



HAL
open science

Localisation et cartographie simultanées de l'environnement à bord de véhicules autonomes : analyse de solutions fondées sur le filtrage de Kalman

François Chanier

► **To cite this version:**

François Chanier. Localisation et cartographie simultanées de l'environnement à bord de véhicules autonomes : analyse de solutions fondées sur le filtrage de Kalman. Automatique / Robotique. Université Blaise Pascal - Clermont-Ferrand II, 2010. Français. NNT : 2010CLF22021 . tel-00719341

HAL Id: tel-00719341

<https://theses.hal.science/tel-00719341>

Submitted on 19 Jul 2012

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

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

Université Blaise Pascal - Clermont II

Ecole Doctorale

Sciences Pour L'Ingénieur De Clermont-Ferrand

Thèse

présentée par

François CHANIER

pour obtenir le grade de

Docteur d'Université

(Spécialité : Vision pour la robotique)

**Localisation et cartographie simultanées de l'environnement
à bord de véhicules autonomes :
analyse de solutions fondées sur le filtrage de Kalman**

Soutenue publiquement **12 mars 2010** devant le jury :

S. MAMMAR	Professeur Université d'Evry	Président
P. BONNIFAIT	Professeur Université de technologie de Compiègne	Rapporteur
S. LACROIX	Directeur de Recherche LAAS	Rapporteur
P. CHECCHIN	Maître de Conf. Université Blaise Pascal	Examineur
C. BLANC	Maître de Conf. Université Blaise Pascal	Examineur
P. SAYD	Ingénieur CEA	Invité
L. TRASSOUDAINÉ	Professeur Université Blaise Pascal	Directeur de thèse

Remerciements

Je remercie tous les membres de mon jury, Monsieur MAMMAR pour l'avoir présidé, Monsieur SAYD, et Messieurs BONNIFAIT et LACROIX pour l'intérêt qu'ils ont porté à mes travaux en acceptant d'être mes rapporteurs.

Je souhaite remercier Laurent TRASSOUDAINE, mon directeur de thèse, Paul CHECCHIN et Christophe BLANC, constituant mon encadrement de thèse, pour m'avoir permis de réaliser cette thèse, pour m'avoir encadré durant toutes ces années, pour leurs précieux conseils, leur aide et la patience dont ils ont fait preuve durant la correction de ce manuscrit. Je souhaite également remercier le conseil général de l'Allier pour le financement de mes travaux de recherche.

Je voudrais témoigner toute ma gratitude envers les personnes qui m'ont apporté leur aide durant les années passées au LASMEA. Je pense notamment à Serge A., Laurent M., François M., Frédéric C., Roland C., Sébastien B. et surtout Christophe D. dont les conseils et les encouragements m'ont été d'une aide précieuse.

Bien sur je remercie l'ensemble des personnes que j'ai pu côtoyer au LASMEA, notamment Laurence ("ma chef"), Jon, Pierre, Lionel, Florent, Christophe, Manuel, Bertrand, Alex, Gaël, Réda, Hala, Nadir, Noël et Fabio dont leur amitié, leur soutien et leur bonne humeur ont rendu mon travail plus agréable au quotidien.

Mes dernières pensées vont à l'ensemble des personnes de ma famille qui m'ont toujours encouragé à poursuivre cette voie, à ma tendre et douce Laetitia, qui m'a soutenu dans les moments difficiles et, plus particulièrement, à mes parents.

Résumé

Pour être autonome, un robot mobile doit être capable de décrire l'environnement dans lequel il évolue. Pour ce faire, le robot doit pouvoir se localiser et construire une carte de l'environnement si celle-ci n'existe pas. Grâce à ces informations, il pourra par exemple éviter les obstacles et donc naviguer en toute sécurité. Cette capacité correspond à une étape obligatoire dans la route vers son autonomie totale.

Le sujet traité dans ce manuscrit répond à cette problématique. Il concerne la localisation d'un véhicule et la cartographie d'un environnement en simultané. Dans la littérature, ce problème correspond à l'acronyme **SLAM** pour *Simultaneous Localization And Mapping*. Pour résoudre ce problème, une multitude de solutions existe. Ces méthodes emploient différentes représentations de l'environnement mais aussi différentes méthodes d'estimation. Les deux premiers chapitres de ce manuscrit proposent un état de l'art non exhaustif mais complet des approches abordées durant la dernière décennie. Cette étude nous a menés à trois constatations importantes lors de l'emploi d'un filtre non-linéaire de Kalman dans un contexte SLAM :

- lorsqu'un télémètre laser est employé comme capteur extéroceptif, dans la majorité des travaux, l'environnement est représenté avec un unique type d'amer. La carte estimée est donc peu représentative de l'environnement exploré par le véhicule ;
- les estimations obtenues avec un filtre EKF-SLAM ne sont pas consistantes. Certains articles le soulignent mais peu ont réalisé une étude complète de ce problème ;
- l'approche EKF-SLAM, telle qu'elle fut introduite, ne répond pas aux critères d'observabilité d'un système.

L'objectif des travaux de cette thèse n'est pas de proposer une application EKF-SLAM mais une étude des différents problèmes cités. Pour chaque problème étudié, une solution est proposée et critiquée à l'aide de résultats de simulation et d'expérimentation.

Représentation de l'environnement

Dans ce manuscrit, une **représentation hétérogène** de l'environnement est adoptée. Les résultats expérimentaux prouvent que l'emploi d'amers avec des géométries différentes apportent plus d'informations sur l'environnement visité et n'altèrent pas la précision des estimations du système. La principale difficulté d'une telle approche réside dans la différenciation des amers mesurés et dans la capacité à les associer. Pour résoudre ce problème, une étape d'association de données est présentée pour détecter simplement les appartenances entre deux amers de types différents.

Consistance des estimations d'une approche EKF-SLAM

Une synthèse des problèmes de consistance des estimations d'une approche EKF-SLAM est réalisée. A travers des scénarii simples de simulation, une étude analytique des raisons de l'inconsistance de l'EKF-SLAM est présentée. Après avoir établi les limites théoriques de convergence de l'EKF-SLAM, nous prouvons que l'application d'un filtre polynomial de Kalman, **PEKF-SLAM**, permet de diminuer les problèmes de consistance et d'améliorer la précision des estimations.

