.1), dans un repère $R^u (O, X, Y, K)$. Le repère $R_1 (O_1, X_1, Y_1, K_1)$ est lié à la roue et le roulement sans glissement se traduit par la vitesse nulle au point I de la roue en contact avec le sol. Une vitesse qui sera exprimée en fonction de la vitesse du centre O_1 de la roue et de la vitesse de rotation ω de la roue, et cela en se basant sur une méthode de description de la vitesse d'un solide dans l'espace fondée sur l'utilisation du torseur cinématique [6]. L'application qui à tout point P fait correspondre un vecteur v_P appelé champ de vecteur. Si ce vecteur est une vitesse on parlera de champ des vitesses, le champ de vitesse est antisymétrique ce qui caractérise un torseur appelé torseur cinématique en P , où v_P est le moment résultant en P , représentant la vitesse absolue de l'origine par rapport au repère R^u tel que :

$$v_P = \frac{d}{dt}(OP) \quad (1)$$

Ω étant la résultante du torseur représentant le vecteur de rotation du corps par rapport à R^u , la connaissance de ces deux éléments permet de calculer la vitesse d'un point A par la relation fondamentale suivante :

$$v_P = v_A + \Omega \wedge \overrightarrow{AP} \quad (2)$$

D'où :

$$\begin{aligned} \vec{V}(I/R) = \vec{0} &= \vec{V}(O_1/R) + \vec{\omega}_{(I/O)} \wedge \overrightarrow{O_1I} \\ &= x_{re} \dot{\vec{X}} + y_{re} \dot{\vec{Y}} + (\dot{\varphi} \vec{K} + \theta(-\sin \varphi \dot{\vec{X}} + \cos \varphi \dot{\vec{Y}})) \wedge (-r \vec{K}) \\ &= (x_{re} \dot{\vec{X}} - r \dot{\theta} \cos \varphi \vec{K}) + (y_{re} \dot{\vec{Y}} - r \dot{\theta} \sin \varphi \vec{K}) \end{aligned}$$

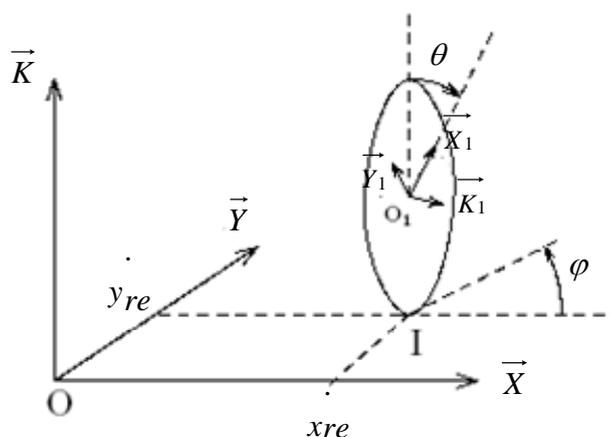


Figure 1.1 : Roulement sans glissement.

Avec r le rayon de la roue et (x, y) les coordonnées du point O_1 dans R , on en déduit donc deux contraintes :

$$x_{re} \dot{\vec{X}} - r \dot{\theta} \cos \varphi = 0 \quad (3)$$

$$y_{re} \dot{\vec{Y}} - r \dot{\theta} \sin \varphi = 0 \quad (4)$$

Que l'on peut transformer pour faire apparaître les composantes de vitesse dans le plan de la roue d'une part et perpendiculairement à la roue d'autre part :

$$-x_{re} \dot{\vec{X}} \sin \varphi + y_{re} \dot{\vec{Y}} \cos \varphi = 0 \quad (5)$$

$$x_{re} \dot{\vec{X}} \cos \varphi + y_{re} \dot{\vec{Y}} \sin \varphi = -r \dot{\theta} \quad (6)$$

Ce système d'équation traduit les deux propriétés suivantes :

La première est que la vitesse du centre de la roue est parallèle au plan de la roue,

et la deuxième est que la vitesse du centre de la roue est $r \dot{\theta}$.

1.3 Les grandes classes de robots mobiles et leurs modèles

1.3.1 Disposition des roues et centre instantané de rotation

C'est la combinaison du choix des roues et de leur disposition qui confère à un robot son mode de locomotion. On rencontre principalement quatre types de roues (voir figure 1.2) :

- les roues fixes dont l'axe de rotation passe par le centre de la roue, tandis que l'axe d'orientation est constant (figure 1.2 (a)).
- les roues centrées orientables dont l'axe d'orientation passe par le centre de la roue. (figure 1.2 (b)).
- les roues décentrées orientables souvent appelées roues folles, pour lesquelles l'axe d'orientation ne passe pas par le centre de la roue (figure 1.2 (c)).
- Les roues suédoises dont la bande de roulement a été remplacée par des galets inclinés par rapport à la normale au plan de la roue. C'est la combinaison de la rotation de la roue avec la rotation libre du galet en contact avec le sol qui permet un déplacement sans glissement dans toutes les directions (figure 1.2 (d)).

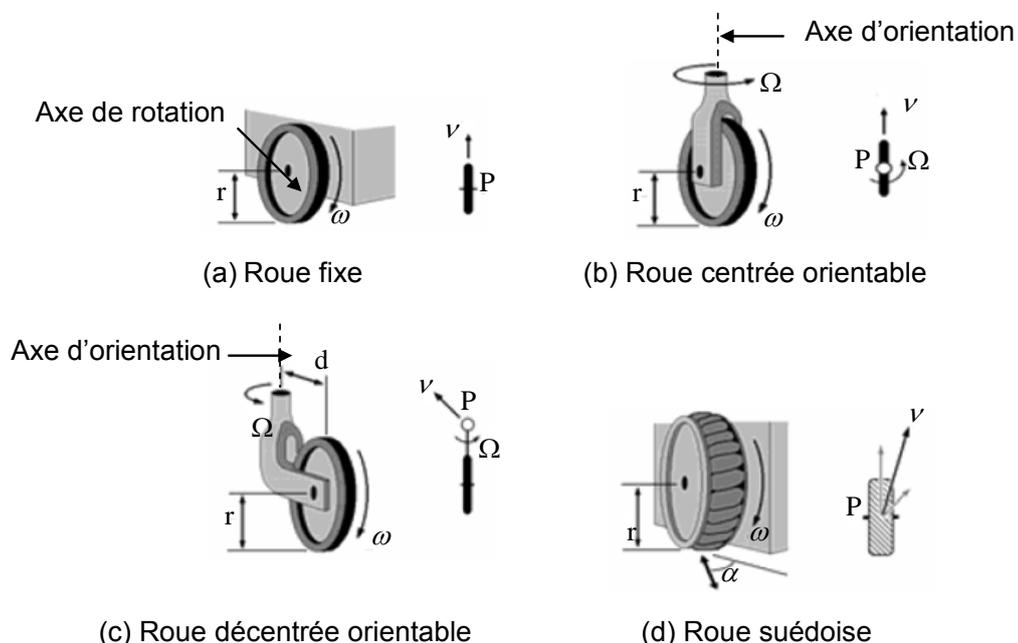


Figure 1.2 : Les principaux types de roues pour robots mobiles

Ces quatre types de roues sont les plus utilisés en robotique mobile et n'empêche qu'il en existe d'autres tel que les roues sphériques connues pour leur propriété

omnidirectionnelle. Comme celles qui seront utilisées dans la plateforme omnidirectionnelle de la figure 1.14.

Bien évidemment, pour un ensemble de roues données, toute disposition ne conduit pas à une solution viable. Un mauvais choix peut limiter la mobilité du robot ou occasionner d'éventuels blocages. Par exemple, un robot équipé de deux roues fixes non parallèles ne pourrait pas aller en ligne droite. Pour qu'une disposition de roues soit viable et n'entraîne pas de glissement des roues sur le sol, il faut qu'il existe pour toutes ces roues un unique point de vitesse nulle autour duquel tourne le robot, en instantanée. Ce point, lorsqu'il existe, est appelé centre instantané de rotation (*CIR*). Les points de vitesse nulle liés aux roues se trouvant sur leur axe de rotation, il est donc nécessaire que le point d'intersection des axes de rotation des différentes roues soit unique. Pour cette raison, il existe en pratique cinq principales catégories de robots mobiles à roues, que l'on va présenter maintenant.

1.3.2 Robots mobiles différentiels

Description

Une des configurations les plus utilisées pour les robots mobiles est la configuration différentielle (differential drive) qui comporte deux roues fixes non orientables commandées indépendamment, une ou plusieurs roues folles sont ajoutées à l'avant ou à l'arrière du robot pour assurer sa stabilité, n'empêche qu'il existe certains robots différentiels avec quatre roues commandées indépendamment sauf qu'ils sont modélisés en deux roues. Le schéma du robot différentiel est présenté à la figure 1.3. On y a omis les roues folles, parce qu'ils n'interviennent pas dans la cinématique, mais ils assurent juste l'équilibre.

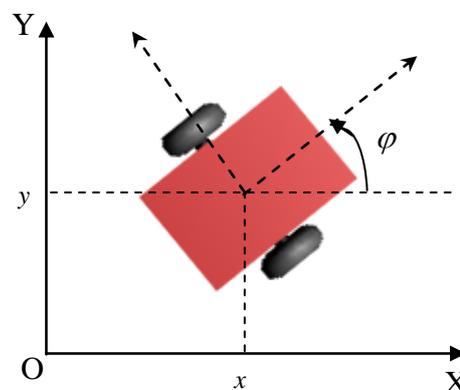


Figure 1.3 Robot mobile différentiel

Ce type de robot est très répandu en raison de sa simplicité de construction et de ses propriétés cinématiques intéressantes, comme sa capacité à tourner sur lui-même. La figure 1.4 présente un exemple de robots mobiles différentiels.



Figure 1.4 : Exemple de robots mobiles différentiels :
(a) Robot mobile Pioneer P3-DX, (b) Robot mobile ATRV2 du CDTA.

Modélisation

Centre instantané de rotation les roues motrices ayant même axe de rotation, le *CIR* du robot est un point de cet axe. Soit R le rayon de courbure de la trajectoire du robot, c'est-à-dire la distance du *CIR* au point O' (figure 1.5). Soit $2L$ la distance qui sépare les deux roues, et Ω la vitesse angulaire du robot par rapport au *CIR*. Alors les vitesses des roues droite et gauche, respectivement notées v_d et v_g vérifient :

$$v_d = (R+L)\Omega \quad (7)$$

$$v_g = (R-L)\Omega \quad (8)$$

Ce qui permet de déterminer R et Ω à partir des vitesses des roues :

$$\Omega = \frac{v_d - v_g}{2L} \quad (9)$$

$$R = L \frac{v_d + v_g}{v_d - v_g} \quad (10)$$

La vitesse linéaire v du robot au point O' est :

$$v = \frac{v_d + v_g}{2} \quad (11)$$

La vitesse de rotation du robot est égale à la vitesse de rotation autour du *CIR* :

$$\Omega = \dot{\varphi} = \frac{v_d - v_g}{2L} \quad (12)$$

L'équation (10) permet de situer le *CIR* sur l'axe des roues. Par ailleurs ces équations expliquent deux propriétés particulières du mouvement des robots différentiels : si $v_d = v_g$, la vitesse angulaire Ω sera nulle et le rayon de courbure R

est infinie donc le robot se déplace en ligne droite ; si $v_d = -v_g$, $\Omega \neq 0$ et R est nulle alors le robot effectue une rotation sur lui-même. Cependant dans le cas où $v_d \neq v_g$ le déplacement du robot est un virage à gauche ou à droite selon que v_d soit supérieur ou inférieur à v_g (dans une direction qui correspond à la vitesse inférieure). L'utilisation de ce mode de locomotion, fournit une solution simple pour amener le robot d'une position à l'autre. C'est sans doute là une des raisons du succès de ce type de robot.

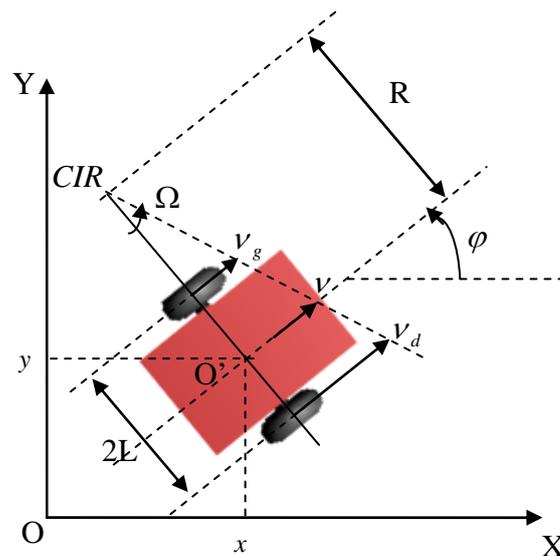


Figure 1.5 : centre instantané de rotation CIR d'un robot mobile différentiel

Le modèle cinématique du robot différentiel est donné par les équations suivantes :

$$\dot{x} = v \cos \varphi \quad (13)$$

$$\dot{y} = v \sin \varphi \quad (14)$$

$$\dot{\varphi} = \Omega \quad (15)$$

Ces équations relient la dérivée de la position (x, y, φ) du robot à la commande $u = (v, \Omega)^T$, avec φ rotation instantanée du robot par rapport au repère (O, X, Y) . De ce fait la position du robot est donnée par :

$$x(t) = \int_0^t v(\sigma) \cos(\varphi(\sigma)) d\sigma \quad (16)$$

$$y(t) = \int_0^t v(\sigma) \sin(\varphi(\sigma)) d\sigma \quad (17)$$

$$\varphi(t) = \int_0^t \Omega(\sigma) d\sigma \quad (19)$$

1.3.3 Robots mobiles tricycles

Description

L'architecture d'un robot mobile tricycle est représentée dans la figure 1.6. Ce robot est constitué de deux roues fixes (passives) de même axe et d'une roue centrée orientable placée sur l'axe longitudinal du robot. Le mouvement est conféré au robot par deux actions: la vitesse longitudinale et l'orientation de la roue orientable.

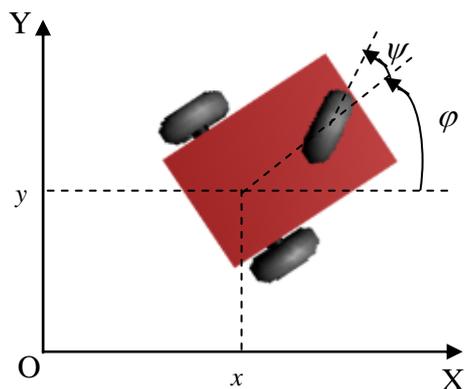
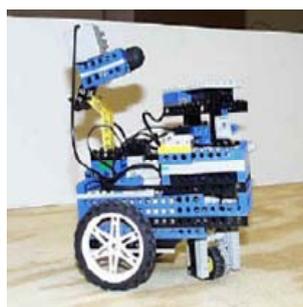


Figure 1.6 Robot mobile tricycle

Un exemple de robots tricycles est représenté dans la figure 1.7. Ce type de robot possède les mêmes propriétés cinématiques que le robot de type bicycle sauf que celui-ci est constitué d'une seule roue fixe passive et une roue centrée orientable.



(a)



(b)

Figure 1.7 : Exemple de robots mobiles de type tricycle

Modélisation

Centre instantané de rotation : le *CIR* du robot se situe à l'intersection des axes des roues fixes et de la roue orientable, comme cela est représenté à la figure 1.8.

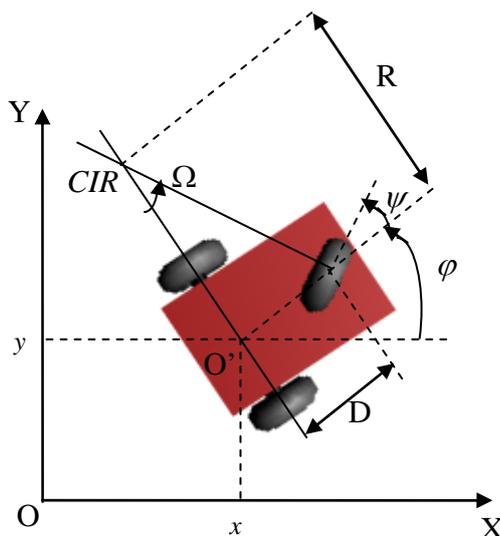


Figure 1.8 : Centre instantané de rotation du robot mobile de type tricycle

On peut déterminer R de manière géométrique à partir de l'angle d'orientation ψ de la roue avant et Ω à partir de la vitesse linéaire v du robot (vitesse en O') et de R :

$$R = \frac{D}{\tan \psi} \quad (20)$$

$$\Omega = \frac{v}{D} \tan \psi \quad (21)$$

La vitesse linéaire v peut être exprimée en fonction de la vitesse linéaire de la roue orientable v_s :

$$v = v_s \cos \psi \quad (22)$$

Ce type de robot peut se diriger en ligne droite pour $\psi = 0$ et théoriquement tourner autour du point O' (sur lui-même) pour $\psi = \frac{\pi}{2}$. Néanmoins, le rayon de braquage de la roue orientable, généralement limité, impose le plus souvent des valeurs de ψ telles que $-\frac{\pi}{2} < \psi < \frac{\pi}{2}$, interdisant cette rotation du robot sur lui-même.

Le modèle cinématique l'écriture des contraintes sur chacune des roues est un raisonnement similaire à celui suivi dans le cas du robot différentiel permettant de déterminer les modèles cinématiques des robots de type tricycle. Toutefois, par un simple raisonnement géométrique, on établit les équations suivantes représentant la dérivée de la position du robot:

$$\dot{x} = v \cos \varphi \quad (23)$$

$$\dot{y} = v \sin \varphi \quad (24)$$

$$\dot{\varphi} = \Omega = \frac{v}{D} \tan \psi \quad (25)$$

$$\dot{\psi} = \Omega_s \quad (26)$$

Où $u = (v, \Omega_s)^T$ est le vecteur de commande cinématique, φ la rotation instantanée du robot par rapport au repère (O, X, Y) , et Ω_s la vitesse d'orientation imposée à la roue orientable. Ces équations sont celles du modèle cinématique.

Donc la position du robot est donnée par :

$$x(t) = \int_0^t v(\sigma) \cos(\varphi(\sigma)) d\sigma \quad (27)$$

$$y(t) = \int_0^t v(\sigma) \sin(\varphi(\sigma)) d\sigma \quad (28)$$

$$\varphi(t) = \int_0^t \Omega(\sigma) d\sigma \quad (29)$$

1.3.4 Robots mobiles de type voiture

Le cas du robot de type voiture est très similaire à celui du tricycle. La différence se situe au niveau du train avant, qui comporte deux roues au lieu d'une. Cela va de soit, on rencontre beaucoup plus souvent ce type de système. Deux réalisations sont montrées à la figure 1.9.



(a) Robot Kanade



(b) Robot Cycab

Figure 1.9 : Exemple de robots mobiles de type voiture

Modélisation

Comme on l'a vu précédemment, l'existence d'un *CIR* unique impose que les axes des roues du robot soient concourants (qui se coupent en un point). Dans le cas du robot de type voiture, cela impose aux roues du train avant d'avoir une orientation différente, comme illustré à la figure 1.10. Le roulement idéal, assurant que le *CIR* est bien unique, est réalisé sur une voiture par un système de braquage différentiel (dit d'Ackerman). Par ailleurs, les trajectoires des roues n'ayant pas même rayon de courbure, leurs vitesses sont également différentes.

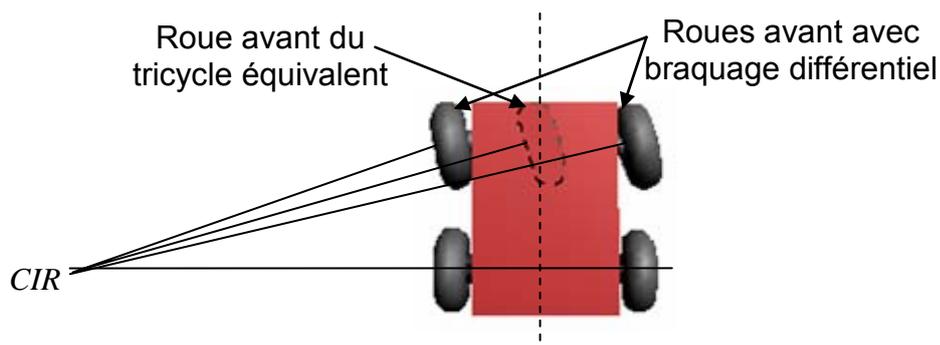


Figure 1.10 : Robot mobile de type voiture et sont *CIR*

L'équivalence entre tricycle et voiture réside dans le fait de figurer une roue virtuelle qui transformerait un robot de type voiture en tricycle en plaçant la roue orientable du tricycle au centre de l'axe des roues avant de la voiture, orientée de sorte que le *CIR* reste inchangé, conformément à la figure 1.10.

De ce fait, le modèle cinématique reste inchangé et les équations de la position du robot seront celles exprimées pour le robot tricycle.

1.3.5 Robots mobiles omnidirectionnels

Description

Un robot mobile est dit omnidirectionnel si l'on peut agir indépendamment sur les vitesses de translation selon les axes \bar{x} et \bar{y} , et la vitesse de rotation autour de \bar{z} . D'un point de vue cinématique cela n'est pas possible avec des roues fixes ou des roues centrées orientables [7]. On peut en revanche réaliser un robot omnidirectionnel ayant recours à un ensemble de trois roues décentrées orientables ou de trois roues suédoises disposées aux sommets d'un triangle équilatéral (voir figure 1.11).

Trois roues sont suffisantes mais dans certain cas une quatrième offre des possibilités d'optimisation et permet de rendre le système plus robuste en évitant le glissement dans le cas d'un sol qui ne serait pas parfaitement plat par exemple.

Notons qu'en ce qui concerne les roues suédoises (figure 1.2.d), la rotation des galets inclinés permet aux roues du robot de rouler dans une direction perpendiculaire à celle dans laquelle elles roulent normalement. C'est ce qui permet au robot de se déplacer dans diverses directions sans qu'il ait à tourner. Donc le corps du robot lui-même n'effectue pas de rotation mais uniquement des translations.

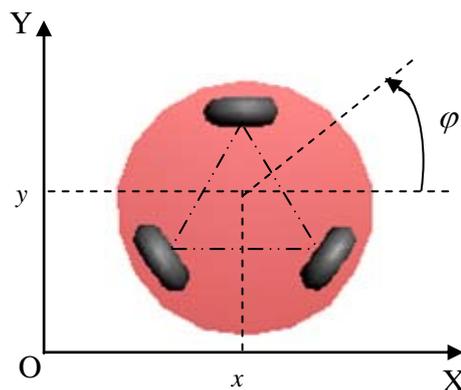


Figure 1.11 Robot mobile omnidirectionnel

Modélisation

Dans ce cas on peut considérer qu'il est possible d'appliquer directement la commande sur le modèle cinématique qui sera défini par les équations suivantes :

$$\dot{x} = u_1 \quad (30)$$

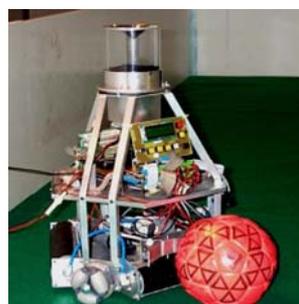
$$\dot{y} = u_2 \quad (31)$$

$$\dot{\varphi} = u_3 \quad (32)$$

Où $u = (u_1, u_2, u_3)^T$ représente le vecteur de commande. On choisit ainsi généralement ce type de robot pour se dispenser des problèmes de planification et de commande liés à la non holonomie. L'avantage d'une cinématique extrêmement simple est cependant à mettre en balance avec les inconvénients liés à une localisation odométrique déficiente et à une grande complexité mécanique, en plus qu'ils sont limités en capacité de franchissement et requièrent un sol très plat. La figure 1.12 représente un exemple de ce type de robots.



(a)



(b)

Figure 1.12 : Exemple de robots mobiles omnidirectionnels :
 (a) Robot omnidirectionnel muni de roues décentrées orientables
 (b) Robot omnidirectionnel muni de roues suédoises.

Une autre technique consiste à utiliser des roues sphériques ou des associations de roues sphériques tronquées, montées orthogonalement les unes par rapport aux autres [8]. Dans ce dernier cas, lors du pivotement de l'axe motorisé, le contact sur le sol se fait en alternance avec l'une ou l'autre des deux roues (sphères) formant l'essieu (figure 1.13).

Les axes de rotation des deux sphères sont perpendiculaires entre eux et également perpendiculaires et concourants avec l'axe longitudinal de l'essieu.

La rotation de cet essieu constitue le moyen de propulsion de la structure mobile (1 moteur par essieu).

La mobilité interne que constitue la rotation libre des sphères, permet un déplacement en translation de l'essieu (et de la structure qui le porte) dans sa direction longitudinale, sans glissement des roues sur le sol (figure 1.14).

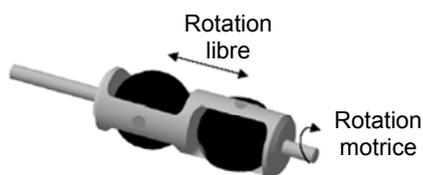


Figure 1.13 : Représentation 3D d'un essieu

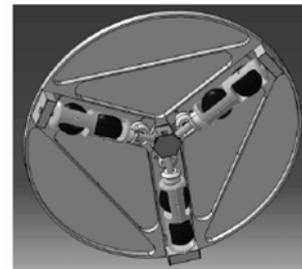


Figure 1.14 : Disposition des trois essieux

1.3.6 Robots mobiles à traction Synchrones

Description

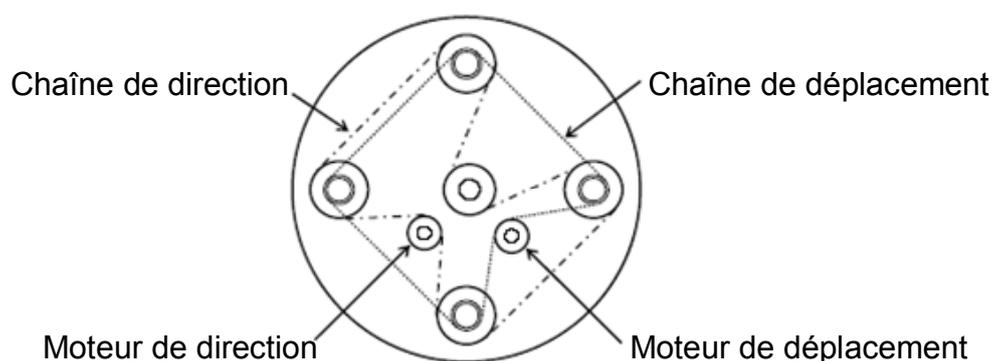


Figure 1.15 : L'architecture d'une plateforme pour robot mobile à traction synchrone

La traction synchrone (synchronous drive) est une technique utilisée pour minimiser l'effet de glissement et augmenter la force de traction.

On rencontre ce type de robot dans l'industrie automobile, et dans les robots tout terrains. La configuration du robot à traction synchrone est similaire à un robot à trois ou quatre roues couplées de façon quelles soient actionnées en même temps, en ayant la même vitesse et la même orientation, ce système est réalisé grâce à deux moteurs un pour la traction et l'autre pour l'orientation où l'ensemble est relié par une chaîne (ou une ceinture) pour assurer que toutes les roues tournent de façon synchrone. La figure 1.15 montre un robot à quatre roues couplées avec des chaînes.

La plateforme mobile utilisée pour notre travail nommée *B21r* est du type à traction synchrone possédant deux ceintures de capteurs à ultrasons, une ceinture de capteurs infrarouges, un Laser, des capteurs tactiles placés le long de ses parois, et une camera CCD (N/B)(figure 1.16).



Figure 1.16 : Robot mobile *B21r* à traction synchrone du CDTA

1.3.6.1 Structure des roues

On a vu qu'à chaque modèle de robot correspond un type de roues selon son architecture et ses tâches à accomplir. Le *B21r* dispose de quatre roues décentrées orientables tournant selon deux axes, une rotation selon l'axe y permettant le roulement des roues pour avoir la translation, et une rotation selon l'axe K permettant de changer leurs orientations (figure 1.17).

Les paramètres de la roue orientée sont :

r = rayon de la roue.

v = vitesse linéaire de la roue.

ω = vitesse angulaire de la roue.

Ω = vitesse d'orientation

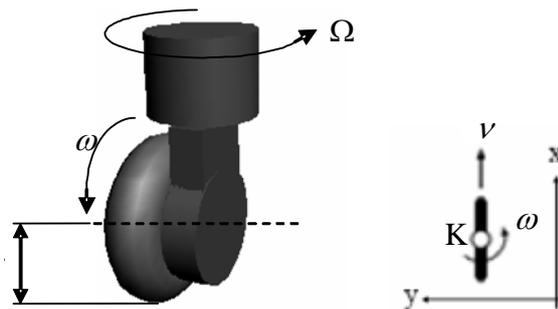


Figure 1.17 : Roue décentrée orientable

D'où on en déduit que le nombre de degré de liberté du robot $B21r$ est :

$$\delta_M = \delta_m + \delta_t = 1 + 1 = 2.$$

δ_m : degré de mobilité (degree of mobility).

δ_t : degré d'orientation (degree of steerability).

1.3.6.2 Centre instantané de rotation

Quand le robot se déplace suivant un arc (un virage), les rayons liés au centre du robot et perpendiculaire à l'axe virtuel défini par son orientation pour chaque position converge vers un point de l'espace, qui est le centre instantané de rotation (CIR) (figure 1.18).

Ayant une vitesse linéaire v et une vitesse angulaire Ω , le robot peut se déplacer de façon rectiligne ($v \neq 0, \Omega = 0$), le CIR se trouve à l'infinie. Alors que lors d'une rotation pure du $B21r$ ($\Omega \neq 0, v = 0$), le CIR se trouve sur son centre (O'). Et enfin dans le cas où le déplacement est un virage c'est-à-dire qu'il se déplace sur un arc de cercle de longueur δs , de rayon R et avec un angle $\delta \varphi$.

Le déplacement δs du robot mobile est :

$$\delta s = R \delta \varphi \quad (33)$$

R et φ peuvent être positifs ou négatifs selon le sens de déplacement du robot (sens du virage).

Comme le robot tourne autour du CIR avec une vitesse linéaire v et une vitesse angulaire $\delta \varphi$, on peut écrire :

$$v = R \delta \varphi \quad (34)$$

Sachant que cette vitesse angulaire du robot mobile $B21r$ est : $\delta \varphi = \Omega$

Donc nous pouvons exprimer la valeur du rayon du centre instantané de rotation CIR par l'équation :

$$R = \frac{v}{\delta\varphi} \quad (35)$$

La position du *CIR* du *B21r* dans le repère $R^U (O, X, Y)$ est donnée par les coordonnées cartésiennes suivantes :

$$x_{CIR} = x - R \sin \delta\varphi \quad (36)$$

$$y_{CIR} = y + R \cos \delta\varphi \quad (37)$$

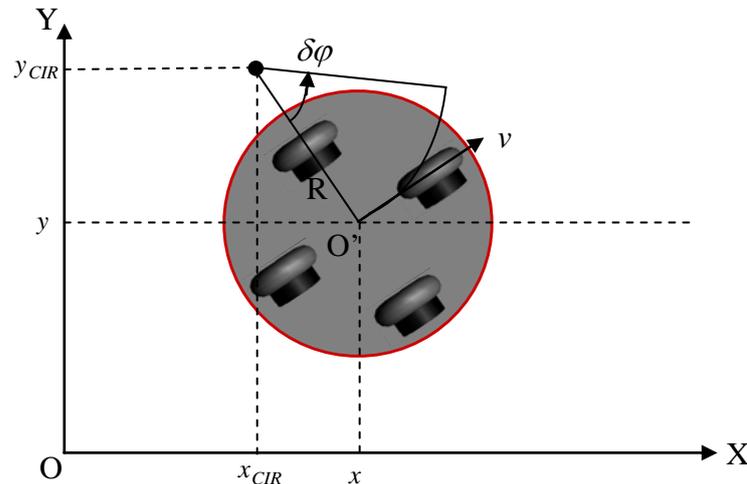


Figure 1.18 : Position du centre instantané de rotation dans le cas d'un virage à gauche

1.3.6.3 La cinématique

Afin de trouver les équations de déplacement du robot *B21r*, on utilisera les deux contraintes déduites du principe de roulement sans glissement présentées dans les équations (3) et (4). En introduisant la vitesse de roulement de la roue $v = r\dot{\theta}$ et Ω sa vitesse de rotation autour de l'axe \vec{K} (voir figure 1.19), on forme le modèle cinématique suivant :

$$\dot{x}_{re} = v \cos \varphi \quad (38)$$

$$\dot{y}_{re} = v \sin \varphi \quad (39)$$

$$\dot{\varphi} = \Omega \quad (40)$$

Puisque les vitesses connues sont celles des roues motrices v_A, v_B, v_D, v_E qui sont égales, nous pouvons calculer la vitesse v_C du centre C du robot mobile *B21r*, tel que :

$$v_C = v_A + \Omega \wedge \overrightarrow{AC} \quad (41)$$

$$v_C = v_B + \Omega \wedge \overrightarrow{BC} \quad (42)$$

$$v_C = v_D + \Omega \wedge \overrightarrow{DC} \quad (43)$$

$$v_C = v_E + \Omega \wedge \overrightarrow{EC} \quad (44)$$

La somme des quatre équations sachant que $v_A = v_B = v_D = v_E = v$ est donnée par :

$$4v_C = 4v + \Omega \wedge (\overrightarrow{AC} + \overrightarrow{BC} + \overrightarrow{DC} + \overrightarrow{EC}) \quad (45)$$

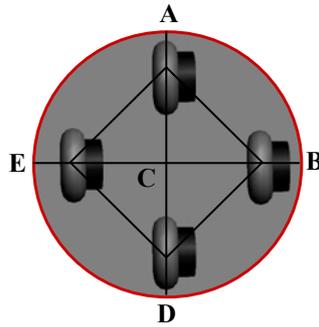


Figure 1.19 : Disposition des quatre roues du robot mobile B21r.

A partir de la figure 1.19, nous pouvons en déduire que les vecteurs \overrightarrow{AC} et \overrightarrow{DC} ainsi que \overrightarrow{EC} et \overrightarrow{BC} sont des vecteurs égaux et opposés, donc :

$$4v_C = 4v, \text{ ce qui signifie que } v_C = v_A = v_B = v_D = v_E = v \quad (46)$$

On en déduit que la vitesse du robot est égale à la vitesse des roues et à la même direction que ces dernières, et donc cette vitesse sera parallèle au plan (O, X, Y).

La vitesse linéaire du robot peut être exprimée en fonction de la variation de ses coordonnées dans le repère univers tel que :

$$v = \sqrt{x^2 + y^2} \quad (47)$$

Le changement d'orientation du robot c'est-à-dire sa vitesse de rotation Ω est exprimé par :

$$\Omega = \dot{\varphi} = \frac{\delta\varphi}{\delta t} \quad (48)$$

Où φ est l'orientation instantanée du robot dans le repère R^u (l'angle φ est représenté par rapport à l'axe horizontal X, mais dans notre travail il sera représenté par rapport à l'axe vertical Y, ce qui nous à été imposé par le système software (*Mobility*) du robot B21r).

Pour le cas de changement de position dans la direction de l'axe X et Y du repère univers R^u , on aura :

$$\dot{x} = v \cos \varphi \quad (49)$$

$$\dot{y} = v \sin \varphi \quad (50)$$

Les équations 48, 49, 50 constituent l'ensemble d'équations cinématiques du robot mobile $B21r$, correspondant à notre modèle cinématique (Figure 1.20).

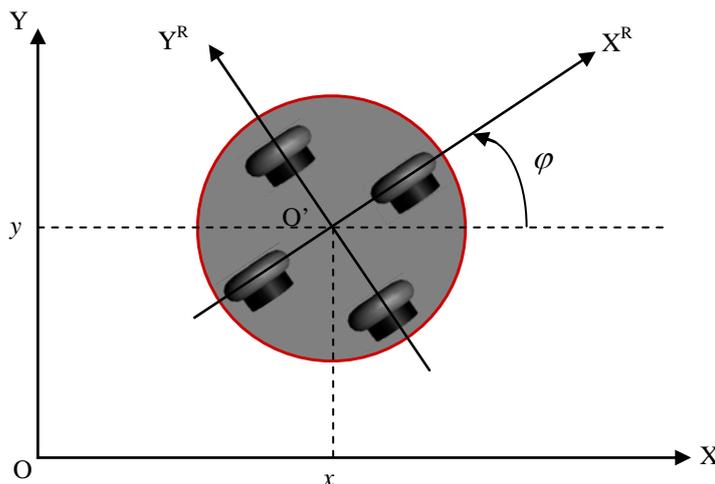


Figure 1.20 : Le modèle cinématique utilisé pour le robot mobile B21r

D'où la position du robot mobile dans le repère absolu est donnée par :

$$x(t) = \int_0^t v(\sigma) \cos(\varphi(\sigma)) d\sigma \quad (51)$$

$$y(t) = \int_0^t v(\sigma) \sin(\varphi(\sigma)) d\sigma \quad (52)$$

$$\varphi(t) = \int_0^t \Omega(\sigma) d\sigma \quad (53)$$

1.4 Holonomie et non holonomie

La capacité d'un mobile à pouvoir se déplacer à partir d'une situation dans n'importe quelle direction est appelée "holonomie". En effet, le mécanisme holonomique permet au robot de manœuvrer dans n'importe quelle direction arbitraire à partir de n'importe quelle configuration arbitraire. Il n'y a pas de contrainte sur les commandes. Cela signifie donc une très grande simplification des problèmes de contrôle.

Un système holonomique est un système qui a le même nombre de déplacements virtuels que le nombre de coordonnées généralisées pour le décrire. Comme nous l'avons vu, nous avons besoin de trois coordonnées généralisées pour décrire de façon unique la configuration d'un robot sur le plan. Un robot holonomique omnidirectionnel est donc un robot pour lequel les trois mouvements planaires indépendants, deux de translation et un de rotation, sont admissibles à une vitesse non nulle à partir d'une configuration quelconque.

Par contre, de nombreux robots mobiles sont des systèmes non holonome tel que les robots différentiels, tricycles, voitures, et à traction synchrone. Pour ce type de robots, bien que l'espace de configuration sur le plan est de trois dimensions, le mouvement est produit par deux déplacements indépendants seulement, et donc le robot ne possède que deux degrés de liberté.

D'un point de vue mathématique, nous pouvons dire que la non holonomie d'un robot est due au fait que ses équations de contraintes ne sont pas intégrables [9]. Pour cette raison, l'analyse cinématique, dynamique et le contrôle sont plus complexes étant donné que les coordonnées de leur représentation ne peuvent être éliminées en utilisant les équations de contrainte [10]. Donc, les systèmes possédant des contraintes non holonomiques nécessitent toujours un nombre plus grand de coordonnées pour leur description que le nombre de degrés de liberté.

Une étude très complète sur la modélisation cinématique et dynamique des robots mobiles à roues a été publiée par Muir [11]. En dynamique, le nombre de degrés de liberté est défini comme le nombre de coordonnées utilisées pour spécifier la configuration du système moins le nombre d'équations de contrainte indépendante [9].

La contrainte de roulement sans glissement impose au point de contact entre la roue et le sol (point I) de se déplacer dans une direction perpendiculaire à l'axe de rotation de la roue ce qui se traduit par la relation suivante :

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0 \quad (54)$$

La relation (54) constitue ce qu'on appelle "contrainte cinématique non holonome". Nous pouvons déduire bien que l'espace des configurations du robot mobile $B21r$ (de type synchro drive) soit de dimension trois, il existe deux paramètres de contrôle à savoir la vitesse linéaire v et la vitesse angulaire Ω , par conséquent le robot est non

holonome et possède deux degrés de liberté du moment que sa trajectoire est toujours normale à l'axe des roues motrices.

1.5 Conclusion

Au cours de ce chapitre, nous avons présenté les différents types de robots mobiles à roues les plus utilisés. Cette variété réside dans leur mode de locomotion qui dépend du type et de la disposition des roues employées. Une étude cinématique spécifique est donc nécessaire pour chaque type de robot avant son utilisation.

Nous avons montré aussi dans ce chapitre, la distinction entre systèmes holonomes et systèmes non holonomes. On a vu que la majorité des robots mobiles à roues sont des systèmes non holonomes sauf dans le cas des robots mobiles omnidirectionnels. Dans notre cas le robot mobile *B21r* est non holonome, de type à traction synchrone. L'architecture matérielle de ce robot mobile a été aussi détaillée dans ce chapitre.

Ces robots mobiles sont dédiés à diverses applications où ils auront à se déplacer dans des environnements d'intérieur ou d'extérieur. Pour cela ils doivent être munis d'un système de perception qui sera mis en œuvre pour rendre le robot complètement autonome. Un état de l'art des capteurs les plus utilisés en robotique mobile ainsi que leurs modèles sera présenté dans le chapitre suivant.

CHAPITRE 2 PERCEPTION

2.1 Introduction

La notion de perception en robotique mobile est relative à la capacité du système à recueillir, traiter et mettre en forme des informations utiles au robot pour agir et réagir dans l'environnement qui l'entoure. Cet environnement peut être structuré, non structuré ou semi structuré. Aussi, pour extraire les informations utiles à l'accomplissement de sa tâche, il est nécessaire que le robot dispose de nombreux capteurs mesurant aussi bien son état interne que l'environnement dans lequel il évolue. Le choix des capteurs dépend bien évidemment de l'application envisagée. Le système de perception d'un robot mobile intégrera le plus souvent plusieurs capteurs pour un enrichissement des informations sensorielles, dans ce cadre, des méthodes de fusion de données seront généralement employées pour conditionner ces informations sensorielles.

Le capteur traduit un état physique en signaux électriques. La modélisation consiste à associer une représentation informatique ou/et mathématique aux informations délivrées par le capteur, ceci dans le but de pouvoir les manipuler avec les outils disponibles.

Pour se focaliser sur le problème de localisation, nous allons nous restreindre dans ce chapitre aux capteurs utiles à cette tâche. Dans un second temps, nous expliquerons comment utiliser les données issues de ces capteurs pour obtenir une représentation fiable de l'état du système et de l'environnement.

2.2 Capteurs de perception

Nous pouvons définir deux catégories de capteurs en robotique mobile : ceux qui délivrent une information caractérisant l'environnement et ceux qui fournissent une information propre au comportement interne du robot. Les premiers sont appelés capteurs extéroceptifs employés pour la localisation absolue à partir de la perception de l'environnement extérieur au robot, et les seconds sont les capteurs proprioceptifs

exploités pour la localisation relative en estimant la distance parcourue par le robot (n'empêche qu'ils peuvent être utilisés pour d'autres applications).

2.2.1 Les capteurs proprioceptifs

Les capteurs proprioceptifs fournissent par intégration des informations élémentaires sur les paramètres cinématiques du robot. Les informations sensorielles gérées dans ce cadre sont généralement des vitesses, des accélérations, des angles de giration, des angles d'altitude. Cependant, ils ne peuvent pas procurer des renseignements lors de l'arrêt du robot mobile.