Observabilité d'une approche EKF-SLAM

Une étude rigoureuse de la matrice d'observabilité d'une approche SLAM prouve qu'un tel système n'est pas observable. Ceci implique que la convergence des estimations ne peut être garantie. Pour répondre à cette problématique, une méthode, nommée **SLAM-contraint multi-cartes**, est présentée dans ce manuscrit. Elle permet de rendre une approche SLAM observable via la connaissance d'une carte de l'environnement. Les résultats obtenus avec une application de cette méthode couplée à un filtre EKF en démontre l'intérêt.

Mots clés : cartographie, consistance, filtre de Kalman, localisation, observabilité, robot mobile

Abstract

In order to be autonomous, a mobile robot must be able to describe the environment where it evolves. To this end, the robot must localize itself and make a map of the environment if no one is available. With this information, it will be able, for example, to avoid obstacles and therefore, to navigate safely. This ability is a compulsory step on the way of its complete autonomy.

The subject handled in this document deals with this problem. It refers to simultaneous vehicle localization and environment mapping (**SLAM**). In order to solve this problem, a lot of solutions exist. These methods use different environment representations but also different estimators. The first and second chapters of this manuscript suggest a state of the art of approaches proposed during the last decade. This study showed three important points when an extended Kalman filter is used in a SLAM context :

- when a range finder is used as exteroceptive sensor, environment is represented with only one feature kind in the majority of works. The estimated map is not representative of the explored environment ;
- estimates obtained with an EKF-SLAM filter are not consistent. Some articles underline this fact but few articles have carried out a complete study of this problem ;
- EKF-SLAM approach, like it has been introduced, does not answer system observability criteria.

The goal of my research works is not to suggest a new EKF-SLAM application but a study of the quoted problems. For each studied problem, a solution is introduced and criticized by means of simulation and experiment results.

Environment representation

In this manuscript, a **heterogeneous representation** of the environment is adopted. The experimental results show that the use of features with different geometries brings more information onto the visited environment and does not change the estimate accuracy of the system. The main difficulty of such approach is to differentiate the measured features and in the ability to associate them. To solve this problem, a data association step is presented to detect easily the belongings of two features with different geometries.

Estimate consistency of EKF-SLAM approach

A synthesis of EKF-SLAM estimate consistency problem is done. Through simple simulation scenario, an analytical study of the EKF-SLAM discrepancy reasons is presented. After the quotation of the EKF-SLAM convergence theoretical limits, we show that the application

of polynomial Kalman filter, **PEKF-SLAM**, enables to reduce consistency problem and to improve estimate accuracy.

EKF-SLAM approach observability

A meticulous study of the SLAM approach observability matrix demonstrates that such system is not observable. This involves that estimate convergence can not be guaranteed. To answer this problem, a method, named **multi-maps Constrained SLAM**, is presented in this manuscript. It allows to make SLAM approach observable with a known environment map. The results obtained with the method application coupled with Kalman filter demonstrate its interest.

Keywords : mapping, consistency, Kalman filter, localization, observability, mobile robot

Glossaire et notations

Approche EKF-SLAM :

Variables du système :

$\mathbf{X} = [\mathbf{X}_v \mathbf{X}_M]^T$: vecteur d'état du système

\mathbf{P} : matrice de covariance associée au vecteur \mathbf{X}

$\mathbf{X}_V = [x_v y_v \theta_v]^T$: vecteur d'état associé au véhicule

x_v : abscisse de la position du véhicule dans le plan 2D

y_v : ordonnée de la position du véhicule dans le plan 2D

θ_v : orientation du véhicule dans le plan 2D

\mathbf{P}_V : matrice de covariance associée au vecteur d'état du véhicule \mathbf{X}_V

\mathbf{X}_M : vecteur d'état associé à la carte de l'environnement

\mathbf{P}_M : matrice de covariance associée au vecteur d'état de la carte de l'environnement \mathbf{X}_M

\mathbf{P}_{VM} : matrice de cross-covariance entre les états du véhicule et ceux de la carte

Variables associées à la carte estimée :

W_g : repère associé à la carte globale

$\mathbf{X}_L = [x_L y_L \theta_L]^T$: vecteur d'état associé à un amer de type droite

x_L : abscisse du repère associé à un amer de type droite

y_L : ordonnée du repère associé à un amer de type droite

θ_L : orientation du repère associé à un amer de type droite

$\mathbf{X}_{ce} = [x_{ce} y_{ce}]^T$: vecteur d'état associé à un amer de type cercle

x_{ce} : abscisse du centre d'un amer de type cercle

y_{ce} : ordonnée du centre d'un amer de type cercle

r_{ce} : rayon d'un amer de type cercle

Prédiction du système :

\mathbf{f} : fonction non-linéaire décrivant l'évolution des états du vecteur \mathbf{X}

\mathbf{U} : vecteur des commandes appliquées sur le véhicule

\mathbf{w} : bruit associé aux commandes du véhicule

\mathbf{Q}_U : matrice de covariance associée aux bruits sur les commandes \mathbf{U}

\mathbf{F}_X : matrice jacobienne de la fonction \mathbf{f} par rapport à \mathbf{X}

\mathbf{F}_U : matrice jacobienne de la fonction \mathbf{f} par rapport à \mathbf{U}

Observation du système :

\mathbf{h} : fonction non-linéaire décrivant l'observation des états du vecteur \mathbf{X}

\mathbf{H} : matrice jacobienne de la fonction \mathbf{h} par rapport à \mathbf{X}

\mathbf{Z} : prédiction des observations du système

\mathbf{P}_Z : matrice de covariance associée au vecteur \mathbf{Z}

\mathbf{w}_m : bruit associé aux observations de la carte de l'environnement

\mathbf{Z}_{mesure} : vecteur de mesure de la carte de l'environnement

$\mathbf{P}_{Z_{mesure}}$: matrice de covariance associée au vecteur \mathbf{Z}_{mesure}

Cartographie du système :

\mathbf{g} : fonction non-linéaire décrivant la cartographie de nouveaux amers dans la carte estimée

\mathbf{G} : matrice jacobienne de la fonction \mathbf{g} par rapport à \mathbf{X}

Variables associées au filtre de Kalman :

v : innovation apportée par la mesure de l'environnement

\mathbf{S} : matrice de covariance associée à l'innovation v

\mathbf{W} : matrice du gain du filtre de Kalman

Approche PEKF-SLAM :Variables du système :