On peut regrouper les capteurs proprioceptifs en deux familles [12] :

- Les capteurs de déplacement qui comprennent les odomètres, les accéléromètres, les radars Doppler,... Cette catégorie permet de mesurer des déplacements élémentaires, des variations de vitesse ou d'accélération sur des trajectoires rectilignes ou curvilignes.
- Les capteurs d'altitude, qui mesurent deux types de données : les angles de cap (direction de déplacement), et les angles de roulis et de tangage. Ils reposent principalement sur les mesures inertielles.

2.2.1.1 Les capteurs de déplacement

2.2.1.1.1 Les odomètres

les systèmes odométriques peuvent fournir l'information concernant le changement de la position du robot mobile, ces informations sont extraites au moyen de capteurs qui comptent le nombre de rotations pour les axes des roues et leurs axes d'orientation, pour cela des codeurs à haute résolution sont utilisés. Dans la plupart des cas ce sont des codeurs optiques incrémentaux, il existe néanmoins d'autres codeurs (magnétiques, inductifs, capacitifs...). L'information sur l'angle est discrétisée et le nombre de tours est compté, normalement la résolution est haute cependant la discrétisation devient un problème quand on mesure des rotations lentes.

Comme tout capteur, le modèle d'odométrie a bien des limitations du moment que l'idée fondamentale de celui-ci est l'intégration d'information incrémentale du mouvement à travers le temps, ce qui mène inévitablement à l'accumulation d'erreurs, et particulièrement l'accumulation d'erreurs d'orientation qui causera une

grande erreur dans la position [13], cette erreur croît proportionnellement avec la distance traversée par le robot.

Quand on cherche à mesurer ou réduire les erreurs de l'odométrie, il est important de faire la différence entre l'erreur systématique et non systématique de l'odométrie. Les erreurs systématiques sont ceux qui sont une partie inhérente à la cinématique du robot ou à ces paramètres de contrôle indépendamment de l'environnement. Les erreurs non systématiques sont celles qui dépendent de l'environnement du robot et diffèrent d'un environnement à un autre.

La distinction entre ces deux groupes est importante car chaque groupe influe différemment sur la plateforme mobile, leurs remèdes sont différents et plus important tous les deux nécessitent différentes techniques de mesure. Borenstein et ses collaborateurs [14-18] étudient les différentes sources des deux types d'erreurs, et catégorisent ces erreurs d'odométrie comme tel :

- Les erreurs systématiques :
 - Diamètres des roues différents.
 - Roues mal positionnées.
 - Une résolution limitée des encodeurs.
 - vitesse d'échantillonnage des encodeurs limitée.
- Les erreurs non systématiques
 - Se déplacer sur un sol raboteux.
 - Se déplacer à travers des objets imprévus sur le sol.
 - Glissement des roues (sol lisse, excès d'accélération, patiner dans un tournant rapide, etc.).
 - Forces externes (interaction avec des corps externes).
 - Forces internes (roue folle).
 - Pas de contact avec le sol.

Tout modèle odométrique aussi bon qu'il soit n'est qu'une approximation du vrai modèle cinématique, et quand l'odométrie est utilisée pour la prédiction de la position, la partie la plus critiquée de l'estimation est la capacité à estimer l'orientation du robot, car même une petite erreur dans l'orientation φ du robot (ce qu'on appelle par drift) induit à une grande erreur dans la position, mais par une modélisation soignée cette erreur systématique pourra être limitée (Borenstein & Feng 1996b, Borenshtein 1998).

Les plateformes d'intérieur ont normalement une qualité odométrique beaucoup plus meilleure que celle d'extérieur, vu la nature plane des surfaces d'intérieur contrairement aux surfaces d'extérieur qui sont généralement rugueuses.

Comme l'odométrie (en plus pas cher) est une partie importante de la localisation et que l'information odométrique est facile à utiliser, c'est devenue une partie centrale de la plupart des systèmes de localisation. L'odométrie est hautement fiable pour des petites distances, mais se dégrade avec la distance.

Au cours de notre étude (détaillée dans le chapitre 4), on va développer un modèle pour le système odométrique du robot mobile *B21r*. Ce modèle odométrique nous servira de modèle d'évolution lors de la phase d'estimation de la position par le filtre de Kalman étendu.

2.2.1.1.2 Les accéléromètres

L'accéléromètre est un capteur qui mesure l'accélération linéaire en un point donné. En pratique, la mesure de l'accélération est effectuée à l'aide d'une masse d'épreuve M , de masse m , reliée à un boîtier du capteur [1]. Le principe de ce capteur est de mesurer l'effort massique non gravitationnel qu'on doit appliquer à M pour le maintenir en place dans le boîtier lorsqu'une accélération est appliquée au boîtier. Le calcul du déplacement élémentaire du robot est obtenu par double intégration de ces informations. Cette double intégration conduit généralement à des accumulations importantes d'erreurs. Ce capteur est plus coûteux que les odomètres.

2.2.1.1.3 Le radar Doppler

Le radar Doppler fournit une estimation instantanée de la vitesse linéaire d'une plateforme mobile par rapport à un objet de la scène en se basant sur l'effet Doppler-Fizeau. Le principe est de diriger un faisceau électromagnétique de fréquence F vers le sol. Le faisceau reçu après rediffusion sur le sol présente un décalage de fréquence ΔF proportionnel à la vitesse V du déplacement relatif du véhicule par rapport au sol (Figure 2.1). L'intégration de la mesure de vitesse fournira une estimation du déplacement du mobile.

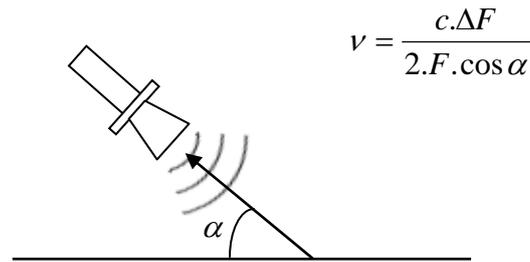


Figure 2.1 Principe du capteur à effet Doppler

v : Vitesse de déplacement de l'émetteur.

c : Vitesse de propagation de l'onde.

ΔF : Différence de fréquence.

F : Fréquence d'émission.

α : Angle d'inclinaison entre le capteur et le sol.

Ce capteur est le plus souvent utilisé en milieu extérieur. Il présente le gros avantage d'être insensible au patinage et aux glissements des roues, raison pour laquelle est employé dans certaines applications pour réduire les effets des erreurs odométriques. En revanche, sa précision se dégrade rapidement avec la diminution de la vitesse. De plus il est généralement lourd à mettre en oeuvre, aussi bien du point de vue électronique que du point de vue du traitement des informations reçues.

2.2.1.2 Les capteurs d'altitude

Les capteurs d'altitude permettent d'estimer les paramètres intrinsèques du robot qui sont les angles de cap, de roulis et de tangage. Ces capteurs sont principalement de type inertiel.

Ces capteurs ont pour point commun d'être généralement coûteux et sensibles au bruit, d'où une intégration moins fréquente dans les systèmes embarqués que les odomètres.

2.2.1.2 .1 Le gyroscope, gyromètre et gyrocompas

Les gyroscopes permettent la mesure de l'orientation du robot mobile (Figure 2.2.a). Ces mesures sont intégrées pour calculer sa position. Ces capteurs sont particulièrement répandus en positionnement de robots mobiles, car ils aident à compenser pour la plupart l'imprécision sur l'orientation fournie par l'odomètre. Ils

sont intéressants en robotique mobile parce qu'ils peuvent compenser les défauts des odomètres. Une erreur d'orientation odométrique peut entraîner une erreur de position cumulative qui peut être diminuée voire compensée par l'utilisation conjointe de gyroscopes. Les gyroscopes très précis sont trop onéreux pour être utilisés en robotique mobile. Cependant, les gyroscopes à fibre optique, connu pour leur grande précision, ont vu leur prix chuter et sont donc devenus une solution attractive [1].

Le gyromètre est un capteur qui permet de mesurer une vitesse angulaire du robot ou l'angle (le cap) (Figure 2.2.b). Il existe plusieurs types de gyromètres : les premiers à avoir fait leur apparition furent mécaniques, aujourd'hui, on utilise surtout des gyromètres laser ou des gyromètres optiques, n'empêche qu'il existe aussi des gyromètres à structures vibrantes, capacitifs, etc. [19].

Le gyrocompas est un capteur qui permet de mesurer le cap. Il est composé d'un gyroscope et d'un compas magnétique. Le gyrocompas conserve le nord magnétique durant tout le déplacement du véhicule, après l'avoir initialement déterminé de façon autonome.

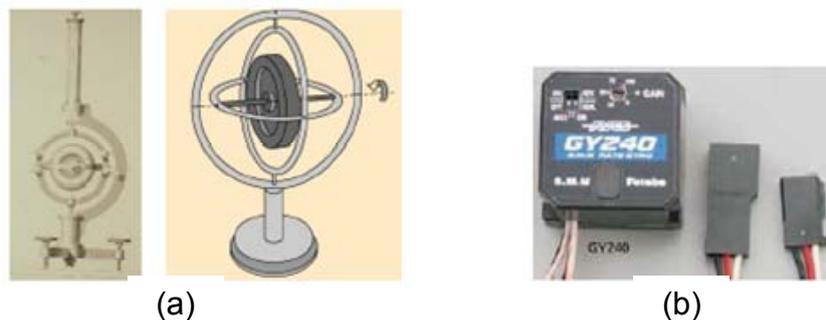


Figure 2.2 Exemples de Gyroscope et gyromètre
 (a) Gyroscope mécanique de Foucault
 (b) gyromètre Futaba à structure vibrante

2.2.1.2 .2 Le magnétomètre ou compas magnétique

Le magnétomètre qui est aussi appelé compas magnétique (ou boussole) mesure la direction du champ magnétique terrestre pour déduire l'orientation du robot. Parmi toutes les technologies adoptées pour ce type de capteur, la mieux adaptée pour la robotique mobile est celle dite à vanne de flux. Elle a l'avantage de consommer peu d'énergie, de n'avoir aucune pièce mobile, d'être résistante aux chocs et aux vibrations et d'être peu coûteuse. Toutefois, les mesures sont perturbées par l'environnement magnétique du robot (comme par exemple les lignes

d'énergie ou les structures en acier). Ceci rend difficile l'utilisation de ce capteur en milieu d'intérieur, c'est pourquoi il est très utilisé sur des robots mobiles évoluant dans la nature en apportant le plus grand soin à leur positionnement sur le robot pour éviter les influences des composants du robot, notamment les moteurs électriques.

La caractéristique principale de ce capteur est sa précision moyenne qui sur un long trajet, est relativement bonne [12].

2.2.2 Les capteurs extéroceptifs

Les capteurs extéroceptifs permettent de percevoir le milieu d'évolution du robot. Ils sont généralement le complément indispensable aux capteurs présentés précédemment. Des méthodes de fusion de données seront alors utilisées pour conditionner et traiter les informations sensorielles de natures différentes. Deux familles de capteurs extéroceptifs embarqués peuvent être identifiées : les capteurs télémétriques et les systèmes de vision.

2.2.2.1 Les capteurs télémétriques

On appelle télémétrie toute technique de mesure de distance par des procédés acoustiques, optiques ou radioélectriques. L'appareil permettant de mesurer les distances est appelé *télémètre*. De même qu'il existe différentes techniques de mesure de distance (mesure du temps de vol d'une onde, triangulation), il existe différentes technologies pour réaliser des télémètres. Nous présentons ici les plus répandues en robotique mobile, en donnant une idée de leur gamme de mesure et d'application.

2.2.2.1.1 Les capteurs à ultrasons

Les capteurs ultrasonores utilisent des ondes sonores dont les fréquences ne sont pas perceptibles par l'oreille humaine. Les fréquences couramment utilisées dans ce type de technologie vont de 20 khz à 200 khz. Les capteurs à ultrasons mesurent le temps de vol d'une onde ultrasonore entre son émission et sa réception après réflexion. A partir de ce temps et connaissant la vitesse de propagation de l'onde, la distance séparant le capteur de l'obstacle qu'il perçoit est directement disponible sans analyses complémentaires. Les acquisitions des mesures sont

rapides, ils peuvent donc être utilisés pour le processus qui commandent un traitement en temps réel comme l'évitement d'obstacle.

Les capteurs ultrasons sont aujourd'hui les capteurs les plus communément employés dans les systèmes de robotique mobile autonome d'intérieur ou d'extérieur en raison de leurs : bas prix, faible consommation, simplicité et compacité. Depuis plusieurs années, ils ont été utilisés dans des domaines tels que la modélisation d'environnement, l'évitement d'obstacle, l'estimation de la position ou la navigation. Puis des travaux ont montrés la possibilité de les employer en milieu d'extérieur [20]. Il existe différents types de capteurs à ultrasons ; les plus utilisés en robotique mobile sont de type polaroid [21], plusieurs exemples sont présentés dans la figure 2.3.

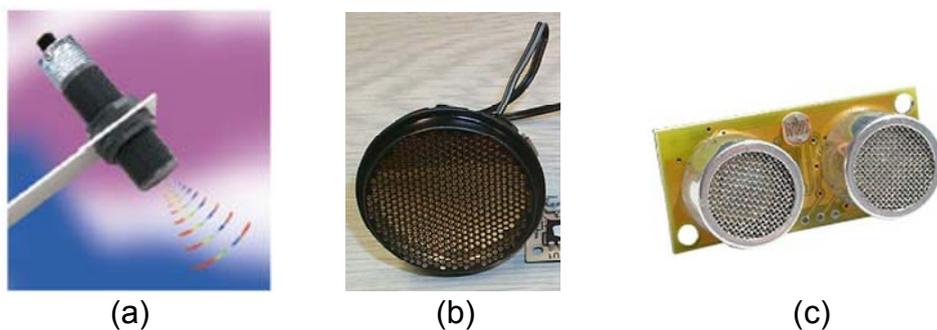


Figure 2.3 Exemples de télémètres ultrasonores

- (a) Télémètre ultrasonore Polaroid Migatron RPS 409 IS
- (b) Télémètre ultrasonore Polaroid série 600.
- (c) Télémètre ultrasonore MSU08

Malgré leurs avantages cités, les capteurs ultrasonores possèdent de nombreux inconvénients, comme leur forte sensibilité aux conditions d'utilisation, telles que la température, l'humidité, les échos parasites, et la nature de la texture de l'obstacle. Ces paramètres influent directement sur la précision de mesure. Une autre caractéristique de ces capteurs est leur cône d'émission assez large (de 20 à 30°) qui ne permet pas de connaître précisément la position de l'obstacle dans le cône d'émission du capteur.

Il est possible de disposer les capteurs à ultrasons en ceinture autour du robot afin de percevoir une grande partie de l'environnement du robot. Par ce dispositif, deux télémètres voisins ne peuvent être utilisés simultanément, car il est impossible de savoir par lequel des deux télémètres une onde réfléchi a été émise (phénomène de "crosstalk"). Un robot possédant plusieurs télémètres doit donc les activer l'un après l'autre.

Ces télémètres possèdent une “zone aveugle”, de quelques dizaines de centimètres, en dessous de laquelle ils ne peuvent détecter les obstacles. Cette zone est due à une temporisation entre l’émission de l’onde sonore et le début de la détection de l’onde réfléchi qui est nécessaire pour ne pas perturber cette mesure (figure 2.4).

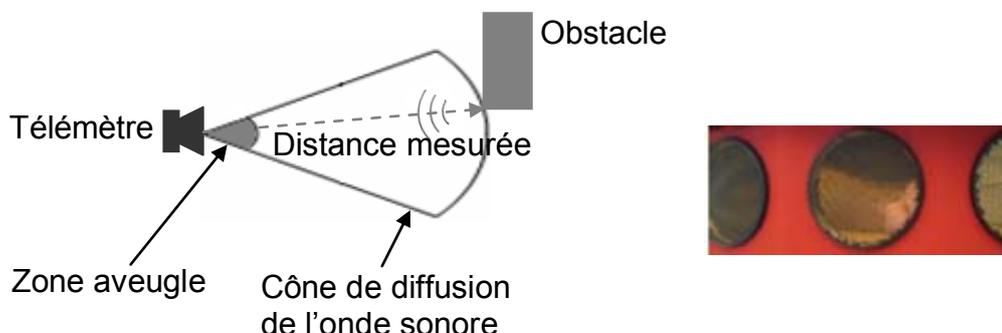


Figure 2.4 Principe du télémètre à ultrasons et un exemple de télémètre réel

2.2.2.1.2 Les capteurs lasers

Il existe deux manières de mesurer une distance grâce à un système laser. Soit il émet en mode continu et la distance est déterminée à partir du déphasage entre l’onde émise et l’onde réfléchi par l’obstacle, soit il émet en mode pulsé et la distance est proportionnelle au « temps de vol » de l’onde entre son émission et sa réception.

Le faisceau d’onde émis est très concentré, ce qui permet d’avoir un cône d’émission très étroit et donc une bonne précision de mesure. Pour obtenir un balayage de l’environnement en deux, voir en trois dimensions, différents systèmes de miroir tournants peuvent être associés au laser [22, 23].

Les lasers existants diffèrent par leurs caractéristiques et leurs domaines d’applications. En effet, la distance maximale mesurable des lasers peut aller d’une dizaine de mètres jusqu’à quelques centaines de mètres.

Les télémètres les plus utilisés à l’heure actuelle pour des applications de cartographie et de localisation sont les télémètres laser à balayage. Ils utilisent un faisceau laser mis en rotation afin de balayer un plan, en général horizontal, et qui permet de mesurer la distance des objets qui coupent ce plan (Figure 2.5).

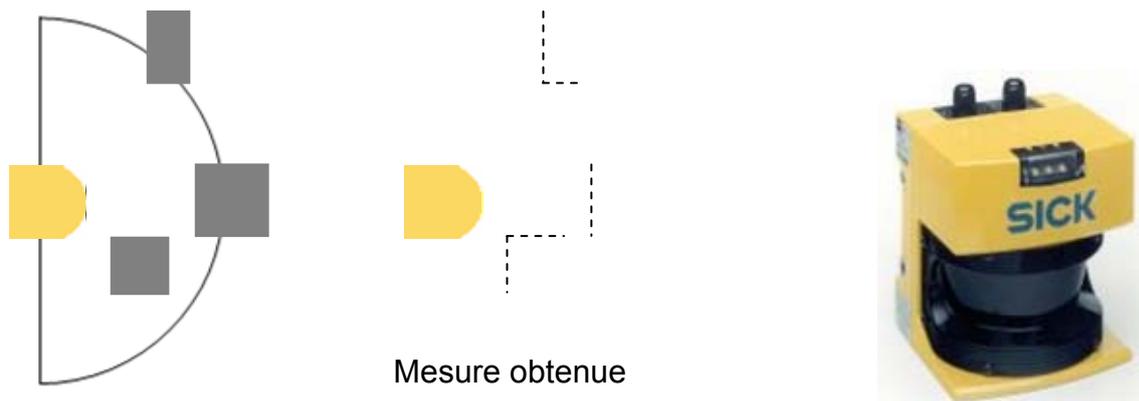


Figure 2. 5 Illustration d'un télémètre laser et un exemple de télémètre laser SICK à balayage fournissant 180 mesures réparties sur 180 degrés

Les télémètres laser les plus courants ont une bonne résolution angulaire car ils permettent d'obtenir une mesure de distance tout les demi ou un degré, sur une zone de 180 à 360 degrés selon les modèles. Ces télémètres sont très utilisés en environnement d'intérieur car ils fournissent des données abondantes et précises sur la position des objets caractéristiques de l'environnement tels que les murs.

Ils possèdent toutefois un certain nombre d'inconvénients. En premier lieu, leur zone de perception est restreinte à un plan et ne permet donc pas la détection d'obstacles situés hors de ce plan (un petit objet posé au sol par exemple). Ils ne peuvent pas non plus détecter les objets ne réfléchissant pas correctement la lumière du laser (tel que les vitres). Pour limiter ces inconvénients, il est possible de les utiliser en conjonction avec des capteurs à ultrasons qui ont un cône de détection plus large et qui peuvent détecter les vitres.

Ces télémètres restent toutefois chers, fragiles et délicats à mettre en œuvre en raison du matériel optique utilisé (lentilles, miroirs, etc.), n'empêche que leurs grandes qualités en terme de précision de mesure, de résolution angulaire et stabilité en température font d'eux l'un des meilleurs télémètres en robotique mobile.

2.2.2.1.3 Les capteurs à infrarouge

Les capteurs à infrarouges fonctionnent suivant le même principe que les capteurs laser et les capteurs à ultrasons. Ils se composent d'un couple émetteur-récepteur. L'émission s'effectue par une diode électroluminescente infrarouge. Le récepteur est une photodiode ou un phototransistor sensible au flux lumineux rétro diffusé par la cible.

Ces télémètres possèdent l'avantage d'avoir un cône de détection beaucoup plus restreint que les capteurs à ultrasons. Sauf qu'ils sont sensibles aux perturbations lumineuses ainsi qu'à la nature des surfaces réfléchissantes (par exemple, ils détectent difficilement les vitres). Leurs portées n'étant que de quelques dizaines de centimètres, ils sont utilisés essentiellement pour détecter les obstacles proches [24].

Un exemple de ces capteurs, est le capteur sharp de la famille GP2 (figure 2.6). Sur des robots mobiles, ces capteurs ne sont pas généralement utilisés pour repérer l'ensemble des corps présents dans l'environnement car leur portée est très faible. Ils servent plutôt à mesurer la proximité de corps proche du robot.

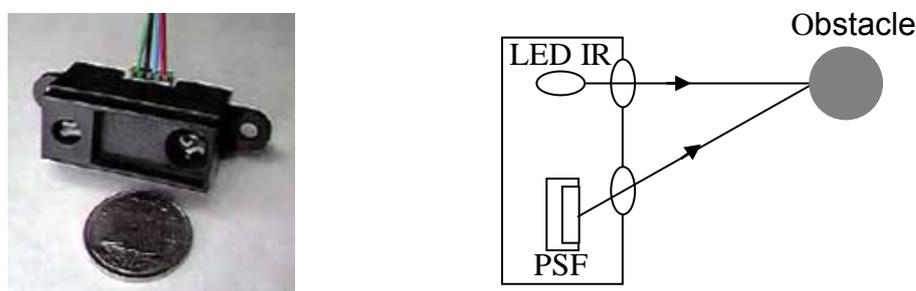


Figure 2. 6 Télémètre infrarouge Sharp

2.2.2.1.4 Le GPS

Les besoins de localisation étant omniprésents dans de très nombreux secteurs de la vie actuelle, l'idée d'avoir un système de localisation le plus universel possible a donné lieu à l'apparition du Global Positioning System (GPS). C'est un système de balises utilisant les ondes radio, et dont on a placé ces balises sur des satellites en orbite terrestre et qui est par conséquent accessible de quasiment partout à la surface du globe. Ce système permet donc d'avoir une mesure de sa position dans un repère global couvrant la terre avec une précision variant de quelques dizaines de mètres à quelques centimètres suivant les équipements. Ce système est cependant loin de résoudre tous les problèmes de localisation des robots mobiles.

Il fonctionne en effet difficilement dans des environnements urbains, et n'est pas utilisable à l'intérieur des bâtiments car les ondes employées ont fortement atténuées en traversant les murs. Sa précision est de plus souvent trop faible pour qu'un robot terrestre puisse utiliser ces informations seules. En pratique, il est souvent couplé à un système inertiel qui permet de palier aux pertes du signal GPS

et il ne remplace de toute façon pas les capteurs du robot qui lui permettent de percevoir son environnement immédiat.

Mis à part son utilisation pour la localisation, ce système est aussi utilisé pour l'aide à la navigation d'avions et de bateaux lors de l'approche d'aéroports ou de ports, ainsi que dans la surveillance de camionnettes de livraison et de véhicules d'urgence, en vue de choisir au mieux leurs itinéraires.

2.2.2.2 Les systèmes de vision

L'utilisation d'une caméra pour percevoir l'environnement est une méthode attractive car elle semble proche des méthodes utilisées par les humains. Le traitement des données volumineuses et complexes fournies par ces capteurs reste cependant difficile à l'heure actuelle, même si cela reste une voie de recherche très explorée.

2.2.2.2.1 Caméras simples

Une caméra standard peut être utilisée de différentes manières. Elle peut être utilisée pour détecter des amers visuels (des points particuliers qui servent de repères, tels que des portes ou des affiches) à partir desquels il sera possible de calculer la position du robot. Si ces amers sont simplement ponctuels, ou de petites tailles, il sera en général simplement possible d'estimer leur direction. Dans le cas où les amers sont des objets connus en deux ou trois dimensions, il sera en général possible d'estimer complètement la position du robot par rapport à la leur. Elle peut également être utilisée pour détecter des "guides" de navigation pour le robot, tels que des routes ou des couloirs.

Il est également possible d'utiliser globalement une image pour caractériser une position ou un point de vue dans l'environnement. Il faudra alors comparer cette image aux nouvelles images acquises par le robot pour savoir si le robot est revenu à cette position. Cette comparaison peut faire appel à de très nombreuses techniques, notamment à celles utilisées dans le domaine de l'indexation d'image. Lorsque le robot est en mouvement, il est également possible de tirer parti du *flot optique* (le mouvement apparent des objets dans l'image), afin d'avoir une estimation de la distance des objets. En effet, les objets les plus proches ont un déplacement apparent plus important que les objets lointains. Cette méthode permet notamment

de réaliser un évitement d'obstacles ou de réaliser une reconstruction tridimensionnelle de l'environnement.

2.2.2.2.2 Caméras stéréoscopiques

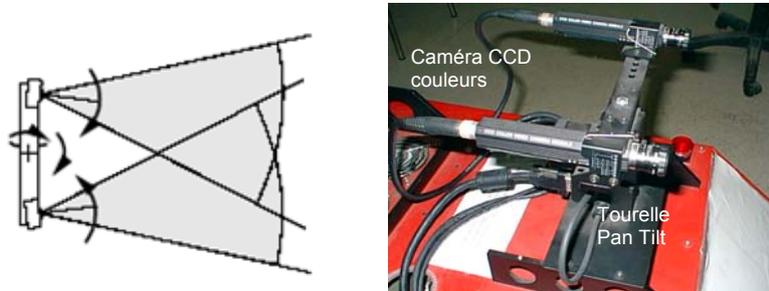


Figure 2. 7 Un exemple de banc stéréoscopique

Lorsque l'on dispose de deux caméras observant la même partie de l'environnement à partir de deux points de vue différents, il est possible d'estimer la distance des objets et d'avoir ainsi une image de profondeur (Figure 2.7), qui peut être utilisée pour l'évitement d'obstacles ou la cartographie. Cette méthode suppose toutefois un minimum d'éléments saillants dans l'environnement (ou un minimum de textures) et peut être limitée, par exemple dans un environnement dont les murs sont peints de couleurs uniformes. La qualité de la reconstruction risque également de dépendre fortement des conditions de luminosité. La résolution et l'écartement des deux caméras imposent également les profondeurs minimums et maximums qui peuvent être perçues, ce qui peut être limitatif pour la vitesse de déplacement du robot.

2.2.2.2.3 Caméras panoramiques

Les caméras panoramiques (catadioptriques) sont constituées d'une caméra standard pointant vers un miroir de révolution (par exemple un simple cône, ou un profil plus complexe qui peut s'adapter à la résolution exacte que l'on veut obtenir sur le panorama) (figure 2.8). L'image recueillie permet d'avoir une vision de l'environnement sur 360 degrés autour de la camera. Le secteur angulaire vertical

observé dépend de la forme du miroir et peut être adapté aux besoins de chaque application.

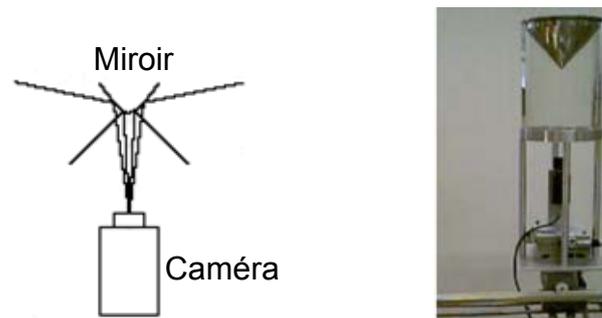


Figure 2.8 Principe des caméras panoramiques

Ce type de caméra est très pratique aussi bien pour la localisation que pour la navigation car une image prise par une caméra panoramique orientée verticalement permet de caractériser une position, indépendamment de la direction du robot. En effet, pour une position donnée et pour deux orientations différentes, la même image sera formée par la caméra, à une rotation autour du centre près, tandis que pour une caméra standard, orientée horizontalement, la scène serait différente.

Ces caméras sont donc très pratiques lorsque l'on caractérise une position de manière globale, mais peuvent aussi être utilisées pour détecter des amers ou pour estimer le flux optique. Dans ce cas, toutefois, comme la géométrie de l'image formée est relativement complexe et comme la résolution obtenue varie énormément selon la direction observée, les algorithmes doivent être adaptés, ce qui pose un certain nombre de problèmes.

Concernant le flux optique, cependant, les caméras panoramiques possèdent l'avantage de contenir toujours le point d'expansion et le point de contraction dans l'image, ce qui rend l'estimation du mouvement beaucoup plus aisée.

Les systèmes de vision peuvent être utilisés pour la modélisation en trois dimensions de l'environnement en s'aidant de mesures de distances de capteurs à ultrasons, ou d'un télémètre laser par exemple. La localisation de robots mobiles est un domaine de recherche dans lequel ce capteur est très employé. N'empêche que les données issues de l'image peuvent être fusionnées avec celles fournies par un odomètre ou avec la donnée d'un laser

2.2.2.3 Autres capteurs

2.2.2.3.1 Les capteurs tactiles

Les robots peuvent être équipés de capteurs tactiles, qui sont le plus souvent utilisés pour des arrêts d'urgence lorsqu'ils rencontrent un obstacle qui n'avait pas été détecté par le reste du système de perception. Ces capteurs peuvent être de simples contacteurs répartis sur le pourtour du robot. Ils ne détectent alors le contact qu'au dernier moment.

2.2.2.3.2 Les balises

Dans certaines applications, il est également possible d'utiliser des balises dont on connaît la position, et qui pourront être facilement détectées par le robot, afin de faciliter sa localisation. Des techniques très diverses peuvent être utilisées pour ces balises. On peut par exemple utiliser un signal radio, émis de manière omnidirectionnelle par la balise. Le robot sera alors équipé d'une antenne directionnelle qui lui permettra de détecter la direction des différentes balises, afin de déduire sa position par triangulation. On peut également utiliser des codes couleurs ou des codes à barres qui pourront être détectés par une caméra.

2.3 Modélisation des mesures

2.3.1 Grilles d'occupation

Ces méthodes consistent à représenter les mesures sous forme d'un ensemble de cellules. Chaque cellule possède plusieurs attributs exprimant chacun une propriété de la zone correspondante, notamment son occupation. Pour des scènes d'intérieur, où le sol est plat, la plupart des chercheurs décrivent l'état de chaque cellule par un attribut unique qui est la probabilité d'occupation. On parle alors de *grille d'occupation*. Une cellule est dite *libre* si sa probabilité d'occupation est inférieure à un seuil choisi et *occupée* dans le cas inverse.

Moravec, *Elfes* et *Matthies* ont été parmi les premiers à utiliser le principe des grilles d'incertitude (ou grille d'occupation) [25, 26]. L'objectif de leurs travaux est de construire de manière autonome la carte de l'environnement d'un robot mobile. Pour cela le robot évolue dans un environnement inconnu non structuré et s'y déplace en

évitant les obstacles, sans disposer d'aucun modèle a priori du lieu où il se trouve. Il doit construire une carte du lieu à partir des seules informations données par une ceinture de 24 capteurs à ultrasons de type polaroid montée sur le robot. Chaque mesure de distance fournie par cette ceinture est interprétée comme définissant un volume d'espace probablement vide et un volume probablement occupé dans le cône d'émission de chaque capteur. Cette information d'occupation est une densité de probabilité que l'on projette verticalement sur une carte à deux dimensions, où l'on voit apparaître des aires élémentaires vides, occupées et inexplorées. Le profil de probabilité utilisé permet de générer deux grilles d'occupations l'une représentant les régions certainement vides, l'autre représentant les régions certainement occupées.

Au fur et à mesure que le robot se déplace, de nouvelles mesures sont intégrées dans ces cartes affinant ainsi les connaissances des zones déjà explorées, et découvrant les autres. La mise à jour des cartes s'effectue en utilisant des formules d'addition de probabilités [20]. Ces formules permettent de calculer les nouvelles probabilités d'occupation et de non occupation à partir des probabilités mémorisées dans la cellule et des nouvelles probabilités fournies par la nouvelle mesure. Deux zones vides superposées se renforcent mutuellement, deux zones occupées également. Une zone occupée et une zone vide superposée donnent une zone soit occupée soit vide suivant la plus forte des probabilités. Les zones vides mettent en évidence les limites des régions occupées. A chaque nouvelle mesure l'incertitude sur les positions des obstacles diminue et la précision de la carte augmente.

Cette méthode peut être utilisée à partir des mesures issues de plusieurs types de capteur [27]. Chaque capteur construit sa carte locale qui va ensuite être intégrée dans une carte locale du robot servant à établir la carte globale de l'environnement (Figure 2.9).

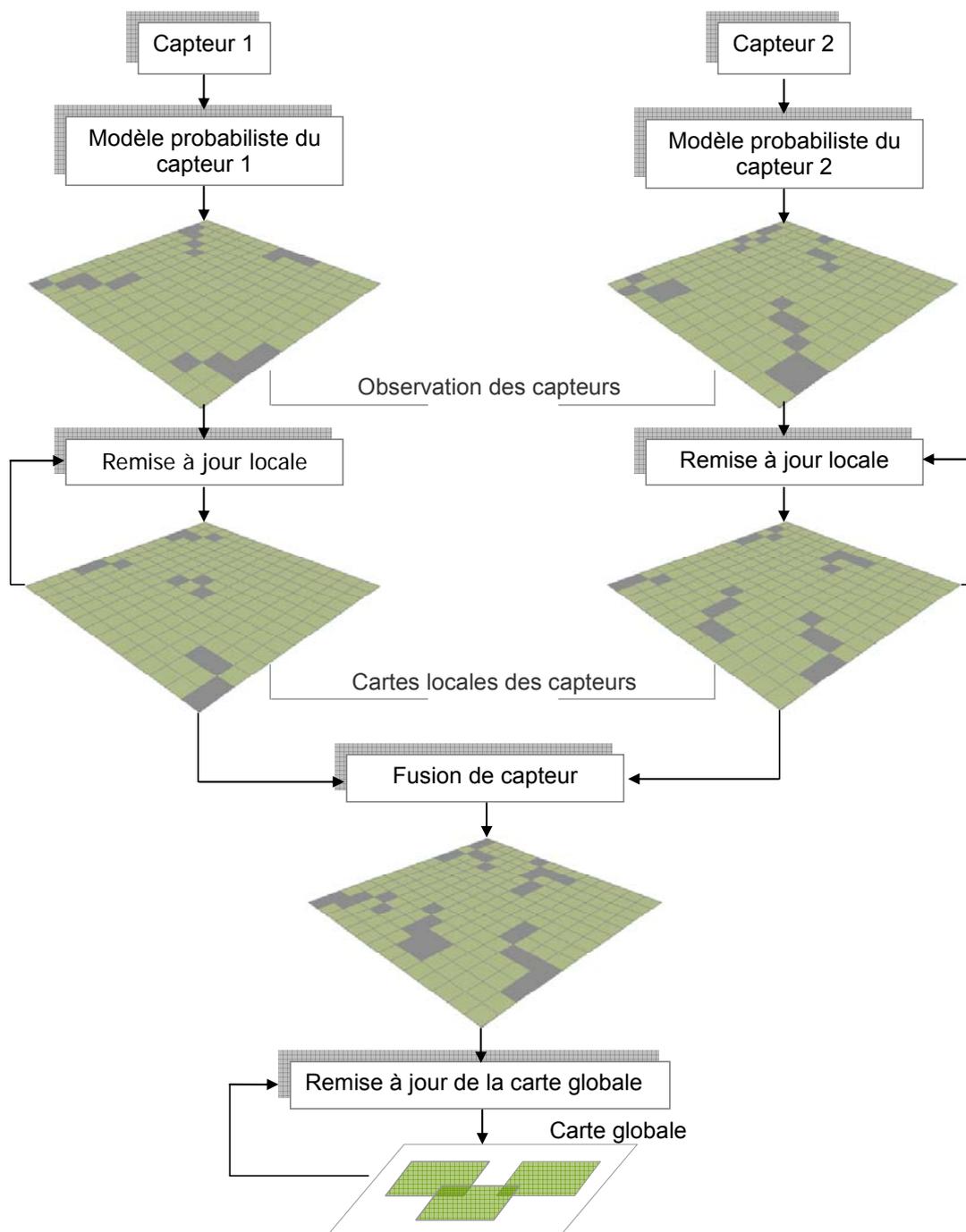


Figure 2.9 Construction d'une carte par fusion de données de plusieurs capteurs (d'après [27])

Un inconvénient de ce principe de mise à jour de la probabilité d'occupation d'une cellule est l'élimination des mesures contradictoires. Dans d'autres travaux [28], un test permet de gérer les mesures contradictoires en les traitant différemment. Quand une cellule libre à l'instant t est perçue comme étant occupée à l'instant $t+l$, la probabilité d'occupation de la cellule est incrémentée.

Ce principe favorise la représentation des obstacles, ce qui est appréciable pour la sécurité du robot, mais cette méthode différencie l'ordre d'arrivée des

mesures. En effet, une cellule d'abord libre puis perçue comme étant occupée n'est pas traitée de la même manière qu'une cellule occupée qui par la suite est perçue comme étant libre. Ce qui peut être gênant puisque pour le même environnement, en reprenant les mêmes configurations du robot en sens inverse, le résultat de la carte sera différent.

Certains travaux utilisent ces grilles d'occupation en remplaçant le seuil de décision unique par deux seuils ; cela permet de donner une valeur neutre (ni libre, ni occupée) aux cellules ayant une valeur proche du seuil unique [29].

D'autres auteurs [21] utilisent une modélisation différente. Chaque cellule de la carte contient deux probabilités : la probabilité P_L que la cellule soit libre et la probabilité P_O que la cellule soit occupée.

La mesure d'un capteur à ultrasons peut être représentée par un cône composé d'un secteur représentant l'espace libre et d'un arc indiquant la présence d'un obstacle. Chaque probabilité d'une cellule est définie par une valeur inversement proportionnelle au nombre de cases recouvertes par le cône. Par exemple s'il y a n cellules sur l'arc et m cellules dans le secteur, les probabilités de chaque cellule de

$$\text{l'arc sont : } \begin{cases} P_L = 0 \\ P_O = \frac{1}{n} \end{cases} \text{ et les probabilités de chaque cellule du secteur sont : } \begin{cases} P_L = \frac{1}{m} \\ P_O = 0 \end{cases}$$

Pour pouvoir ensuite utiliser des règles de probabilité pour la mise à jour de la carte, une cellule doit avoir la somme de ses probabilités égale à 1 ce qui n'est pas le cas puisque pour une cellule $P_L + P_O \neq 1$. On a recours à l'interprétation des données ultrasonores par la théorie de *Dempster-Shafer* [30]. Cette théorie ajoute deux probabilités : la probabilité P_ϕ que la cellule ne soit ni vide ni occupée et la probabilité P_{LO} que la cellule soit vide ou occupée. Par ailleurs, pour chaque cellule de la grille il est impossible qu'une cellule ne soit ni vide ni occupée et donc $P_\phi = 0$, de plus $\sum P = P_\phi + P_L + P_O + P_{LO} = 1$. Dans ces conditions il est seulement nécessaire de déterminer P_L , et P_O puis P_{LO} est calculée en utilisant la relation $P_{LO} = 1 - P_L - P_O$. Pour mettre ensuite à jour les probabilités d'une cellule lors d'une nouvelle mesure, les nouvelles valeurs sont calculées en fonction des trois valeurs mémorisées (cellule) et des trois nouvelles données (mesure) par les règles de combinaison de *Dempster*.

2.3.2 Modèles géométriques

Les modèles géométriques sont des entités telles que des points, des segments de droite, des rectangles, des polygones, etc. obtenues à partir des mesures et constituant une représentation des différents éléments de l'environnement. La représentation la plus utilisée pour la localisation de robots mobiles est celle par segments de droite, que l'on obtient en effectuant un prétraitement des points de mesure qui consiste à les filtrer et à les regrouper.

Ces modèles sont obtenus soit à partir des mesures de capteurs télémétriques qui donnent la limite des obstacles, soit des modèles établis à partir d'une image d'un capteur de vision qui donne une autre perception de l'environnement.

2.3.2.1 Par télémétrie

Il existe différents filtrages et regroupements de points de mesure. Par exemple la méthode proposée par [31] consiste à prendre les mesures d'un capteur à ultrasons tournant une par une, et à créer un segment dès que trois mesures sont suffisamment proches et alignées. Pour qu'un nouveau point puisse appartenir à un segment existant, il faut que la distance du point au segment soit plus petite que l'incertitude sur la mesure.

Au lieu de regrouper les points de mesure suivant leur alignement pour avoir une représentation par segments, il est possible de les réunir en fonction de la distance entre le capteur (tournant) et les points, ce qui constitue une région conique appelée RCD (Regions of Constant Depth) [13]. Ainsi, quand plusieurs mesures consécutives renvoient une distance semblable, elles sont regroupées en un arc défini par la distance δ entre le capteur et l'obstacle, l'orientation β de l'axe central du RCD et l'ouverture angulaire η (Figure 2.10).

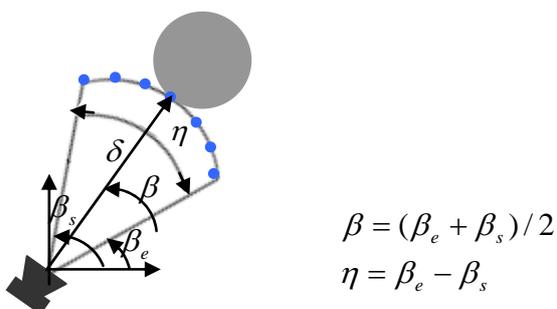


Figure 2.10 Création d'une RCD (Regions of Constant Depth)

2.3.2.2 par vision

Pour obtenir une représentation géométrique de l'environnement à partir d'un capteur de vision il faut réaliser un prétraitement de l'image. Ce traitement consiste à extraire les contours des objets présents dans l'image grâce à un filtre de *sobel* par exemple utilisé dans [32] ou encore un filtre de *Deriche* utilisé dans [33]. Dans la plupart des travaux de localisation de robots mobiles, des algorithmes utilisant uniquement les contours verticaux pour établir une liste de segment, ainsi dans les travaux de recherche sur l'estimation de l'orientation du robot par vision monoculaire [1], les segments sont obtenus en recherchant les contours verticaux ayant un nombre de pixels consécutif supérieur à un seuil fixé. En travaillant avec les contours extraits de deux images (par un capteur de vision stéréoscopique), il est possible d'obtenir une carte 3D composée de segments [23].

2.4 Modélisation de l'environnement

2.4.1 Les Cartes topologiques

Les cartes topologiques permettent de représenter l'environnement du robot sous forme de graphe. Les noeuds du graphe correspondent à des lieux, c'est-à-dire des positions que le robot peut atteindre. Les arêtes liant les noeuds marquent la possibilité pour le robot de passer directement d'un lieu à un autre et mémorisent en général la manière de réaliser ce passage (figure 2.11).

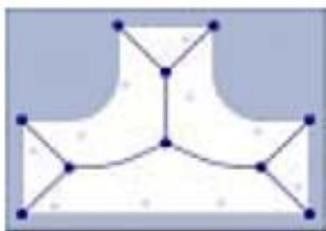


Figure 2.11 Un exemple de la modélisation topologique [d'après 34]

La détection et la mémorisation des lieux reposent en général sur deux procédures qui utilisent les perceptions. La première permet simplement de comparer deux perceptions et donc de reconnaître un lieu de la carte ou de détecter un lieu nouveau en faisant appel aux données proprioceptives en plus des

perceptions. La seconde procédure permet de mémoriser un nouveau lieu ou d'adapter la définition d'un lieu lors des passages successifs du robot en ce lieu.

Les données mémorisées dans les arêtes du graphe sur les relations de voisinage entre lieux proviennent, pour leur part, des données proprioceptives. Cela est caractéristique des cartes topologiques, dans lesquelles les perceptions ne sont en général pas utilisées pour estimer les positions relatives des lieux visités, mais seulement pour reconnaître un lieu. Ces données peuvent être des informations sur les positions relatives des noeuds, ou des informations sur les actions à effectuer pour parcourir cette arête.

Un avantage important des cartes topologiques est qu'elles ne requièrent pas de modèle métrique des capteurs pour fusionner les données proprioceptives et les perceptions au sein d'une représentation unifiée de l'environnement. Cela est avantageux car le fait de ne pas fusionner les deux sources d'informations permet de séparer les influences des erreurs correspondantes.

La carte est très proche des données brutes perçues par le robot, il est en général assez simple de comparer et de mémoriser des lieux de l'environnement. Cette proximité avec les données brutes conduit en général la représentation topologique à utiliser beaucoup moins de concepts de haut niveau que les représentations métriques, tels que des objets ou des obstacles.

La discrétisation de l'environnement correspondant au choix des lieux représentés dans la carte est un autre point fort des cartes topologiques. Cet avantage est encore plus important lorsque les lieux représentés dans la carte correspondent à des structures humaines telles que les portes, les couloirs ou les pièces. La discrétisation permet alors de décrire et de résoudre les problèmes de manière naturelle pour les humains, par exemple en donnant l'ordre au robot d'aller à une salle portant un numéro, plutôt que d'aller à une position définie par des coordonnées (x,y) .

Mais cette méthode présente aussi plusieurs inconvénients, tel qu'on la déjà mentionné l'utilisation directe des perceptions sans modèle métrique empêche d'estimer ces données pour des positions non visitées. En conséquence, les cartes topologiques nécessitent en général une exploration très complète de l'environnement pour le représenter avec précision. Cela peut être un inconvénient

dans le cas où les lieux représentés ont une grande densité spatiale car l'exploration complète de l'environnement demandera un temps important.

L'absence de modèle métrique des capteurs peut poser des problèmes pour la réutilisation de la carte, et donc ne permettra pas d'adapter la carte à un robot avec des capteurs légèrement différents.

La reconnaissance des lieux de l'environnement peut également être difficile dans le cas de capteurs très bruités, ou d'environnements très dynamiques. Elle est, de plus, très sensible au problème de *perceptual aliasing* (dans le cas où le système de perception donne une même mesure pour deux amers différents). Ces difficultés conduisent à des problèmes de fausse reconnaissance, c'est-à-dire à la reconnaissance d'un lieu donné alors que le robot se trouve dans un autre lieu. À leur tour, ces fausses reconnaissances conduisent à une mauvaise topologie de la carte. Il devient alors très difficile d'estimer correctement la position du robot au sein de cette carte et de lui ajouter de nouvelles informations sans erreurs.

2.4.2 Les Cartes métriques

Dans une carte métrique, l'environnement est représenté par un ensemble d'objets auxquels sont associées des positions dans un espace métrique, généralement en deux dimensions. Cet espace est, la plupart du temps, celui dans lequel s'exprime la position du robot estimée par les données proprioceptives. Les perceptions permettent, en utilisant un modèle métrique des capteurs, de détecter ces objets et d'estimer leur position par rapport au robot. La position de ces objets dans l'environnement est alors calculée en utilisant la position estimée du robot. La fusion des deux sources d'information au sein d'un même cadre de représentation est caractéristique des cartes métriques.

L'avantage principal des cartes métriques est de permettre de représenter l'ensemble de l'environnement, et non un petit sous-ensemble de lieux comme le font les cartes topologiques. Cette représentation complète permet ainsi d'estimer avec précision et de manière continue la position du robot sur l'ensemble de son environnement. De plus, cette représentation complète ne se limite pas aux positions physiquement explorées, mais s'étend à toutes les zones que le robot a pu percevoir depuis les lieux qu'il a visités. Cette propriété peut permettre la construction d'une carte plus exhaustive de l'environnement en un temps plus court.

Un autre avantage des cartes métriques est lié au fait que la position du robot est définie de manière non ambiguë par ses coordonnées au sein de l'espace dans lequel la carte est représentée. Il s'ensuit une utilisation simple et directe de toutes les informations métriques fournies par les données proprioceptives ou les perceptions. Cela est un avantage par rapport aux cartes topologiques où les positions possibles du robot sont limitées aux noeuds présents dans la carte et sont donc relativement imprécises.

Contrairement aux cartes topologiques, une telle représentation permet une réutilisation plus facile de la carte sur des robots différents équipés de capteurs différents, puisque l'adaptation se fait simplement au niveau du modèle du capteur sans modification de la carte elle-même.

L'inconvénient majeur des cartes métriques est que lors de leurs utilisations, les données proprioceptives ont en général une importance supérieure à celle qu'elles ont dans l'utilisation d'une carte topologique. Par conséquent, une odométrie plus fiable peut être requise, sinon ces données odométriques peuvent être corrigées par une méthode d'estimation tel que le filtre de Kalman. Un modèle métrique des capteurs peut être difficile à obtenir. Les problèmes liés au bruit des capteurs et à la difficulté de modéliser de manière fiable leur relation avec l'environnement constituent donc un point faible des cartes métriques.

L'environnement du robot mobile est construit à partir des modèles métriques des capteurs, ce qui mène à avoir deux méthodes métriques de modélisation de l'environnement ; les modèles géométriques et les modèles basés sur les grilles d'occupations.

Pour mémoriser les informations sous forme de carte. La première méthode s'attache à représenter directement l'espace libre accessible au robot et les zones d'obstacles qu'il ne peut pas franchir, sans avoir recours à l'identification d'objets individuels. La seconde méthode consiste à extraire explicitement des objets des perceptions et à les enregistrer dans la carte avec leur position estimée. Les objets peuvent être de types très variés et situés à différents niveaux d'abstraction.

2.4.2.3 Grille d'occupation

La modélisation par grille d'occupation est une approche qui permet de modéliser l'environnement en grille ou en ensemble de cellules. La présentation par

grille utilise une matrice pour stocker les différentes informations concernant les cellules de la carte. La taille de la carte est définie par la taille de la matrice ainsi que la taille de la surface incluse dans cette carte. Chaque élément de la matrice porte différentes informations sur la cellule correspondante de l'espace physique, ces informations consistent à ce que la cellule soit occupée par un objet ou non (figure2.12).

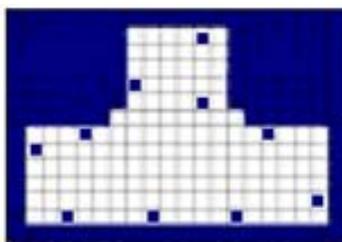


Figure 2.12 Un exemple de modélisation par grille d'occupation [d'après 34]

La mise à jour de la carte se fait au cours du déplacement du robot dans son environnement. Pour cela le robot utilise ces capteurs pour prendre des mesures, il peut utiliser par exemple un télémètre laser. Les mesures prises par le laser sont modélisées en une position dans la matrice relative à la position du robot. Tout en se déplaçant, de nouvelles mesures sont prises par le robot mobile, la certitude augmente dans les éléments ayant une grande probabilité de contenir un obstacle. La surface entre le robot et les mesures est représentée comme étant libre.

Une fois la modélisation terminée, la carte contiendra toutes les informations concernant l'environnement du robot. Les séquences d'images présentées dans la figure 2.13 illustrent la mise à jour de la carte d'environnement basée sur les grilles d'occupation d'un robot mobile. La première image de la figure 2.13 représente l'environnement du robot divisé en cellules, montrant comment les objets doivent occuper les différentes cellules de la grille, les autres images montrent ce qui est visible au robot et comment les airs inconnus sont explorés grâce au capteur laser qui prend des mesures à partir de son scanne.

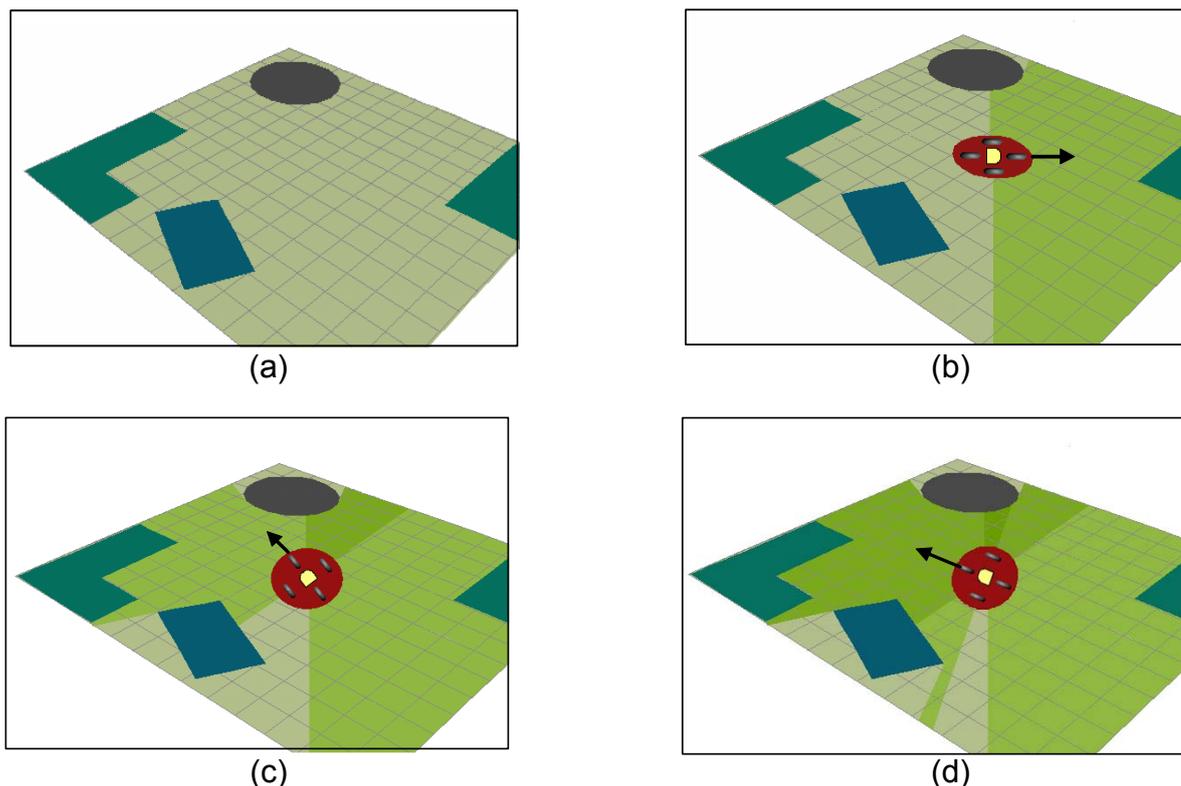


Figure 2.13 Quatre étapes montrant comment l'environnement est graduellement exploré au cours du déplacement du robot.

Le champ de vue du laser (180 degrés) et en vert, les zones les plus foncées représentent les surfaces de chevauchement entre chaque scanne, où la certitude de cellule est plus grande.

Après la mise à jour, les cellules contiendront l'information d'être libre, occupée ou non explorée. C'est bien pour cette raison que ces cartes sont connues sous l'appellation grilles d'occupation.

L'avantage de cette représentation est le fait que c'est une méthode facile à implémenter et extrêmement robuste selon Thrun et Brenstein [35], c'est ce qui la rend très utilisée pour modéliser un environnement. Les grilles d'occupation permettent de représenter de grandes densités d'informations et sont adaptées à des environnements de forme quelconque. Elles fournissent en outre une estimation statistique de la confiance dans les données, de plus, contrairement aux représentations topologiques, elles fournissent des informations «de plein et de vide» puisqu'elles indiquent directement où sont placées les obstacles. Concernant leurs méthodes de constructions, les grilles d'occupation sont plutôt économiques en ressources de calcul : leur mise à jour s'avère rapide et facile. Enfin, les grilles d'occupation permettent une intégration aisée des divers types de capteurs :

l'algorithme de fusion est immédiat. Elles sont de plus bien adaptées à des capteurs bruités (vision stéréoscopique, sonars, radars, etc.) où les primitives géométriques sont assez difficiles à extraire du fait de l'incertitude sur les données [36]. Toutefois, seules certaines approches permettent de bien prendre en compte les modèles de capteurs, comme chez Thrun [37].

L'inconvénient est que si la carte demande une grande résolution à travers une grande surface, la matrice devient grande, et ça conduit à des problèmes si on utilise un système avec mémoire ou capacité de calcul restreinte. De plus, la finesse de la discrétisation étant prédéfinie, les grilles d'occupation ne sont pas capables de s'adapter automatiquement à la densité ou à la taille des obstacles. Cela peut s'avérer un problème pour certains algorithmes de navigation mais il existe des solutions tel qu'agrandir l'espace de configuration des obstacles. En revanche cela se prête bien au algorithme de localisation, dont le plus grand souci est d'estimer la position du robot capable d'éviter les obstacles comme une moindre tâche.

2.4.2.4 Modélisation géométrique

Au lieu d'utiliser une matrice, une liste ou un vecteur de tous les objets de l'environnement est utilisé. Cette liste contient des informations sur les objets tel que leurs natures, position et orientation.

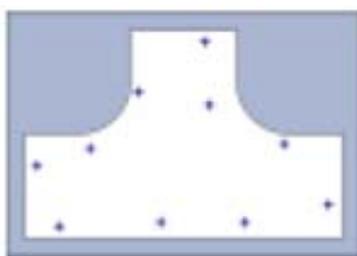


Figure 2.14 Un exemple de modélisation géométrique [d'après 34]

En se déplaçant, le robot prend des mesures de son environnement en utilisant son système de perception, afin d'extraire des informations sur l'éventualité de la présence d'une primitive, c'est ce qu'on appelle ; " extraction de primitive". Ces primitives peuvent être de différentes formes géométriques tel que les points, les lignes, les polygones, etc.

Les méthodes employées dépendent de l'équipement sensoriel dont il est muni le robot et le type de primitives détectées. Un robot disposant d'un capteur laser peut mettre en œuvre la transformée de Hough pour l'extraction des lignes de droite, ou la méthode d'estimation par les moindres carrés des points du scanne. Chaque primitive trouvée est ajoutée à la liste des primitives, mais elle doit être comparée à la précédente, dans le cas où elle existe déjà elle sera ignorée.

Différentes primitives dans l'environnement sont détectées et stockées dans une liste, cette liste est mise à jour tant que le robot se déplace et prend de nouvelles mesures. Les informations concernant les primitives ne sont pas fixées et peuvent changer plusieurs fois pour les mettre à jour, c'est ce qui rend la carte géométrique plus flexible que la carte à grille d'occupation, mais en revanche elle est plus robuste.

Une fois la carte géométrique construite, elle peut être utilisée par le robot pour sa localisation dans l'environnement. Cette méthode de modélisation est largement utilisée dans les environnements d'intérieur structurés.

Cette solution est avantageuse pour sa grande résolution en plus du fait qu'elle ne nécessite pas une grande capacité mémoire. La position des objets peut être stockée avec une grande précision du moment qu'on a besoin juste des positions x et y des objets. Quoique l'espace mémoire requis pour le stockage augmente avec la taille de l'environnement du robot.

L'inconvénient est que la bonne détection des primitives est une tâche difficile pour un système autonome, même si des primitives simples peuvent être détectées, il est difficile d'en faire la distinction. C'est une méthode compliquée quant à la manière dont les primitives dans la carte sont connectées et comment le robot se déplace par rapport à ces primitives.

La modélisation géométrique est plus difficile à implémenter que la modélisation par grille d'occupation, car cette dernière a pour avantage d'utiliser directement les valeurs des capteurs de distance afin de mettre à jour les probabilités d'occupation des cellules. Elle permet donc de supprimer la phase d'extraction d'objet qui est souvent coûteuse en temps de calcul et soumise à une forte incertitude.

2.5 Conclusion

Le paradigme de modélisation de l'environnement nécessite de s'intéresser obligatoirement aux éléments de la chaîne de perception qui sont les capteurs et les méthodes de modélisation. C'est ce que nous avons fait de façon la plus synthétique possible dans ce chapitre. Les constats pouvant être dégagés sont multiples. Par rapport aux systèmes de perception utilisés en robotique mobile, deux familles de capteurs peuvent être utilisées. Nous avons pu constater que celles-ci sont plus complémentaires que concurrentes. C'est pour cette raison qu'un système de localisation sera généralement basé sur l'exploitation des informations sensorielles émanant de ces deux catégories de capteurs. Nous pouvons étendre cette remarque à la problématique spécifique qu'est la perception du milieu d'évolution du robot : l'emploi d'un unique capteur extéroceptif pour cette mission sera généralement insuffisant. Ainsi, l'association de plusieurs capteurs extéroceptifs, permettra d'obtenir un modèle sensoriel robuste et hautement descriptif. Ce modèle servira à construire la carte d'environnement du robot. Quant aux capteurs proprioceptifs, ils serviront à estimer la position du robot dans cet environnement.

L'état de l'art présenté laisse apparaître plusieurs méthodes de modélisation d'environnement, chacune ayant des avantages et inconvénients. Pour notre travail, notre choix s'est porté sur la modélisation par grille d'occupation, du moment que cette méthode est facile et assez robuste à implémenter en comparaison avec la modélisation géométrique, qui nécessite une phase d'extraction de primitives. Vu que l'un de nos objectifs est la fusion des données issues des différents capteurs (ultrasons, infrarouges, laser), cette méthode est la mieux adaptée pour intégrer les données issues de tous les capteurs embarqués. Ce qui est une tâche rude pour la modélisation topologique. Mais ce qui peut être un inconvénient de la modélisation par grille d'occupation est la nécessité d'une grande résolution en présence d'une grande surface. Cela aurait posé un problème si le robot mobile disposait d'un système avec mémoire ou capacité de calcul restreinte.

CHAPITRE 3 LOCALISATION

3.1 Introduction

La localisation est la détermination de la situation, position et orientation, du robot mobile par rapport à un repère de référence. La localisation est absolue si le repère est fixe, lié à son environnement. La localisation est relative lorsque le repère est lié à une position précédente du robot mobile. Nous pouvons aussi parler de localisation statique lorsque le calcul de position s'effectue à l'arrêt ou de localisation dynamique lorsque celle-ci est évaluée durant le mouvement.

Par analogie avec la classification des capteurs, nous pouvons distinguer deux grands systèmes de localisation :

- La localisation à l'estime (ou Dead Reckoning) ou relative, basée sur l'utilisation des capteurs proprioceptifs.
- La localisation absolue, basée sur l'utilisation des capteurs extéroceptifs.
- La localisation dite *hybride*, basée sur l'utilisation conjointe des capteurs proprioceptifs et des capteurs extéroceptifs.

En dépit des capteurs utilisés par le robot pour sa localisation. Les méthodes utilisées peuvent être soit locales ou globales. Les méthodes locales sont celles qui estiment la position actuelle du robot par rapport à sa position précédente, c'est le cas pour la localisation relative et le suivi de position qui est une méthode hybride qui a la capacité de mettre à jour une estimation existante de la position au vu des données proprioceptives et de perceptions nouvellement acquises. Toutefois, cette méthode est problématique car elle repose sur une estimation initiale de la position qui doit souvent être fournie par une source extérieure.

Tandis que les méthodes de localisation globales sont plus générales et permettent de retrouver la position du robot sans qu'aucune estimation initiale ne soit fournie. Cette capacité est très importante du point de vue de l'autonomie, car elle permet au robot de trouver sa position initiale, dans toutes les conditions, sans intervention extérieure. Elle permet, par exemple, de couper l'alimentation d'un robot à des fins

de maintenance, puis de remettre ce robot dans une position quelconque de l'environnement sans se soucier d'initialiser correctement son estimation de la position. Parmi les méthodes globales, y a celles qui font appel uniquement aux perceptions disponibles en une position donnée comme pour certaines méthodes de localisation absolue, et celles qui fusionnent au contraire les informations proprioceptives et les perceptions (méthodes hybride), comme le fait le suivi de position. Toutefois, au lieu de restreindre la recherche par une estimation précédente de la position, elle estime parmi toutes les positions possibles au sein de la carte celle qui correspond le mieux aux données présentes et passées recueillies par le robot. Les méthodes de cette catégorie reposent, d'une façon ou d'une autre, sur le suivi de plusieurs hypothèses de position.

On va tirer profit des avantages et inconvénients de toutes ces approches pour désigner l'approche la plus appropriée pour notre travail.

3.2 Les différentes méthodes de localisation

3.2.1 Localisation relative

La localisation relative, aussi appelée localisation à l'estime (ou Dead reckoning) consiste à déterminer la position du robot mobile par intégration des informations fournies par des capteurs proprioceptifs. L'intégration se fait par rapport au point de départ du robot. Ces données peuvent être des informations sur les variations de position ou mesures de vitesse fournies par des odomètres, ou des mesures d'accélération fournies par des accéléromètres ou gyromètres.

La technique de localisation basée sur les odomètres est peu coûteuse et simple à mettre en œuvre, ces caractéristiques en font un système de localisation couramment utilisé en robotique mobile. En outre, l'odométrie permet d'obtenir une estimation de la position à une cadence relativement élevée. L'inconvénient est qu'elle a une précision très médiocre sur des distances importantes, à cause des erreurs cumulatives. L'odométrie permet de déterminer la position du robot mobile par intégration de ses déplacements élémentaires, et ce, par rapport à un repère lié à sa configuration initiale.

L'algorithme de localisation est basé sur le comptage des impulsions générées par des codeurs durant une période d'échantillonnage connue. Connaissant le rayon des

roues et la distance entre chaque roue, il est possible d'exprimer de manière récursive la position du robot mobile.

La technique inertielle (accéléromètre, gyromètre) plus coûteuse constitue le haut de gamme des systèmes de localisation à l'estime. La connaissance de la position référencée à l'origine du déplacement nécessite une double intégration de l'accélération. Ces calculs entraînent une inévitable accumulation d'erreurs qui constitue une dérive d'estimation dans le temps. Un recalage est alors indispensable. Un radar à effet Doppler pourrait être aussi utilisé pour l'estimation de la position, par intégration de la vitesse mesurée, à savoir la vitesse linéaire du robot par rapport au sol (la réflexion d'une onde émise sur le sol subit une variation de fréquence proportionnelle à la vitesse). Ce dispositif est particulièrement bien adapté dans le cas où il existe des glissements importants ou bien quand la surface de roulement est bosselée (une distance mesurée à l'aide d'une roue est alors surestimée).

La technique de localisation à l'estime présente l'immense avantage d'être indépendante de l'environnement. Les seules erreurs qu'elle peut générer sont celles dues à son mode de fonctionnement interne. Par contre, l'inconvénient majeur est l'accumulation d'erreurs due aux intégrations d'erreur de mesure et de modèle au fur et à mesure du temps et/ou de la distance parcourue.

3.2.2 Localisation absolue

La localisation absolue est une technique qui permet à un robot de se repérer directement dans son milieu d'évolution, que ce soit en environnement d'extérieur (mer, espace, terre, etc.), ou en environnement d'intérieur (ateliers, immeubles, centrales nucléaires, etc.). Ces méthodes de localisation sont basées sur l'utilisation de capteurs extéroceptifs.

Elle nécessite toujours une représentation de l'environnement. Le robot possède donc une banque de données regroupant les éléments caractéristiques de son milieu d'évolution (balises par exemple), et pour sa localisation absolue, il doit déduire de la perception de ces éléments caractéristiques sa position dans son environnement. L'avantage principal de cette technique est d'obtenir des localisations qui ne dérivent pas dans le temps. La précision dépend des capteurs et de la configuration des amers par rapport au système. Par contre, le temps de traitement des informations

des capteurs est, en général, plus long qu'avec des capteurs proprioceptifs, ce qui engendre des difficultés lorsque le robot mobile est en mouvement.

Ces méthodes peuvent être utilisées pour estimer la position d'un robot à l'aide des seules perceptions. Dans le cas où l'environnement est exempt de *perceptual aliasing*, cette étape suffit à déterminer la position du robot de manière unique. Cette méthode est alors une méthode de localisation globale. Dans le cas où le *perceptual aliasing* est présent, ces méthodes sont également utilisées mais elles serviront à repérer plusieurs positions possibles pour le robot au sein de l'environnement. Les méthodes de suivi d'une hypothèse unique ou de suivi de plusieurs hypothèses qui seront présentées dans les paragraphes suivants doivent alors être utilisées pour sélectionner la position correcte.

3.2.2.1 Localisation par balises

La localisation à l'aide de balises peut se faire soit par la mesure des distances par télémétrie séparant le robot mobile des différentes balises basée sur la méthode de trilatération (ou multilatération), soit par la mesure de l'angle sous lequel le robot perçoit chaque balise au moyen de la goniométrie, basée sur la triangulation. Chacune de ces méthodes peut utiliser trois types de balises dont les positions sont connues, les balises actives émettent un signal, les balises passives qui réfléchissent le signal envoyé par un instrument embarqué et les repères naturels ou amers qui sont des caractéristiques de l'environnement.

3.2.2.1.1 Différents types de balises

3.2.2.1.1.1 Les repères artificiels

Les repères artificiels sont des balises caractéristiques qui sont ajoutées au milieu d'évolution du robot et dont les positions sont connues. L'inconvénient de ce type de techniques réside essentiellement dans son manque de souplesse et dans sa lourdeur d'utilisation. En effet un domaine d'évolution vaste nécessitera un investissement lourd en équipement. En outre tout changement de configuration de l'environnement impliquera une remise en cause du réseau de balises. En revanche cette technique a le gros avantage d'être précise, robuste et surtout elle satisfait la contrainte temps réel.

Les balises artificielles peuvent être; actives (elles émettent des signaux) ou passives (elles ne peuvent pas émettre de signaux).

- *Balises actives*

Les balises actives sont généralement interactives avec le système de perception de l'engin mobile puisque ce sont elles qui émettent l'onde captée par l'équipement de mesure du robot, ce qui servirait à calculer la position du robot mobile. Les deux principaux types d'organes de transmission utilisés sont les émetteurs de sources lumineuses (infrarouge par exemple) [38] et les antennes émettrices hyperfréquences. Ces ondes peuvent être émises en continu ou être déclenchées par ordre provenant de l'équipement embarqué sur le mobile. Ils ont été utilisés depuis plusieurs siècles pour la navigation maritime. En effet, les étoiles peuvent être considérées comme des balises actives, ainsi que les lumières des phares. Dans ce contexte, les balises doivent être disposées à des emplacements connus de l'environnement de travail. Ainsi elles seront facilement détectées par le robot mobile et avec un faible coût calculatoire.

Notons qu'une source sonore peut aussi être une balise active. En mesurant les différents temps de réception de chaque balise, la position du robot peut être calculée.

Un des systèmes les plus connus de ces dernières années est le Système de Positionnement Global (GPS). ce système est basé sur le repérage de balises actives que sont les satellites. Chaque satellite du GPS émet un signal contenant des informations telles que son identification, l'heure d'émission du signal et les paramètres définissant son orbite donc sa position. En connaissant sa distance par rapport à trois satellites, ainsi que la localisation de chaque satellite au moment de l'émission du signal, le récepteur peut calculer sa position en trois dimensions en utilisant des méthodes avancées de trilatération [39].

L'inconvénient du GPS se situe au niveau des signaux satellites qui peuvent ne pas parvenir au récepteur à cause du relief du terrain et de l'environnement (bâtiments hauts et ponts en villes, reliefs montagneux, etc.).

- *Balises passives*

Contrairement aux balises actives, les balises passives se contentent de réfléchir un signal provenant de l'équipement de mesure. Il peut s'agir par exemple de miroirs ou de catadioptrés.

Ces balises peuvent également comporter des informations plus évoluées (code-barres). Donc la différence se situe au niveau de l'identification de la balise qui doit être faite par le système de perception. Ainsi le problème de la mise en correspondance est généralement facilité par un codage au niveau du marquage de la balise.

Comme précédemment ce type de méthode est basé sur le repérage de balises artificielles placées à des positions connues dans le milieu d'évolution du robot mobile. Ces systèmes ont l'avantage d'être moins lourds à installer que les systèmes précédents.

Au moyen d'un capteur de vision, un panneau dont la position est connue et sur lequel sont dessinées des bandes verticales blanches et noires peut être utilisé comme repère [40]. Une caméra CCD vise le panneau et l'algorithme proposé détermine la position du robot en analysant les déformations des bandes vues dans l'image.

En environnement d'intérieur, les repères peuvent être aussi des réflecteurs placés sur le mur. La localisation du robot est alors réalisée par une triangulation pour laquelle les angles séparant le robot des balises sont mesurés. Ces angles peuvent être mesurés à partir d'un capteur ultrasons tournant et de réflecteurs acoustiques [41], ou d'un goniomètre laser et de réflecteurs de lumières [42].

3.2.2.1.1.2 Les repères naturels

Les amers naturels sont des éléments de l'environnement distincts que le robot peut reconnaître aisément grâce à ses capteurs extéroceptifs. Ces amers peuvent être par exemple des formes géométriques (rectangles, lignes, cercles, etc.). Ce sont des éléments déjà existants dans l'environnement. Ils présentent l'avantage de ne pas modifier cet environnement. Cependant, leur détection et leur identification peuvent être plus difficiles et les risques d'ambiguïté ou de fausses détections sont plus élevés. Cette technique consiste à utiliser ces éléments de l'environnement pour estimer la position du robot.

L'intérêt de ces méthodes est leur souplesse d'utilisation puisqu'elles ne nécessitent pas d'aménager le milieu d'évolution du robot. Leurs inconvénients sont la portée réduite et la difficulté d'identification.

Par exemple. Dans une application utilisant la vision pour déterminer la position du robot par un calcul de triangulation, de nombreux repères (lignes verticales, cadres, posters, etc.) sont disponibles dans l'environnement et sont utilisés pour ce fait [43].

Lorsque cette méthode est utilisée avec des cartes ne comportant pas de *perceptual aliasing*, chaque amer est unique et cette méthode permet d'estimer directement de manière non ambiguë la position du robot. Cette méthode est donc une méthode de localisation globale. En cas de *perceptual aliasing*, certains amers ne peuvent être distingués et il faut tenir compte de l'estimation précédente de la position afin de pouvoir identifier correctement les différents amers et estimer correctement la position. Donc cette méthode sera utilisée pour repérer plusieurs positions possibles pour le robot au sein de l'environnement.

3.2.2.1.2 Méthodes de calcul de la position du robot

Il est à noter que le point commun de toutes ces méthodes citées ci-dessus est le calcul de la configuration du robot basée systématiquement sur l'une des deux méthodes suivantes :

- La trilatération (triangulation par télémétrie)
- La triangulation (triangulation par goniométrie)

3.2.2.1.2.1 La trilatération (ou multilatération)

Le calcul de la position se fait grâce à la connaissance des distances séparant le robot de plusieurs balises placées dans l'environnement. Le temps de vol de l'onde émise peut par exemple être utilisé pour le calcul de ces distances. On peut calculer sa position par l'intersection de cercles en (2D) ou de sphères (en 3D). Par exemple, en deux dimensions, l'intersection des trois cercles centrés sur trois balises donne la position du robot mobile (Figure 3.1).

Soient $M(x_M, y_M)$ la position du robot et d_i la distance mesurée entre le robot et la balise $B_i(x_i, y_i)$, la position de M est calculée en cherchant la solution (x, y) du système composé des équations des cercles C_i :

$$C_i = (x - x_i)^2 + (y - y_i)^2 = d_i^2 \quad \text{pour } i=1..3$$

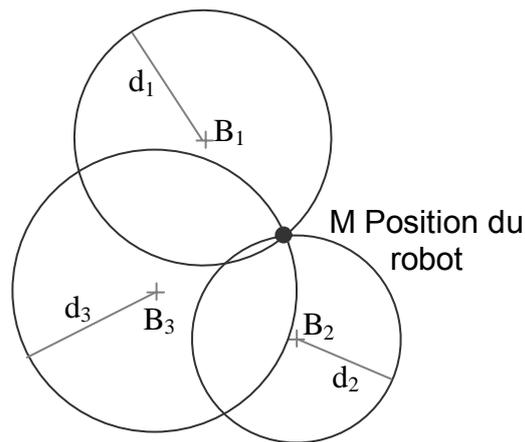


Figure 3.1 Localisation par mesure de distances de trois balises (trilatération)

Si l'on prend en compte les erreurs de mesures. On pourra alors considérer des anneaux à la place des cercles. Et on obtient ainsi une zone à la place d'un point.

3.2.2.1.2.2 La triangulation

Au lieu d'employer des distances, cette méthode utilise la mesure d'angles séparant l'axe du robot mobile des directions sous lesquelles il perçoit les balises. Par triangulation on en déduit la position du robot (Figure 3.2).

Soit $M(x_M, y_M)$ la position du robot et φ son orientation. Soit α_i , l'angle sous lequel le robot perçoit la balise B_i . Sachant les positions de chaque balise dans l'environnement, les distances $B_i B_j$, les angles β_i et l'angle δ sont connus. A partir des α_i , les angles $B_3 M B_2$ et $B_1 M B_2$ sont déterminés. Les propriétés de trigonométrie dans un triangle nous donnent les relations :

$$\frac{MB_2}{\sin(\beta_3 + \theta_3)} = \frac{B_2 B_3}{\sin(B_3 M B_2)} \quad \text{et} \quad \frac{MB_2}{\sin(\beta_1 + \theta_1)} = \frac{B_2 B_1}{\sin(B_1 M B_2)}$$

En isolant MB_2 , on obtient une relation où les seuls inconnus sont θ_1 et θ_3 .

En utilisant la relation $B_3 M B_2 + B_1 M B_2 + \theta_1 + \theta_3 = \pi$, on exprime θ_1 en fonction de θ_3 , puis on substitue θ_1 dans la relation obtenue précédemment, ce qui permet de déterminer θ_3 . Connaissant MB_2 et θ_3 on détermine l'angle β par la relation $\beta = B_3 M B_2 + \theta_3 + \beta_3 - \delta$ et on obtient les coordonnées x_M et y_M du robot et son orientation φ par :

$$\begin{cases} x_M = x_{B_2} - MB_2 \cos \beta \\ y_M = y_{B_2} - MB_2 \sin \beta \\ \theta = -\frac{\pi}{2} + \beta + \alpha_2 \end{cases}$$

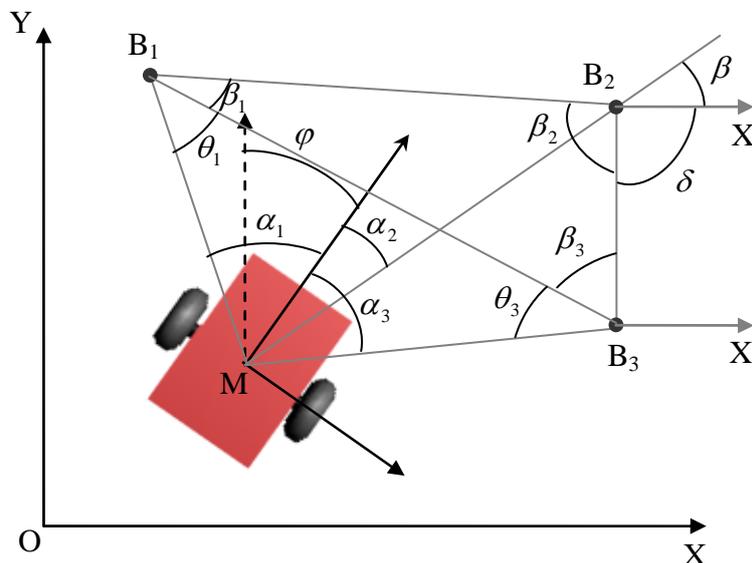


Figure 3.2 Localisation par triangulation

3.2.2.2 Mise en correspondance de modèle

La méthode de localisation consiste dans ce cas à trouver la meilleure correspondance entre deux modèles. Le premier, représente l'environnement partiellement ou dans sa totalité, appelé carte globale. Le second, correspondant à la partie de l'environnement perçue par le robot depuis sa position, appelé carte locale. La carte locale est formée à partir des informations acquises par les capteurs embarqués sur le robot, et dépend de la position du robot. En revanche la carte globale est connue soit par intégration de plusieurs cartes locales lors de l'exploration d'un environnement inconnu, soit par une connaissance à priori de l'environnement. Dans le cas où seules certaines parties de l'environnement total sont intéressantes pour la tâche à effectuer. Il est utile d'avoir recours à une carte topologique. Ce type de carte représente l'environnement par un graphe dont les nœuds représentent des lieux particuliers de l'environnement et les arcs désignent le chemin entre deux de ses lieux. C'est le cas par exemple d'un environnement constitué d'un couloir donnant accès à différentes salles de travail. Seules les salles ont un intérêt pour la localisation (chaque salle représente un nœud du graphe), le

couloir ne servant qu'à relier les salles (chaque chemin entre deux salles représente un arc du graphe) [44].

Comme nous l'avons vu dans la partie précédente, il existe principalement deux méthodes pour représenter les mesures : la méthode géométrique et la méthode par grilles. Ces deux méthodes induisent donc deux techniques de localisation distinctes. Une troisième technique liant les deux méthodes est aussi possible.

Cette méthode est souvent utilisée sur un espace de recherche restreint par une estimation initiale de la position. Elles peuvent cependant être utilisées pour la localisation globale [45, 46].

3.2.2.2 .1 Les cartes géométriques

Pour établir la carte géométrique locale, il est souvent nécessaire en raison de l'imprécision sur les mesures des capteurs de modéliser les incertitudes. Dans le cas de la localisation dynamique, de nombreux travaux utilisent alors le filtre de Kalman généralisé qui permet d'estimer successivement après chaque acquisition des mesures la position du robot. Lorsque la carte globale de l'environnement est construite au fur et à mesure de l'exploration du robot, ce filtre permet aussi d'estimer la position de chaque élément géométrique introduit dans la carte.

Une autre approche permet de réaliser une mise en correspondance de cartes sachant qu'il existe une incertitude sur ces dernières en utilisant la méthode des moindres carrés [18]. Le plan de l'environnement fourni est constitué d'un ensemble de segments et à partir de l'odométrie, la position du robot est approximativement connue dans ce plan. Un capteur laser monté sur une tourelle, permet au robot d'acquérir un panorama complet de l'environnement. La mise en correspondance des mesures avec le plan se fait par association du point de mesure au segment du modèle le plus proche. La somme des distances au carré séparant les points de mesures des segments associés est calculée, et la position du robot est obtenue en recherchant celle qui minimise cette somme.

3.2.2.2 .2 Les grilles

Une première méthode de mise en correspondance de deux grilles est décrite dans [27] ; les cartes globales et locales sont constituées de cellules dont les valeurs

indiquent la présence ou l'absence d'un obstacle. Pour ce faire l'auteur affecte des valeurs positives aux cellules représentant des obstacles et des valeurs négatives aux cellules correspondant aux espaces libres. En superposant les deux grilles, le produit de deux cellules superposées indique s'il y a correspondance ou non puisque le produit de deux cellules identiques est positif et celui de deux cellules opposées est négatif. Ce produit est calculé sur toute la grille et la somme de tous ces produits est évaluée. Ainsi en calculant cette somme pour différentes positions et orientations de la carte locale par rapport à la carte globale, l'emplacement qui fournit la somme maximale et donc qui donne la meilleure correspondance entre les deux grilles permet de localiser le robot.

Une autre méthode à base de grilles utilise une représentation topologique pour décrire l'environnement [44] ; chaque nœud de la carte topologique correspond à une pièce de travail, le couloir reliant deux pièces étant représenté par un arc. Une première phase d'exploration permet au robot de construire la carte topologique en représentant chaque pièce par trois grilles construites à partir des mesures issues d'un type de capteurs : capteur à ultrasons, caméra et capteurs à infrarouges. Les grilles sont créées par la méthode *HIMM* (Histogrammic In-Motion Mapping) introduite par *Borenstein* et *Koren* dans [47], puis sont décrites de façon unique en calculant des paramètres basés sur des valeurs des cellules de la grille. Pour déterminer la pièce de travail au cours de la navigation, le robot construit de nouvelles grilles par cette même méthode. En comparant les paramètres extraits des grilles actuelles à ceux extraits des grilles construites lors de l'exploration, la région de travail (couloir ou différentes salles) dans laquelle le robot se trouve est déterminée. Cette localisation ne sera donc que grossière puisqu'elle n'indique que la région de travail et ne précise pas le positionnement exact du robot dans cette région.

3.2.3 Localisation hybride

Les chercheurs se sont rendus compte de l'intérêt à utiliser conjointement la localisation à l'estime et la localisation absolue. On parle alors de recalage dynamique. Mais les méthodes hybrides intègrent aussi celles qui s'appuient sur la coopération de plusieurs capteurs proprioceptifs ou extéroceptifs.

- *Le recalage dynamique*

Cette méthode consiste à fusionner les deux types de localisation, absolue et à l'estime, dans le but de cumuler les avantages et de pallier les inconvénients de chacun pris séparément. Les principaux avantages des systèmes de localisations à l'estime sont le fonctionnement continu ainsi que l'autonomie et les inconvénients sont les accumulations d'erreurs. Les inconvénients des systèmes de localisation absolue sont le fonctionnement non autonome et les pertes dans le cas des systèmes utilisant des balises ou le problème de calcul en temps réel dans le cas de la mise en correspondance, tandis que l'avantage principal est la précision de la position absolue.

Les techniques de fusions sont multiples mais la méthode de filtrage statistique (filtrage de Kalman) présente de nombreux avantages et est fréquemment employée.