\mathbf{Z}_V : vecteur d'état agrandi associé à la position du véhicule

\mathbf{A} : matrice jacobienne de la fonction \mathbf{f} par rapport à \mathbf{Z}_V

\mathcal{Z} : vecteur d'état du système reformulé

\mathcal{Z}_V : vecteur d'état étendu avec les puissances de Kronecker associé à la position du véhicule

\mathcal{Z}_{pi} : vecteur d'état étendu avec les puissances de Kronecker associé à la position d'un point ou d'un cercle

\mathcal{Z}_{li} : vecteur d'état étendu avec les puissances de Kronecker associé à la position d'une droite

Approche SLAM-contraint multi-cartes :Variables associées à la carte connue :

W_c : repère associé à la carte connue

\mathbf{X}_C : vecteur d'état associé à la carte connue a priori

\mathbf{P}_C : matrice de covariance associée au vecteur \mathbf{X}_C

$\mathbf{X}_{W_c}^{W_g} = [x_c y_c \theta_c]^T$: vecteur d'état définissant la position du repère de la carte connue W_c dans une carte globale W_g

$\mathbf{C}_{X_{W_c}^{W_g}}$: matrice de covariance associée au vecteur d'état $\mathbf{X}_{W_c}^{W_g}$

Variables associées à une carte locale :

W_l : repère associé à la carte locale

\mathbf{X}_l : vecteur d'état associé à la carte connue a priori

\mathbf{P}_l : matrice de covariance associée au vecteur \mathbf{X}_C

$\mathbf{X}_{W_c}^{W_l}$: vecteur d'état définissant la position du repère de la carte connue W_c dans une carte locale W_l

$\mathbf{C}_{X_{W_c}^{W_l}}$: matrice de covariance associée au vecteur d'état $\mathbf{X}_{W_c}^{W_l}$

Abréviations :

SLAM : Simultaneous Localization And Mapping

EKF : Extended Kalman Filter

PEKF : Polynomial Extended Kalman Filter

IEKF : Iterated Extended Kalman Filter

UKF : Unscented Kalman Filter

NIS : Normalized Innovation Squared

PPV : Plus Proches Voisins

JCBB : Joint Compatibility Branch and Bound

NEES : Normalised Estimated Error Squarred

GPS : Global Positioning System

Table des matières

Table des matières	10
Table des figures	15
1 Introduction et état de l'art	19
1.1 Introduction générale	19
1.1.1 Le véhicule intelligent : au début, une fiction !	19
1.1.2 La réalité rattrape la fiction	20
1.1.3 Les véhicules intelligents « auvergnats »	21
1.2 Présentation de l'étude	23
1.2.1 Problème de la localisation d'un véhicule	23
1.2.2 Problème de la cartographie d'un environnement à bord d'un véhicule	24
1.2.3 Résolution simultanée des problèmes de localisation et de cartographie à bord d'un véhicule	25
1.2.4 Approche SLAM retenue	30
1.3 Résoudre le problème du SLAM avec un filtre EKF	31
1.3.1 Approches EKF-SLAM existantes	31
1.3.2 Problèmes observés avec une application EKF-SLAM	33
1.4 Description du manuscrit	36
1.4.1 Objectif et positionnement des travaux	36
1.4.2 Organisation du manuscrit	37
2 EKF-SLAM avec une représentation hétérogène de l'environnement	39
2.1 Introduction	39
2.2 Algorithme général de l'EKF-SLAM	40
2.2.1 Espace d'état	40
2.2.2 Transition d'état	41
2.2.3 Fonction d'observation	42
2.2.4 Fonction de cartographie	43
2.2.5 Algorithme général de l'EKF-SLAM	44
2.3 Représentation 2D de l'environnement	45
2.3.1 Principales représentations stochastiques de l'environnement	45
2.3.2 Segmentation d'un balayage télémétrique	47
2.3.3 Extraction et paramétrisation des droites	49

2.3.4	Extraction et paramétrisation des cercles	52
2.3.5	Application de la méthode de segmentation sur des données réelles . . .	53
2.4	Association de données	53
2.4.1	Etat de l'art des méthodes existantes	55
2.4.2	Mise en correspondance des amers et méthode des plus proches voisins	58
2.4.3	Méthode des plus proches voisins	61
2.4.4	Algorithme de « Joint Compatibility Branch and Bound »	62
2.4.5	Comparaison des algorithmes PPV et JCBB	65
2.4.6	Gestion de la carte	66
2.5	SLAM avec cartes hétérogènes : résultats expérimentaux	68
2.5.1	Environnement intérieur : hall d'aéroport	69
2.5.2	Environnement extérieur : Parc du Cemagref	71
2.6	Conclusion	73
3	Etude de la convergence et la consistance des estimations du filtre polynomial de Kalman	75
3.1	Introduction	75
3.2	Consistance et convergence des estimations de l'EKF-SLAM	76
3.2.1	Définition de la consistance	76
3.2.2	Travaux existants	80
3.2.3	Etude des limites de convergence de l'EKF-SLAM	81
3.2.4	Conclusion sur les limites de convergence de l'EKF-SLAM	90
3.3	Application du filtre polynomial de Kalman dans un contexte SLAM : PEKF-SLAM	91
3.3.1	Présentation du filtre polynomial de Kalman	91
3.3.2	Présentation du filtre PEKF-SLAM	95
3.4	Simulations : évaluation des propriétés du filtre PEKF-SLAM	101
3.4.1	Comportement des filtres sur des cas d'études simples	101
3.4.2	Comportement non-linéaire : incertitude sur l'orientation importante . .	103
3.4.3	Analyses et études des corrélations	105
3.4.4	Analyse de la fermeture de la trajectoire du véhicule	107
3.4.5	Trajectoire du véhicule sans bouclage	108
3.4.6	Conclusion sur les résultats de simulation	110
3.5	Expérimentations	111
3.6	Conclusion	112
4	Méthode SLAM contrainte par une carte connue de l'environnement	115
4.1	Introduction	115
4.2	Observabilité d'un système	116
4.2.1	Définition de l'observabilité	116
4.2.2	Observabilité d'un système linéaire	116
4.2.3	Observabilité d'un système non-linéaire	116
4.3	Observabilité d'une approche SLAM	118
4.3.1	Observabilité d'un amer point	118