- *La coopération multi-capteur*

Le recalage dynamique n'est qu'un cas particulier d'association possible de capteurs. En effet, d'autres combinaisons, telles qu'un ensemble de capteurs extéroceptifs fournissant des données complémentaires, sont intéressantes à utiliser pour la perception en robotique mobile. Un capteur utilisé seul ne peut relever complètement toutes les caractéristiques de l'environnement. Il est généralement nécessaire de combiner des données provenant de plusieurs capteurs extéroceptifs. Deux types de combinaison peuvent être réalisés dans les problèmes de fusion de données ; une combinaison d'informations redondantes où l'intégration et la fusion de l'information redondante aide alors à accroître la précision et le degré de confiance d'une estimation obtenue avec plusieurs observations distinctes. Et une combinaison d'informations complémentaires où l'intégration et la fusion de données complémentaires permettent généralement de fournir une information ne pouvant être obtenue séparément avec chacun des capteurs.

De nombreux travaux utilisent la coopération de capteurs pour contribuer aux problématiques que sont la localisation et la modélisation de l'environnement. Dans [48] un capteur de vision omnidirectionnel et un capteur ultrasonore sont utilisés conjointement. Le système de vision omnidirectionnel donne l'angle d'azimut des amers verticaux de l'environnement. Le capteur ultrasonore fournit une confirmation de l'espace libre entre deux amers verticaux. D'autres travaux utilisent pour la localisation du robot la coopération entre un télémètre laser et un système de vision monoculaire [32]. Une segmentation du relevé laser lui permet d'obtenir un ensemble

de primitives de type segment. L'application de la transformée de Hough sur l'image CDD fournit les angles des amers verticaux devant le robot. La carte théorique de l'environnement est composée d'une liste de segments et d'une liste d'amers verticaux. L'appariement des deux types d'objets est réalisé grâce à un filtre de Kalman étendu.

Les méthodes de localisation hybrides peuvent être locales (méthodes de suivi d'une hypothèse unique), ou globales (méthodes de suivi de plusieurs hypothèses).

3.2.3.1 Méthode de Suivi d'une hypothèse unique

Lorsque les perceptions ne suffisent pas pour estimer la position de manière unique, une seconde source d'estimation de la position du robot est nécessaire pour lever l'ambiguïté.

Cette seconde estimation provient, d'une part de la position déterminée lors de la précédente phase de localisation et, d'autre part, des données proprioceptives recueillies depuis cet instant. Les méthodes présentées dans cette section utilisent cette seconde estimation pour sélectionner ou calculer à chaque instant, la position qui est la plus cohérente vis-à-vis de cette estimation.

Pour ces méthodes locales, généralement la position initiale du robot est connue, mais celle-ci pourra être retrouvée par l'une des méthodes de localisation globales décrites dans la section précédente.

Toutefois, il est également possible d'utiliser une des méthodes de localisation globales (utilisant de manière optimale les informations proprioceptives et extéroceptives), qui seront décrites par la suite et qui permettent de ne plus dépendre d'une estimation initiale correcte de la position. Cette seconde solution ne requiert pas l'utilisation séparée de deux méthodes.

Les méthodes de suivi de positions les plus utilisées sont les filtres de Kalman, qu'on va illustrer dans cette section.

3.2.3.1.1 Le filtre de Kalman

Le filtre de Kalman est une façon d'intégrer des données mesurées à l'intérieur d'un estimateur en admettant qu'une mesure est intrinsèquement bruitée et

que toute mesure ne devrait pas avoir le même degré de confiance et que certaines mesures devraient être ignorées ou avoir un poids plus faible dans l'établissement de l'état du système. Le filtre lisse les données et atténue l'effet du bruit dans l'estimation de l'état du système en essayant d'utiliser plus d'informations fiables que d'informations moins fiables. En informant le filtre du niveau de bruit du système (mesure et modèle) celui-ci calcule une estimation de l'état du système en tenant compte du bruit. Il est moins probable que le filtre laisse passer une mauvaise mesure si la confiance en cette mesure est grande que si elle est faible.

L'algorithme du filtre de Kalman facilite aussi la combinaison de mesures provenant de différentes sources, exp. caméra, encodeur. En plus d'estimer le vecteur d'état du système, le filtre estime l'état du vecteur d'incertitude de l'estimation.

Le filtre de Kalman permet d'estimer la progression de la sortie d'un procédé en projetant dans le temps une prédiction de son état à partir de mesures et d'un modèle de celui-ci. Il utilise une boucle de rétroaction pour réintroduire dans le contrôle la valeur déterminée après filtrage. Essentiellement, le cycle de l'algorithme de filtrage (Figure 3.3) peut se diviser en deux parties : la *prédiction* de l'état à venir et la mesure et correction de l'état actuel [49].

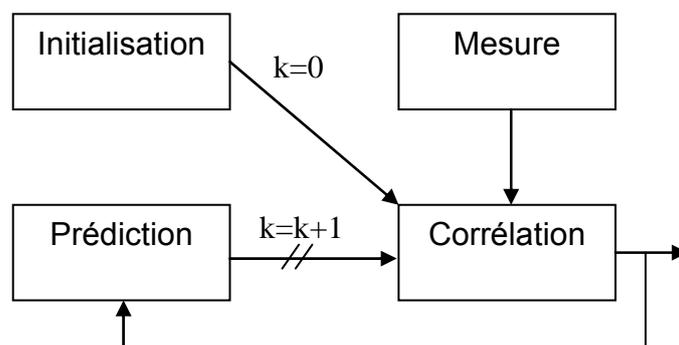


Figure 3.3 Cycle de rétroaction du filtre de Kalman.
La prédiction projette l'estime de l'état à la prochaine itération,
la correction ajuste la valeur de l'état actuel en tenant compte
de la prédiction et la mesure effectuée.

Pour expliquer le principe du filtre de Kalman, on commencera par le cas le plus simple, c'ad le cas du filtre de Kalman à une dimension (1D).

Nous cherchons à estimer une **variable** X_k pour chaque état k (itération k). Où le filtre de Kalman doit disposer de deux modèles linéaires ; le modèle du système et le modèle de mesure.

- Le modèle du système s'exprime par l'équation de récurrence stochastique suivante :

$$X_k = aX_{k-1} + W \quad (55)$$

Où W est le bruit du modèle du système, on suppose qu'il est de distribution gaussienne, de variance q et de moyenne nulle.

- Afin de déterminer la variable X_k on fait une mesure m_k , cette mesure est reliée à X_k via le modèle linéaire suivant :

$$m_k = hX_k + V \quad (56)$$

V est le bruit de mesure, de distribution gaussienne, de variance r et de moyenne nulle.

- Pour déterminer l'estimé final X_k , on doit calculer avant la prédiction de X_k (X_k^-). cette valeur prédite sera exprimée en fonction de la variable X_{k-1} (à l'état $k-1$) comme le montre l'équation suivante :

$$X_k^- = aX_{k-1} + W \quad (57)$$

Le bruit W n'affecte pas cette prédiction puisque sa moyenne est nulle, donc :

$$X_k^- = aX_{k-1}$$

- La variance P_k^- de la prédiction X_k^- est donnée par :

$$P_k^- = \text{var}(X_k^-) = \text{var}(aX_{k-1} + W) = a^2 \text{var}(X_{k-1}) + \text{var}(W)$$

Notons P_{k-1} la variance de X_{k-1} , dans ce cas :

$$P_k^- = a^2 P_{k-1} + q \quad (58)$$

La variance P_k^- est une valeur qui nous indique quel degré de confiance est attribué à une prédiction. En effet, plus la variance P_k^- est grande, moins la certitude d'avoir fait une bonne estimation est élevée, c'est logique, parce qu'une grande variance indique qu'on a une grande différence entre l'état mesuré et l'état prédit.

- Afin de déterminer la valeur de X_k , on combine les valeurs (incertaines) de la mesure m_k avec les valeurs (incertaines aussi) de la prédiction X_k^- . Donc l'estimé final X_k s'exprimera par l'équation :

$$X_k = X_k^- + K_k (m_k - hX_k^-) \quad (59)$$

Où K_k est le gain de Kalman. La partie $(m_k - hX_k^-)$ de l'équation (59) mesure la différence entre la mesure m_k et la prédiction hX_k^- , et c'est ce qui est appelé l'innovation ou le résiduel. Lorsque le résiduel est zéro, alors la mesure est identique à la prédiction qui avait été faite à l'itération précédente.

- Pour déterminer l'expression du gain de Kalman K_k :

- On commence par exprimer X_k différemment, à partir de l'équation (59) on peut écrire X_k tel que :

$$X_k = X_k^- + K_k m_k - K_k hX_k^-$$

$$X_k = (1 - K_k h)X_k^- + K_k m_k$$

- Dans ce cas la variance P_k de X_k est :

$$P_k = \text{var}(X_k) = \text{var}((1 - K_k h)X_k^- + K_k m_k)$$

$$P_k = (1 - 2K_k h + K_k^2 h^2)P_k^- + K_k^2 r$$

$$P_k = P_k^- - 2K_k h P_k^- + K_k^2 (h^2 P_k^- + r) \quad (60)$$

- Le gain de Kalman est un facteur qui permet de donner plus ou moins de poids à la prédiction antérieure pour l'estimation de l'état.

Le calcul du gain est effectué de telle sorte qu'il minimise la variance P_k , alors :

$$\frac{dP_k}{dK_k} = 0 \Rightarrow \frac{d}{dK_k} (P_k^- - 2K_k h P_k^- + K_k^2 (h^2 P_k^- + r)) = 0$$

$$\Rightarrow -2h P_k^- + 2K_k (h^2 P_k^- + r) = 0$$

Donc :

$$K_k = \frac{h P_k^-}{(h^2 P_k^- + r)} \quad (61)$$

- en utilisant les équations (60) et (61), on pourra exprimer la variance P_k comme tel :

$$P_k = P_k^- - 2K_k h P_k^- + K_k^2 \left(\frac{h P_k^-}{K_k} \right)$$

(On a remplacé $(h^2 P_k^- + r)$ par $\left(\frac{h P_k^-}{K_k} \right)$).

$$P_k = P_k^- - 2K_k h P_k^- + K_k h P_k^-$$

$$P_k = P_k^- - K_k h P_k^- \quad (62)$$

Pour résumer, les équations du Filtre de Kalman à 1D se traduit par :

$$X_k = aX_{k-1} + W$$

$$m_k = hX_k + V$$

Avec W et V respectivement le bruit du modèle et le bruit de mesure, de variance q et r , et de moyenne nulle.

$$X_k^- = aX_{k-1}$$

$$P_k^- = a^2 P_{k-1} + q$$

$$K_k = \frac{hP_k^-}{(h^2 P_k^- + r)}$$

$$X_k = X_k^- + K_k (m_k - hX_k^-)$$

$$P_k = P_k^- - K_k h P_k^-$$

Dans notre cas, la valeur de X_k qu'on cherche à estimer est un **vecteur** de i éléments (avec $i=3$) puisque c'est une position (x, y, φ) .

Donc il ne s'agira plus de traiter des variables mais des vecteurs, et on aura plus à calculer des variances mais des **covariances**.

Dans ce cas, les équations du filtre de Kalman seront exprimées comme tel :

- Le modèle du système : $X_k = AX_{k-1} + W$ (63)

- Le modèle de mesure : $m_k = HX_k + V$ (64)

Où W est le bruit du modèle et V le bruit de mesure, tous deux indépendants l'un de l'autre. L'hypothèse sur les bruits W et V est qu'ils sont de distribution gaussienne de moyenne nulle, respectivement de **covariance** Q et R . La matrice A fait le lien entre l'état de X à l'itération $k-1$ et à l'itération k en l'absence du bruit du modèle mais en fonction du signal de control U . La matrice H dans l'équation de mesure lie l'état du procédé X à la mesure m .

- L'état prédit (ou l'estimé à priori) est :

$$X_k^- = AX_{k-1} \quad (65)$$

- La covariance P_k^- de la prédiction X_k^- est :

$$P_k^- = AP_{k-1}A^T + Q \quad (66)$$

- L'estimé final X_k est :

$$X_k = X_k^- + K_k (m_k - HX_k^-) \quad (67)$$

- Le gain de Kalman est le facteur K_k . Tout comme dans le cas du filtre de Kalman à 1D, afin de déterminer son expression on exprime avant X_k par l'équation suivante :

$$X_k = (I - K_k H) X_k^- + K_k m_k$$

Dans ce cas la covariance P_k de X_k est :

$$P_k = P_k^- - 2K_k H P_k^- + K_k K_k^T (H P_k^- H^T + R) \quad (68)$$

Pour déterminer la valeur du gain K_k qui minimise P_k , on le calcul pour $\frac{dP_k}{dK_k} = 0$, on

obtient alors :

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (69)$$

En analysant la forme de ce gain, les conclusions suivantes sont faites:

- $\lim_{R \rightarrow 0} K_k = H^{-1}$, donc plus la covariance du bruit de mesure diminue, plus le gain augmente, jusqu'à H^{-1} . Ainsi, moins l'incertitude sur la mesure est grande, plus la mesure m_k sera prise en compte au détriment de la prédiction. En effet, si K_k est remplacé avec $R=0$ dans l'équation (59b), $X_k = H^{-1} m_k$ est obtenu où les termes référant à l'état de l'étape précédente X_k^- ont été éliminés.
- $\lim_{P_k^- \rightarrow 0} K_k = 0$, évidemment, si la covariance de l'erreur estimée a priori est nulle, cela signifie que l'état actuel correspond à l'état prédit à l'étape précédente donc $X_k = X_k^-$ et qu'aucune correction n'a été apporté par le gain.

Donc, lorsque la covariance du bruit de mesure (R) diminue, le filtre fait d'autant plus confiance à la mesure m_k , alors que lorsque la covariance de l'erreur estimée à priori (P_k^-) diminue, c'est la prédiction X_k^- qui obtient la faveur du filtre.

- La covariance P_k de l'état estimé final X_k est :

$$P_k = P_k^- - K_k H P_k^- \quad (70)$$

Pour résumer le filtre de Kalman à plusieurs dimensions est basé sur les équations suivantes :

$$X_k = AX_{k-1} + W$$

$$m_k = HX_k + V$$

Avec W et V respectivement le bruit du modèle et le bruit de mesure, de covariance Q et R , et de moyenne nulle.

$$X_k^- = AX_{k-1}$$

$$P_k^- = AP_{k-1}A^T + Q$$

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}$$

$$X_k = X_k^- + K_k (m_k - HX_k^-)$$

$$P_k = P_k^- - K_k HP_k^-$$

Enfin, pour initialiser le filtre il faut un état initial pour l'ensemble des variables. Ainsi l'état initial X_0 doit être fourni au départ, quant à sa covariance P_0 la précision n'est pas importante il suffit de choisir une valeur assez grande pour englober toutes les erreurs possibles (et non nulle pour que le filtre ne tombe pas dans un état instable) puisque cette valeur diminue automatiquement avec la convergence du filtre.

Notons que, la matrice de covariance P_0 est diagonale dans le cas où les éléments du vecteur d'état X_k sont indépendants puisque la $\text{cov}(X_i, X_j) = 0$ pour $i \neq j$, et sa correspond à notre cas où le vecteur X_k est une position (x, y, φ) dont les variables x , y et φ sont indépendantes.

Les covariances du bruit Q et R sont constantes, et peuvent être fournies à partir de tests expérimentaux. Dans certaines conditions, la covariance du bruit de mesure R et du modèle Q n'est pas constante. Par exemple, dans un autre registre que le notre lors du suivi temporel de la position d'un marqueur en 3D, plus l'objet suivi s'éloigne du système de mesure, plus l'erreur sur la mesure de la position sera grande, et plus la covariance R sera grande aussi. Par ailleurs, la modélisation du procédé peut aussi changer dynamiquement durant le suivi. Ou par exemple, dans le cas du suivi de la position d'un visage, le sujet peut bouger la tête, lentement ou rapidement, ainsi, si l'utilisateur bouge rapidement, l'incertitude sur le modèle ou sur l'intention du sujet peut être ajustée en augmentant Q .

Bien que tout ça semble compliqué, voici à la figure 3.4 les équations qui implémentent l'algorithme du filtre de Kalman, divisées en deux parties comme à la figure 3.3 par des équations de *prédiction* et des équations de *mesure et correction*.

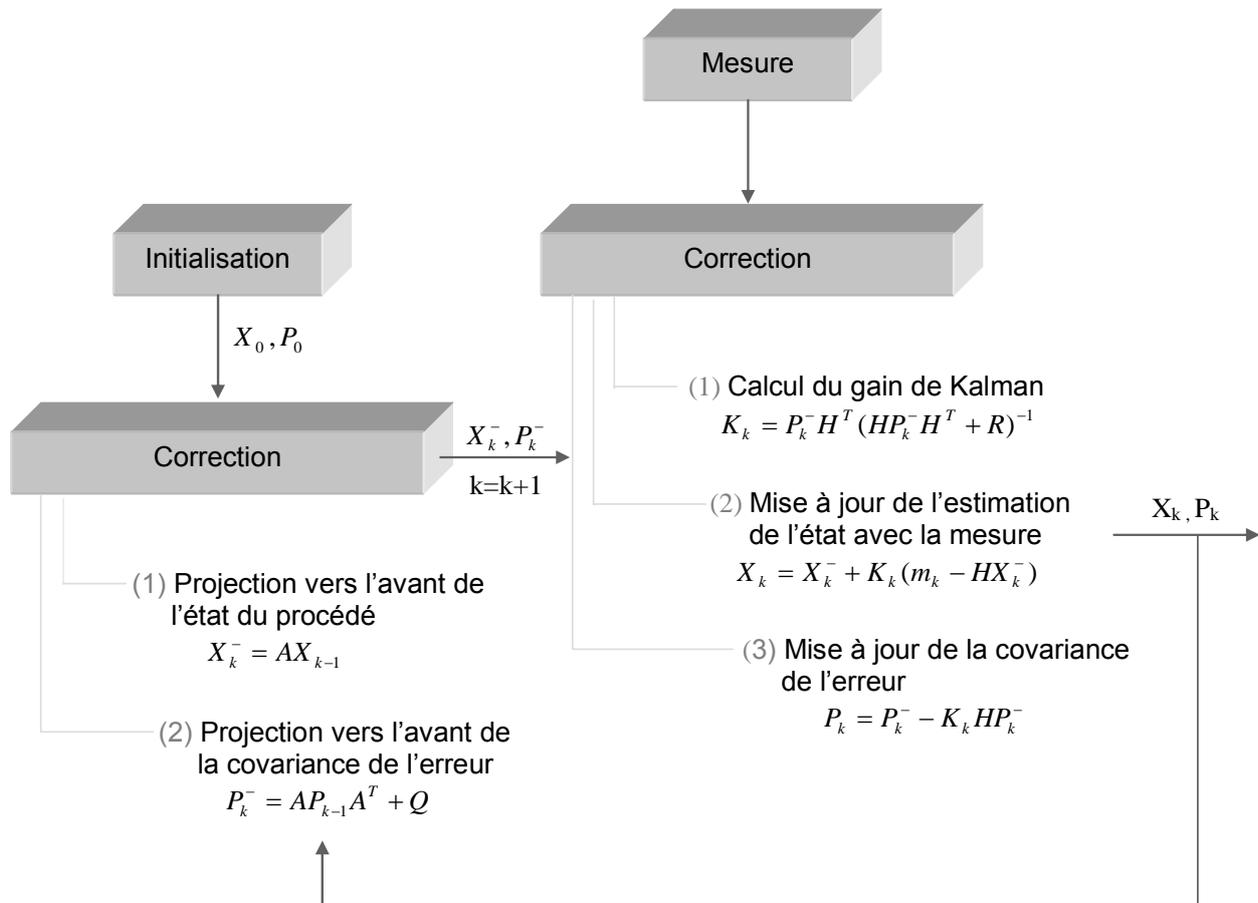


Figure 3.4 Le cycle complet du filtre de Kalman avec les équations de prédiction et de mesure et correction du filtre de Kalman

3.2.3.1.2 Le filtre de Kalman étendu (Extended Kalman Filter ou EKF) [50, 51]

Le filtre de *Kalman* tel que présenté dans la section précédente permet d'estimer dans le temps l'état d'un procédé défini par une équation linéaire. Cependant, dans la réalité, l'hypothèse de linéarité d'un procédé ne peut pas toujours être vérifiée. Le filtre de Kalman ne peut donc pas être utilisé, mais ça ne veut pas dire que tout doit être jeté de ce qui a été proposé. En effet, une astuce permet d'ajuster le filtre pour un procédé avec une équation non linéaire pour linéariser les variables dont le filtre a besoin. L'idée est de linéariser localement sur la moyenne et la variance de la sortie du procédé. Avec les séries de *Taylor*, une linéarisation de l'estimation peut être effectuée en utilisant les dérivées partielles des fonctions du procédé f et de mesure h . Cette linéarisation utilise l'hypothèse que l'erreur sur la dérivée est petite, ainsi, la série de *Taylor* est tronquée dès le premier ordre.

Les équations (65) et (66) du modèle et la mesure peuvent se mettre sous la forme suivante :

$$X_k = f(X_{k-1}, U, W) \quad (71)$$

$$m_k = h(X_k, V) \quad (72)$$

Où les fonctions non linéaires f et h relient respectivement l'état et la mesure d'une étape k à l'étape suivante et prennent la place des matrices A et H des équations (63) et (64).

Comme le bruit W et V ont une moyenne nulle selon notre hypothèse, alors une estimation (prédiction) de l'état et de la mesure peut être exprimée comme étant :

$$X_k = f(X_{k-1}, U, 0) \quad (73)$$

$$m_k = h(X_k, 0) \quad (74)$$

Enfin, pour linéariser notre modèle la matrice A devient la matrice Jacobienne des dérivées partielles de la fonction f de l'équation (73) par rapport au vecteur d'état X_k , et H la matrice Jacobienne des dérivées partielles de la fonction h de l'équation 72 par rapport X_k aussi, les deux matrices seront exprimées comme tel :

$$A = \nabla f_{X_k} = \frac{\delta X_k}{\delta X_{k-1}} \quad (75)$$

$$H = \nabla h_{X_k} = \frac{\delta m_k}{\delta X_k} \quad (76)$$

De ce qui est des équations du filtre de Kalman étendu, elles restent les mêmes que celles du filtre de Kalman sauf que la matrice A et/ou la matrice H prendront pour valeurs celles exprimées dans les équations (75) et (76), pour avoir enfin les équations suivantes :

$X_k = f(X_{k-1}, U, W)$	}	Modèle
$m_k = h(X_k, V)$		
$X_k^- = f(X_{k-1}, U, 0)$	}	Prédiction
$P_k^- = AP_{k-1}A^T + Q$		
$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}$	}	Gain
$X_k = X_k^- + K_k(m_k - HX_k^-)$		
$P_k = P_k^- - K_k H P_k^-$	}	Estimé final

Avec W et V respectivement le bruit du modèle et le bruit de mesure, de covariance Q et R , et de moyenne nulle.

Un diagramme illustrant l'algorithme du filtre de Kalman étendu est présenté à la figure 3.5.

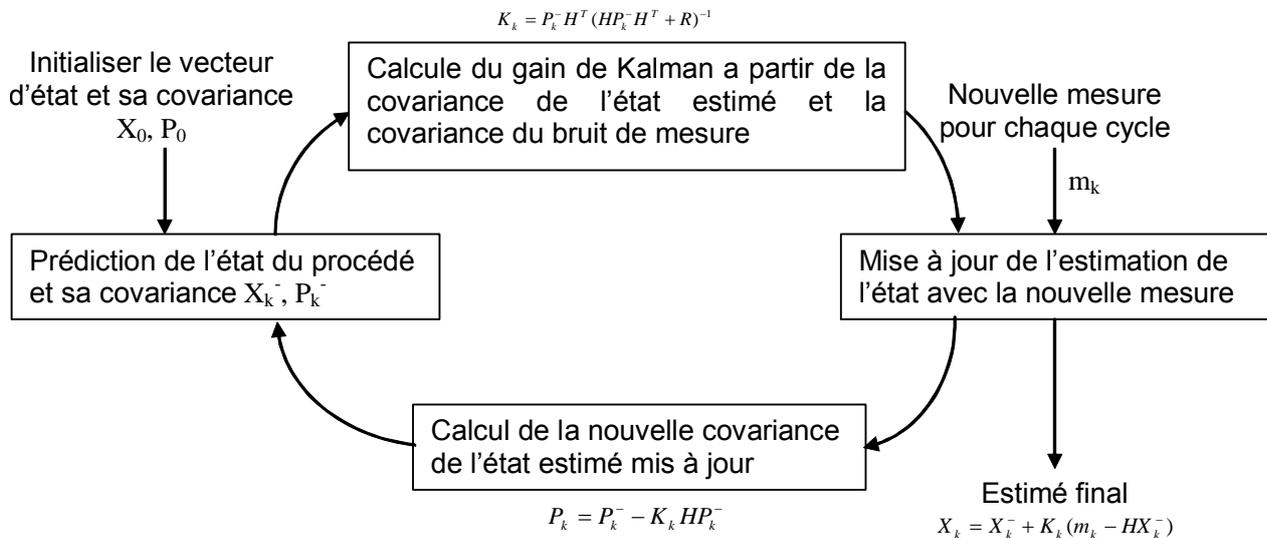


Figure 3.5 L'algorithme du filtre de Kalman étendu

3.2.3.1.3 Le filtre de Kalman inodore (Unscented Kalman Filter ou UKF) [52-58]

Cette nouvelle fragrance du filtre de Kalman propose de laisser le modèle tel quel au lieu d'essayer de le linéariser et d'estimer plutôt la distribution qui est utilisée pour prédire l'état prochain, c'est-à-dire selon Jutier et Uhlmann développeurs du UKF [52], c'est plus facile d'approximer une distribution gaussienne que d'approximer une fonction non linéaire dont on aura à calculer les matrices jacobiniennes pour arriver à la linéariser. Pour ce faire, la transformée inodore (*unscented transform*) est utilisée pour estimer les statistiques des variables aléatoires ayant subi une transformation non linéaire. La transformation inodore est ensuite utilisée avec le filtre de Kalman original vu plus haut.

3.2.3.1.3.1 La transformé inodore (unscented transform)

Considérons une variable aléatoire x de dimension n , qui est transformée par un procédé non linéaire g , tel que $y=g(x)$ où x a une moyenne \bar{x} et une variance P_x . Pour calculer les statistiques de y , une matrice \aleph de dimension $(2n+1)$ vecteur sigma \aleph_i , chacun de poids W_i , est formée selon les équations suivantes :

$$\aleph_0 = \bar{x} \quad i=0$$

$$\aleph_i = \bar{x} + (\sqrt{(n+\lambda)P_x})_i \quad i=1, \dots, n \quad (77)$$

$$\aleph_{i+n} = \bar{x} - (\sqrt{(n+\lambda)P_x})_i \quad i=1, \dots, n$$

Avec :

$$W_0 = \frac{\lambda}{n+\lambda}$$

$$W_i = \frac{1}{2(n+\lambda)} \quad (78)$$

$$W_{i+n} = \frac{1}{2(n+\lambda)}$$

Où $\lambda \in \Re$ est un facteur d'échelle permettant d'ajuster l'étendue du choix des points sigma autour de \bar{x} et $(\sqrt{(n+\lambda)P_x})_i$ est la i ème rangée de la matrice des racines carrées de $(n+\lambda)P_x$. Le poids W_0 correspondant au point sigma \aleph_0 , (ou \bar{x}) a le poids le plus élevé et plus le point sigma s'éloigne de ce point central, plus le poids est faible.

La procédure de transformation comprend les étapes suivantes :

1. Transformer chaque point sigma \aleph_i en le faisant passer dans le procédé non linéaire, tel que

$$y_i = g(\aleph_i) \quad (79)$$

2. Calculer la moyenne pondérée des points transformés

$$\bar{y} = \sum_{i=0}^{2n} W_i y_i \quad (80)$$

3. Calculer la covariance pondérée des points transformés :

$$P_{yy} = \sum_{i=0}^{2n} W_i [y_i - \bar{y}][y_i - \bar{y}]^T \quad (81)$$

Certaines propriétés de cet algorithme sont énumérées ici :

1. Puisque c'est la distribution de x qui est approximée plutôt que le procédé, la série de Taylor sous-jacente n'a pas à être tronquée à un ordre spécifique, ce qui peut aider à obtenir des résultats plus précis.
2. Les points sigma ont la même moyenne et covariance quel que soit le choix de la matrice de racine carrée choisie.
3. La moyenne et la covariance sont calculées par des opérations matricielles usuelles. Ceci implique que l'algorithme fonctionnera peu importe le choix du modèle du procédé (facilement dérivable ou non) et l'implémentation est rapide

car elle n'implique pas l'évaluation des matrices Jacobiennes que l'EKF nécessite.

4. Le paramètre λ permet d'ajuster les moments d'ordres supérieurs et de réduire l'erreur de prédiction globale. Quand l'hypothèse d'une distribution gaussienne est émise, il est suggéré d'utiliser $\lambda + n = 3$.

3.2.3.1.3.2 Le filtre de Kalman avec la transformée inodore

Quelques modifications aux équations du filtre de Kalman doivent être apportées.

D'abord, le vecteur d'état X est augmenté avec les termes du bruit du procédé,

$$X_k^a = \begin{bmatrix} X_k \\ W \end{bmatrix} \quad (82)$$

De dimension $n^a = n + q$, où n est la taille du vecteur d'état X et q la taille du vecteur du bruit W .

Le modèle du procédé est exprimé en fonction de X^a ,

$$X_k = f(X_{k-1}^a, U, k) \quad (83)$$

On introduit maintenant la transformé inodore, qui prend ses $2n^a + 1$ points à partir de :

$$X_k^a = \begin{pmatrix} X_k \\ 0_{q \times 1} \end{pmatrix} \quad (84)$$

Et de

$$P_k^a = \begin{bmatrix} P_k & P_{xw_k} \\ P_{xw_k} & Q \end{bmatrix} \quad (85)$$

P_{xw_k} est la covariance entre l'état et le bruit du procédé. Le fait que le bruit du procédé soit aussi introduit dans le calcul de la covariance implique que l'impact du bruit dans ce calcul aura la même précision que l'incertitude de l'état. Par ailleurs, l'implémentation d'un modèle où le procédé a une source qui génère du bruit corrélé peut facilement être implémenté à l'aide de ce modèle.

On commencera par calculer un ensemble de points sigma stocké dans une matrice \mathfrak{S}_k de dimension $n^a \times (2n^a + 1)$ avec n^a dimensions du vecteur d'état X_k , cette matrice est donnée par :

$$\begin{aligned}
(\mathfrak{S}_k)_0 &= X_k \\
(\mathfrak{S}_k)_i &= X_k + (\sqrt{(n^a + \lambda)P_k})_i \quad i=1, \dots, n^a \\
(\mathfrak{S}_k)_{i+n^a} &= X_k - (\sqrt{(n^a + \lambda)P_k})_i
\end{aligned} \tag{86}$$

Avec $\lambda = \alpha^2(n^a + \kappa) - n^a$, où α est un paramètre d'échelle qui détermine la vitesse des points sigma et κ le paramètre d'échelle secondaire.

Une fois \mathfrak{S}_k calculé, on améliore l'étape de prédiction en propageant chaque \mathfrak{S}_k dans le temps en utilisant :

$$(\mathfrak{S}_k)_i = f((\mathfrak{S}_{k-1})_i, U, k), \quad i=0 \dots 2n^a \tag{87}$$

Avec $(\mathfrak{S}_k)_i$ calculé, l'état estimé devient :

$$X_k^- = \sum_{i=0}^{2n^a} W_i^{(m)} (\mathfrak{S}_k)_i \tag{88}$$

$$\begin{aligned}
\text{Où : } W_0^{(m)} &= \frac{\lambda}{n^a + \lambda} \\
W_i^{(m)} &= \frac{1}{2(n^a + \lambda)}, \quad i=1 \dots 2n^a.
\end{aligned}$$

Comme dernière partie de l'étape de prédiction, on calcul la covariance d'erreur :

$$P_k^- = \sum_{i=0}^{2n^a} W_i^{(c)} [(\mathfrak{S}_k)_i - X_k^-][(\mathfrak{S}_k)_i - X_k^-]^T + Q \tag{89}$$

Où Q est la matrice de covariance d'erreur du processus et les poids sont définis par:

$$\begin{aligned}
W_0^{(c)} &= \frac{\lambda}{n^a + \lambda} + (1 + \alpha^2 + \beta) \\
W_i^{(c)} &= \frac{1}{2(n^a + \lambda)}, \quad i=1 \dots 2n^a.
\end{aligned}$$

Notant que β est un paramètre utilisé pour introduire toute information concernant la distribution de X.

Pour calculer l'étape de corrélation, on doit transformer \mathfrak{S}_k à travers la fonction de mesure :

$$(m_k)_i = h((\mathfrak{S}_k)_i), \quad i=0 \dots 2n^a \tag{90}$$

$$m_k^- = \sum_{i=0}^{2n^a} W_i^{(m)} (m_k)_i \tag{91}$$

En utilisant le vecteur de mesure transformé, on calcul l'état estimé final :

$$X_k = X_k^- + K_k (m_k - m_k^-) \tag{92}$$

Avec K_k le gain de Kalman. Or dans la formulation de UKF :

$$K_k = P_{X_k m_k} P_{m_k m_k}^T \quad (93)$$

Où :

$$P_{m_k m_k} = \sum_{i=0}^{2n^a} W_i^{(c)} [(m_k)_i - m_k^-][(m_k)_i - m_k^-]^T + R \quad (94)$$

$$P_{X_k m_k} = \sum_{i=0}^{2n^a} W_i^{(c)} [(X_k)_i - X_k^-][(m_k)_i - m_k^-]^T \quad (95)$$

Sachant que R est la matrice de covariance du bruit de mesure (comme dans EKF).

Finalement, dernier calcul à faire dans l'étape de corrélation est la covariance de l'état estimé final :

$$P_k = P_k^- - K_k P_{m_k m_k} K_k^T \quad (96)$$

Les équations de prédiction du filtre de Kalman inodore ainsi que les modifications à l'étape de correction sont décrites à la figure 3.6.

On remarque qu'en plus de la partie ayant directement rapport au calcul des points sigma, une différence frappante avec l'EKF est l'apparition de la prédiction sur la mesure m_k^- et le calcul un peu différent du résiduel, qui utilise la covariance pondérée de l'erreur sur la prédiction de la mesure et qui affecte le calcul du gain de Kalman.

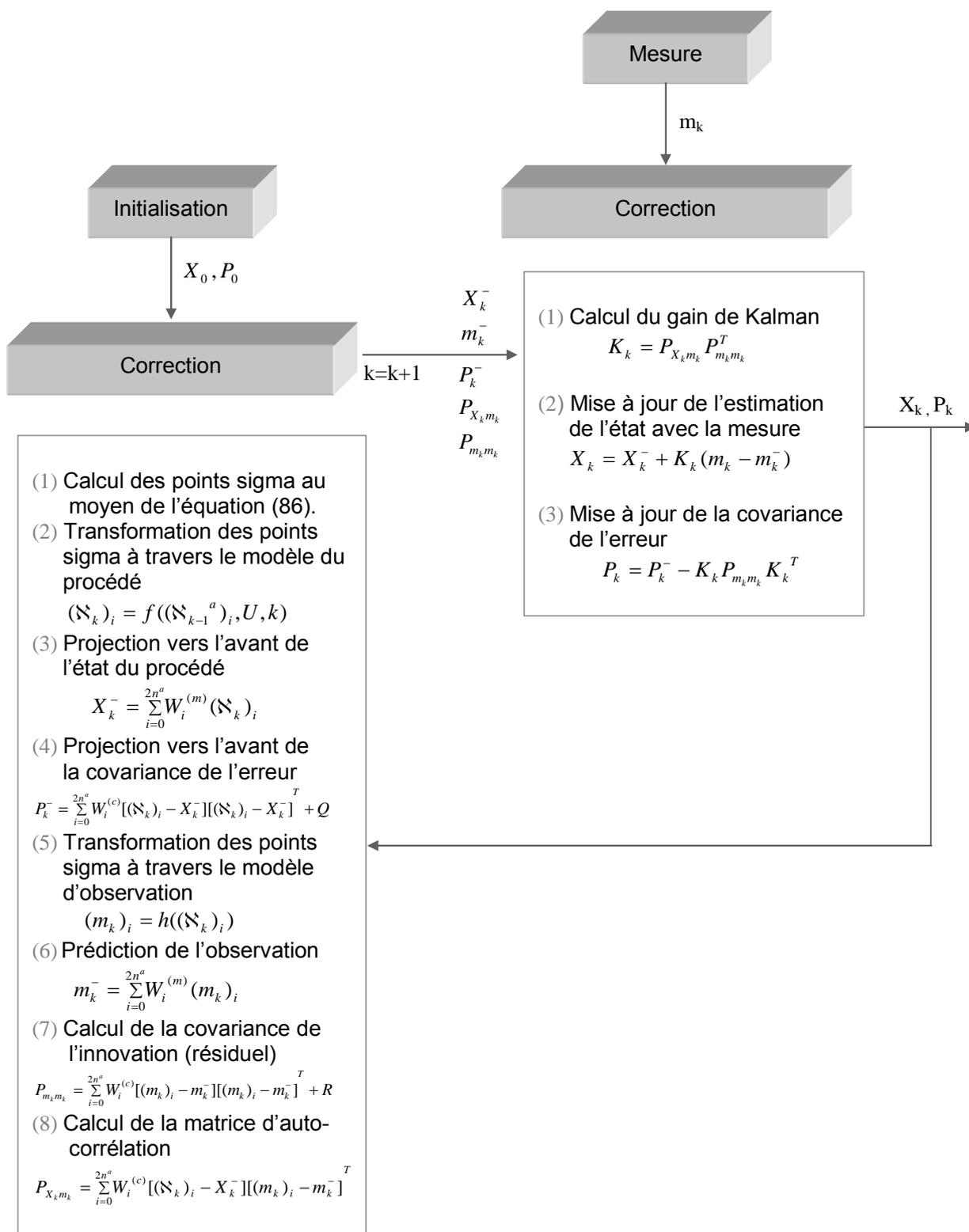


Figure 3.6 Cycle de prédiction-corréction du filtre de Kalman inodore (UKF).

Principalement, EKF a deux inconvénients le premier est la dérivation des matrices jacobienes, l'approximation linéaire des fonctions non linéaires peut être complexe ce qui induit à une implémentation ardue, et le second est que cette linéarisation peut causer l'instabilité du filtre si la période d'échantillonnage n'est pas suffisamment petite.

Bien que beaucoup de travaux ont été élaborés [54, 59, 60] dans un sens où ils favorisent l'UKF pour sa précision dans la prédiction de l'état, et puis que celui-ci est plus facile à implémenter que l'EKF pour les raisons citées auparavant, ces systèmes marchent le plus souvent avec le suivi de position et ne prend pas en considération l'orientation, c'est déjà un inconvénient pour notre travail qui consiste à estimer la position du robot et son orientation.

Par ailleurs, toute une étude comparative entre les deux filtres EKF et UKF a été faite dans [58], et les résultats prouvent que les deux filtres ont des performances très similaires. Puisque expérimentalement parlant il a été démontré que la précision de l'estimation est presque la même pour les deux filtres. En plus la vitesse d'exécution de l'algorithme de l'EKF est plus rapide que pour l'UKF, raison pour laquelle l'algorithme de l'UKF prend plus de temps pour trouver l'estimé puisqu'il doit traiter tous les points sigma, ce qui est un inconvénient majeur pour notre application surtout que le robot doit se localiser tout en se déplaçant où le temps de calcul est très important. Finalement l'un des principaux avantages de l'UKF est qu'il ne nécessite en aucun cas le calcul des matrices jacobienes, dans certaines applications l'évaluation des matrices jacobienes rend l'implémentation de l'algorithme difficile. Or que dans notre application le calcul de la matrice jacobienne est plutôt facile basé sur la structure du modèle.

C'est pourquoi, l'UKF ne nous fournit aucun bénéfice de plus dans notre cas, donc l'EKF est le plus approprié à l'utiliser.

D'autres filtres développés en parallèle par d'autres chercheurs appartiennent à la même famille des « *Sigma Point Kalman Filters* » (SPKF) d'estimation Bayésienne sans dérivation, cette famille comprend :

- « *Unscented Kalman filter* » (UKF) [51-55]
- « *Central difference Kalman filter* » (CDKF) [51, 55]
- « *Square-root SPKFs* » [51, 57]

- « *Sigma-point particle filter* » [51]
- « *Gaussian mixture sigma-point particle filter* » [51,56]

Les filtres SPKF autres que l'UKF sont très similaires dans l'intention, i.e. estimer la distribution plutôt que de linéariser le modèle, leur différence résidant principalement dans la manière de choisir les points sigmas (c'est pourquoi on se contente de les citer).

3.2.3.2 Suivi de plusieurs hypothèses

Contrairement au suivi de position qui utilise l'estimation précédente de la position pour sélectionner l'une des positions caractérisées par les perceptions et ignorer les autres. Les méthodes de suivi de plusieurs hypothèses tiennent compte de toutes ces positions possibles à chaque étape dans son traitement. Ces positions conduisent à des hypothèses qui peuvent être mises à jour en fonction des données proprioceptives et qu'il faut comparer, afin de choisir la plus pertinente à chaque étape. C'est ce qui fait de ces méthodes des méthodes de localisation globales.

Ces méthodes regroupent un grand nombre d'approches connues sous différents noms, mais les plus réputés pour leurs robustesses et leurs usages sont les filtres de Kalman à plusieurs hypothèses, la localisation de Markov (Markov localization) et les filtres particuliers. Ces méthodes partagent les mêmes fondations mathématiques (filtre bayésien).

▪ *Filtres de Kalman à plusieurs hypothèses*

Ce suivi de plusieurs hypothèses peut être réalisé en gérant une liste des hypothèses en question. Ces différentes hypothèses sont représentées par une distribution de probabilité de présence du robot sur l'ensemble des positions possibles de la carte, qui est une combinaison de plusieurs distributions gaussiennes unimodales [61, 62]. Cette représentation permet donc de considérer chacune des positions au sein de la carte comme une position possible du robot dont il faut évaluer la crédibilité.