4.3.2	Observabilité d'un amer droite	121
4.3.3	Conclusion et travaux existants	123
4.4	Approche SLAM contrainte par une carte connue	126
4.4.1	Présentation de l'approche	126
4.4.2	Carte connue <i>a priori</i> de l'environnement	127
4.4.3	Variables et équations du système SLAM-contraint	129
4.5	Observabilité de l'approche SLAM-contraint	131
4.5.1	Première étude : 1 unique amer	131
4.5.2	Deuxième étude : 3 amers présents sur la carte connue	134
4.5.3	Conclusions	136
4.6	Méthode SLAM-contraint dans une approche multi-cartes	137
4.6.1	Présentation de l'approche	137
4.6.2	Initialisation du vecteur des cartes locales	138
4.7	Résultats de simulation	139
4.7.1	Séries de simulations simples	140
4.7.2	Simulation sur une trajectoire bouclée	142
4.8	Résultats expérimentaux	144
4.9	Conclusions	146
5	Conclusions et perspectives	149
5.1	Bilan général	149
5.2	Perpectives	152
A	Loi gaussienne et description du filtre de Kalman	157
A.0.1	Loi gaussienne à une dimension	157
A.0.2	Loi gaussienne à n dimensions	158
A.1	Estimateurs probabilistes	158
A.1.1	Pourquoi utiliser un estimateur ?	158
A.1.2	Méthodes d'estimation d'une variable aléatoire	159
A.1.3	Filtre de Kalman linéaire : phase d'estimation	160
A.1.4	Filtre de Kalman linéaire : phase de prédiction	162
A.1.5	Filtre de Kalman non-linéaire	163
B	Modélisation mathématique du véhicule et des observations	165
B.1	Modélisation du déplacement du véhicule	165
B.1.1	Système d'équations	165
B.1.2	Vecteur de commande avec l'angle de braquage des roues avant	166
B.1.3	Vecteur de commande uniquement composé des vitesses des roues arrières	167
B.2	Opération de composition entre deux entités géométriques dans un plan 2D	168
B.2.1	Opérateur « \oplus »	169
B.2.2	Opérateur « \ominus »	169

C	Description des méthodes d'extraction des amers ligne et cercle	171
C.1	Introduction	171
C.2	Paramétrisation d'une droite	171
C.2.1	Algorithme de découpage récursif	172
C.2.2	Estimation des paramètres de la droite	173
C.2.3	Fusion des droites identiques	174
C.2.4	Gestion des extrémités des segments	175
C.2.5	Discussions sur la méthode de détection des droites	175
C.3	Paramétrisation d'un amer cercle	175
C.3.1	Méthode de Guivant	176
C.3.2	Méthode de Gauss-Newton : Formulation du critère à minimiser	178
C.3.3	Algorithme de Levenberg-Marquardt	179
C.3.4	Comparaison des trois méthodes	180
D	Description des méthodes d'analyse de la consistance des estimations et formules du filtre PEKF-SLAM	183
D.1	Outils pour la description de la consistance des estimées	183
D.1.1	Limite de convergence pour un amer de type droite	183
D.1.2	Robot stationnaire observant un amer	187
D.1.3	Robot mobile observant un amer	188
D.2	Formulation du filtre PEKF-SLAM	189
D.2.1	Modélisation du déplacement du véhicule	189
D.2.2	Equations d'évolution des états du système	191
D.2.3	Equations d'observation	194
D.2.4	Equations de cartographie d'un nouvel amer	196
D.3	PEKF-SLAM : étape d'association de données	201
E	Simulateurs et contextes expérimentaux	203
E.1	Présentation des simulateurs	203
E.1.1	Simulateur de base	203
E.1.2	Simulateur de cartes hétérogènes	203
E.2	Contexte expérimental	204
E.2.1	Véhicule expérimental : CYCAB	204
E.2.2	Présentation des lieux d'expérimentation	206
F	Méthodes SLAM fondées sur la théorie de Kalman	211
F.1	Introduction	211
F.2	Approche EKF-SLAM <i>Worldcentric</i>	211
F.2.1	Présentation	211
F.2.2	Fonctions non linéaires régissant le système	212
F.3	Approche EKF-SLAM <i>RobotCentric</i>	214
F.3.1	Présentation	214
F.3.2	Fonctions régissant le système	215
F.3.3	Remarque sur cette approche	217

F.4	Filtrage de Kalman Itératif	218
F.5	Filtrage sans parfum dans une application SLAM	219
F.5.1	Principe de la transformation <i>unscented</i>	219
F.5.2	Application de l’UKF dans un contexte SLAM	220
F.6	Approche multi-cartes	221
F.6.1	Présentation de l’approche	221
F.6.2	Systèmes d’état	222
F.6.3	Fonctions régissant la fusion des cartes locales	223
F.6.4	Remarques sur cette approche	224
Bibliographie		225

Table des figures

1.1	Voiture de la série télévisuelle américaine K2000	19
1.2	DARPA GrandChallenge : Véhicule de l'université de Stanford	20
1.3	DARPA Urban Challenge : Véhicule de l'université de Carnegie Mellon	21
1.4	VELAC : Véhicule expérimental du LASMEA	21
1.5	Convoi de véhicules urbains nommés CyCab	22
1.6	Localisation d'un véhicule dans une carte connue de l'environnement	23
1.7	Cartographie d'un environnement à bord d'un véhicule	25
1.8	Représentations de l'environnement	25
1.9	Algorithme général du SLAM	26
2.1	Les différentes étapes de l'algorithme EKF-SLAM	39
2.2	Description des éléments estimés par le filtre EKF-SLAM	41
2.3	Description de l'extraction des amers	47
2.4	Balayage télémétrique	48
2.5	Représentation d'une droite	50
2.6	Associations ambiguës de droites	51
2.7	Représentation d'un cercle	52
2.8	Méthode de segmentation d'un balayage télémétrique	54
2.9	Résultats de l'extraction d'amers	55
2.10	Application de la distance de Mahalanobis	59
2.11	Mise en correspondance de deux droites	59
2.12	Mise en correspondance d'une droite et d'un point	61
2.13	Méthode des plus proches voisins	62
2.14	Méthode d'association de données JCBB	63
2.15	Arbre d'interprétation des mises en correspondance d'amers	64
2.16	Résultats de simulation avec la méthode JCBB	65
2.17	Comparaison des méthodes d'association de données : PPV et JCBB	66
2.18	Description de l'élimination d'un amer	69
2.19	Résultats avec gestion de la carte estimée	70
2.20	Résultats de simulation avec la méthode JCBB couplée à une approche de gestion de la carte	70
2.21	Contexte d'expérimentation : aéroport d'Aulnat	71
2.22	Résultats de l'expérimentation dans l'aéroport d'Aulnat	71
2.23	Evolution de l'air de l'ellipse représentant une incertitude sur la position du véhicule	72

2.24	Vue aérienne de l'expérimentation réalisée au Cemagref	72
2.25	Résultats de la manipulation dans un environnement extérieur	73
3.1	Exemples d'application du test de consistance des estimations : vérité de terrain . . .	78
3.2	Exemples d'application du test de consistance des estimations : première série de résultats	79
3.3	Exemples d'application du test de consistance des estimations : deuxième série de résultats	80
3.4	Simulation : véhicule stationnaire observant un amer	84
3.5	Simulation : véhicule stationnaire observant deux amers	86
3.6	Simulation : Véhicule stationnaire observant une droite	88
3.7	Simulation : le véhicule se déplace et observe un unique amer	102
3.8	Simulation : véhicule stationnaire observant un amer avec une incertitude initiale grande	104
3.9	Matrices de corrélation d'un véhicule se déplaçant et observant un seul amer : comparaison approche EKF-SLAM et PEKF-SLAM	106
3.10	Simulation : trajectoire du véhicule bouclée	107
3.11	Résultats de simulation : trajectoire du véhicule non bouclée	108
3.12	Résultats de simulation : trajectoire du véhicule bouclée	109
3.13	Vérité de terrain de l'expérimentation réalisée sur site PAVIN	111
3.14	Résultats expérimentaux des filtres EKF-SLAM et PEKF-SLAM	112
3.15	Résultats expérimentaux des filtres EKF-SLAM et PEKF-SLAM	112
4.1	Exemples de cartes numériques	124
4.2	Description de l'approche SLAM contrainte par une carte connue	126
4.3	Description de l'algorithme de la méthode SLAM-contraint	127
4.4	Exemples de cartes connues d'environnement	127
4.5	Observabilité de l'approche SLAM-contraint	135
4.6	Approche SLAM-contraint multi-cartes	137
4.7	Description de l'algorithme de SLAM-contraint multi-cartes	138
4.8	Scénario des séries de simulation décrivant l'observabilité des systèmes	140
4.9	Observabilité de l'approche SLAM-contraint lorsque la carte est connue parfaitement	141
4.10	Observabilité de l'approche SLAM-contraint lorsque la carte n'est pas connue parfaitement	142
4.11	Simulation d'une trajectoire bouclée avec un environnement hétérogène	142
4.12	Résultats de simulation de l'approche SLAM-contraint	143
4.13	Résultats de simulation de l'approche SLAM-contraint	143
4.14	Présentation de l'expérimentation pour l'approche SLAM-contraint multi-cartes . . .	144
4.15	Résultats expérimentaux de l'approche SLAM-contraint multi-cartes	145
4.16	Résultats expérimentaux de l'approche SLAM-contraint multi-cartes	145
5.1	Présentation approche EKF couplée avec un estimateur de perturbation	154
5.2	Résultats de simulations avec une approche EKF couplé avec un estimateur de perturbation	154

A.1	Distribution gaussienne	157
B.1	Modèle du tricycle	166
B.2	Angle de braquage des roues avant du véhicule	167
C.1	Représentation cartésienne d'une droite	171
C.2	Extraction de droites d'un balayage télémétrique	172
C.3	Fusion de deux droites dans un balayage télémétrique	174
C.4	Représentation d'un cercle	176
C.5	Méthode de Guivant pour déterminer les paramètres d'un cercle	177
C.6	Exemple de fausse détection d'un cercle	180
C.7	Estimation des paramètres d'un cercle : descente du gradient	182
C.8	Comparaison des trois algorithmes de détermination des paramètres d'un cercle	182
E.1	Présentation du premier simulateur employé	204
E.2	Description du deuxième simulateur	205
E.3	Véhicule expérimental : CYCAB	205
E.4	Photographies de l'expérimentation réalisée à l'aéroport d'Aulnat	207
E.5	Plan de l'aéroport	208
E.6	Vue aérienne du parc du Cemagref	208
E.7	Présentation du site d'expérimentations PAVIN	209
F.1	Représentaiton <i>WorldCentric</i> de l'environnement	212
F.2	Représentaiton <i>RobotCentric</i> de l'environnement	214
F.3	Approche SLAM multi-cartes	221
F.4	Mise à jour de la carte globale dans une approche multi-cartes	222

Chapitre 1

Introduction et état de l'art

1.1 Introduction générale

1.1.1 Le véhicule intelligent : au début, une fiction !



Figure 1.1: Voiture de la série télévisuelle américaine *K2000*.

Mon enfance fut bercée par la diffusion de séries de science-fiction comportant des véhicules et des machines tous irréalistes et inconcevables à l'époque. L'une de ces séries m'aura plus marqué que les autres. Il s'agissait de la série télévisuelle *K2000*¹. Le héros de cette série était une voiture qui se nommait *KITT* (cf. la figure 1.1) et qui était dotée d'une intelligence artificielle et de nombreux gadgets. Elle était capable de réaliser de nombreux exploits, comme par exemple, des bonds de plusieurs mètres, de rouler sur deux roues ou de traverser des murs. Il lui était surtout possible de se localiser, de détecter des obstacles et des personnes dans son environnement et de conduire de manière totalement automatique. Une telle fiction peut-elle devenir une réalité ou rester tout simplement un rêve ?

1. Série télévisuelle américaine tournée entre les années 1982 et 1989 dont le titre original est « *Knight Rider* ».

1.1.2 La réalité rattrape la fiction

En ce début de nouveau millénaire, la réalité tend à rejoindre la fiction. Les exemples les plus probants correspondent aux compétitions de véhicules autonomes lancées par l'organisme central de recherche et de développement du département d'Etat de la défense américaine, DARPA². Ces épreuves ont été développées dans l'espoir que les technologies mises en œuvre soient utiles dans la création de véhicules militaires autonomes. Ces engins permettront de remplacer l'homme dans des environnements dangereux ou inaccessibles.