Lorsque des données proprioceptives sont disponibles, chaque hypothèse est simplement mise à jour de manière à refléter le déplacement du robot en déplaçant les probabilités d'une position à une autre (figure 3.7). Lorsque de nouvelles perceptions sont disponibles, l'ensemble des positions de la carte susceptibles de

correspondre à ces perceptions est déterminé. Cet ensemble est ensuite comparé à l'ensemble des hypothèses. Si une hypothèse correspond à une position perçue, cette hypothèse est alors mise à jour en utilisant les perceptions et cela par le filtre de Kalman. Les positions perçues qui ne correspondent à aucune hypothèse sont utilisées pour créer de nouvelles hypothèses associées à la position correspondante. Lorsqu'on utilise une telle distribution de probabilité, la position du robot calculée est en général donnée par l'hypothèse ayant la plus forte probabilité

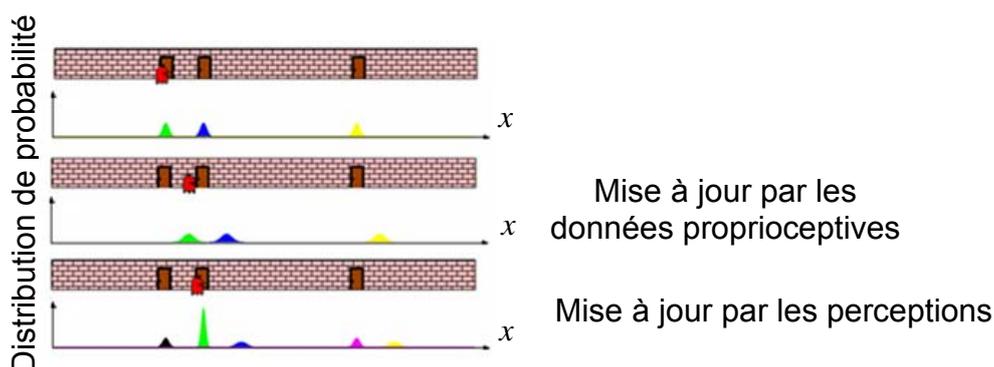


Figure 3.7 Illustration de la méthode de suivi de plusieurs hypothèses

- *Localisation de Markov*

Cette approche consiste à discrétiser l'environnement et à donner à chaque position discrète une probabilité approximant la valeur de la distribution continue. Donc la densité de probabilité est approximée par des histogrammes ("*piecewise constante fonction*"), ce qui permet de représenter des distributions multimodales.

Pour expliquer l'idée basique de la localisation de Markov [61, 63], on pourra utiliser un environnement 1D où le robot se déplace horizontalement et ne tourne pas (figure 3.8). Supposant que le robot est placé à une position aléatoire et que la carte de l'environnement est donnée.

Au début, comme il n'existe pas de probabilité concernant la position du robot en cet état, la localisation de Markov représente cet état en utilisant une distribution uniforme (la figure 3.8 (a)). En se déplaçant, le robot fait usage de ses capteurs. Quand celui-ci découvre la porte (la figure 3.8 (b)), la localisation de Markov augmente la probabilité des positions proches des portes et diminue celle des autres positions, et cela en utilisant la carte de l'environnement. Donc la probabilité résultante est multimodale.

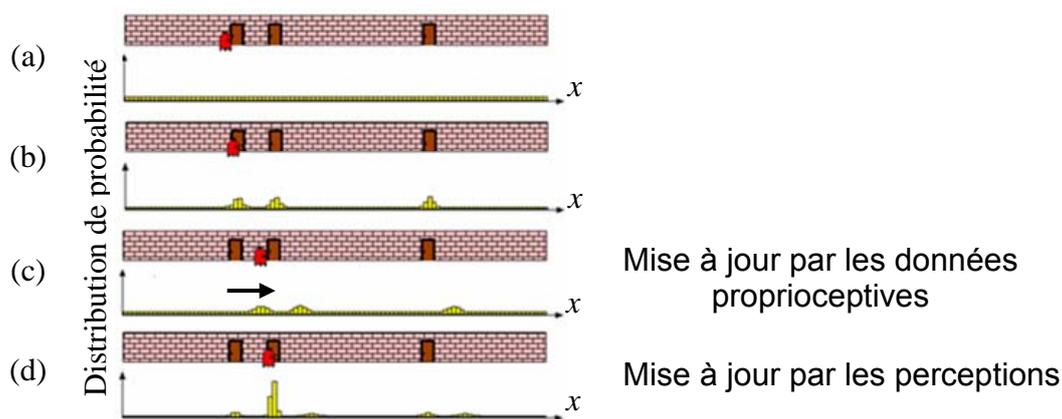


Figure 3.8 L'idée basic de la localisation de Markov

Notons que les positions qui ne sont pas proches des portes, elles gardent des probabilités non nulle, car les mesures des capteurs sont incomplètes.

Supposant encore que le robot se déplace vers la droite, la localisation de Markov déplace alors la probabilité (selon la figure 3.8 (c)) à cause des incertitudes propres au déplacement du robot. Les probabilités deviennent moins certaines que celles du diagramme précédent. Finalement, quand le robot observe la deuxième porte par usage de ses capteurs cette observation est multipliée par la probabilité actuelle. Cette multiplication donne la probabilité finale (comme illustré dans la figure 3.8 (d)), et la probabilité sera centrée devant la deuxième porte. Ainsi le robot s'est localisé soit même et sans connaissance a priori de la position initiale.

L'inconvénient majeur de cette méthode est sa complexité de calcul et sa difficulté d'implémentation, c'est ce qui a conduit à utiliser les filtres particulaires.

- *Filtres particulaires*

Les filtres particulaires sont généralement connus sous l'appellation "*localisation de Monté Carlo*" (monte Carlo Localisation ou LMC). La localisation de Monté Carlo utilise une technique d'échantillonnage aléatoire pour estimer la densité de probabilité des positions du robot [61,64]. LMC représente cette densité de probabilité par un ensemble d'échantillons pondérés, aussi appelés particules (figure 3.9). Et cela sans avoir à discrétiser l'environnement comme c'était le cas pour la localisation de Markov.

Chaque particule sera considérée comme une position possible du robot dans l'environnement. Tous comme pour la localisation de Markov, la distribution de

probabilité sera mise à jour par les données proprioceptives en déplaçant la probabilité d'une position à une autre. Et pour chaque perception, l'ensemble des positions susceptible de correspondre à ces perceptions est déterminé.

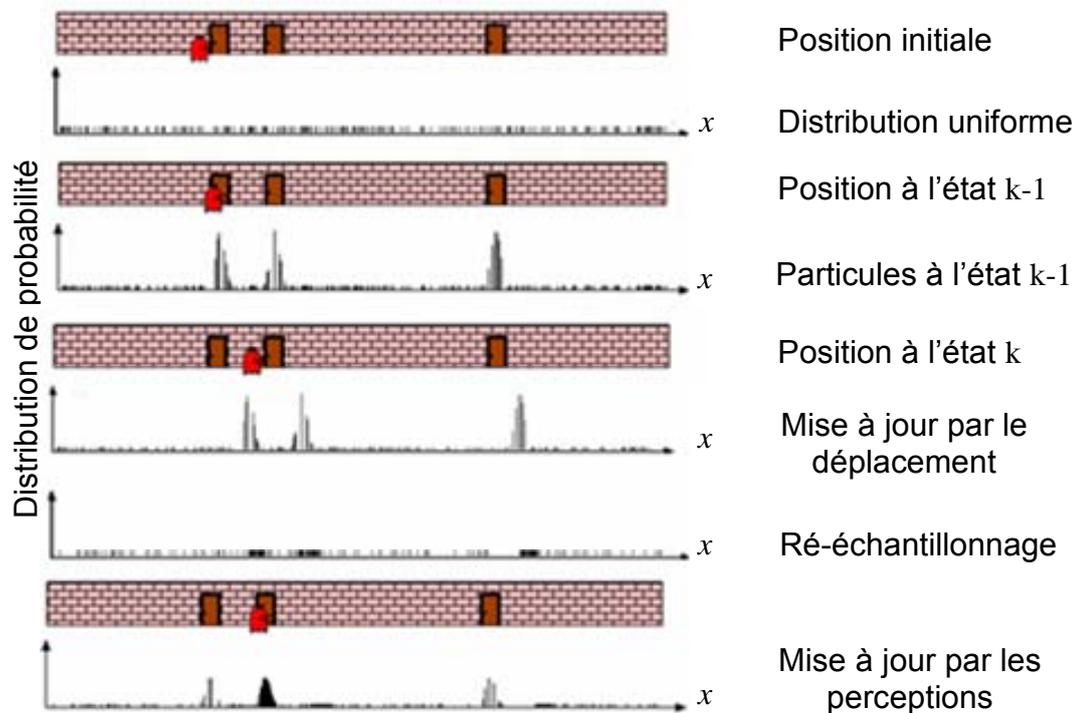


Figure 3.9 Illustration du fonctionnement de l'algorithme de filtrage particulaire

Ces méthodes de suivi de plusieurs hypothèses sont extrêmement robustes en pratique et mettent en oeuvre un système de localisation complètement autonome, ne dépendant d'aucune intervention extérieure et dépendant faiblement des conditions initiales. Ces qualités reposent toutefois de manière importante sur le fait que la carte de l'environnement est complète. En effet, les systèmes de suivi de plusieurs hypothèses nécessitent une estimation correcte des crédibilités des différentes positions possibles. Or une carte partielle de l'environnement rend une telle estimation à partir des perceptions difficile. Pour cette raison, ces systèmes sont en général utilisés pour la localisation sur des cartes construites dans une phase préalable.

L'utilisation de la localisation de Markov ou la localisation par filtre de Kalman à plusieurs hypothèses permet la localisation globale mais elle conduit à des implémentations complexes et difficiles à maîtriser. Les filtres particulaires sont une

bonne solution en particulier pour leur robustesse et leur simplicité d'implémentation, mais peuvent être difficiles à régler et lourds en temps de calcul. Ce qui est un inconvénient majeur pour notre application en temps réel.

3.4 Conclusion

Nous avons vu au cours de ce chapitre une grande variété de méthodes de localisation qui diffèrent par les capteurs utilisés, les modèles d'environnement et le mode d'association entre les mesures et le modèle. Les choix sont établis en fonction des applications auxquelles sont dédiées. La localisation à l'estime est utilisable en environnement d'extérieur ou d'intérieur mais la distance de déplacement doit être réduite pour limiter les accumulations d'erreurs. La technique par balisage est employée indifféremment à l'intérieur et à l'extérieur mais oblige le mobile à rester dans une zone prédéfinie. La mise en correspondance par carte nécessite la connaissance exacte de la topologie de l'environnement, sauf dans le cas où la carte de l'environnement est mise à jour au fur et à mesure de son exploration. Enfin, les méthodes de localisation hybride qui fusionnent les données proprioceptives et les données extéroceptives les rend plus précises et plus riches en information, mais des fois elles peuvent être complexes, difficiles à implémenter et même lourdes en temps de calcul et cela selon l'algorithme utilisé.

En vu des avantages et inconvénients des différentes méthodes abordées au cours de ce chapitre, et des outils dont nous disposons (capteurs utilisés pour la perception, modèle d'environnement utilisé, etc.), notre choix s'est porté sur la méthode de mise en correspondance. C'est une méthode précise qui fait partie des méthodes les plus appropriées pour un robot mobile se déplaçant dans un environnement modélisé en grille d'occupation. Une autre méthode qui pourrait aussi très bien s'adapter à notre cas est le filtre de Kalman. C'est un algorithme simple à implémenter et n'est pas lourd en temps de calcul.

CHAPITRE 4

ALGORITHME DE LOCALISATION DEVELOPPE

4.1 Introduction

Après avoir survolé les méthodes de localisation les plus utilisées en robotique mobile, nous nous focalisons dans ce chapitre sur deux méthodes, celles les plus adaptées pour notre cas. À savoir, la méthode de la mise en correspondance et le filtre de Kalman étendu. Elles sont étudiées en détail, puis implémentées sur le robot mobile de type à traction synchrone B21r, muni de deux ceintures à ultrasons, une ceinture d'infrarouge et un laser. Ces capteurs sont modélisés selon un modèle basé sur les grilles d'occupation et qui va servir pour la reconstruction de l'environnement lors de la mise en correspondance des grilles. Enfin les résultats expérimentaux obtenus suite aux deux méthodes de localisation y sont présentés.

La localisation par la mise en correspondance des grilles consiste à trouver la meilleure correspondance entre la grille locale (l'espace perceptible par le robot depuis sa position) et la grille globale (l'environnement). Elle passe par les étapes suivantes :

- Construire la grille globale en utilisant le système de perception du robot pendant la phase d'exploration d'un environnement inconnu.
- Construire la grille locale par usage des capteurs embarqués sur le robot. Elle représente la partie de l'environnement qui l'entoure.
- Faire la mise en correspondance entre la carte locale et la carte globale pour toutes les translations et orientations possibles de la grille locale (limitée par une zone de recherche définie par l'utilisateur), afin de trouver celle qui correspond le mieux au robot.

Pour la création des grilles, une modélisation des capteurs utilisés et de l'environnement est nécessaire. La méthode citée est une méthode de localisation absolue (utilisant des capteurs extéroceptifs), n'empêche qu'elle peut avoir recours à des capteurs proprioceptifs tel que les odomètres pour avoir une estimation de la position autour de laquelle elle va définir sa zone de recherche. Mais ces capteurs

sont généralement connus pour leurs erreurs accumulatifs, ce qui peut fausser les résultats obtenus lors de la phase de localisation.

Une autre méthode de localisation qu'on pourrait aussi bien appliquer et qui peut remédier aux problèmes des odomètres est le filtre de Kalman étendu. C'est une méthode de localisation hybride qui fusionne les données issues des capteurs proprioceptifs et celles issues des capteurs extéroceptifs pour fournir une position estimée du robot.

4.2 Le robot mobile B21r

Le robot mobile B21r que nous utilisons est une plateforme expérimentale construite par la société *iRobot* pouvant se déplacer dans un milieu d'intérieur.

4.2.1 Architecture matérielle

Le robot synchro drive B21r se déplace grâce à ses quatre roues disposées en forme de carré. Il a une forme cylindrique de dimension de 52.5 x 106cm, pèse environ 122.5kg et peut supporter une charge supplémentaire de 90kg, et se déplace à une vitesse max de 90cm/s.

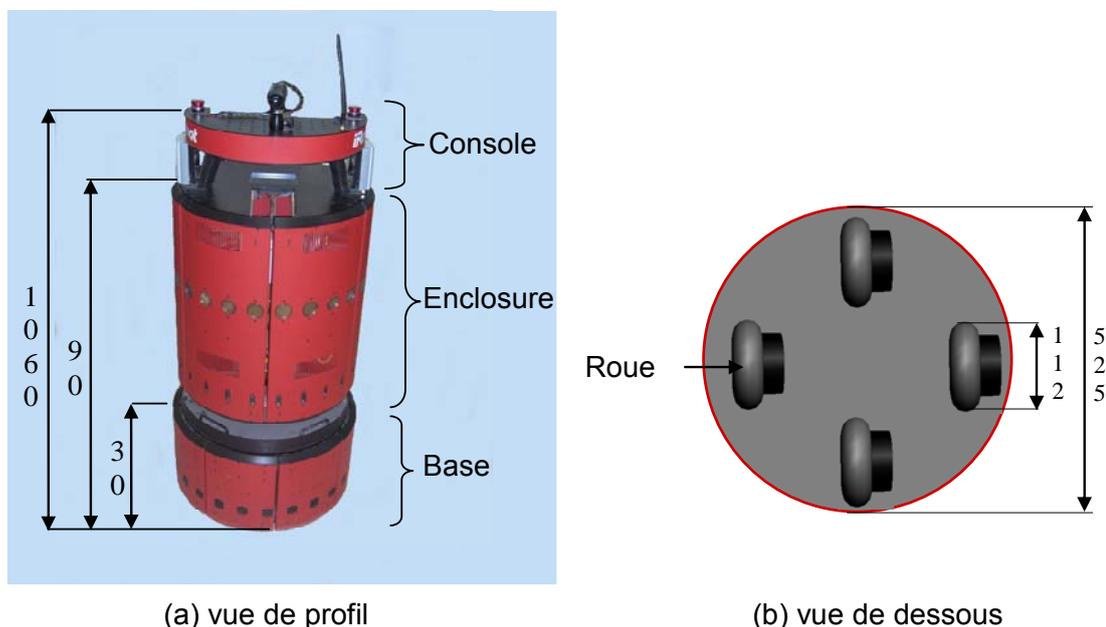


Figure 4.1 la géométrie du robot mobile B21r

Il est constitué de trois parties : la base, l'enclosure, et la console. L'enclosure et la console sont physiquement attachées, alors que l'enclosure et la base sont reliées par une articulation se qui permet une rotation libre de l'enclosure indépendamment

de la base. L'enclosure comporte deux ceintures de capteurs, une ceinture de capteurs à ultrasons de type polaroid et une ceinture de capteurs infrarouges de type GP2D02, la base quant à elle comporte la deuxième ceinture de capteurs à ultrasons, et la console c'est la partie du robot qui supporte la caméra CCD (N/B) et l'interface *rFLEX*. Un télémètre laser est aussi embarqué sur le robot, celui-ci est placé dans la partie inférieure de l'enclosure, donc l'orientation du laser suivra celle du robot.

Le robot mobile *B21r* est aussi muni d'odomètres qui sert à mesurer le déplacement et l'orientation effectués par le robot. Les données provenant de ces capteurs sont présentées de façon à donner la position du robot mobile en X et Y ainsi que son orientation, de ce fait la position du robot peut être directement lue.

Dans ce type de robots, toutes les roues tournent en même temps et roulent de la même façon, c'est ce qui mène à un comportement *non holonome*. De sorte que les quatre roues pointent dans la même direction et roulent à la même vitesse. Ceci est typiquement réalisé par l'utilisation d'une collection complexe de ceintures qui lie physiquement les roues ensemble et synchronise leur mouvement, comme illustré dans la section 1.3.6. Pour la locomotion, le robot possède deux moteurs indépendants (un pour la translation et l'autre pour la rotation). L'ensemble du mécanisme : moteurs, roues et la collection de ceintures se situent dans la base, qui ne tourne pas lors de la rotation des roues, contrairement à l'enclosure qui tourne d'une façon synchrone avec l'orientation des roues.

Les vitesses (rotation et translation) du robot à traction synchrone (synchro drive) peuvent être positive ou négative, cela signifie qu'il peut se déplacer en avant ou en arrière, tourner sur lui-même ou en suivant un virage.

Tous les capteurs et actionneurs sont connectés au bus *FARnet* d'*iRobot* et contrôlés par un ordinateur embarqué. Ce dernier est un Pentium PC avec dual 800Mhz processeurs. Il est utilisé pour sauvegarder et exécuter tous les programmes et données sur le robot mobile. Il utilise aussi LINUX comme système d'exploitation et communique avec un autre ordinateur extérieur par l'intermédiaire de (figure 4.2) :

- Deux antennes BreezeCom pour une communication Ethernet sans fils.
- Un port Ethernet (RJ45) pour une connexion directe à un réseau local.
- Un port série pour une communication série avec un autre ordinateur.



Figure 4.2 Console du Robot *B21r*

4.2.2 Architecture logicielle

Le robot *B21r* est contrôlé par le système *rFlex* de la société *iRobot* fournissant des contrôles standardisés et une interface utilisateur facile à utiliser et conviviale permettant l'interaction avec l'utilisateur à travers un ensemble de menus permettant la gestion des capteurs ultrasonores, les moteurs et autres systèmes embarqués. Ce système est considéré comme le système nerveux central du robot mobile *B21r* où les câbles, nœuds et Hubs sont facile à configurer et son réseau est très optimisé pour lier les capteurs embarqués et les actionneurs à son ordinateur du traitement central.

Le contrôle software se fait par *MobilityTM*, logiciel distribué, toolkit orienté objet pour la construction de programmes de contrôle d'un seul ou plusieurs robots de la famille *iRobot*. Le logiciel *Mobility* utilisé est basé sur CORBA, il collecte les données en provenance des capteurs embarqués, les interprète et les exécute. En utilisant CORBA tous les capteurs et actionneurs peuvent être vu comme des ressources Internet. C'est une librairie composée de programmes écrits en C et C++ contrôlant le hardware du robot mobile *B21r*.

La connexion avec les différents capteurs embarqués se fait en exécutant le programme serveur qui réagit réciproquement avec le capteur. Ce dernier se connectera à son tour au name server. Le capteur qui peut être donc connecté et contrôlé par un simple programme écrit en C incluant les fonctions et procédures fournies avec *Mobility*.

4.2.3 Comportement du B21r

La direction et l'orientation de déplacement du robot mobile B21r dépendent de sa vitesse linéaire v et sa vitesse de rotation Ω comme c'est illustré dans les cas de la figure 4.3.

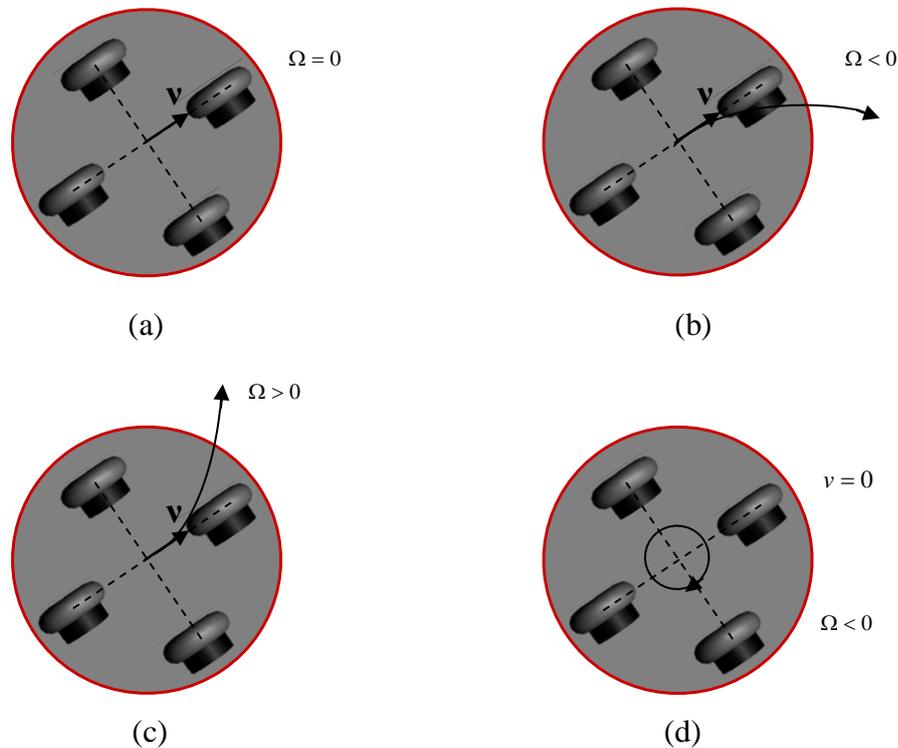


Figure 4.3 Quatre différents types de mouvement réalisé par le robot mobile B21r.

Le robot peut se déplacer en ligne droite si $v \neq 0$ et $\Omega = 0$, en avant ou en arrière selon que v soit positive ou négative (Figure 4.3.a), et le centre instantané de rotation (*CIR*) se trouve à l'infinie. Il peut aussi faire une rotation pure sur place (le robot tourne sur lui-même) quand $v = 0$ et $\Omega \neq 0$ dans le sens trigonométrique (si $\Omega > 0$), et dans le sens des aiguille d'une montre (si $\Omega < 0$) (Figure 4.3.d), dans ce cas le *CIR* se trouve au centre de gravité (*C*) du robot. Enfin, il peut se déplacer suivant un virage dans le cas où $v \neq 0$ et $\Omega \neq 0$, à droite (avec $\Omega < 0$) (Figure 4.3.b) et à gauche (avec $\Omega > 0$) (Figure 4.3.c), et cela peut être en avant ou en arrière selon que la vitesse linéaire soit positive ou négative, où le robot tourne autour du *CIR* avec un rayon $R = \frac{v}{\Omega}$.

4.3 Modélisation de l'environnement

4.3.1 Modélisation de l'environnement basée sur les capteurs à ultrasons

4.3.1.1 Le Capteur à ultrasons

Comme nous l'avons décrit au chapitre 2, le capteur à ultrasons présente l'avantage de donner directement une information de distance mais son inconvénient est que cette mesure est assez imprécise. En effet, l'angle d'ouverture (de 20 à 30°) introduit un facteur d'incertitude concernant la direction dans laquelle se situe l'obstacle perçu. Pour une mesure d donnée par le capteur, en considérant un espace à deux dimensions et en négligeant les incertitudes sur la mesure, la région correspondant aux positions possibles de l'obstacle détecté prend la forme d'un arc centré sur le capteur, avec un rayon égal à d . Les dimensions de l'objet sont aussi masquées par cette propriété du capteur à ultrasons ; il peut s'agir d'un mur ou seulement d'un objet de petite taille. La figure 4.4 illustre la situation où deux obstacles possibles produisent la même mesure.

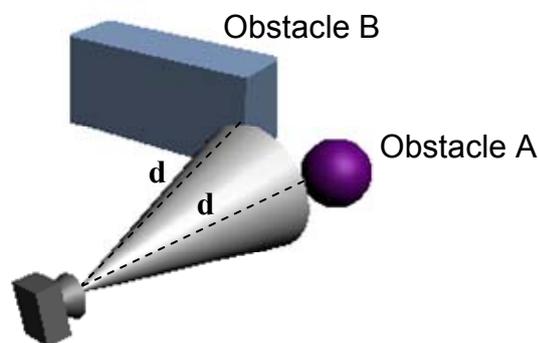


Figure 4.4 Deux obstacles renvoyant la même mesure

Une incertitude sur la distance mesurée, appelée également incertitude radiale est aussi présente. Elle provient des phénomènes atmosphériques (température, courants d'air, etc.) pouvant modifier la vitesse de l'onde. Pour les capteurs utilisés comme émetteur et récepteur tels que ceux que nous utilisons (Polaroid), une distance minimale détectable est définie. Cette distance est déduite du temps

nécessaire pour que la membrane du capteur puisse se stabiliser après l'émission de l'onde.

Dans le domaine de perception du capteur, des phénomènes tels que la spécularité, les réflexions multiples ou la diaphonie peuvent intervenir, nous allons en décrire les causes et les effets.

- *La spécularité*

La spécularité, encore appelée effet miroir peut provoquer la non détection des obstacles. En effet, la fiabilité des mesures ultrasonores est très dépendante de la texture de la surface réfléchissante de l'objet. Les surfaces rugueuses retournent l'onde ultrasonore produite par le capteur quel que soit l'angle d'incidence, contrairement aux surfaces lisses qui ne renvoient que les ondes ayant un angle d'incidence proche d'un angle droit (figure 4.5).

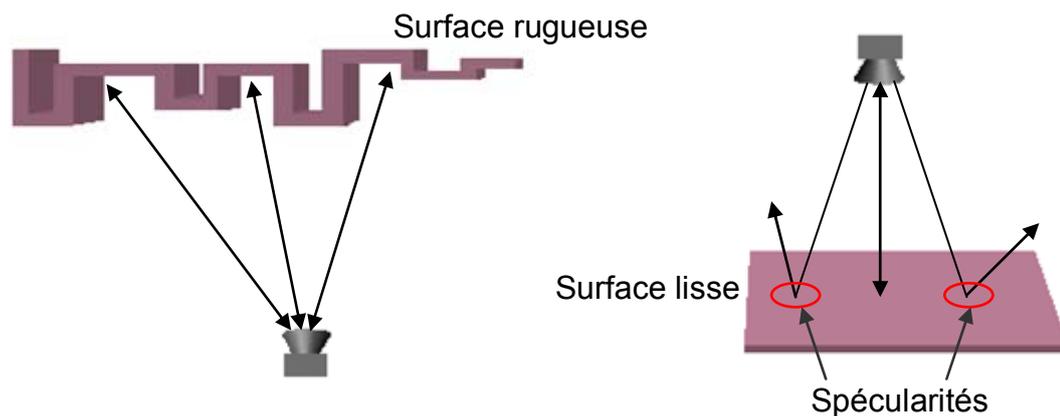


Figure 4.5 Phénomène de spécularité

Si une spécularité intervient, la mesure issue du capteur concerné est inexploitable, dans ce cas l'onde ne se réfléchit pas sur un autre obstacle pour revenir sur le capteur. Par contre, si la spécularité est suivie d'une ou plusieurs réflexions sur les obstacles, nous avons alors des réflexions multiples.

- *Les réflexions multiples*

Le phénomène de spécularité peut se produire plusieurs fois, il s'ensuit des réflexions multiples. Le signal ultrasonore peut donc se réfléchir sur plusieurs surfaces avant de retourner au capteur. Par ce fait, l'obstacle paraît plus loin que son emplacement réel (figure 4.6). Les mesures issues de réflexions multiples ne

correspondent pas à un obstacle de l'environnement et risquent de fausser la localisation en associant la mesure à un autre obstacle situé plus loin.

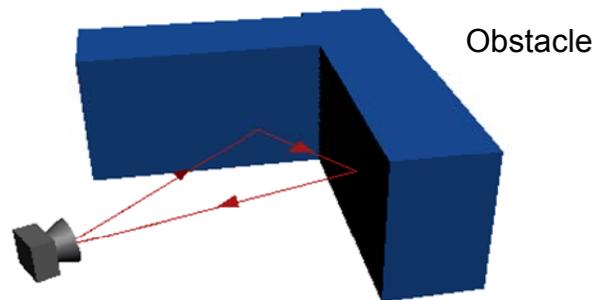


Figure 4.6 Exemple de réflexions multiples

- *La diaphonie*

La diaphonie intervient quand un capteur perçoit l'onde émise par un autre. Supposons qu'un capteur x émette une onde ultrasonore. Si le capteur y se trouve en mode de réception, il interprétera l'onde émise par le capteur x et renvoyée par l'obstacle comme étant la sienne, et percevra un objet plus proche qu'il ne devrait l'être. Le capteur y pourrait recevoir cette onde par un chemin direct, tout comme il pourrait la recevoir par un chemin critique indirect (réflexions multiples) (figure 4.7). La encore ce type de mesures représente des obstacles fictifs et peut fausser la localisation.

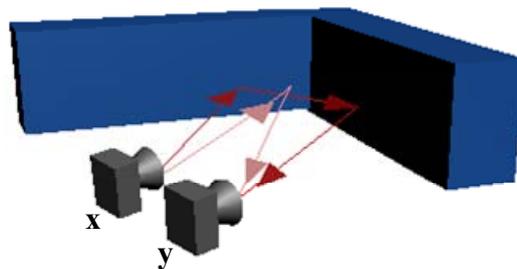


Figure 4.7 Exemples de diaphonie

La spécularité, les réflexions multiples et la diaphonie sont des phénomènes parasites plus ou moins imprévisibles contrairement aux incertitudes angulaires qui sont toujours présentes dans la mesure.

4.3.1.2 Modélisation des mesures ultrasonores

Un moyen de modéliser le capteur à ultrasons consiste à déterminer la fonction densité de probabilité $p(r/d)$ de la mesure de profondeur r connaissant la distance d entre le capteur et l'obstacle.

En supposant le capteur idéal, la mesure est interprétée comme étant la distance au plus proche obstacle dans la direction de l'axe acoustique (figure 4.8).

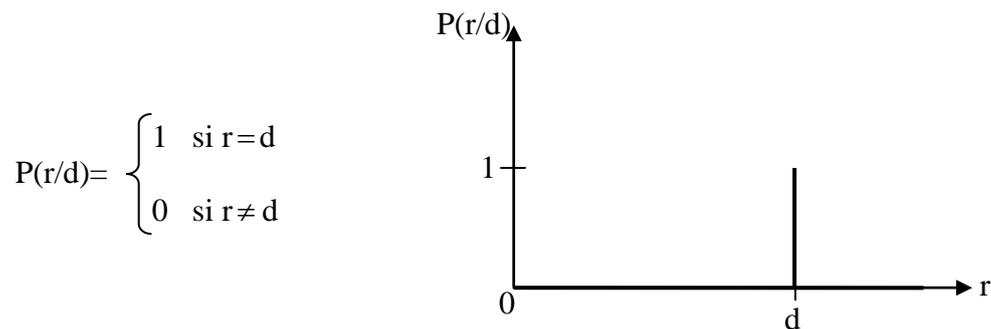


Figure 4.8 Fonction de probabilité d'un capteur idéal

En enregistrant ce model dans une grille, seule la cellule située sur l'axe acoustique du capteur à la distance mesurée est modifiée (figure 4.9). De cette façon, l'incertitude sur la distance mesurée peut être contenue dans la cellule.

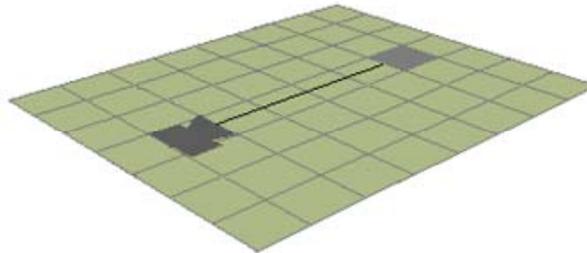


Figure 4.9 Grille pour capteur idéal.

Ce model a été utilisé pour effectuer des tâches d'évitement d'obstacle pour un robot mobile [65]. Les cellules de la grille représentant l'environnement sont initialisées à la valeur 0. Puis, au cours du déplacement du robot, chaque obstacle détecté incrémente la valeur de la cellule correspondante. Ainsi au fur et à mesure du déplacement, les obstacles sont repérés par des cellules dont les valeurs augmentent, ce qui permet de les éviter.

La modélisation de ce modèle, prenant en compte l'espace libre entre le capteur et l'obstacle, a permis de réaliser la localisation d'un robot mobile [66]. Comme précédemment, la valeur initiale de chaque cellule de la grille est nulle. Puis, lors d'une acquisition, les valeurs des cellules correspondant à l'espace libre sont décrémentées tandis que la valeur de la cellule représentant l'obstacle est augmentée de la valeur 3, les valeurs des cellules restantes n'étant pas modifiées (figure 4.10).

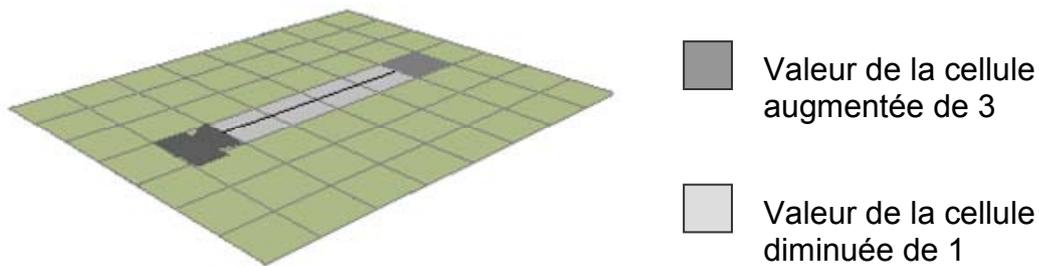


Figure 4.10 Grille pour un capteur idéal prenant en compte l'espace libre.

Le problème de ces modèles est qu'ils ignorent l'incertitude angulaire des capteurs à ultrasons. Un capteur de type polaroïd a un cône d'émission de 20-30°, les réflexions retournées au capteur peuvent provenir d'objets situés dans la périphérie du cône. Ramener la mesure simplement sur l'axe de tir, considérant le capteur à ultrasons comme un capteur ponctuel (comme le laser), peut donc être source d'erreurs car beaucoup d'obstacles seront mal ou pas perçus (figure 4.11).

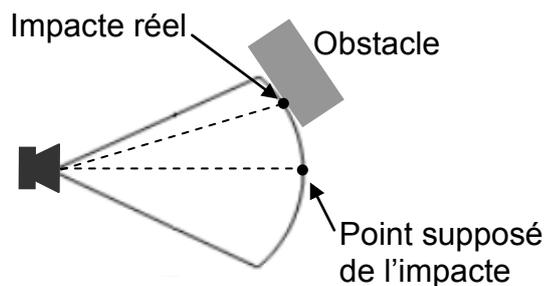


Figure 4.11 Exemple d'obstacle mal perçu

L'incertitude sur les mesures intervient donc sur la profondeur r , mais aussi sur l'angle θ par rapport à l'axe de tir. Une première approche utilisée pour la détection d'obstacles lors de la navigation de robots mobiles consiste à modifier le premier modèle (figure II-7) en un arc de cercle (figure 4.12) [67] ; la valeur attribuée à chaque cellule étant inversement proportionnelle au nombre de cellules recouvertes par l'arc entier.

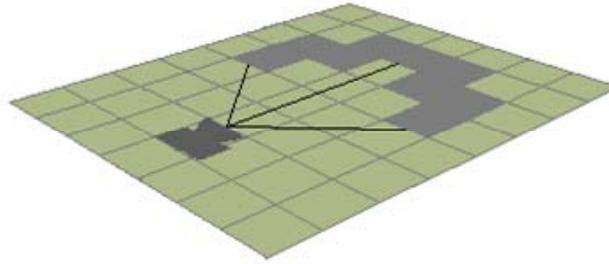


Figure 4.12 Grille prenant en compte l'ouverture angulaire du capteur

Une autre méthode pour représenter la mesure sur le secteur entier consiste à estimer que si un objet est détecté par le capteur à ultrasons, il est plus probable que cet objet soit proche de l'axe acoustique que de la périphérie du cône [20]. Ce qui favorise l'axe acoustique par rapport à la périphérie du cône ultrasonore en affectant des valeurs plus grandes aux cellules situées au centre.

En considérant que sans informations complémentaires sur la mesure ultrasonore tous les points de l'arc sont équivalents pour une position possible de l'obstacle, il nous paraît logique d'attribuer la même valeur à chacune des cellules représentant le même état (libre ou occupé). Notre méthode utilise cette idée et attribue aux cellules une des trois valeurs définies par :

$C_L=0$: valeur d'une cellule sur laquelle aucune information n'est disponible.

$C_L=C_L-1$: valeur d'une cellule représentant l'espace libre avant l'obstacle détecté.

$C_L=C_L+1$: valeur d'une cellule représentant un obstacle.

Le modèle du capteur à ultrasons ainsi que la grille correspondante sont représentés par la figure 4.13.

Ce modèle a l'avantage d'éviter le traitement des mesures à l'aide de lois de probabilités lourdes à gérer. La zone occupée est une limite donnée par les capteurs à ultrasons, c'est pourquoi l'arc n'est réalisé que sur une épaisseur d'une ligne ; du point de vue du robot, la zone à l'intérieur des obstacles, bien qu'occupée est considérée comme inconnue. Une seule carte est nécessaire pour représenter l'espace occupé et l'espace libre ; la différence se fait sur le signe de la valeur de la cellule.

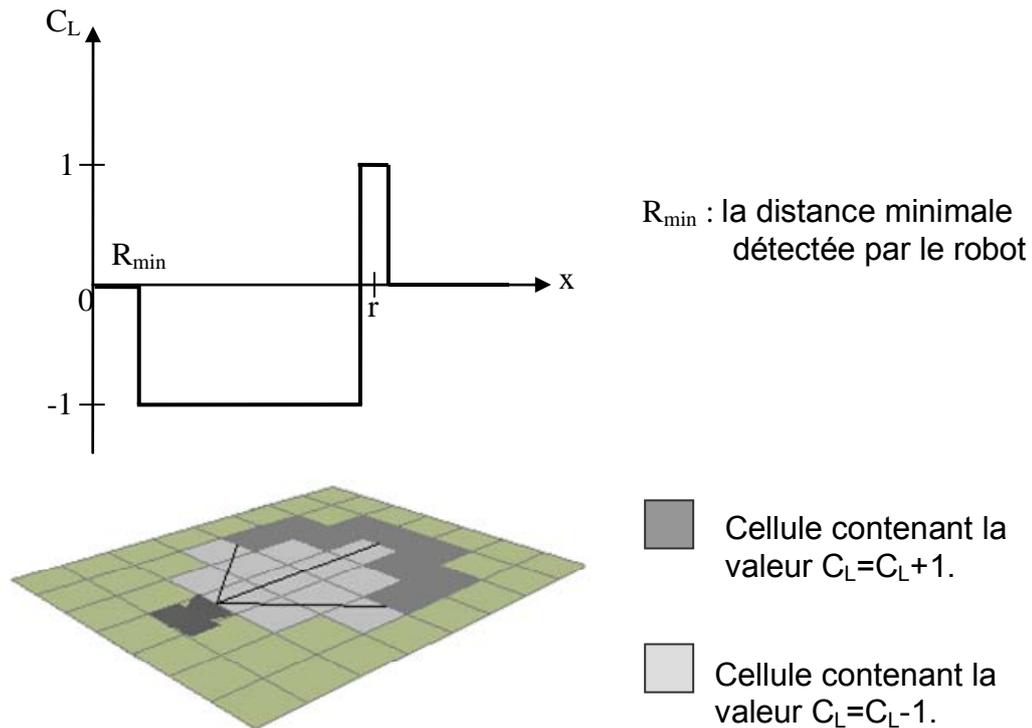


Figure 4.13 Modélisation du capteur à ultrasons

4.3.1.3 Création de la grille

Pour construire la grille, le robot utilise les deux ceintures de capteurs ultrasonores comportant chacune 24 capteurs. La donnée fournie par le capteur à ultrasons de type polaroid est une distance. Elle représente la distance entre l'obstacle détecté et le capteur en question.

Notre but est de trouver à partir de cette distance d (comme seule information disponible), la cellule correspondante à la position de l'obstacle. Nous commençons par trouver la relation qui lie la position de l'obstacle au repère mobile (repère lié au robot mobile). Puis nous déterminerons le lien entre le repère mobile et le repère univers (lié à l'environnement) où la position de chaque cellule est connue. Enfin nous déduisons la position de l'obstacle par rapport au repère univers pour trouver la cellule qu'il occupe.

Dans le cas du robot *B21r*, la position de chaque capteur à ultrasons par rapport au repère mobile R^R (O' , Y^R , X^R) est connue (voir Appendice A). Cette position est définie par les paramètres r_c et β_c (Figure 4.14).

Le cône de chaque capteur décrit une ouverture de $\left[-\frac{\alpha_c}{2}, \frac{\alpha_c}{2}\right]$ avec $\alpha_c = 30^\circ$.

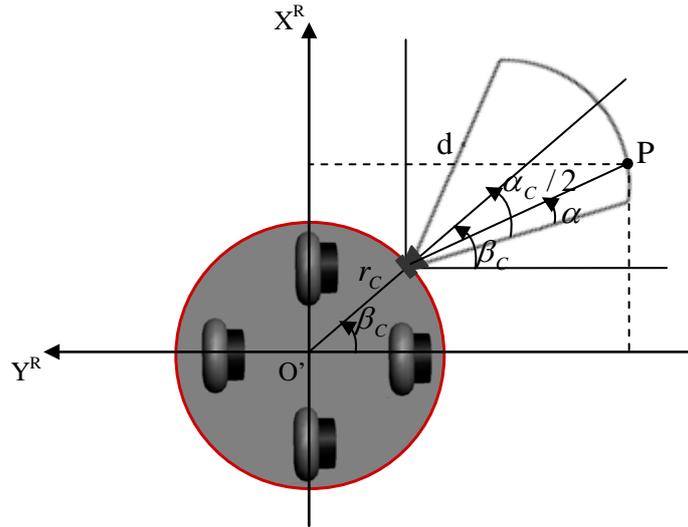


Figure 4.14 Configuration du cône d'émission d'un capteur à ultrasons et de sa position par rapport au robot mobile

Ainsi pour chaque capteur, la position de l'obstacle détecté par rapport au repère mobile R^R est donnée par la relation suivant :

$$\begin{cases} y_c^R = -r_c \cos \beta_c - d \cos(\beta_c - \frac{\alpha_c}{2} + \alpha) \\ x_c^R = r_c \sin \beta_c + d \sin(\beta_c - \frac{\alpha_c}{2} + \alpha) \end{cases} \quad (97)$$

Cette position peut être exprimée dans le repère univers grâce à la relation de passage du repère mobile R^R (O' , Y^R , X^R) vers le repère univers R^u (O , X , Y) (voir Appendice B) donnée par :

$$\begin{cases} X = x_0 - yr \cos \varphi_0 - xr \sin \varphi_0 - X^R \sin(\theta r + \varphi_0) - Y^R \cos(\theta r + \varphi_0) \\ Y = y_0 - yr \sin \varphi_0 + xr \cos \varphi_0 + X^R \cos(\theta r + \varphi_0) - Y^R \sin(\theta r + \varphi_0) \end{cases} \quad (98)$$

Avec (x_0, y_0, φ_0) la position initiale du robot mobile et $(xr, yr, \theta r)$ la position fournie par le système odométrique du robot.