Figure 1.2: DARPA GrandChallenge 2005 : Véhicule présenté par l'université de Stanford en Californie.

Les deux premiers challenges ont démontré la possibilité de créer des véhicules autonomes capables de naviguer en dehors d'un contexte routier (*offroad*). Lors de la première édition, en 2004, aucun véhicule n'avait effectué l'épreuve dans son intégralité. En revanche, l'année suivante, le *Grand Challenge* fut relevé par plusieurs équipes. Il fut remporté par l'équipe de l'université de Stanford en Californie avec le véhicule présenté sur la figure 1.2, après avoir parcouru 212,4 kilomètres en moins de sept heures dans le désert californien sans aucune intervention humaine.

La dernière compétition eut lieu récemment en novembre 2007. Les véhicules ont dû parcourir une distance de 96 kilomètres dans un environnement urbain dans un délai de 6 heures. La principale difficulté était de respecter les règles de circulation et de prendre en compte les autres véhicules qui circulaient. Ce challenge fut remporté par l'équipe de l'université de Carnegie Mellon (cf. la figure 1.3). Le véhicule équipé de radars, de télémètres laser, de caméras et d'un système GPS³ avec une précision d'un mètre, réalisa l'exploit d'être autonome. Toutes les informations provenant de ces capteurs ont été exploitées par une intelligence artificielle. Ce système a ainsi pu se localiser, détecter les objets mobiles de l'environnement et respecter les lois de circulation routière. Grâce à cette compréhension, il a pu adapter ses commandes pour éviter tous les obstacles, adapter sa conduite à tous les croisements et parcourir la trajectoire GPS fournie par les organisateurs de la course. La réussite d'un tel défi est la preuve que des véhicules autonomes ne sont plus une fiction mais une réalité à venir dans un horizon proche.

2. Defense Advanced Research Projects Agency.

3. Global Positioning System.



Figure 1.3: DARPA Urban Challenge 2007 : Véhicule présenté par l'université de Carnegie Mellon.

1.1.3 Les véhicules intelligents « auvergnats »

Les universités américaines ne sont pas les seules à travailler dans le domaine des véhicules intelligents. En Auvergne, l'université Blaise Pascal de Clermont Ferrand, via le laboratoire LASMEA⁴ qui est une entité du CNRS⁵, réalise des recherches et des actions dans ce domaine. Le groupe GRAVIR⁶ au sein du LASMEA est engagé depuis une vingtaine d'années sur des recherches liées à la perception de l'environnement et à la commande dans le secteur de l'automobile. Il est impliqué dans de nombreux projets nationaux et européens.



Figure 1.4: Véhicule expérimental du LASMEA : VELAC.

Le premier objectif du groupe GRAVIR est de mettre au point des systèmes dits intelligents. A l'origine, le but n'était pas de rendre les véhicules autonomes mais capables d'aider le

4. Laboratoire des Sciences et Matériaux pour l'Electronique, et d'Automatique.

5. Centre National de la Recherche Scientifique.

6. Groupe Automatique en Vision et en Robotique.

conducteur en lui fournissant des informations sur son environnement routier. Entre les années 1987 et 1995, le groupe a notamment participé au programme européen PROMETHEUS ⁷ dans le cadre des projets ProLab1 et ProLab2. Fort de cette expérience, le LASMEA s'est équipé en 1998 d'un véhicule expérimental VELAC ⁸ (cf. la figure 1.4). Ce démonstrateur permet de développer et d'expérimenter des systèmes de détection d'obstacles. Ces détections couplées avec un système d'alerte informent le conducteur lorsqu'un obstacle observé représente un danger immédiat. Ces travaux sont rattachés aux projets ARCOS ⁹ et PAROTO ¹⁰ du programme PREDIT ¹¹.



Figure 1.5: Véhicules urbains CyCab commandés en convoi.

Dans le domaine des véhicules intelligents, le deuxième objectif de GRAVIR est de proposer une alternative de transport urbain propre et totalement autonome. Plusieurs scénarii sont possibles, le but étant de diminuer le trafic dans les centres-villes en développant ces flottes de véhicules en libre service. Par exemple, une personne pourrait utiliser un de ces véhicules pour se déplacer dans le centre puis le laisserait au coin d'une rue. Ensuite, ce véhicule, de manière totalement autonome, retournerait à son point de départ.

De telles approches sont testées sur des petites voitures, électriques équipées de nombreux capteurs embarqués, nommées Cycab (cf. la figure 1.5). Pour prouver la faisabilité et la viabilité d'un tel défi, deux axes de recherche sont menés en parallèle :

- le premier axe correspond à la commande de ces véhicules urbains seuls [153] ou en convoi [18], avec dans ce cas, un véhicule de tête guidé par un capteur CP-GPS ¹² ou par vision [135]. Ces recherches sont menées dans le cadre du programme Mobivip ¹³ ;
- le second axe de recherche correspond à des travaux menés sur la localisation et la cartographie simultanées par un véhicule évoluant dans un milieu inconnu. La finalité est

7. *PRogram for an European Traffic with Highest Efficiency and Unpressed Safety.*

8. *Véhicule Expérimental pour l'Aide à la Conduite.*

9. *Action de Recherche pour une CONduite Sécurisée.*

10. *Projet Anticollision Radar et Optronique pour l'auTOMobile*

11. *Programme de REcherche et D'innovation dans les Transports terrestres.*

12. *Carrier Phase Differential Global Positioning System.*

13. *Véhicules Individuels Publics pour la Mobilité en centre ville.*

d'accroître l'autonomie des robots mobiles. De telles applications permettent aux robots d'utiliser uniquement des informations issues de capteurs embarqués. Mes travaux de recherche se situent dans cette problématique. Ils sont appliqués sur les véhicules expérimentaux de type CyCab. Les lieux d'expérimentation sont des environnements d'intérieur et d'extérieur : le hall de l'aéroport d'Aulnat près de Clermont Ferrand, un parc situé sur le campus universitaire et un site équivalent à un mini centre ville nommé PAVIN.

Sur le long terme, l'objectif ultime est de coupler ces travaux avec les lois de commande implémentées sur ces véhicules urbains. Un tel système permettra de rendre totalement autonome des véhicules, à l'image de ceux présentés lors des challenges DARPA, mais en s'affranchissant des données GPS.