Par conséquent, la position de l'obstacle par rapport au repère univers R^u pour

$\alpha \in \left[-\frac{\alpha_c}{2}, \frac{\alpha_c}{2}\right]$ est donnée par :

$$\begin{cases} x_c = x_0 - yr \cos \varphi_0 - xr \sin \varphi_0 + r_c \cos(\beta_c + \theta r + \varphi_0) + d \cos(\theta r + \varphi_0 + \beta_c - \frac{\alpha_c}{2} + \alpha) \\ y_c = y_0 - yr \sin \varphi_0 + xr \cos \varphi_0 + r_c \sin(\beta_c + \theta r + \varphi_0) + d \sin(\theta r + \varphi_0 + \beta_c - \frac{\alpha_c}{2} + \alpha) \end{cases} \quad (99)$$

Enfin, pour déterminer la cellule correspondante à la position (x_c, y_c) on associe à chaque position la cellule d'indice (i, j) , avec :

$$i = \text{la partie entière de } \left(\frac{x_c}{X_l} \right).$$

$$j = \text{la partie entière de } \left(\frac{y_c}{Y_l} \right).$$

Où X_l et Y_l sont les dimensions de la cellule selon l'axe X et Y (respectivement), on a choisi une résolution de 100x100mm.

Les cellules de la grille situées dans le cône d'émission d'un capteur à ultrasons avant la distance mesurée ont leur valeur C_L décrétementée de 1 pour représenter l'espace libre. Puis pour représenter l'obstacle, nous recherchons toutes les cellules se trouvant sur l'arc de cercle décrivant la limite du cône et nous incrémentons la valeur C_L de la cellule de 1 (Figure 4.15). Cette opération sera répéter pour chaque capteur.

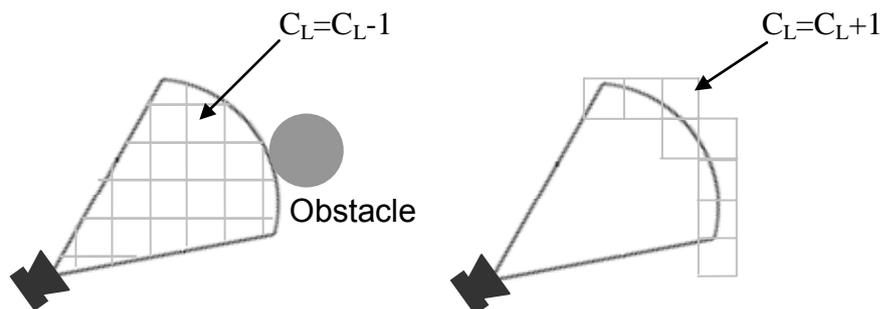


Figure 4.15 Cellules correspondantes à une mesure d'un capteur à ultrasons

Pour nos tests préliminaires, le robot *B21r* va se déplacer dans un couloir (figure 4.16.a), où il utilisera ces deux ceintures d'ultrasons (celles de l'enclosure et la base) pour construire la carte de l'environnement qu'il perçoit (figure 4.16.b).

Les capteurs de ces deux ceintures sont actionnés en même temps. Les problèmes de la diaphonie et les réflexions multiples sont très visibles quand on affiche les mesures ultrasonores, cela engendre des erreurs de perception. Donc on ne pourra pas se fier à la carte construite

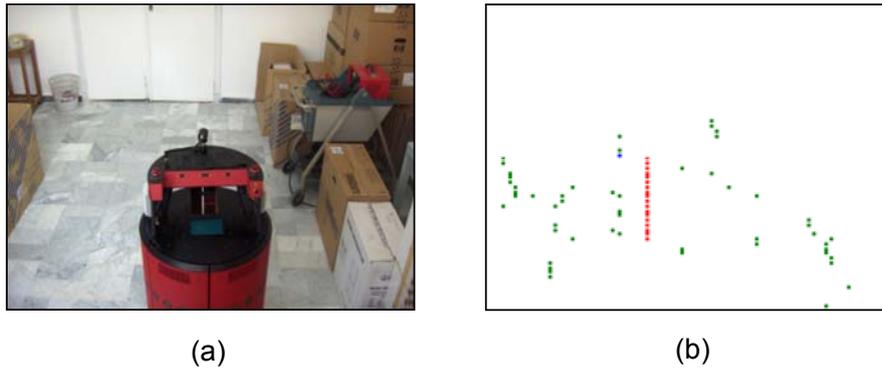


Figure 4.16 Création de la carte de l'environnement utilisant les deux ceintures à ultrasons
 (a) Environnement réel
 (b) Environnement reconstruit.

Une des solutions pour résoudre ce problème, est le déclenchement des capteurs à ultrasons par groupes, tel que chaque groupe comporte les capteurs qui ne sont pas adjacents.

Malheureusement, cette solution n'est pas prise en compte dans notre travail à cause des contraintes imposées par le système software du robot B21r qui impose le déclenchement de tous les capteurs au même moment.

4.3.2 Le capteur infrarouge

Le robot *B21r* dispose d'une ceinture de 24 télémètres infrarouges de la famille Sharp (GP2D02). Ces capteurs ont une portée de quelques dizaines de centimètres (il peut détecter un obstacle entre 10 et 80cm). Ces capteurs ont un cône d'émission très restreint, c'est pourquoi ils peuvent être modélisés comme des capteurs directs. En dépit de la précision des capteurs infrarouges, ils sont limités quant à leurs portées très faible (inconvenient majeur), et leur sensibilité aux perturbations lumineuses. Par conséquent, ils ne sont pas adaptés dans notre cas pour la construction des grilles d'occupations.

Pour construire nos cartes on utilisera le capteur laser en raison de sa précision et sa portée qui atteint les sept mètres. Le champ de vision de notre capteur laser se limite à la partie avant du robot, en plus il se trouve à une certaine hauteur du sol. Donc il pourrait ne pas détecter certaine forme d'objet. Pour remédier à ce problème, on utilisera les deux ceintures d'ultrasons et la ceinture d'infrarouge pour l'arrêt d'urgence du robot, afin d'éviter sa collision avec des obstacles proches du robot.

4.3.3 Modélisation de l'environnement basée sur le capteur laser

4.3.3.1 Le capteur laser

Le capteur laser embarqué sur le robot *B21r* est un Sick LMS-219. Il est placé à trente centimètres du sol sur la face avant du *B21r*, il balaye un demi-plan horizontal en donnant une mesure tous les degrés à savoir la distance mesurée par le capteur. La portée de ce capteur peut atteindre jusqu'à sept mètres (le constructeur prévoit une erreur de mesure inférieure à 70mm).

L'avantage principal de ce type de capteur est la qualité des données recueillies. En effet, les données issues des capteurs infrarouges ou ultrasonores sont généralement entachées d'une grande incertitude (comme cité auparavant). Au contraire, les données issues du télémètre laser ont de nombreux avantages : précision des distances et des angles mesurés, grande portée et projection aisée des données dans l'espace de travail du robot.

4.3.3.2 Modélisation des mesures laser

Etant donné que le laser est un capteur directive dans notre cas réparti en 180 directions, la mesure est interprétée comme étant la distance au plus proche obstacle pour chacune des directions. Donc les mesures seront modélisées comme celles d'un capteur directive et cela pour chaque direction α du capteur laser.

Comme pour les capteurs à ultrasons, chaque cellule se verra attribuer une des trois valeurs :

$C_L = 0$ si aucune information n'est disponible.

$C_L = C_L - 1$ si la cellule est libre.

$C_L = C_L + 1$ si la cellule est occupée.

De manière à ce que les cellules représentant l'obstacle seront incrémentées de 1, tandis que les cellules représentant l'espace libre et qui se trouve entre le capteur et l'obstacle seront décrémentées de 1.

Le modèle du capteur laser ainsi que la grille correspondante sont représentés dans la figure 4.17.

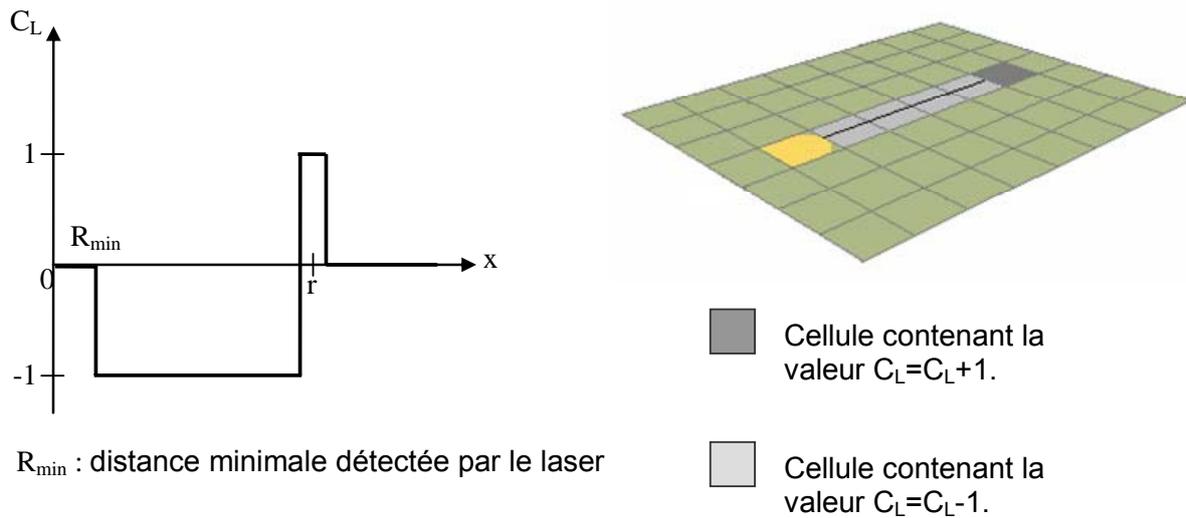
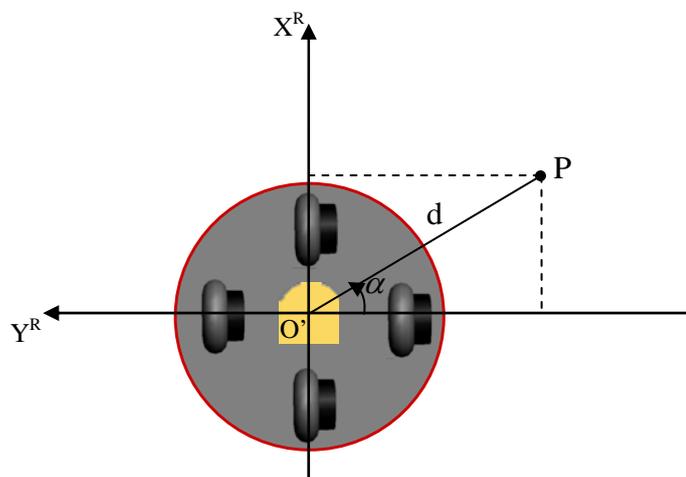


Figure 4. 17 Modélisation du capteur laser

4.3.3.3 Création de la grille

Pour chaque mesure, le capteur laser fournit un vecteur de 180 données. Ces données représentent les distances entre les obstacles et le capteur suivant ses 180 directions. Pour définir les cellules correspondantes aux positions des obstacles dans l'environnement, on suit les mêmes étapes utilisées avec les capteurs à ultrasons.

Figure 4.18 Configuration de la position d'un obstacle P par rapport au repère mobile R^R avec un champ de vision du laser $\alpha \in [0,180^\circ]$

Comme le capteur laser se trouve au centre du robot $B21r$, il est centré sur l'origine du repère mobile R^R (Figure 4.18). Ainsi pour chaque direction du capteur orientée tous les directions (itération sur α) avec $\alpha \in [0^\circ,180^\circ]$, la position de l'obstacle détecté par rapport au repère mobile R^R (O' , Y^R , X^R) est définie par :

$$\begin{cases} y_c^R = -d \cos \alpha \\ x_c^R = d \sin \alpha \end{cases} \quad (100)$$

En utilisant la relation de passage du repère mobile R^R vers le repère univers R^u (équation 98), la position de l'obstacle par rapport au repère univers devient :

$$\begin{cases} x_c = x_0 - yr \cos \varphi_0 - xr \sin \varphi_0 + d \cos(\theta r + \varphi_0 + \alpha) \\ y_c = y_0 - yr \sin \varphi_0 + xr \cos \varphi_0 + d \sin(\theta r + \varphi_0 + \alpha) \end{cases} \quad (101)$$

Notre environnement est divisé en cellules de résolution $X_l \times Y_l = 100 \times 100 \text{mm}$, donc la cellule correspondante à la position (x_c, y_c) est celle portant les indices i =la partie entière de $\left(\frac{x_c}{X_l}\right)$ et j =la partie entière de $\left(\frac{y_c}{Y_l}\right)$.

Dans ce cas on garde le même environnement de la figure 4.16.a. Tout en se déplaçant le robot *B21r* construit la carte de l'environnement en utilisant uniquement les données fournies par le capteur laser. Les résultats obtenus sont représentés dans la figure 4.19.



Figure 4.19 Création de la carte d'environnement utilisant le capteur laser
 (a) Environnement réel
 (b) Environnement reconstruit.

En comparant ces résultats avec ceux de la figure 4.16.b, on voit clairement que la carte d'environnement reconstruite par le télémètre laser est beaucoup plus précise et proche de l'environnement réel du robot que celle reconstruite par le télémètre ultrasonore.

4.4 Méthodes de localisation appliquées au *B21r*

Pour la localisation du robot *B21r*, on ne peut se baser sur les données odométriques fournies par celui-ci à cause de toutes les erreurs citées au paravent dans le chapitre 2, c'est pourquoi on fait une estimation de cette position pour se rapprocher au plus de la position réelle du robot. Dans notre cas, deux méthodes pourraient bien être utilisées pour calculer cette estimation en vu des outils dont on dispose (voir chapitre 3). A savoir la méthode de mise en correspondance des grilles et le filtre de Kalman étendu (EKF).

4.4.1 Localisation par la mise en correspondance des grilles

4.4.1.1 Définition de la grille locale

La taille de la grille locale est ajustée sur la plus grande mesure laser détectée, la portée maximale du LMS embarqué sur le robot mobile *B21r* est de 7m et son champ de vision est de 180° , donc la grille locale aura pour dimension 14mx7m.

Les cellules libres seront décrémentées de la valeur 1, tandis que celles occupées par un obstacle perçu seront incrémentées de la valeur 1. Un exemple de grille locale est donné à la (figure 4.20).



Figure 4.20 Création d'une carte locale basée sur le capteur laser

(a) Situation du robot *B21r*.

(b) Carte locale reconstruite.

4.4.1.2 Définition de la grille globale

La grille globale définit l'environnement dans lequel le robot va se mouvoir. Dans notre cas, l'environnement du robot *B21r* est semi connu. La mise à jour de la

grille globale définissant cet environnement se fera pendant son déplacement. Pour construire sa grille globale, le robot commence par initialiser les cellules de la grille par les valeurs $C_G=0$. Puis il effectue une exploration de l'environnement durant laquelle il décrémente de 1 les valeurs des cellules libres et incrémente de 1 les valeurs des cellules occupées.

La figure 4.21 montre un exemple de reconstruction de la carte globale de l'environnement de notre robot mobile pendant son déplacement rectiligne.

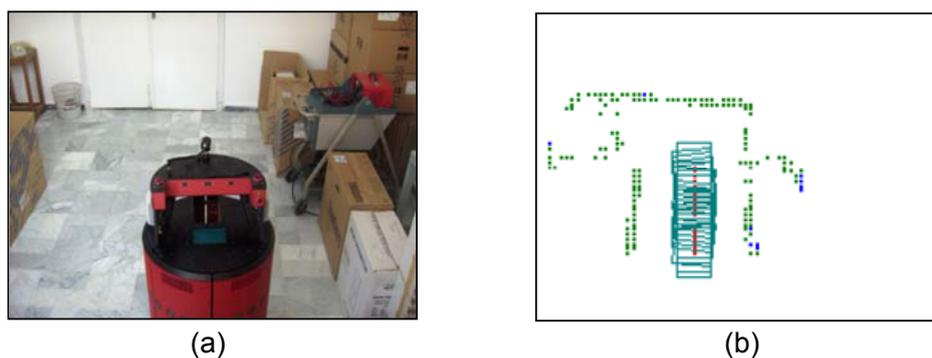


Figure 4.21 Reconstruction de la carte globale en utilisant le capteur laser
 (a) Environnement réel exploré par le robot *B21r*.
 (b) Carte globale reconstruite (les points rouges représentent le centre de gravité du robot).

4.4.1.3 Principe

La mise en correspondance a pour but de déterminer la position et l'orientation du robot dans l'environnement. Le robot mobile se trouvant dans une position quelconque dans la grille globale, il reconstruit sa carte locale en utilisant le capteur laser. Partant d'une position et orientation données, nous recherchons la translation et la rotation qui fera coïncider au mieux ces deux grilles.

La position du robot est estimée en trouvant l'emplacement de la grille locale dans la grille globale ce qui revient à calculer a et b , l'orientation du robot quant à elle est évaluée en mesurant l'angle de rotation θ de la grille locale par rapport à la grille globale (figure 4.22).

Les valeurs a , b et θ sont les valeurs qui maximisent le critère de mise en correspondance faisant apparier toutes les translations et rotations possibles de la grille locale par rapport à la grille globale.

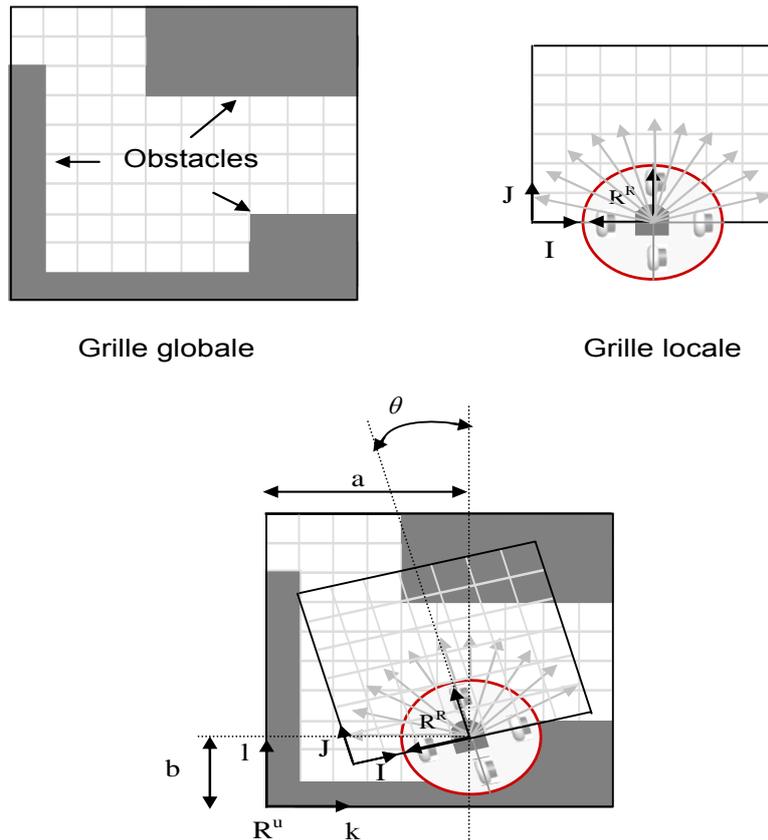


Figure 4.22 Mise en correspondance des grilles

4.4.1.4 Critère de mise en correspondance

Considérons une translation et une rotation données, par cette transformation la cellule $C_L(I,J)$ de la grille locale se superpose à une cellule $C_G(k,l)$ de la grille globale. Soit P le produit des valeurs des deux cellules superposées. Le signe de P indique si les deux cellules ont le même état (occupé ou libre) et donc si il y a bonne correspondance entre les cellules ou non (figure 4.23).

C_G	Occupée	Libre
C_L		
Occupée	>0	<0
Libre	<0	>0

Fig. 4.23 Signe du produit des cellules

Le produit est effectué sur toutes les cellules de la grille locale, la somme S de tous ces produits est calculée et constitue "le critère de mise en correspondance". Il

informe sur une bonne superposition des grilles puisqu'il est maximal quand le plus grand nombre de cellules de même nature sont superposées.

La position du robot et son orientation seront données par a , b et θ pour lesquels la somme $S(a,b,\theta)$ est maximale, avec :

$$S(a,b,\theta) = \sum_{I,J \in \text{grille locale}} C_L(I,J) \cdot C_G(k,l) \quad (102)$$

Les valeurs de a , b et θ pourraient être déterminées après avoir exprimé les coordonnées (k,l) de la grille globale en fonction des coordonnées (I,J) de la grille locale. Tel que :

$$\begin{cases} k = a + (I - i_0) \cos \theta - J \sin \theta \\ l = b + (I - i_0) \sin \theta + J \cos \theta \end{cases} \quad (103)$$

Avec :

- $(i_0,0)$ sont les indices de la cellule de la grille locale (avec i_0 centré) contenant l'origine du repère R^L et sur laquelle est centrée la rotation. (voir Appendice C)
- k et l sont les indices de la cellule de la grille globale couverte par la cellule d'indice I, J de la grille locale.

k et l doivent être entiers, or la relation ci-dessus fournit des valeurs réelles. Il est alors nécessaire de préciser la cellule qui servira à réaliser la mise en correspondance.

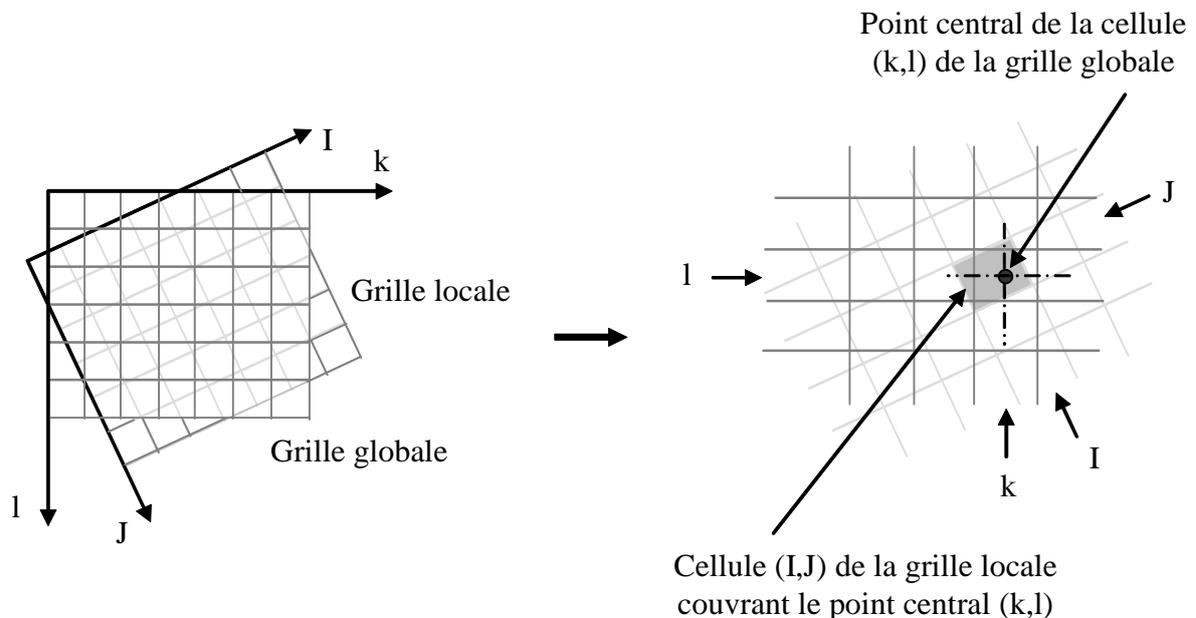


Figure 4.24 Mise en correspondance des grilles

Comme le montre la figure 4.24, une cellule de la grille locale sur laquelle a été appliquée la rotation ne coïncide pas exactement avec une seule cellule de la grille globale mais en chevauche plusieurs. Pour notre application, nous levons l'ambiguïté en considérant qu'une cellule de la grille globale est associée à la cellule de la grille locale qui couvre son point central, ce qui revient à arrondir k et l à l'entier le plus proche.

4.4.1.5 Zone de recherche

Seule information disponible sur la position du robot est celle fournie par le système odométrique du robot *B21r*, c'est une estimation de la position exprimée par le vecteur (x_r, y_r, θ_r) . Nous effectuons la recherche de la mise en correspondance dans une zone de $\pm 5m$ autour de la position estimée et pour des orientations comprises entre ± 45 degrés autour de l'orientation estimée (figure 4.25), ces valeurs ont été choisies d'une façon à ce que la zone de recherche contient forcément la position réelle du robot et fixées expérimentalement.

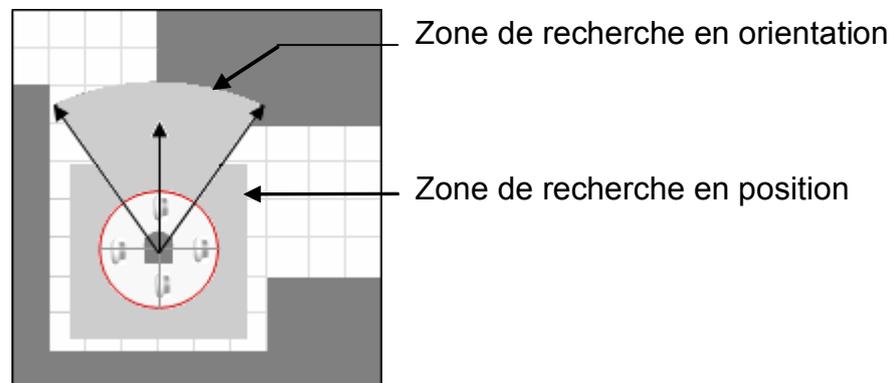


Figure 4.25 Exemple de zone de recherche pour une position donnée

4.4.1.6 L'algorithme

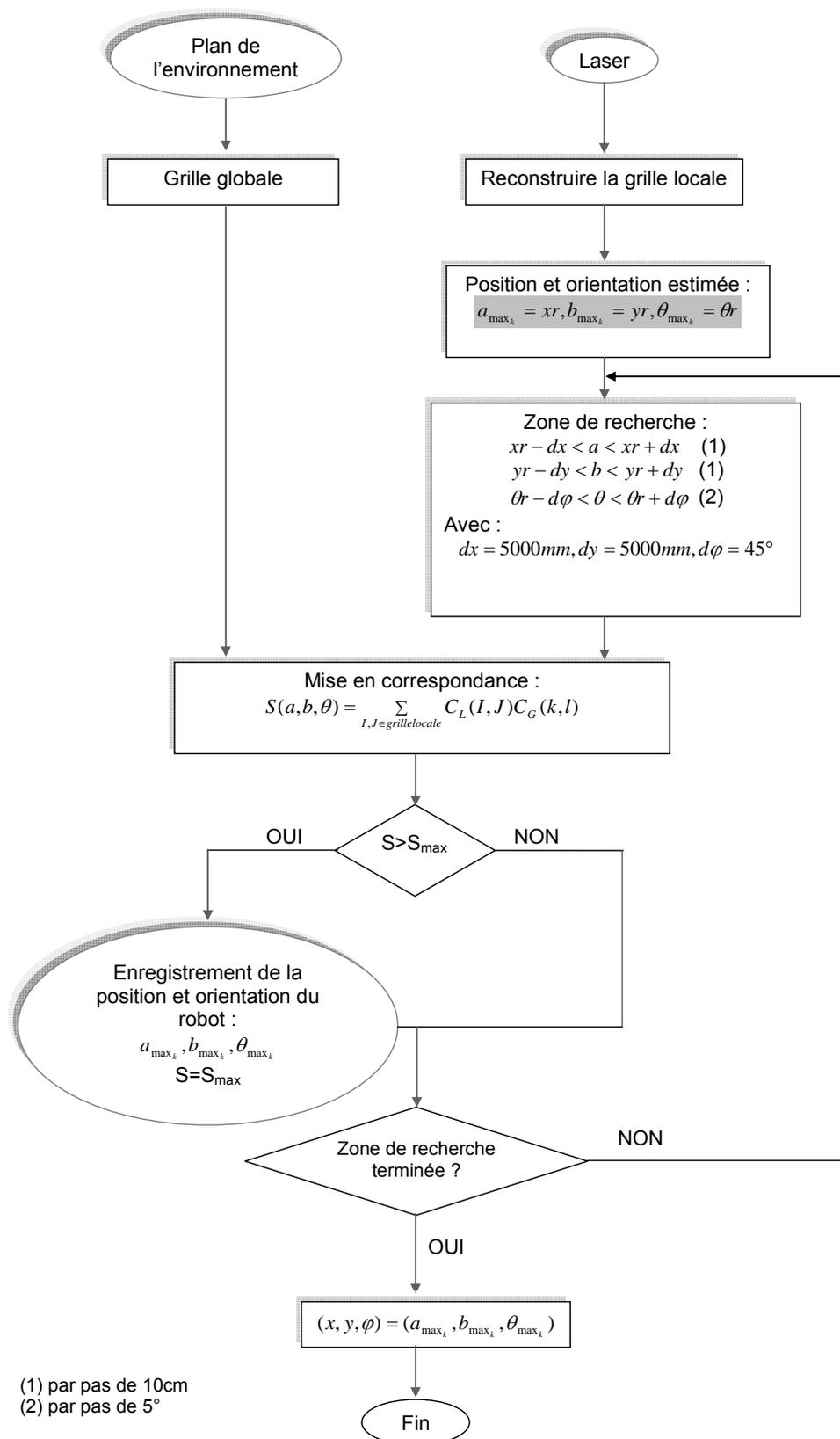


Figure 4.26 Algorithme de mise en correspondance

Après avoir construit la grille locale au moyen du capteur laser, on fait la mise en correspondance entre cette grille et la grille globale pour toutes les translations possibles selon l'axe X et l'axe Y du repère univers R^u , et pour toutes les orientations possibles après avoir initialisé la position (a, b, θ) du robot par la position fournie par le système odométrique. La somme S_{max} est initialisée à 0, puis dès que la somme $S(a, b, \theta)$ est maximale, la position et l'orientation du robot seront enregistrées et la valeur de S constitue la nouvelle valeur de S_{max} . Une fois que la zone de recherche est entièrement balayée, la position (a, b, θ) pour laquelle S est maximale représentera la position du robot $((x, y, \varphi) = (a, b, \theta))$. L'algorithme de localisation est donné par la figure 4.26.

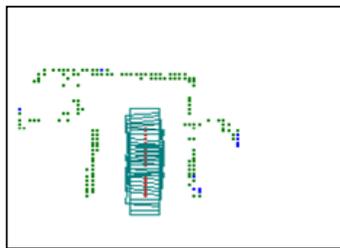
4.4.1.7 Résultat

La carte de l'environnement est reconstruite en faisant explorer le robot *B21r* dans cet environnement. Une fois terminé, elle sera utilisée dans la phase de la mise en correspondance, où le robot fera correspondre la carte locale représentant la partie de l'environnement qu'il perçoit à partir de sa position et la carte globale pour déterminer la position où il se trouve.

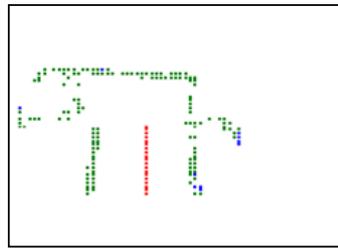
Comme résultats de la localisation par la mise en correspondance des grilles, les deux figures 4.27 et 4.28 sont deux exemples du déplacement du robot dans un milieu d'intérieur. Dans la première figure, le robot *B21r* suit une trajectoire rectiligne et dans la deuxième il suit une trajectoire courbe.



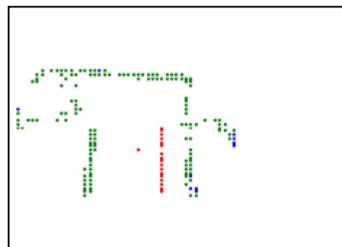
(a)



(b)



(c)



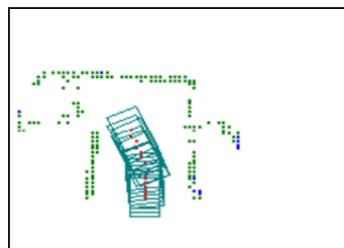
(d)

Figure 4.27 Résultat de la localisation par mise en correspondance des grilles d'occupation pour une trajectoire rectiligne.

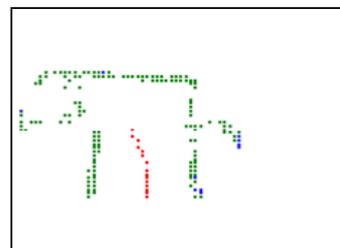
- (a) Déplacement du robot *B21r*.
- (b) Carte globale reconstruite (les points rouges représentent le centre de gravité du robot).
- (c) Positions du *B21r* fournies par les odomètres
- (d) Positions du *B21r* fournies par la mise en correspondance



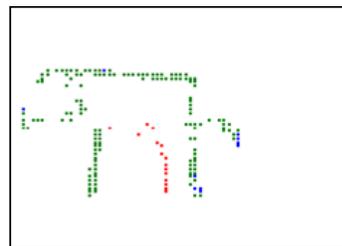
(a)



(b)



(c)



(d)

Figure 4.28 Résultat de la localisation par mise en correspondance des grilles d'occupation pour une trajectoire courbe.

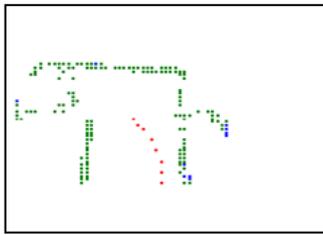
- (a) Déplacement du robot *B21r*.
- (b) Carte globale reconstruite (les points rouges représentent le centre de gravité du robot).
- (c) Positions du *B21r* fournies par les odomètres
- (d) Positions du *B21r* fournies par la mise en correspondance

Les résultats obtenus confirment la précision de la méthode qui corrige les erreurs cumulatives des odomètres. Notons aussi qu'on a admis lors des deux tests une erreur dans la position initiale du robot, suite à cette erreur toutes les positions odométriques ont été calculées par rapport à elle (figure 4.28.C), par conséquent on a l'impression que la position du robot se trouve au milieu, ce qui n'est pas le cas réel (figure 4.28.a). Mais pour la mise en correspondance cette erreur a été corrigée car dans ce cas la position du robot correspond au meilleur appariement

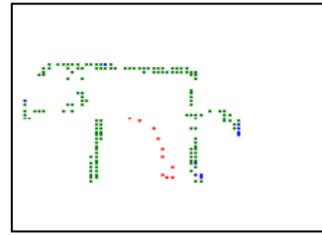
(figure4.28.d) (et la position initiale admise n'est qu'une estimation autour de laquelle se fera la recherche).

On refait les mêmes tests pour des zones de recherche différentes (figure4.29), on essaie de suivre la même trajectoire dans chaque cas (pour avoir une même distance) avec une même vitesse constante sur toute la trajectoire, afin de pouvoir comparer tous les cas.

- La zone de recherche = 1*1mètres

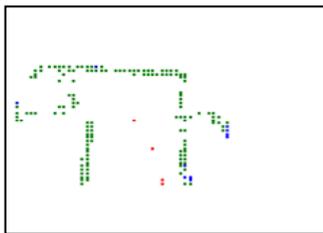


Positions fournies
par les odomètres

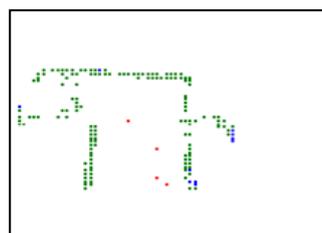


Positions fournies par la
mise en correspondance

- La zone de recherche = 2*2mètres

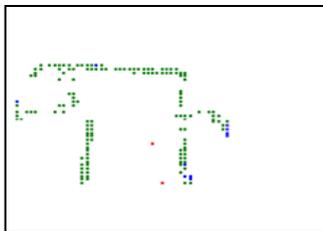


Positions fournies
par les odomètres

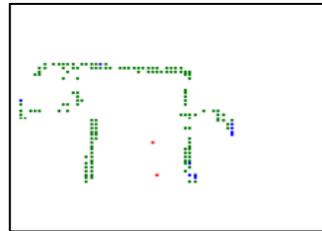


Positions fournies par la
mise en correspondance

- La zone de recherche = 3*3mètres



Positions fournies
par les odomètres



Positions fournies par la
mise en correspondance

Figure 4.29 Résultats de la localisation par la mise en correspondance des grilles pour trois dimensions différentes de la zone de recherche.

On remarque que lorsque la dimension de la zone de recherche augmente, le nombre d'échantillons diminue. Cela est dû au temps de calcul qui augmente.

Pour donner un ordre de grandeur du temps mis par la méthode de la mise en correspondance pour estimer la trajectoire du robot, On a fait déplacer le robot en ligne droite avec une même vitesse (0,2m/s) et sur une même distance (2,5mètres) pour chaque cas correspondant à une dimension différente de la zone de recherche. Mais on a fait en sorte d'arrêter le robot lors du calcul de la position pour ne pas perdre d'information, on a trouvé :

Pour une zone de recherche de 3x3mètres, le temps mis est de 7minutes.

Pour une zone de recherche de 2x2mètres, le temps mis est de 3minutes.

Pour une zone de recherche de 1x1mètres, le temps mis est de 40secondes.

Les résultats obtenus lors de la phase expérimentale montrent que bien que cette méthode donne de bon résultats côté précision, elle dépend fortement du temps de traitement du robot et du temps d'exécution du programme. Car si la vitesse du robot est grande, on pourra pas faire la mise en correspondance et donc on pourra pas estimer la position. Cette méthode dépend aussi de la fenêtre de recherche qui doit être précise, Car tout comme la position réelle du robot pourrait se trouver proche de la position estimée avec une grande zone de recherche donc une perte de temps de calcul, elle pourrait se trouver loin de la position estimée avec une petite zone de recherche et dans ce cas la mesure sera ratée. Donc on ne peut prévoir la dimension optimale de la zone de recherche.

Vu toutes ces contraintes, l'utilisation du filtre de Kalman étendu est la solution la plus appropriée dans notre cas pour localiser le robot mobile. Cet algorithme fusionnera les données odométriques et les données laser pour déterminer la position la plus proche à la position réelle. En plus il optimise l'algorithme de la mise en correspondance qui servirait à déterminer la mesure (position du robot fournie par le télémètre laser).

4.4.2 La localisation par le filtre de Kalman étendu (localisation dynamique avec un odomètre et un capteur laser)

4.4.2.1 Formulation mathématique

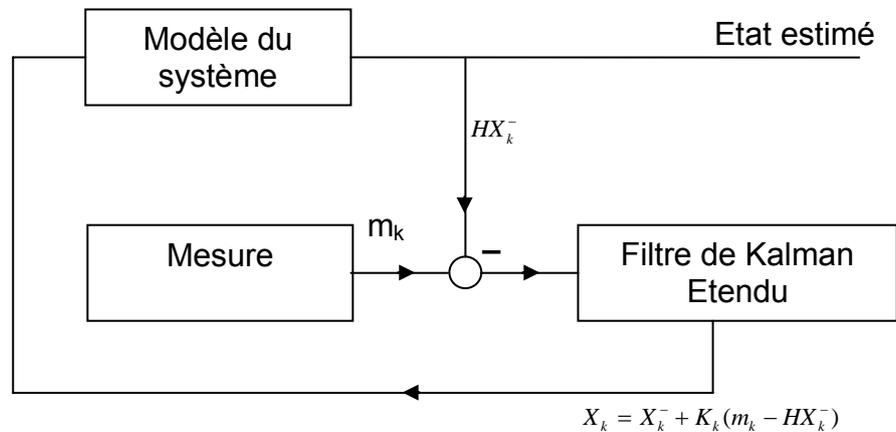


Figure 4.30 Le filtre de Kalman étendu représenté comme une configuration de rétroaction

L'algorithme de localisation sera implémenté sur notre plateforme de recherche à savoir le robot mobile B21r. Celui-ci est non holonome (comme illustré lors de l'étude cinématique dans le chapitre 1) et par conséquent le modèle du système est non linéaire, de ce fait un filtre de Kalman étendu [2] sera utilisé.

Dans notre cas le modèle de mesure est linéaire, donc la linéarisation ne portera que sur l'équation du modèle d'évolution (modèle du système). Dans ce qui suit on présentera la formulation des différents modèles nécessaires pour le calcul du gain de Kalman, ainsi que les résultats de la prédiction [3-5].

4.4.2.2 Le modèle d'évolution (modèle du système)

L'odométrie est une technique de localisation qui permet de déterminer la position (x, y, φ) du robot par rapport au repère univers en intégrant des translations et rotations élémentaires. On appelle "odomètre" un dispositif qui permet de mesurer grâce à des rotation de roues, ses translations et rotations élémentaires. Il existe différentes techniques pour les mesurer et différents algorithmes pour les intégrer.

Dans notre cas, le robot mobile B21r fournit directement une position (x_r, y_r, θ_r) pour chaque échantillon, cette donnée sera utilisée par la suite pour

exprimer la rotation élémentaire $\delta\varphi$ et la translation élémentaire δd du robot (Figure 4.31).