1.2 Présentation de l'étude

Rendre totalement autonome un véhicule correspond au « Saint Graal » de tous les chercheurs travaillant dans le domaine de la robotique mobile. Comme le prouvent les exemples précédents, la compréhension de la scène par ces véhicules est essentielle pour qu'ils accèdent à une autonomie totale. En outre, la localisation du robot est importante pour effectuer une construction incrémentale d'une carte décrivant son environnement, d'où la nécessité de résoudre en simultanément ces deux problèmes largement étudiés en robotique.

1.2.1 Problème de la localisation d'un véhicule

Pour naviguer, un véhicule doit répondre à la question « où suis-je ? ». Dans notre cas d'étude, sa position est déterminée par la comparaison d'informations provenant de capteurs embarqués à bord du véhicule et d'informations connues *a priori* sur l'environnement dans lequel évolue le véhicule.

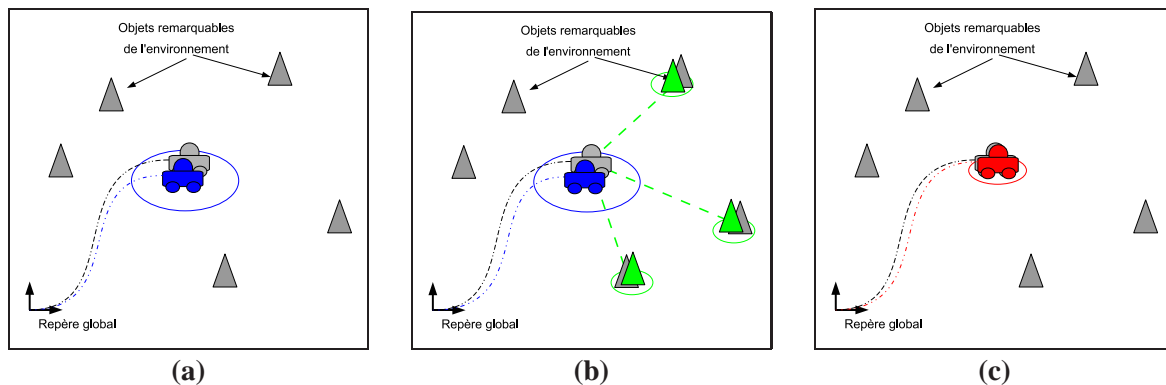


Figure 1.6: Localisation d'un véhicule dans une carte connue de l'environnement : (a) prédiction de la position du véhicule, (b) acquisition de mesures sur l'environnement et (c) estimation de la position.

Le problème de la localisation peut se résoudre par une méthode d'estimation récursive. La figure 1.6 présente les différentes étapes effectuées à chaque itération. Le véhicule, dont la

position réelle est représentée en gris, évolue dans un environnement connu *a priori*, représenté par des triangles de couleur grise. La localisation de celui-ci est réalisée en trois étapes (cf. la figure 1.6) :

- *prédiction de la position (a)* : le déplacement du véhicule est estimé à partir des données issues de capteurs dits proprioceptifs¹⁴. En règle générale, ces données proprioceptives ne sont pas assez précises pour une bonne localisation. L'incertitude sur la position du véhicule est importante (ellipse bleue) ;
- *acquisition de mesures sur l'environnement (b)* : une meilleure localisation exige l'utilisation de données extéroceptives¹⁵ représentées par les tirets de couleur verte. Les données provenant de ces capteurs permettent d'obtenir une carte locale de l'environnement (triangles verts) avec une précision importante (ellipses vertes). La mise en correspondance de cette carte locale et de la carte connue *a priori* (triangles gris) permet d'obtenir une correction de la position du véhicule ;
- *estimation finale de la position (c)* : une nouvelle estimation plus précise de la position du véhicule est obtenue (robot rouge) en réalisant la correction calculée lors de l'étape (b).

Les principales approches mathématiques, employées pour estimer les variables de notre système en combinant des mesures proprioceptives¹⁴ et extéroceptives¹⁵, sont présentées dans l'annexe B.

1.2.2 Problème de la cartographie d'un environnement à bord d'un véhicule

La cartographie est le problème inverse de la localisation (cf. la figure 1.7). Le véhicule se déplace dans un environnement inexploré avec une connaissance parfaite de sa position par rapport à un repère (robot de couleur grise sur la figure 1.7). Le but est d'estimer la position des éléments constituant l'environnement par rapport à ce repère global. La localisation du véhicule, qui est connue et donc très précise, permet de corriger les erreurs des mesures extéroceptives. Cette fusion d'informations est, comme dans le cas de la localisation, réalisée par des estimateurs (cf. l'annexe A).

En outre, la construction d'une carte exige de choisir un type de représentation de l'environnement [149]. Trois types de modèles, référencés dans l'article [39], sont principalement utilisés (cf. la figure 1.8) :

- *le modèle topologique (a)* : graphe qui contient toutes les informations sur la connexité et l'ordre des lieux visités par le véhicule [93] ;
- *la carte ou grille d'occupation (b)* : représentation métrique qui décompose l'espace en un ensemble de cellules. A chacune de ces cellules est associée une probabilité d'occupation estimée à partir des données extéroceptives¹⁵ [59] ;
- *le modèle géométrique (c)* : représentation métrique utilisant des primitives géométriques 2D/3D. Plusieurs types d'amers géométriques peuvent être distingués : des points [53][117],

14. Capteurs embarqués donnant uniquement des informations sur le déplacement du véhicule : centrales inertielles, gyromètres, odomètres, capteurs fournissant l'angle de braquage des roues avant, etc. Une liste précise est donnée dans le document [115].

15. Capteurs embarqués qui fournissent une information sur l'environnement dans lequel évolue le véhicule : télémètre laser, radar, sonar, caméra vidéo, caméra infra-rouge, etc.