Pour un déplacement entre deux instants d'échantillonnages $k-1$ et k , qui leurs correspond respectivement les mesures odométriques $(x_{r_{k-1}}, y_{r_{k-1}}, \theta_{r_{k-1}})$ et $(x_{r_k}, y_{r_k}, \theta_{r_k})$ on a :

La translation élémentaire du robot B21r est :

$$\delta d = \sqrt{(x_{r_k} - x_{r_{k-1}})^2 + (y_{r_k} - y_{r_{k-1}})^2} \quad (104)$$

Le déplacement angulaire du robot est :

$$\delta\varphi = \theta_{r_k} - \theta_{r_{k-1}} \quad (105)$$

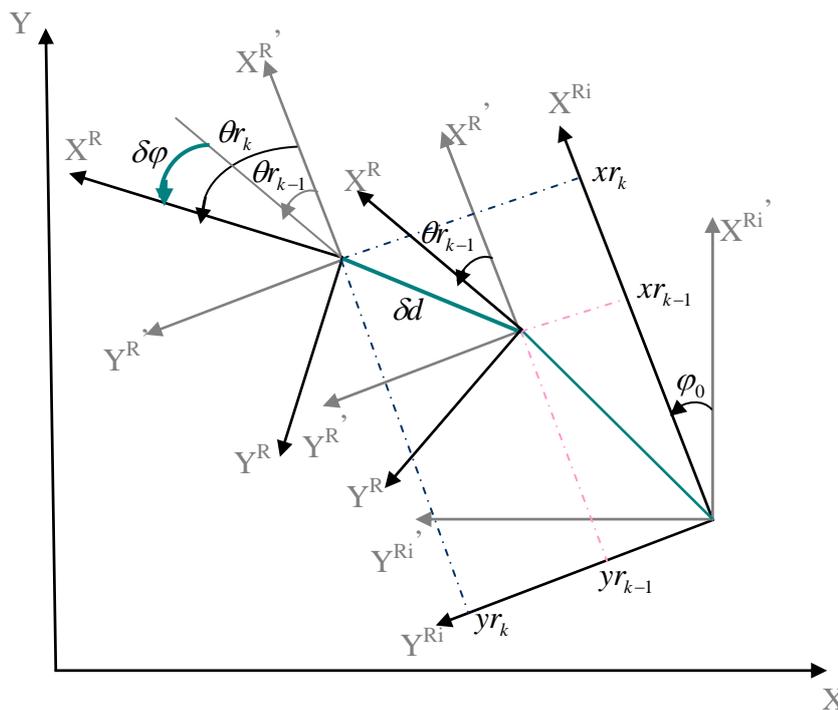


Figure 4.31 Position du robot dans le repère universel R^u en fonction des données odométriques

On cherche à exprimer de manière récurrente la position du robot (x_k, y_k, φ_k) à l'instant " k " en fonction de la position $(x_{k-1}, y_{k-1}, \varphi_{k-1})$ à l'instant " $k-1$ " et des déplacements élémentaires mesurés.

Une première approche consiste à discrétiser le modèle cinématique qui était objet d'étude dans le chapitre 1 (où v et Ω sont les vitesses linéaire et angulaire). Sauf que l'angle φ sera représenté par rapport à l'axe vertical Y (au lieu de l'axe horizontal X) se qui nous a été imposé par système software (*Mobility*) du robot mobile *B21r*, le modèle cinématique devient alors:

$$\begin{aligned}\dot{x} &= -v \sin \varphi \\ \dot{y} &= v \cos \varphi \\ \dot{\varphi} &= \Omega\end{aligned}\quad (106)$$

Il s'agit d'un modèle non linéaire, pour lequel il n'existe pas de méthode générale de discrétisation. Dans le cas particulier qui nous intéresse, on peut approcher \dot{x} par une différence finie, si T_e est petite (même chose pour \dot{y}) :

$$\dot{x}_{(t=k)} = \frac{x_k - x_{k-1}}{T_e} \quad (107)$$

La discrétisation du modèle (106) est alors :

$$\begin{cases} x_k = x_{k-1} - v_{k-1} T_e \sin \varphi_{k-1} \\ y_k = y_{k-1} + v_{k-1} T_e \cos \varphi_{k-1} \\ \varphi_k = \varphi_{k-1} + \Omega_{k-1} T_e \end{cases} \quad (108)$$

On obtient donc en faisant apparaître les mesures odométriques δd et $\delta \varphi$, le modèle:

$$\begin{cases} x_k = x_{k-1} - \delta d \sin \varphi_{k-1} \\ y_k = y_{k-1} + \delta d \cos \varphi_{k-1} \\ \varphi_k = \varphi_{k-1} + \delta \varphi \end{cases} \quad (109)$$

Une autre façon d'aborder le problème revient à émettre des hypothèses sur le déplacement. Ce faisant des relations géométriques permettant de déterminer le déplacement élémentaire du robot correspondant aux distances parcourues par chacune des roues.

Les formules les plus simples sont obtenues en supposant que le robot se déplace en ligne droite suivant la direction donnée par " φ_{k-1} " sur une longueur " δd ", puis effectue une rotation sur place de " $\delta\varphi$ " [5]. Le modèle obtenu est alors identique à la discrétisation proposée ci-dessus du modèle cinématique (équation 109).

Une autre méthode, plus précise lorsque la trajectoire est une courbe, consiste à supposer que le mouvement se fait localement suivant un arc de cercle de longueur δd . Il existe plusieurs façons d'aboutir aux équations (106, 107). Même s'il s'agit d'un résultat classique, nous proposons une démonstration sensiblement différente qui présente l'avantage d'être extrêmement simple [5].

Sous l'hypothèse d'un mouvement circulaire, on a :

$$\delta d = \rho \delta\varphi \quad (110)$$

Où ρ représente le rayon de courbure (figure 4.32).

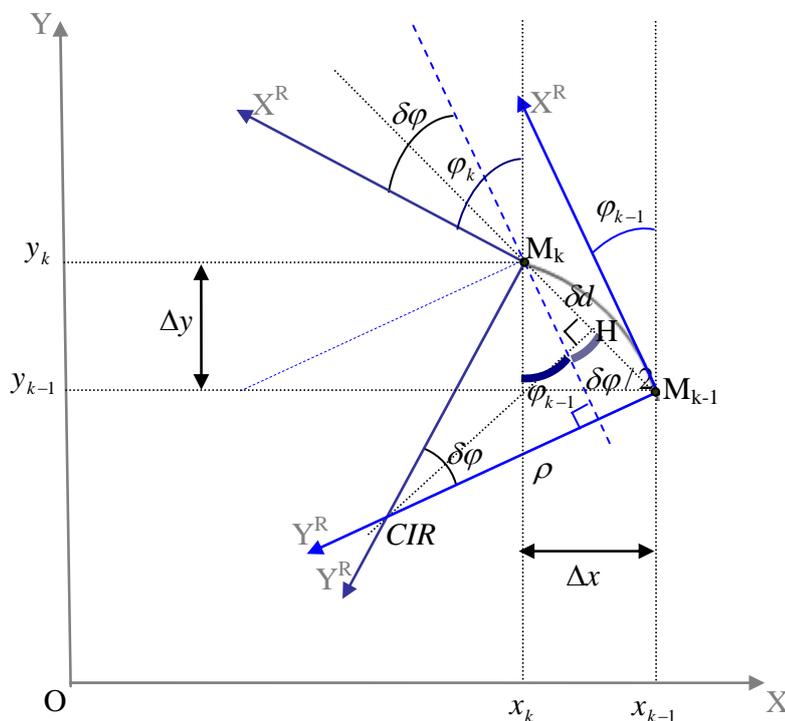


Figure 4.32 Déplacement circulaire du robot entre deux instants d'échantillonnage

$$\|M_{k-1}H\| = \|HM_k\| \approx \rho \sin(\delta\varphi/2)$$

$$\|M_{k-1}M_k\| \approx 2\rho \sin(\delta\varphi/2) = 2 \frac{\delta d}{\delta\varphi} \sin(\delta\varphi/2) = \delta d \frac{\sin(\delta\varphi/2)}{\delta\varphi/2}$$

Les variations Δx et Δy sont définies par le vecteur $M_{k-1}M_k$ dont l'angle avec l'axe vertical Y est donné par :

$$\widehat{(M_{k-1}M_k, oY)} = \varphi_{k-1} + \delta\varphi/2$$

On obtient alors :

$$\begin{cases} x_k = x_{k-1} - \frac{\sin(\delta\varphi/2)}{\delta\varphi/2} \delta d \sin(\varphi_{k-1} + \delta\varphi/2) \\ y_k = y_{k-1} + \frac{\sin(\delta\varphi/2)}{\delta\varphi/2} \delta d \cos(\varphi_{k-1} + \delta\varphi/2) \\ \varphi_k = \varphi_{k-1} + \delta\varphi \end{cases} \quad (111)$$

Si de plus $\delta\varphi$ est petit, ce qui suppose que la période d'échantillonnage soit suffisamment petite par rapport aux dynamiques de l'engin, alors :

$$\begin{cases} x_k = x_{k-1} - \delta d \sin(\varphi_{k-1} + \delta\varphi/2) \\ y_k = y_{k-1} + \delta d \cos(\varphi_{k-1} + \delta\varphi/2) \\ \varphi_k = \varphi_{k-1} + \delta\varphi \end{cases} \quad (112)$$

Ce dernier modèle odométrique est l'un des plus utilisés. Par rapport au modèle (109), il apparaît une correction " $\delta\varphi/2$ " dans les termes trigonométriques. C'est donc sur des trajectoires circulaires, que ce modèle est plus précis.

En tenant compte principalement des contraintes odométriques, on introduit dans nos équations le bruit du modèle d'évolution (normally distributed noise) $W=(0;Q)$ c.a.d sa moyenne est nulle et sa variance est Q .

Donc le modèle que nous utiliserons est le suivant :

$$\begin{aligned} x_k &= x_{k-1} - \delta d \sin(\varphi_{k-1} + \delta\varphi/2) + W_x \\ y_k &= y_{k-1} + \delta d \cos(\varphi_{k-1} + \delta\varphi/2) + W_y \\ \varphi_k &= \varphi_{k-1} + \delta\varphi + W_\varphi \end{aligned} \quad (113)$$

Enfin pour faire apparaître les mesures odométriques, on remplace δd et $\delta\varphi$ par leurs expressions, le modèle devient alors :

$$\begin{cases} x_k = x_{k-1} - \sqrt{(xr_k - xr_{k-1})^2 + (yr_k - yr_{k-1})^2} \sin(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) + Wx \\ y_k = y_{k-1} + \sqrt{(xr_k - xr_{k-1})^2 + (yr_k - yr_{k-1})^2} \cos(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) + Wy \\ \varphi_k = \varphi_{k-1} + (\theta r_k - \theta r_{k-1}) + W\varphi \end{cases} \quad (114)$$

Par analogie ce système peut avoir une forme matricielle qui est la suivante :

$$\begin{bmatrix} x_k \\ y_k \\ \varphi_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \varphi_{k-1} \end{bmatrix} + \begin{bmatrix} -\sqrt{(xr_k - xr_{k-1})^2 + (yr_k - yr_{k-1})^2} \sin(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) \\ \sqrt{(xr_k - xr_{k-1})^2 + (yr_k - yr_{k-1})^2} \cos(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) \\ (\theta r_k - \theta r_{k-1}) \end{bmatrix} + \begin{bmatrix} Wx \\ Wy \\ W\varphi \end{bmatrix} \quad (115)$$

On note $X_k = [x_k, y_k, \varphi_k]^t$ la posture du robot (vecteur d'état), et $U = (xr, yr, \theta r)$ le vecteur obtenu à partir des encodeurs pour une période d'échantillonnage entre l'état k-1 et k. Finalement le système précédent peut se mettre sous la forme vectorielle :

$$X_k = f(X_{k-1}, U, W) \quad (116)$$

Où la fonction vectorielle f est définie de $R^3 \times R^2 \rightarrow R^3$. Elle est continue et dérivable sur $R^3 \times R^2$.

En se référant au fait que le robot mobile B21r est non holonome, le modèle du système est dans ce cas non linéaire c'est-à-dire que la fonction f est non linéaire, ainsi pour linéariser le système on calcule le jacobien, tel que :

$$A = \nabla f_{X_k} = \frac{\delta X_k}{\delta X_{k-1}} \quad (117)$$

$$A = \begin{bmatrix} \nabla f_{11} & \nabla f_{12} & \nabla f_{13} \\ \nabla f_{21} & \nabla f_{22} & \nabla f_{23} \\ \nabla f_{31} & \nabla f_{32} & \nabla f_{33} \end{bmatrix} = \begin{bmatrix} \frac{\delta x_k}{\delta x_{k-1}} & \frac{\delta x_k}{\delta y_{k-1}} & \frac{\delta x_k}{\delta \varphi_{k-1}} \\ \frac{\delta y_k}{\delta x_{k-1}} & \frac{\delta y_k}{\delta y_{k-1}} & \frac{\delta y_k}{\delta \varphi_{k-1}} \\ \frac{\delta \varphi_k}{\delta x_{k-1}} & \frac{\delta \varphi_k}{\delta y_{k-1}} & \frac{\delta \varphi_k}{\delta \varphi_{k-1}} \end{bmatrix} \quad (118)$$

Donc :

$$A = \begin{bmatrix} 1 & 0 & -\delta d \cos(\varphi_{k-1} + \delta\varphi/2) \\ 0 & 1 & -\delta d \sin(\varphi_{k-1} + \delta\varphi/2) \\ 0 & 0 & 1 \end{bmatrix} \quad (119)$$

En intégrant les mesures odométriques :

$$A = \begin{bmatrix} 1 & 0 & -\sqrt{(xr_k - xr_{k-1})^2 + (yr_k - yr_{k-1})^2} \cos(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) \\ 0 & 1 & -\sqrt{(xr_k - xr_{k-1})^2 + (yr_k - yr_{k-1})^2} \sin(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) \\ 0 & 0 & 1 \end{bmatrix} \quad (120)$$

4. 4.2.3 Le modèle de mesure

La mesure est une position (x, y, φ) du robot dans l'environnement, elle est exprimée en fonction du vecteur d'état X_k . Ce dernier est une position (x_k, y_k, φ_k) avec un bruit V additif. Ce lien se traduit par l'équation suivante :

$$m_k = h(X_k, V) \quad (121)$$

Selon le fait que le bruit a une moyenne nulle, l'équation (121) devient :

$$m_k = h(X_k, 0) \quad (122)$$

Cette même relation pourra prendre la forme matricielle :

$$\begin{bmatrix} x_k \\ y_k \\ \varphi_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \varphi_k \end{bmatrix} \quad (123)$$

Donc la matrice H a pour valeur :

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (124)$$

On distingue clairement des équations de mesure (122 et 123), que la fonction de mesure h est linéaire, c'est pourquoi on n'aura pas à calculer le jacobien de la matrice H , et dans ce cas la mesure prend la forme :

$$m_k = HX_k \quad (125)$$

Cette mesure m_k représente la position du robot calculée au moyen du télémètre laser embarqué sur le robot *B21r*, par conséquent elle sera fournie par l'algorithme de mise en correspondances des grilles.

4. 4.2.4 Le modèle d'incertitude

Le modèle d'incertitude représente les bruits qui excitent le système. En général, on a pas de connaissance à priori des corrélations des différentes sources d'erreurs, et puis chacune de la matrice de covariance du bruit du processus Q et la matrice de covariance de l'erreur de mesure R sont supposées diagonales. Q et R sont exprimées comme tel :

$$Q = \text{diag}[\sigma_{Qx}^2 \quad \sigma_{Qy}^2 \quad \sigma_{Q\phi}^2] \quad (126)$$

$$R = \text{diag}[\sigma_{Rx}^2 \quad \sigma_{Ry}^2 \quad \sigma_{R\phi}^2] \quad (127)$$

Les covariances sont déterminées de façon expérimentale. Pour la matrice de covariance Q, on fait une comparaison entre les résultats lus sur les odomètres et les positions réelles du robot, on prend la moyenne d'erreur comme covariance sur la position, donc on pose $\sigma_{Qx} = \sigma_{Qy} = 200\text{mm}$ ce qui équivaut à deux cellules de la grille globale et $\sigma_{Q\phi} = 10^\circ$.

Tandis que pour la matrice de covariance R, on compare les positions fournies par la mesure (laser) et les positions réelles. On prend la moyenne d'erreur et on pose $\sigma_{Rx} = \sigma_{Ry} = 100\text{mm}$ c'est-à-dire une cellule de la grille globale et $\sigma_{R\phi} = 5^\circ$. Ces valeurs correspondent aux pas avec lesquels on fait translater la grille locale par rapport à la grille globale, et qui sera détaillé par la suite.

4. 4.2.5 L'algorithme de localisation basé sur le filtre de Kalman étendu

Connaissant les modèles requis pour le calcul du gain de Kalman, l'explication de la procédure de localisation devient plus simple, où le cycle de localisation est résumé dans l'organigramme de la figure 4.33, dont les différentes étapes seront détaillées dans ce qui suit.

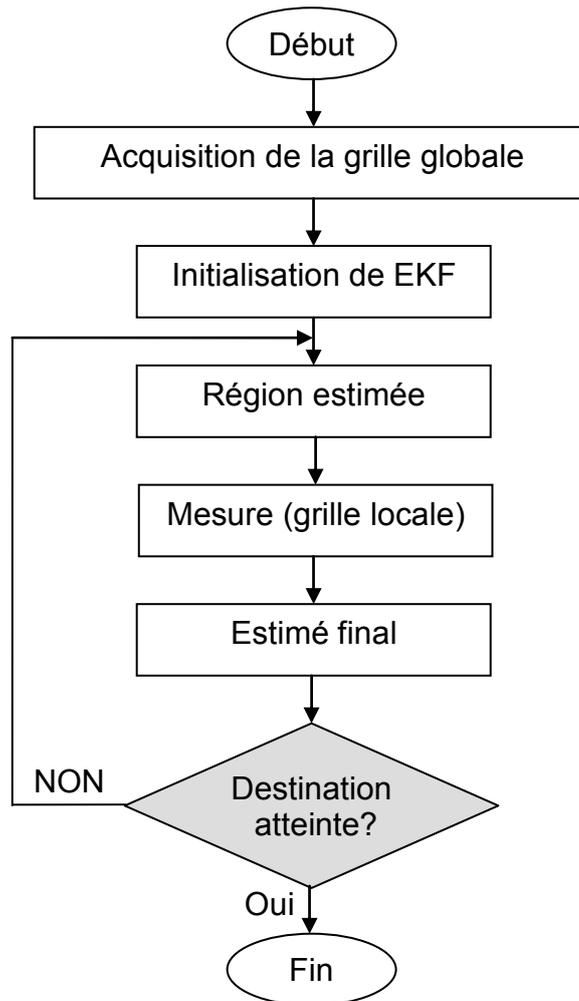


Figure 4.33 Algorithme de localisation

4. 4.2.5.1 Prédiction de la position du robot

Elle consiste à calculer la position et l'orientation estimée du robot B21r à l'état k , connaissant sa position et orientation à l'état $k-1$, ainsi que les données lues à partir des encodeurs du B21r. la position estimée à l'état k peut être écrite tel que:

$$X_k^- = f(X_{k-1}, U, 0) \quad (128)$$

Sachant que selon nos suppositions la moyenne du bruit blanc gaussien est nulle.

Donc

$$\begin{bmatrix} x_k^- \\ y_k^- \\ \varphi_k^- \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \varphi_{k-1} \end{bmatrix} + \begin{bmatrix} -\sqrt{(x r_k - x r_{k-1})^2 + (y r_k - y r_{k-1})^2} \sin(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) \\ \sqrt{(x r_k - x r_{k-1})^2 + (y r_k - y r_{k-1})^2} \cos(\varphi_{k-1} + (\theta r_k - \theta r_{k-1})/2) \\ (\theta r_k - \theta r_{k-1}) \end{bmatrix} \quad (129)$$

Sa covariance sera aussi utile, et sa valeur est définie par l'équation :

$$P_k^- = AP_{k-1}A^T + Q. \quad (130)$$

Durant l'état initial, la position initiale X_0 est connue, tandis que sa covariance P_0 est estimée de quelques centimètres pour la position et quelques degrés pour l'orientation. On pose $\sigma_x = \sigma_y = 200mm$, ce qui équivaut à deux cellules de la grille globale et $\sigma_\varphi = 10^\circ$. En pratique, la valeur de P_0 est choisie assez grande pour englober toutes les erreurs possibles, du moment que la valeur de la covariance P diminue avec la convergence du filtre.

4. 4.2.5.2 La mesure

La mesure est déterminée à partir des données recueillies par le capteur laser et modélisée dans une grille d'occupation (carte locale) pour être mise en correspondance avec la grille globale représentant l'environnement afin de déterminer la position du robot. Donc on va reprendre en partie l'étude faite dans la section 4.4.1.

- *Zone de recherche*

Au départ, la position du robot est donnée par le vecteur d'état estimé X_k^- fourni par le filtre de kalman étendu, et qui représente la position et l'orientation estimées du robot à chaque moment d'échantillonnage k , en plus de cette donnée on a la matrice de covariance de l'état estimé P_k^- , où la covariance σ^2 en x , y et φ représente l'intervalle d'erreur dans lequel se trouve cette position estimée, donc on est sûr que la position réelle du robot se trouve dans un disque de rayon 2σ autour de la position estimée, c'est pourquoi on fera une recherche de mise en correspondance dans une zone de $\pm 2\sigma_x$ et $\pm 2\sigma_y$ autour de la position estimée respectivement selon l'axe des abscisses, et l'axe des ordonnées (du repère locale R^L), et dans une zone de recherche de $\pm 2\sigma_\varphi$ autour de l'orientation estimée.

- *L'algorithme*

Comme dans la section 4.4.1, on fait la mise en correspondance entre la grille locale et la grille globale pour toutes les translations possibles selon l'axe X et l'axe Y du repère univers R^u , et pour toutes les orientations possibles. Dans ce cas la position (a, b, θ) du robot est initialisée par la position estimée par le filtre de kalman étendu (X_k^-). La somme S_{\max} est initialisée à 0, et dès que la somme $S(a, b, \theta)$ est maximale, la position et l'orientation du robot sont enregistrées et la valeur de S constitue la nouvelle valeur de S_{\max} . Une fois que la zone de recherche est entièrement balayée, la position (a, b, θ) pour laquelle S est maximale représentera la mesure m_k qui servira par la suite pour le calcul de l'estimé final dans l'algorithme du filtre de Kalman étendu. Et le nouveau algorithme de mise en correspondance des grilles est représenté dans la figure 4.34.

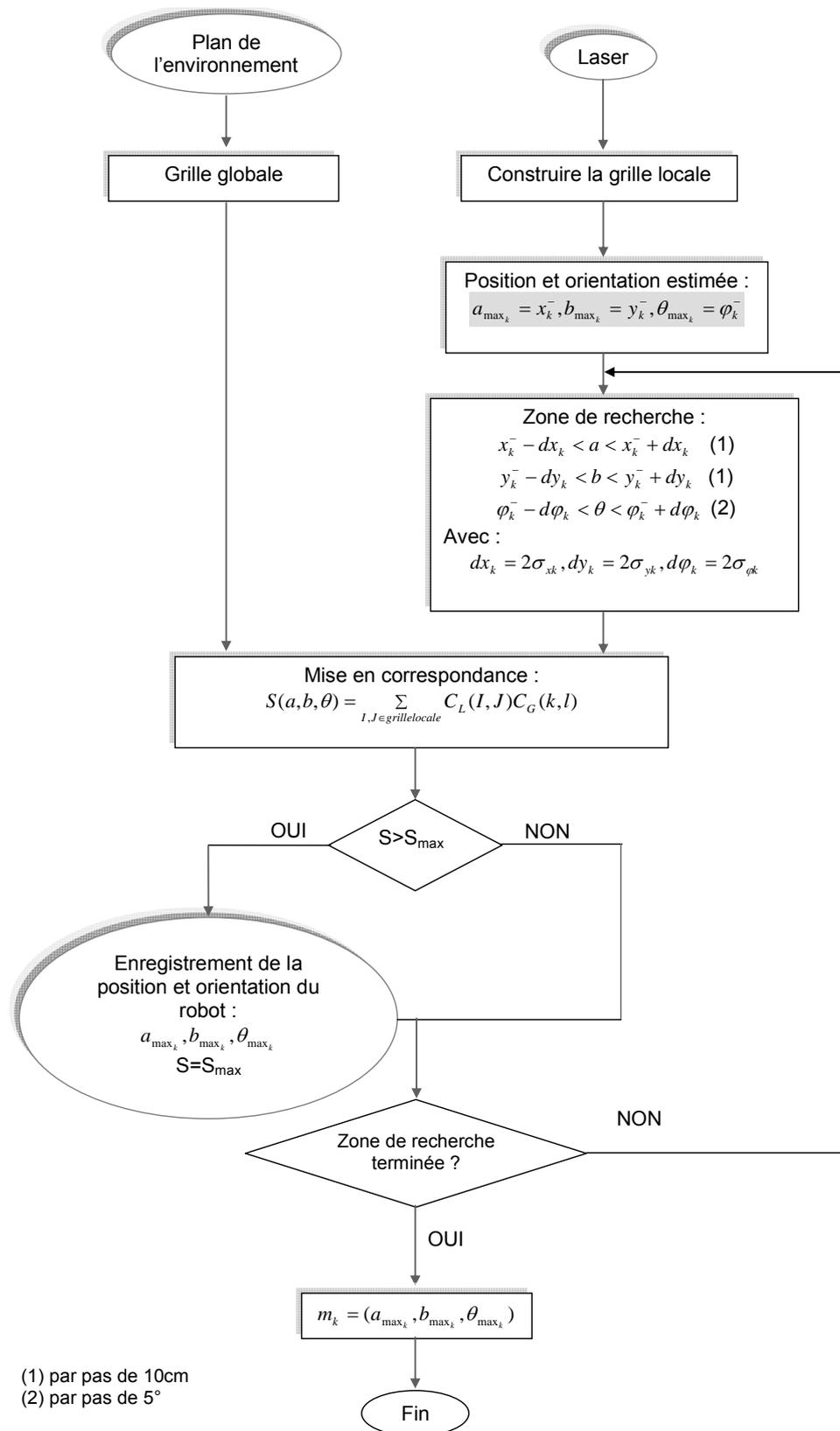


Figure 4.34 Algorithme de mise en correspondance des grilles pour déterminer la mesure

4. 4.2.5.3 Estimation finale de la position

Finalement, pour calculer l'estimé final de la position du robot X_k on utilise le gain de Kalman,

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (128)$$

La position estimée finale devient :

$$X_k = X_k^- + K_k (m_k - HX_k^-) \quad (129)$$

On distingue clairement que cette expression combine entre les résultats incertains de la mesure m_k et les résultats incertains aussi de la prédiction X_k^- , où $(m_k - HX_k^-)$ est l'innovation (résiduel).

Il est aussi important de calculer la covariance de l'état estimé final, qui va être utilisé dans le prochain cycle de localisation.

$$P_k = P_k^- - K_k HP_k^- \quad (130)$$

L'état estimé X_k et sa covariance P_k doivent converger vers un résultat meilleur en évoluant d'un cycle à l'autre.

Ces étapes sont répétées de façon périodique. On suppose que sa période d'échantillonnage T_e est choisie suffisamment petite pour que, d'une part, les approximations effectuées pour établir les équations odométriques soient correctes, et que d'autre part, les conditions suivantes nécessaires pour un bon fonctionnement de l'algorithme soient remplies :

- La période d'échantillonnage doit être petite en comparaison des dynamiques du système.
- l'acquisition et le traitement des mesures doivent être réalisés en une période d'échantillonnage.

Grâce à la première condition, on obtient une bonne intégration des données de la mesure, donc une bonne navigation à l'estime. Grâce aux deux conditions, on admet que le système a suffisamment peu évolué pour que la mesure soit considérée comme "valable".

On peut résumer ces deux conditions de fonctionnement en une seule : entre l'instant où les données du capteur laser sont relevées et l'instant où la mesure est

utilisée, il faut que le système ait suffisamment peu évolué pour que la mesure puisse être considérée comme valable.

On tient à faire remarquer qu'une fois l'état prédit calculé, il est possible que l'on ne peut calculer le gain de Kalman si la matrice inverse $(HP_k^-H^T + R)^{-1}$ dans l'équation du gain de Kalman n'existe pas (son déterminant est nul), dans ce cas on utilise alors la dernière prédiction X_k^- c'est-à-dire que l'estimé final $X_k = X_k^-$.

On pourra aussi être confronté à un cas où δd ou $\delta\varphi$ sont nulles, alors on pourra pas calculer l'état estimé X_k^- , l'estimé final sera égal à la position lue par les odomètres.

4. 4.2.6 Résultats obtenus

Dans une première phase, le robot *B21r* doit disposer de sa carte d'environnement. Celle-ci pourrait soit être lue à partir d'un fichier, sinon elle est reconstruite après exploration de l'environnement, et c'est bien notre cas où le robot utilise le capteur laser pour reconstruire sa grille globale. Cette carte servira lors de la mise en correspondance des cartes locales construites pendant le déplacement du robot et cette carte d'environnement. Après la mise en correspondance la position du robot à partir des données laser est déterminée, donc cette position représente la mesure. L'EKF mettra en œuvre cette mesure en plus de la position fournie par le système odométrique pour estimer la position du robot.

On a fait déplacer le robot suivant une trajectoire droite dans un premier temps en estimant sa position par l'EKF, puis on a refait la même expérience mais en modifiant sa trajectoire. Dans ces deux cas on a introduit une erreur dans la position initiale. Les deux tests sont présentés dans la figure 4.35 et la figure 4.36.

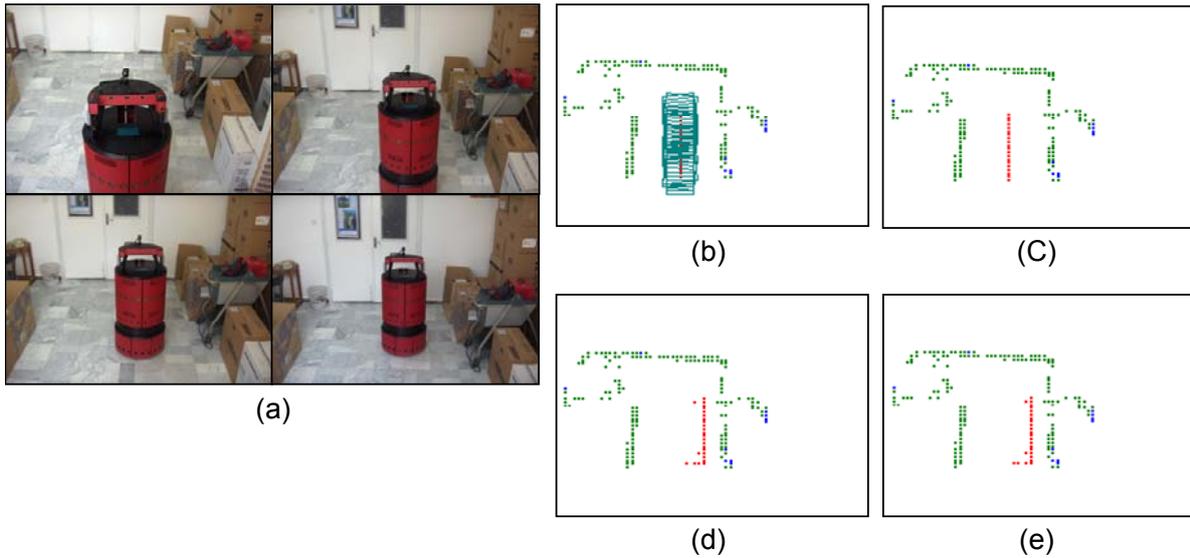


Figure 4.35 Résultat de la localisation par le filtre de Kalman étendu (EKF) pour une trajectoire rectiligne.

- (a) Déplacement du robot *B21r*.
- (b) Carte globale reconstruite (les points rouges représentent le centre de gravité du robot).
- (c) Positions du *B21r* fournies par les odomètres
- (d) La mesure (positions du *B21r* fournies par la mise en correspondance)
- (e) Positions du *B21r* fournies par EKF.

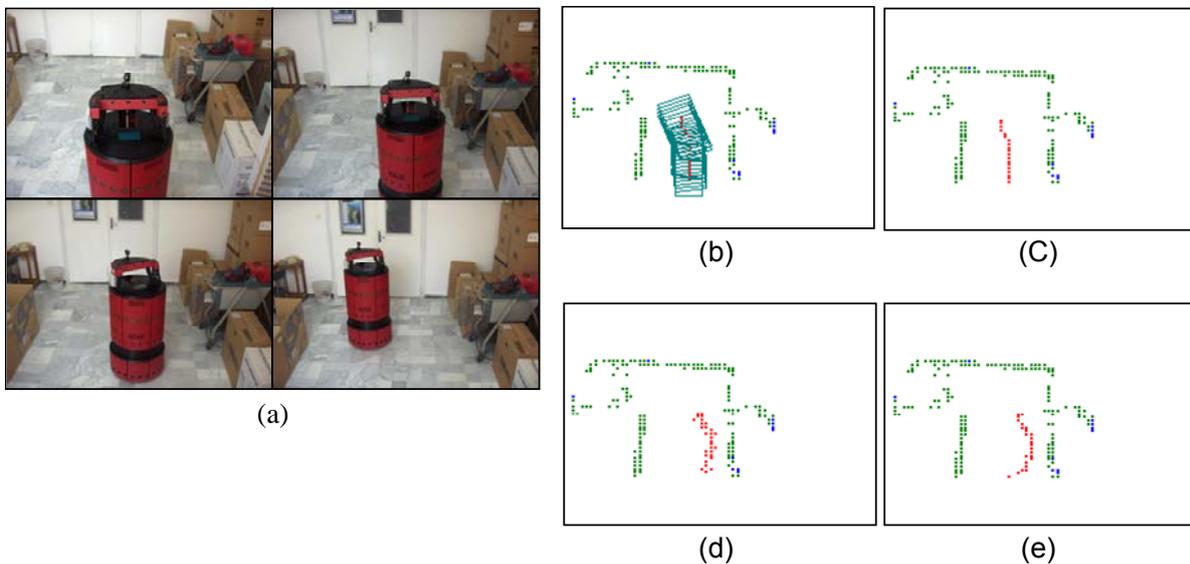


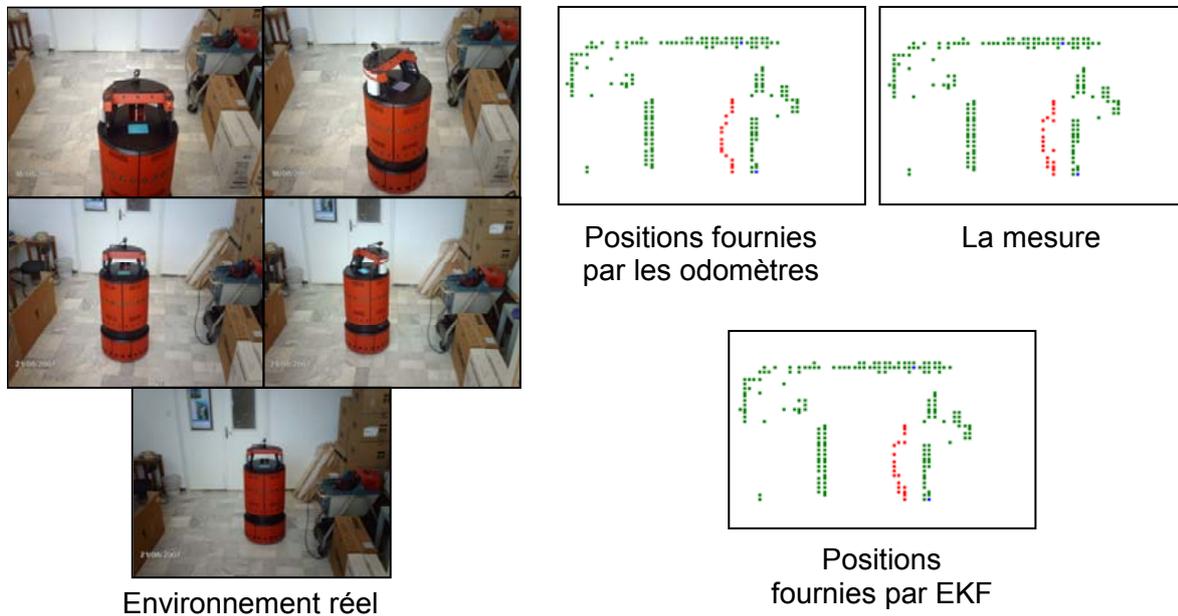
Figure 4.36 Résultat de la localisation par le filtre de Kalman étendu (EKF) pour une trajectoire courbe.

- (a) Déplacement du robot *B21r*.
- (b) Carte globale reconstruite (les points rouges représentent le centre de gravité du robot).
- (c) Positions du *B21r* fournies par les odomètres
- (d) La mesure (positions du *B21r* fournies par la mise en correspondance)
- (e) Positions du *B21r* fournies par EKF.

Dans les deux tests, l'EKF fournit une position imprécise au départ mais avec le déplacement du robot la précision augmente et le filtre converge. Celui-ci corrige l'erreur odométrique et bénéficie de la précision de la méthode de la mise en correspondance du moment que l'EKF optimise sa zone de recherche, pour donner une estimation de la position réelle du robot.

On remarque aussi que suite à l'erreur sur la position initiale, l'EKF met en cause les données odométriques et corrige cette erreur.

Voici dans la figure 4.37 d'autres exemples de trajectoires parcourues par le robot mobile *B21r*.



Trajectoire (a)

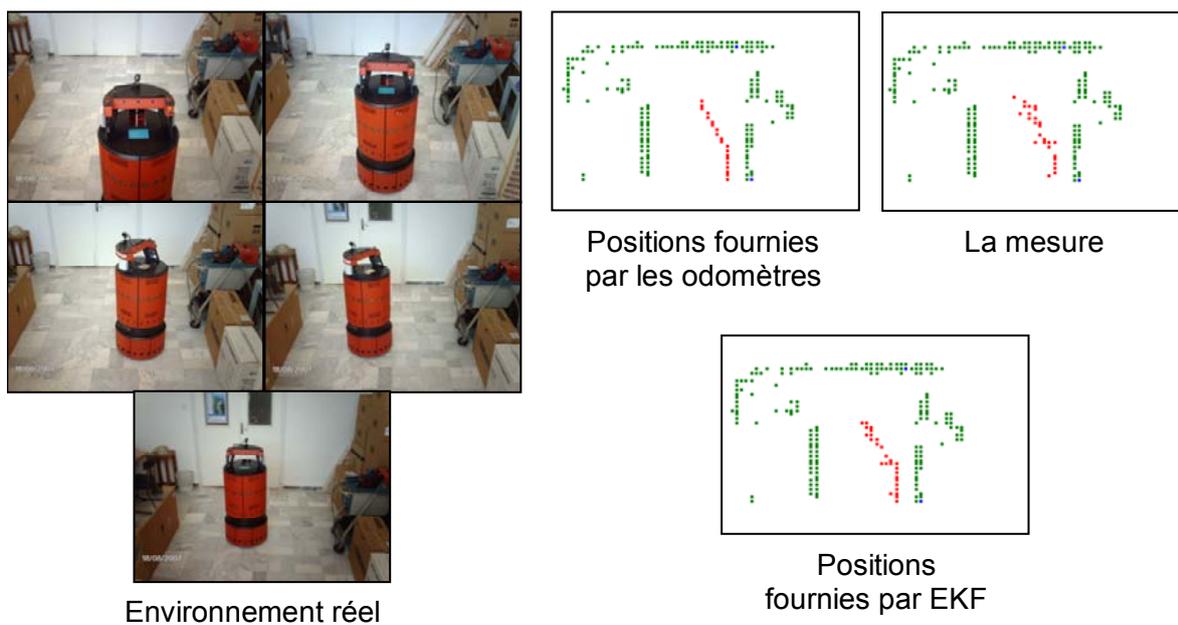
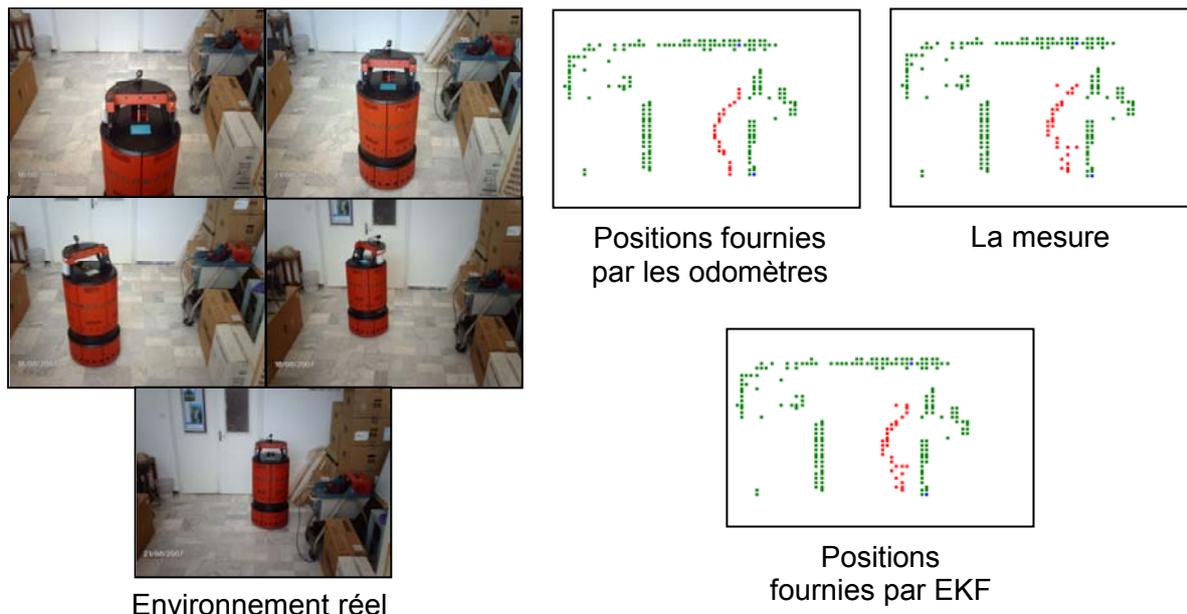


Figure 4. 37 Résultat de la localisation par le filtre de Kalman étendu (EKF) pour différentes trajectoires.

On voit bien que l'EKF augmente la précision de la mesure tout en convergeant.

Pour mieux voir la différence entre les positions fournies par l'EKF, les odomètres et la mesure (fournie par la mise en correspondance), on représente ces positions suivant les coordonnées x , y puis φ en fonction du nombre d'échantillons.

On a pris comme exemples les trajectoires (a) et (b) de la figure 4.37. Les résultats de la trajectoires (a) sont :

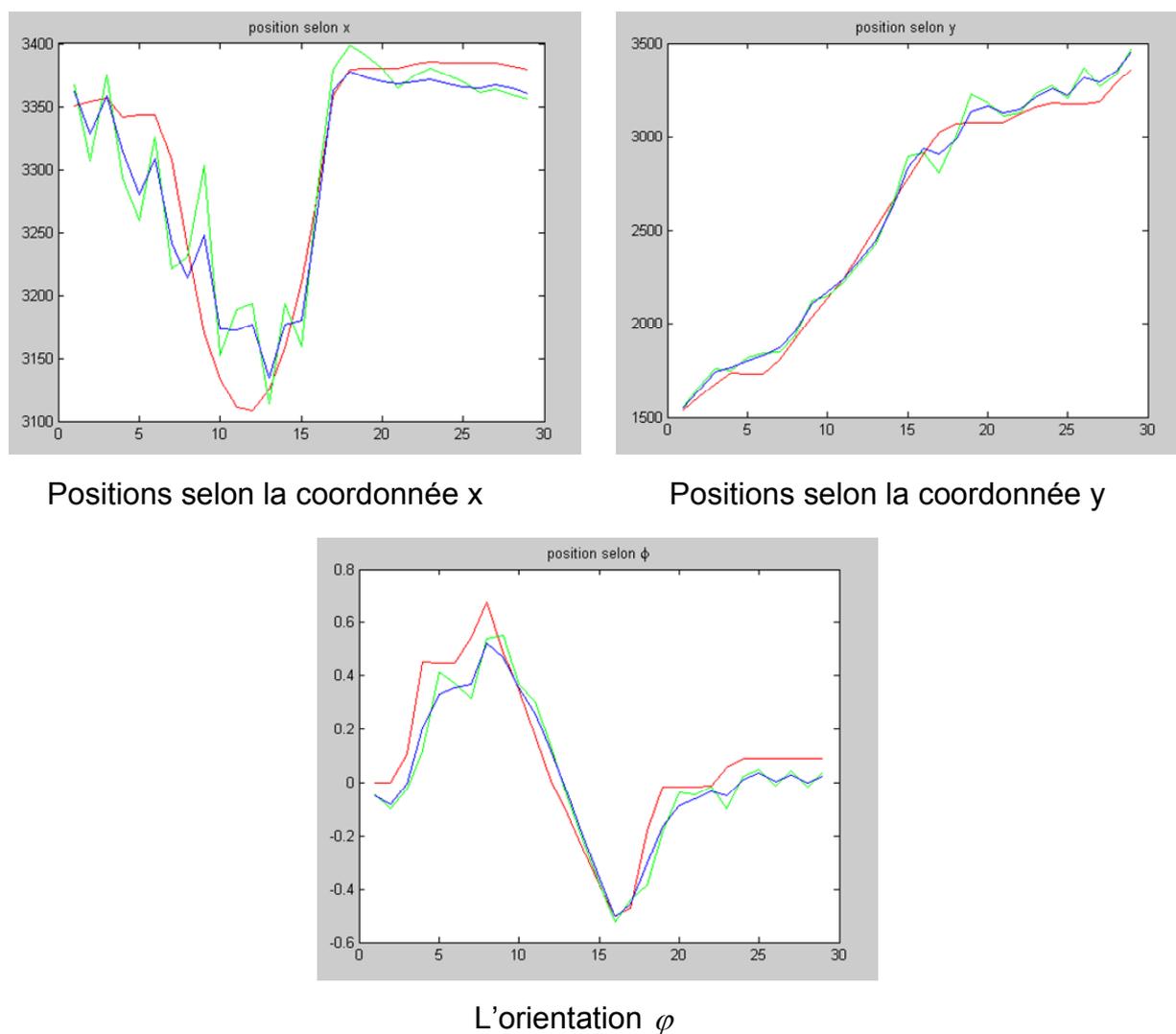


Figure 4.38 Comparaison entre les positions fournies par les odomètres, l'EKF et la mesure (trajectoire a)
 Courbe rouge : positions fournies par les odomètres
 Courbe verte : la mesure
 Courbe bleue : positions fournies par EKF

Pour la position en x, la forme de la courbe EKF indique que la sortie a des oscillations rapides et grandes au départ, mais leur amplitude diminue au fil des itérations. Ce qui veut dire que l'EKF est imprécis au départ mais sa précision augmente quand le nombre d'itérations augmente.

Pour la position en y, on remarque que les trois courbes présentes une même position initiale, donc il n'y a pas d'erreur sur cette position (cela est confirmé par

l'EKF). Ce qui n'est pas le cas pour l'orientation φ , où l'EKF corrige l'erreur présente sur la position initiale, puis corrige le reste des positions (ce qui est visible à la fin de la trajectoire).

La sortie fournie par EKF bien qu'ayant presque la même forme générale que la courbe de la mesure, elle est plus stable avec des oscillations d'amplitude plus petite. L'EKF augmente la précision de la mesure et l'utilise pour corriger les erreurs odométriques.

Pour le deuxième exemple (la trajectoire (b) de la figure 37) on a les résultats suivants:

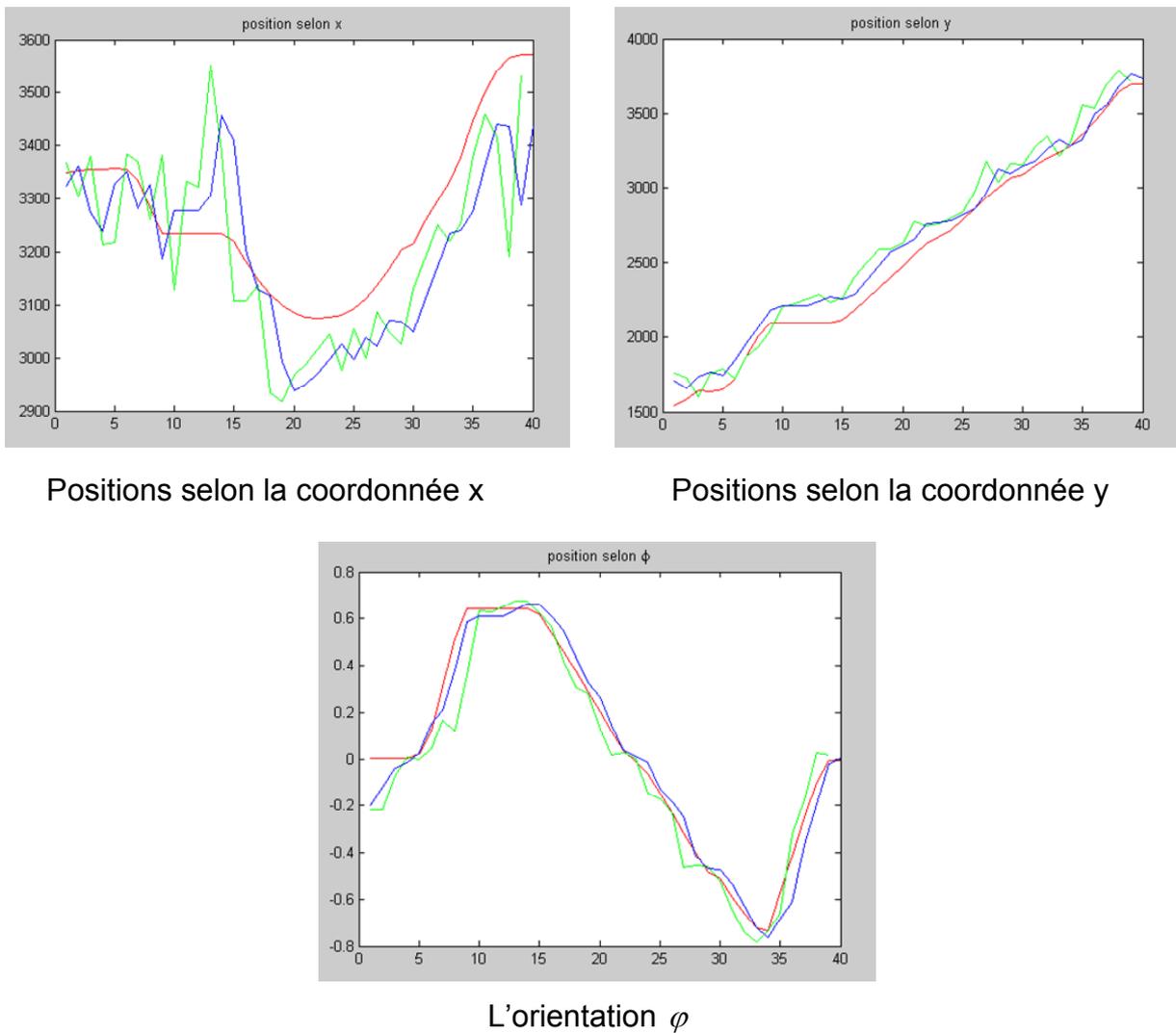


Figure 4.39 Comparaison entre les positions fournies par les odomètres, l'EKF et la mesure (trajectoire b)

Courbe rouge : positions fournies par les odomètres

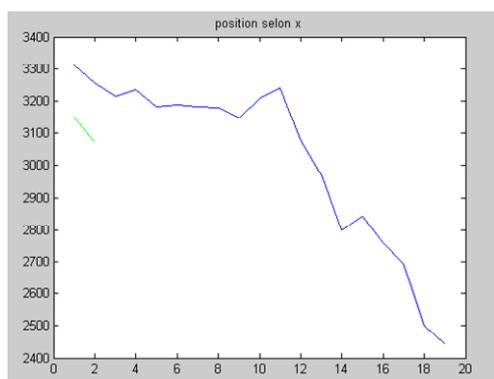
Courbe verte : la mesure

Courbe bleue : positions fournies par EKF

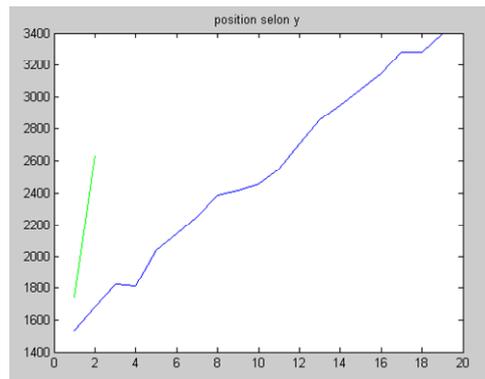
Dans ce cas, il y a une erreur sur la position odométrique initiale au niveau des trois coordonnées (x, y, φ) , et d'autres erreurs dans la trajectoire. L'EKF utilise la mesure pour corriger ces erreurs et se rapprocher le plus de la position réelle.

Durant son implémentation, on remarque que l'EKF est beaucoup plus rapide que la mise en correspondance, déjà parce que lui-même est un algorithme qui ne nécessite pas un grand coût calculatoire, et puis utilisant la mise en correspondance pour déterminer la mesure, celui-ci optimise la dimension de sa zone de recherche qui diminue avec la convergence de l'algorithme. C'est dans ce contexte qu'on veut comparer les deux méthodes (EKF et mise en correspondance). On a essayé de faire déplacer le robot suivant une même trajectoire pour les deux algorithmes (afin de garder la même distance parcourue), et avec une même vitesse constante pendant toute la trajectoire.

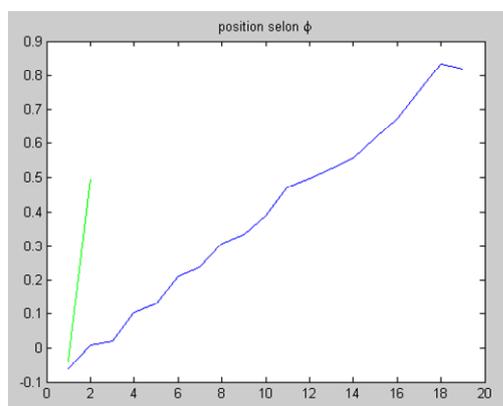
(a) Zone de recherche = 3×3 m



Positions selon la coordonnée x

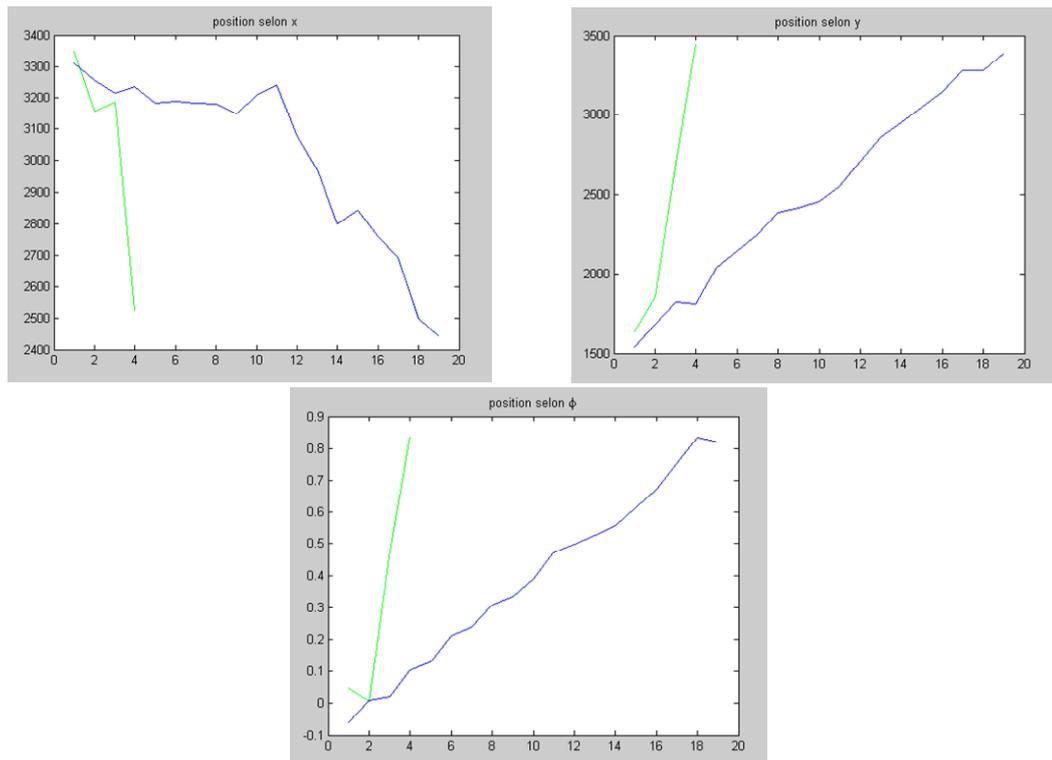


Positions selon la coordonnée y



Positions selon la coordonnée φ

(b) Zone de recherche = 2*2m



(c) Zone de recherche = 1*1m

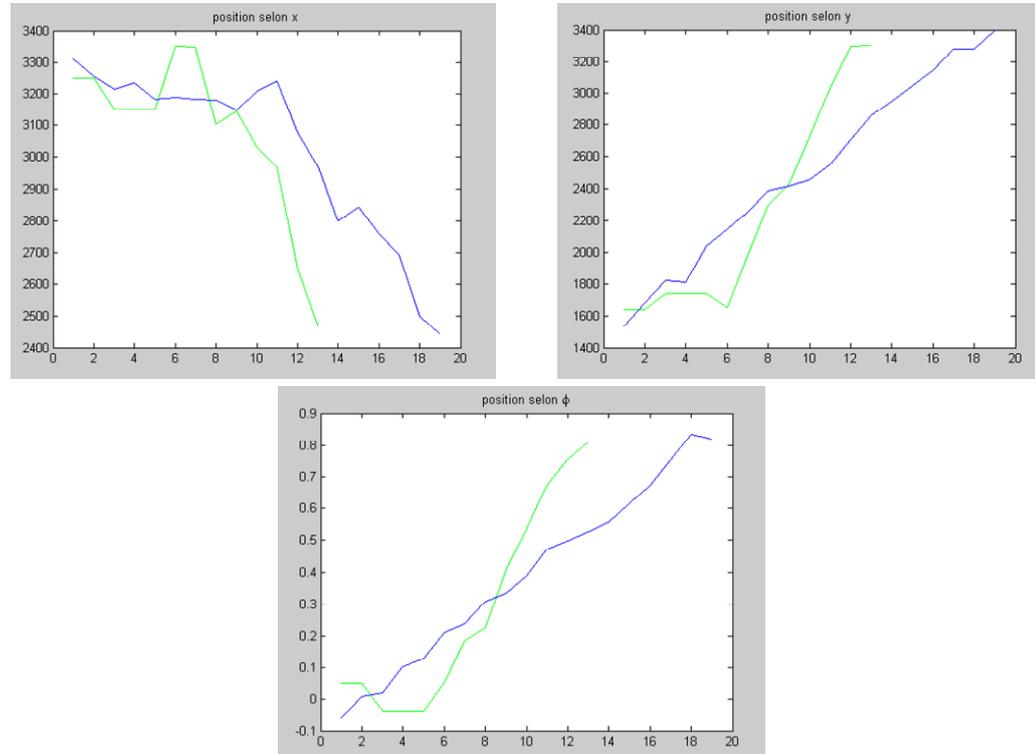


Figure 4.40 comparaison entre EKF (en bleu) et la mise en correspondance (en vert) avec des zones de recherche différentes.

Pour une mise en correspondance avec une zone de recherche de 3x3mètres (figure 40.a), l'EKF est neuf fois plus rapide. Donc l'EKF résout bien le problème du grand coût calculatoire que nécessite la mise en correspondance. C'est un problème dû en partie à la dimension de sa zone de recherche, on voit bien dans les figures 40.b et 40.c que le nombre d'échantillons augmente quand la dimension de la zone de recherche diminue, c'est ce qui nous a conduit bien sur à utiliser l'EKF pour l'estimation de cette zone.

4.5 Conclusion

Dans ce chapitre, nous avons décrit en premier lieu la manière de modéliser les capteurs embarqués sur le robot à savoir les capteurs ultrasonores, infrarouges et le laser, afin de les mettre en œuvre pour construire la grille globale et la grille locale. Après expérimentation, il s'est avéré qu'on ne peut se fier aux données US pour la création des cartes, à cause des problèmes qui lui sont associés (réflexions multiples, diaphonie, etc.) vu leurs dispositions en ceinture et le fait qu'ils sont actionnés au même temps. Quant aux capteurs infrarouges, ils ne peuvent être utilisés à cause de leurs portées assez limités et leurs sensibilités à la lumière. C'est bien pour ces raisons qu'on a choisit pour la création des cartes le capteur laser réputé pour sa précision. Les capteurs US et IR seront utilisés pour l'arrêt d'urgence du robot, sinon pour balayer la zone non perçue par le laser.

Cette partie est utilisée dans la phase de localisation par la mise en correspondance de ces deux grilles pour déterminer la position et l'orientation du robot. Cette méthode a été présentée en détail et après implémentation sur le robot *B21r*, les tests ont révélés que bien que cette méthode fournit des résultats précis, elle présente le grand inconvénient d'être très lourde en temps de calcul (surtout pour des grands espaces), ce qui dépend de la dimension de la zone de recherche. C'est pour cette raison qu'on a eu recours à l'algorithme du filtre de Kalman étendu pour la localisation du robot *B21r*. celui-ci utilise la position fournie par les odomètres et la position fournie par la mise en correspondance des grilles (basée sur le laser) pour déterminer une estimation de la position réelle du robot. Donc l'EKF corrige les erreurs odométriques d'une part, et optimise la méthode de la mise en correspondance en réduisant sa zone de recherche d'autre part.

CONCLUSION

L'objectif de ce travail était de développer un algorithme de localisation fiable qui permet de déterminer la position du robot mobile *B21r* dans un environnement d'intérieur. Pour ce faire il nous a fallu résoudre un certain nombre de problématiques dont celle de la localisation. Mais le préalable aux étapes de localisation et de modélisation est la construction d'un modèle sensoriel robuste de l'environnement. Il nous a donc fallu choisir un système de perception permettant de gérer des primitives sensorielles les plus pertinentes possibles. Après expérimentation des capteurs embarqués, ce choix s'est porté sur le capteur laser, ce capteur a la particularité de fournir des données sur 180 degrés en une seule acquisition, en plus de ses grandes qualités de précision. Pour construire le modèle de l'environnement, il nous a fallu développer une méthode de modélisation basée sur les grilles d'occupation grâce à ce télémètre laser.

Pour la localisation du robot mobile, on a choisit la mise en correspondance des deux grilles ; la première représente l'environnement du robot (carte globale), et la deuxième représente la partie de l'environnement perçue à partir de la position du robot (carte locale). Après appariement des deux cartes pour toutes les translations et rotations possibles de la carte locale par rapport à la carte globale, la meilleure correspondance est évaluée pour déterminer la position du robot. A l'issue de l'implémentation de cette méthode, les résultats obtenus étaient satisfaisants du point de vue précision, mais on a été confronté au problème du temps de traitement, qui est en fonction de la taille de la zone de recherche. En effet quand sa taille augmente la précision augmente mais le temps de calcul augmente aussi.

Pour remédier à ce problème, nous avons utilisé le filtre de Kalman étendu qui fusionne les données odométriques (modèle du système) et les données laser (mesure) afin d'estimer la position du robot. Pour déterminer la mesure, il utilise la méthode de la mise en correspondance des grilles tout en estimant sa zone de recherche et la position autour de laquelle elle effectue la recherche. L'implémentation de cet algorithme laisse apparaître que l'EKF corrige les erreurs

odométriques d'une part et optimise la méthode de la mise en correspondance d'autre part, ce qui le rend très adapté pour des applications en temps réel.

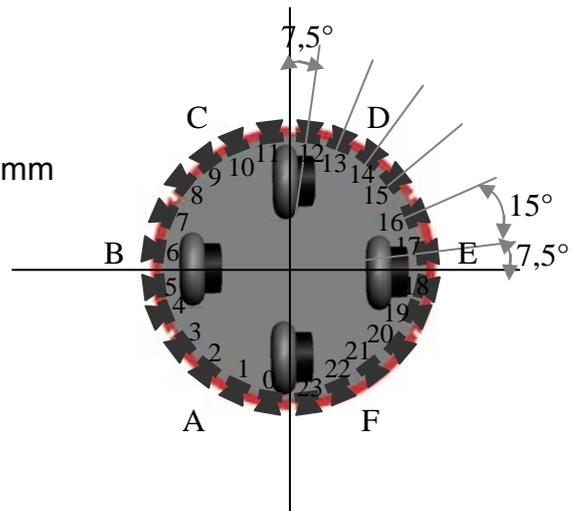
D'un point de vue pratique, l'implémentation des algorithmes développés sur un robot réel permettrait d'évaluer de façon plus précise les performances en termes de temps d'exécution. Concernant l'aspect théorique du travail, les équations mathématiques élaborées ont bien été vérifiées dans la phase implémentation. Ce qui vient confirmer la fiabilité des résultats obtenus.

Enfin, comme perspective de notre travail. Celui-ci va être utilisé dans les algorithmes de navigation développés au sein de la division robotique (au CDTA), il peut être amélioré en enrichissant la perception du robot et en augmentant la précision de la localisation en utilisant d'autres capteurs tel que la caméra CCD (N/B).

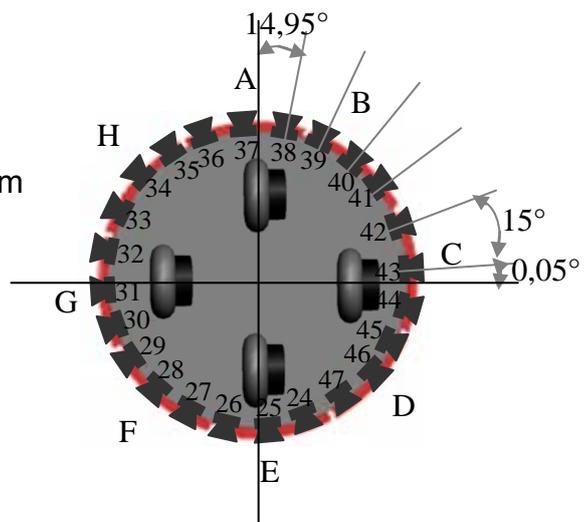
APPENDICE A

L'enclosure

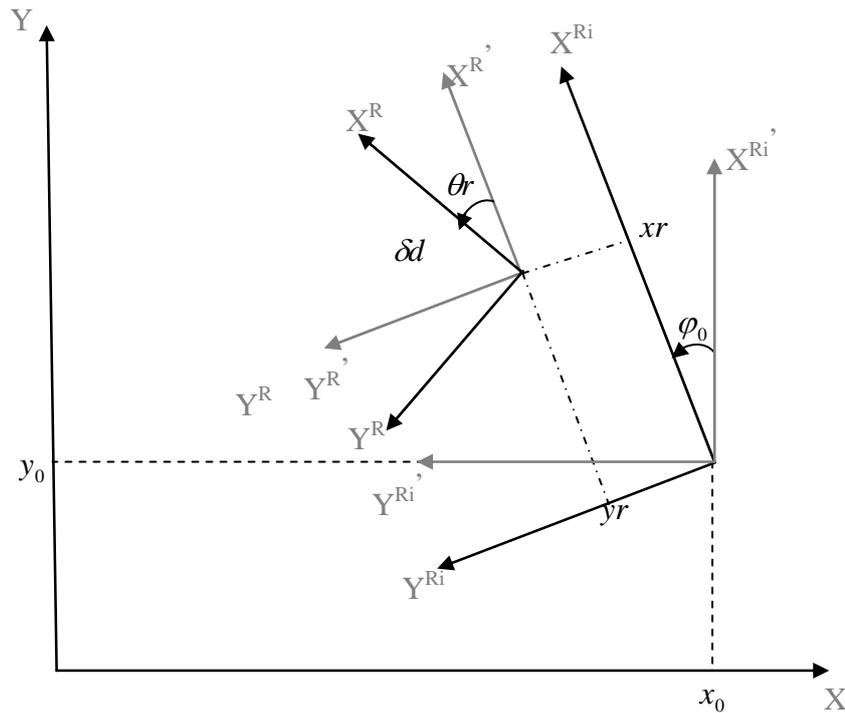
Périmètre = 1580mm
 Rayon = 252mm
 Diamètre_{US} = 40mm
 Espace entre deux US = 25mm

La base

Périmètre = 1700mm
 Rayon = 270mm
 Diamètre_{US} = 29mm
 Espace entre deux US = 40mm



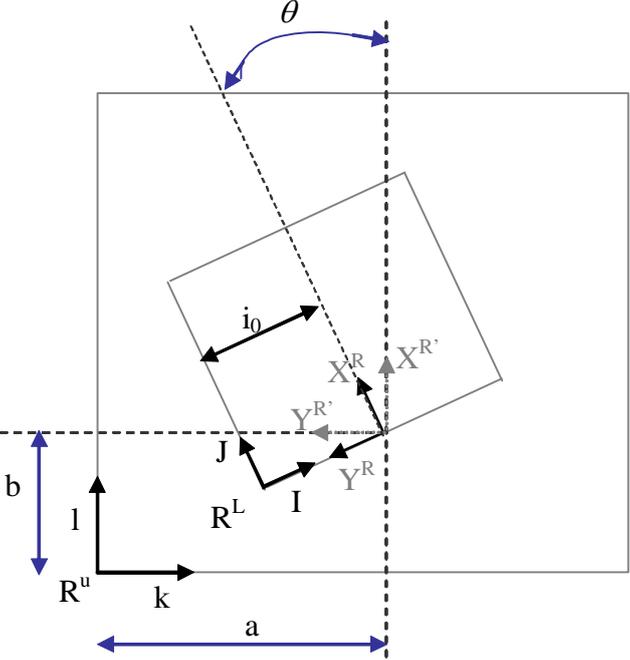
APPENDICE B
Formules de changement de repères :
Repère mobile et repère du monde



$R^R (X^R, Y^R)$: repère mobile (lié au robot mobile).
 $R^u (X, Y)$: repère univers (lié à l'environnement).
 $R^{Ri} (X^{Ri}, Y^{Ri})$: repère du robot à la position initiale.

<p>Passage du repère R^R à $R^{R'}$</p> $Y^{R'} = X^R \sin \theta_r + Y^R \cos \theta_r$ $X^{R'} = X^R \cos \theta_r - Y^R \sin \theta_r$	<p>Passage du repère $R^{Ri'}$ à R^u</p> $X = x_0 - Y^{Ri'}$ $Y = y_0 + X^{Ri'}$
<p>Passage du repère $R^{R'}$ à R^{Ri}</p> $Y^{Ri} = y_r + y^{R'}$ $X^{Ri} = x_r + X^{R'}$	<p>Passage du repère R^{Ri} à R^u</p> $X = x_0 - Y^{Ri} \cos \varphi_0 - X^{Ri} \sin \varphi_0$ $Y = y_0 - Y^{Ri} \sin \varphi_0 + X^{Ri} \cos \varphi_0$
<p>Passage du repère R^{Ri} à $R^{Ri'}$</p> $Y^{Ri'} = Y^{Ri} \cos \varphi_0 + X^{Ri} \sin \varphi_0$ $X^{Ri'} = -Y^{Ri} \sin \varphi_0 + X^{Ri} \cos \varphi_0$	<p>Passage du repère R^R à R^u</p> $X = x_0 - y_r \cos \varphi_0 - x_r \sin \varphi_0 - X^R \sin(\theta_r + \varphi_0) - Y^R \cos(\theta_r + \varphi_0)$ $Y = y_0 - y_r \sin \varphi_0 + x_r \cos \varphi_0 + X^R \cos(\theta_r + \varphi_0) - Y^R \sin(\theta_r + \varphi_0)$

APPENDICE C
Formules de changement de repère :
Grille locale et grille globale

<p><u>Passage de R^L à R^u</u></p> $y^R = i_0 - I$ $x^R = J$	 <p style="text-align: center;">Le repère du robot dans le repère Local et le repère univers</p>
<p><u>Passage de R^R à $R^{R'}$</u></p> $y' = y \cos \theta + x \sin \theta$ $x' = -y \sin \theta + x \cos \theta$	
<p><u>Passage de $R^{R'}$ à R^u</u></p> $k = a - y'$ $l = b + x'$	
<p><u>Passage de R^L à R^u</u></p> $k = a + (I - i_0) \cos \theta - J \sin \theta$ $l = b + (I - i_0) \sin \theta + J \cos \theta$	

REFERENCES

1. A. Courcelle, « Localisation d'un robot mobile : Application à l'aide à la mobilité des personnes handicapées moteur », Doctorat de l'université de METZ, France, Janvier 2000.
2. Lim Chee Wang, Lim Ser Yong, « *Mobile Robot Localization for Indoor Environment* », SIMTech Technical Report, Singapore Institute of Manufacturing Technology, *Mechatronics Group*, 2002.
3. Richard Thrapp, Cristian Westbrook and Devika Subramanian, « Robust localization algorithms for an autonomous campus tour guide », department of computer science, Rice University, Houston TX77005, USA, February 15, 2001.
4. Raj Madhavan, Kingsley Fregene and Lynne E. Parker, « Distributed heterogeneous outdoor multi-robot localization », Center Engineering Science Advanced Research, Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831-6355, USA, 2002.
5. Philippe Bonnifait, « Localisation précise en position et attitude des robots mobiles d'extérieur à évolutions lentes », thèse, Ecole Doctorale Science pour l'ingénieur de Nantes, Spécialité : Automatique et Informatique Appliquée, le 24 Novembre 1997.
6. W. Khalil, E. Dombre, « Modélisation identification de l'environnement et commande des robots », 2^e édition revue et augmentée Paris, Hermès science publications, 1999.
7. G. Campion, G. Bastin and B. D'Andréa-Novel, « Structural Properties and Classification of Kinematics and Dynamic Models of Wheeled Mobile Robots », IEEE Transactions on Robotics and Automation, vol. 12, no. 1, pages 47–62, Atlanta, USA, 1996.
8. B. Raucent, P. Sente, H. Buyse, et J-C Samin, « Robot mobile à trois degrés de liberté », revue générale de l'électricité, 6, pp. 35-43, Paris, France, 1989.
9. Y. Zhao, S.L. Bement, « Kinematics, Dynamics and Control of Wheeled Mobile Robots », IEEE International Conference on Robotics and Automation, Nice, France, May 1992, pp. 91-96.
10. D. T. Greenwood, « Principles of Dynamics », Second Edition, Englewood, New Jersey, Prentice-Hall, Inc., 1988.
11. P. F. Muir, « Modelling and control of Wheeled Mobile Robots », Ph. D. Thesis, Carnegie Mellon University, Pittsburgh, Aug. 1988.

12. G. Frappier, « *Système inertiels de navigation pour robots mobiles* », Séminaire "Les robots mobiles", EC2, Paris, 1990.
13. J. Borenstein, « Internal Correction of Dead-reckoning Errors With the Compliant Linkage Vehicle », *Journal of Robotic Systems*, Vol. 12, No. 4, pp. 257-273, University of Michigan, Ann Arbor, USA, April 1995.
14. J. Borenstein, « Internal Correction of Dead-reckoning Errors with the Smart Encoder Trailer », *International Conference on Intelligent Robots and Systems*, vol1, pp. 127–134, Munich, Germany, 1994.
15. J. Borenstein, « The CLAPPER: A dual-drive Mobile Robot with Internal Correction of Dead-reckoning Errors », *International Conference on Robotics and Automation*, vol. 3, pp. 3085–3090, University of Michigan, Ann Arbor, USA, 1995.
16. J. Borenstein, L. Feng, « Correction of systematic odometry errors in mobile robots », *International Conference on Intelligent Robots and Systems*, vol. 3, pp. 569–574, Pittsburgh, Pennsylvania, 1995.
17. J. Borenstein, L. Feng, « Measurement and correction of systematic odometry errors in mobile robots », *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 869–880, University of Michigan, Ann Arbor, USA, 1996.
18. J. Borenstein, « Experimental results from internal odometry error correction with the OmniMate mobile robot », *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 963– 969, University of Michigan, Ann Arbor, USA, 1998.
19. J. Vaganay, « *Conception d'un système multisensoriel de localisation dynamique 3D pour robot mobile* », Thèse de doctorat, LIRMM, Montpellier, juillet 1993.
20. A. Elfes, « Sonar-Based Real-World Mapping and Navigation », *IEEE Journal of Robotics and Automation*, vol RA-3, n°3, pp.233-249, USA, June 1987.
21. POLAROUND 1991, « Ultrasonic ranging System », Product Literature. Polaroid Corporation, 784Memorial Drive, Combridge, Ma02139, 617-3863964.
22. Y. D. Kwon, J.S. Lee, « A stochastic Environment Modelling Method for Mobile Robot by using 2-D Laser scanner », *Proceedings of International Conference IROS'97*, pp. 1688-1693, Albuquerque, USA.
23. S. Uchida, S. Maeyama, A. Ohya, S. Yuta, « Position Correction Using Elevation Map for Mobile robot on Rough Terrain », *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'98*, Victoria, Canada, October 13-17, 1998, pp. 582-587.
24. C. Gourley, M. Trivedi, « Sensor Based Obstacle Avoidance and Mapping for Fast Mobiles Robots », *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1306-1311, San Diego, USA, 1994.

25. P. Moutarlier, R. Chatila, « *An Experimental System for Incremental Environment Modelling by an Autonomous Mobile Robot* », Experimental Robotics I, Springer Verlag, Vol 139, 1989, pp. 327-346.
26. A. Elfes, L. Matthies, « *Sensor Integration for Robot Navigation Combining Sonar and Stereo Range Data in a Grid Based Representation* », IEEE Conference on Decision and Control, Los Angeles CA, December 1987.
27. A. Elfes, « *Using Occupancy Grids for Mobile Robot Perception and Navigation* », IEEE Computer, pp.46-57, Los Alamitos, USA, 1989.
28. H. Bulata, « *Modélisation d'un environnement structuré et localisation sur amers pour la navigation d'un robot mobile autonome* », Thèse de l'université Paul Sabatier de Toulouse, mai 1996.
29. T. Pannerec, M. Oussaleh, H. Maaref, C. Baaret, « *Absolute Localisation of a Miniature Mobile Robot Using Heterogeneous Sensors Comparison Between Kalman Filter and Possibility Theory* », IMACS Multiconference on Computational Engineering in Systems Applications, Nabeul-Hammamat, Tunisie, April 1-4, 1998, pp.265-267.
30. P. Weckesser, R. Dillmann, « *Modeling unknown environments with a mobile robot* », Robotics and Autonomous Systems, vol 23, pp.293-300, Germany, 1998.
31. J.L. Crowley, « *World modelling and position estimation for a mobile robot using ultrasonic ranging* », Proceedings of the IEEE International Conference on Robotics and Automation. 1998.
32. K.O. Arras, N. Tomatis, « *Improving Robustness and Precision in Mobile Robot Localisation by Using Laser Range Finding and Monocular Vision* », 3rd European Workshop on Advanced Mobile Robots (Eurobot'99), Zurich, Switzerland, September 6-8, 1999.
33. T. Duckett and U. Nehmzow, « *Experiments in evidence based localisation for a mobile robot* », In D. Corne and J. L. Shapiro, editors, *Proceedings of the AISB 97 workshop on Spatial Reasoning in Animals and Robots*, Springer, 1997.
34. G. Dedeoglu, M. Mataric, and G. S. Sukhatme, « *Incremental online topological map building with a mobile robot* », In *Proceedings of Mobile Robots XIV - SPIE*, pages 129–139, 1999.
35. S. Thrun, « *Robotic Mapping* », A Survey, School of computer Science, Carnegie Mellon University, Pittsburgh, February 2002.
36. C.Wang and C. Thorpe, « *A hierarchical object-based representation for simultaneous localization and mapping* », In International Conference on Intelligent Robots and Systems, (IROS'04), Sendai (Japon), septembre - octobre 2004.
37. S. Thrun, « *Learning occupancy grid maps with forward sensor models* », In *Autonomous Robots*, volume 15, pages 111{127, Pittsburgh, Pennsylvania, 2003.

38. E.Brassart, « Localisation absolue d'un robot mobile autonome par des balises actives et un système de vision monoculaire », Thèse de l'université de Compiègne, France, Janvier 1995.
39. S.Li, A. Hayashi, « Robot Navigation in Outdoor Environments by using GPS Information and Panoramic Views », Proceedings of IEEE/RS International Conference on Intelligent Robots and Systems, IROS'98, Victoria, Canada, October 13-17, 1998, pp. 570-575.
40. J.H. Kim, H.S. Cho, « Experimental investigation for the estimation of a mobile robot's position by linear scanning of a landmark », Robotics and Autonomous Systems, vol 13, pp.39-51, Amsterdam, Pays Bas, 1994.
41. J.H. KO, S.D. Kim, M.J. Chung, « A method of indoor mobile robot navigation using acoustic landmark », Proceedings of IEEE International Conference on Robotics and Automation, New Mexico, 1997, pp. 1726-1731.
42. U.D. Hanebeck, G. Schmidt, « Set theoretic localization of fast mobile robot using an angle measurement technique », Proceedings of IEEE International Conference on Robotics and Automation, New Mexico, 1996, pp. 1387-1394.
43. C.B. Madsen, C.S. Andersen, « Optimal landmark selection for triangulation of robot position », Robotics and Autonomous systems, vol 23, pp. 277-292, Alborg university, Denmark, 1998.
44. L. Delahoche, C. Pegard, B. Marhic, P. Vasseur, « A Navigation System Based on an Omnidirectional Vision Sensor », Proceedings of the International Conference IROS'97, pp. 718-724, Grenoble, France.
45. C. F. Olson, « Probabilistic self-localization for mobile robots », *IEEE Transactions on Robotics and Automation*, 16(1), California institute of technology, Pasadena, USA, 2000.
46. L. J. Guibas, R. Motwani, and P. Raghavan, « The robot localization problem », *Algorithmic Foundations of Robotics*, pages 269–282, 1995
47. J. borenstein, J. Evans, « The omniMate Mobile robot – Design, Implementation, and Experimental Results », Proceedings of the IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, April 1997, pp.3505-35 10.
48. Y. Yagi, Y. Nishizawa, M. Yachida, « *Map-based navigation for a mobile robot with omnidirectional image sensor COPIS* », *IEEE Transactions on Robotics and Automation*, Vol. 11, n°5, p. 634-648, Osaka University, Japan, October 1995.
49. G. Welch, G. Bishop, « An Introduction to the Kalman Filter », University of North Carolina at Chapel Hill, Department of Computer Science, TR 95-041, September 1997.

50. D. A. Forsyth, J. Ponce, «Computer Vision A Modern Approach », Livre, (a) Chapter 1: An Introduction To Probability (b) Chapter2: Tracking with Non-Linear Dynamic Models, August 2002, New Jersey, USA.
51. R. van der Merwe, E. A. Wan, « Sigma-Points Kalman Filters For Probabilistic Inference in Dynamic State-Space Models », Workshop on Advances in Machine Learning, Montréal, Canada, June 2003.
52. S. J. Julier, J. K. Uhlmann, « A New Extension of the Kalman Filter To Non-Linear systems », International Symposium Aerospace/Defense Sensing, Simulation and Controls, Orlando, FL, 1997.
53. S. J. Julier, « Sequential State Estimation using the Unscented Transformation and the Kalman Filter », Workshop on New Directions in Signal Processing in the XXI century. Lake Louise, Alberta, Canada, October 5-10 2003.
54. E. A. Wan, R. van der Merwe, « The Unscented Kalman Filter for Nonlinear Estimation », Proceedings of Symposium 2000 on Adaptive Systems for Signal Processing, Communication and Control (AS-SPCC), IEEE, Lake Louise, Alberta, Canada, October 2000.
55. M. Nørgaard, N. K. Poulsen, O. Ravn, « Advances in Derivative-Free State Estimation for Non-Linear Systems », Tech. Rep. IMM-REP-1998-15, Tech. Univ. of Denmark, 2000.
56. R. van der Merwe, E. Wan, « Gaussian Mixture Sigma-Point Particle Filters for Sequential Probabilistic Inference in Dynamic State-Space Models », Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Hong Kong, April 2003.
57. R. van der Merwe, E. Wan, « The Square-Root Unscented Kalman Filter For State and Parameter Estimation », International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, Utah, May, 2001.
58. J. Laviola, « A Comparison of Unscented and Extended Kalman Filtering for Estimating Quaternion Motion », In the Proceedings of the 2003 American Control Conference, IEEE Press, June 2003, pp. 2435-2440.
59. Peihua, Li and Tianwen Zhang, « Unscented Kalman Filter for Visual Curve Tracking », In *Proceedings of Statistical Methods in Video Processing*, June, 2002.
60. Stenger, B., P. R. S, « Mendonc,a, and R. Cipolla. Model-Based Hand Tracking Using an Unscented Kalman Filter », In *Proceedings of the British Machine Vision Conference*, 63-72, September 2001.
61. Patric. Jensfelt, « Approaches to mobile robot localization in Indoor Environment», thèse de doctorat, department de signal, capteurs et systèmes, institut royale de technologie, School of Computer Science Carnegie Mellon University, Pittsburgh, PA 15213, on 2001

62. P. Jensfelt and S. Kristensen, « Active global localisation for a mobile robot using multiple hypotheses tracking », *IEEE Trans. on Robotics and Automation*, 17(5):748-760, October 2001.
63. I. Song, « Probabilistic Localization Methods for a Mobile Robot Navigation », Systems Design Engineering, University of Waterloo, Waterloo, ON May 13, 2002
64. Sebastian, Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert, « Robust Monte Carlo Localization for Mobile Robots », School of Computer Science Carnegie Mellon University, Pittsburgh, PA 15213 April 2000, CMU-CS-00-125
65. J. Borenstein, Y. Koren, « The Vector Field Histogram–Fast Obstacle-Avoidance for Mobile Robot », *IEEE Transactions on Robotics and Automation*, vol 7, n°3, pp.278-288, University of Michigan, Ann Arbor, USA, June 1991.
66. J.D. Courtney, A.K. Jain, « Mobile Robot Localisation via Classification of Multisensor Maps », *Proceedings of the IEEE International Conference on Robotics and Automation*, 1994, pp. 1672-1678.
67. P. Menezes, J. Dias, H. Araujo, A.D. Almeida, « Using Sonar Sensors to Perform Obstacle detection », *Revue d'Automatique et de productique appliquées*, 1995.