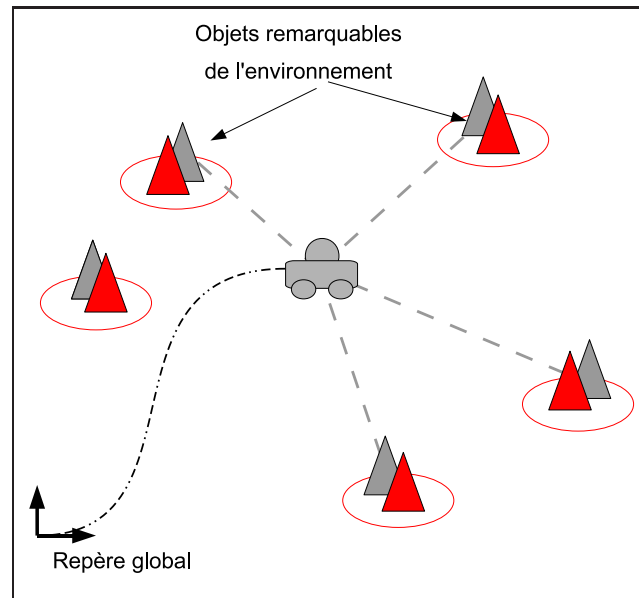


Figure 1.7: Cartographie d'un environnement à bord d'un véhicule : carte estimée de l'environnement (triangles rouges), positions réelles du véhicule et des amers (éléments gris) et mesures extéroceptives (tirets gris).

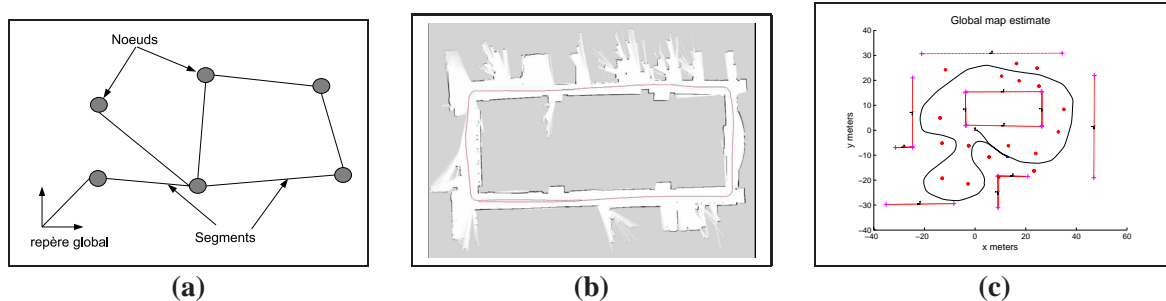


Figure 1.8: Représentations de l'environnement : (a) modèle topologique, (b) carte ou grille d'occupation et (c) carte métrique avec des amers de type droite et point.

des coins [108], des segments représentés en règle générale par leurs droites avec des informations sur la position géométrique de leurs extrémités [27][44][120][130], des cercles [78][165], des polygones, des portions de courbes ou arcs de cercle.

1.2.3 Résolution simultanée des problèmes de localisation et de cartographie à bord d'un véhicule

La résolution simultanée des deux problèmes précédents est appelée SLAM¹⁶ ou CML¹⁷. Ce terme fut introduit pour la première fois dans la littérature par Durrant-Whyte et al. [100].

16. *Simultaneous Localization And Mapping.*

17. *Concurrent Mapping and Localization.*

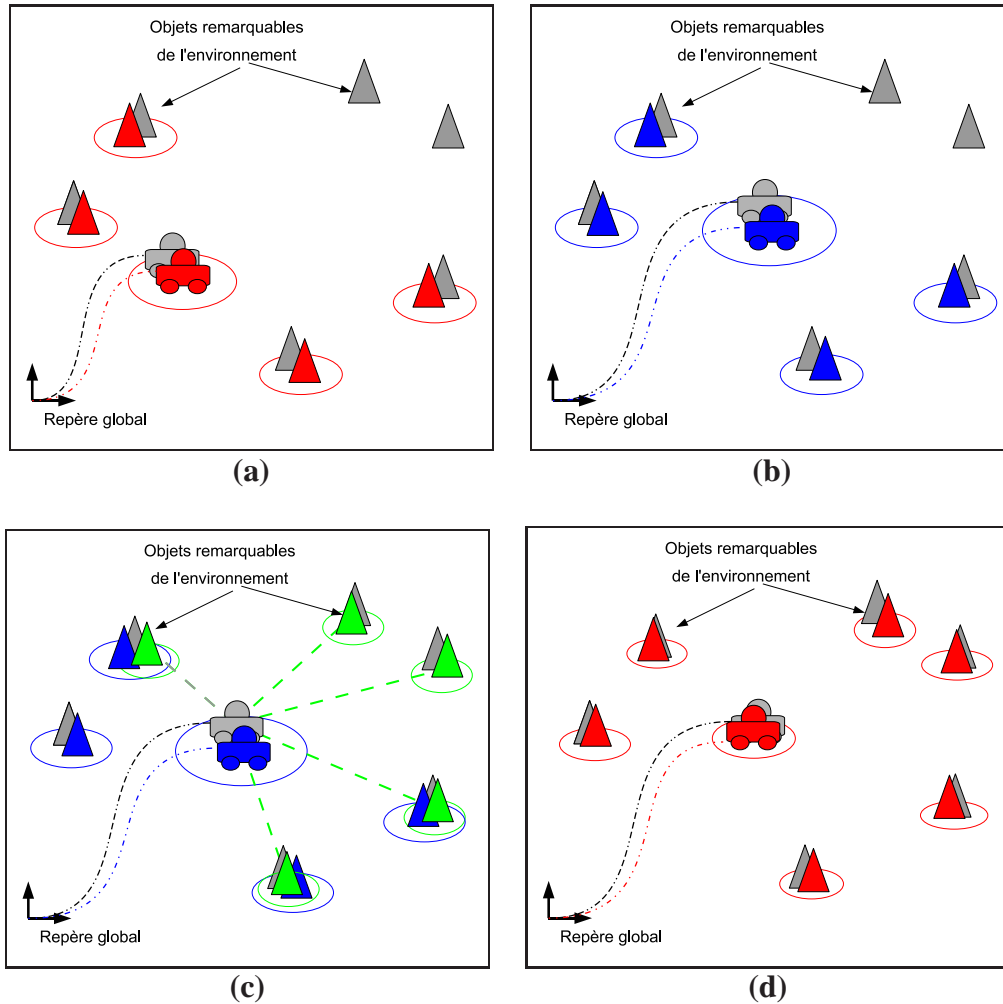


Figure 1.9: *Algorithme général du SLAM : (a) phase initiale, (b) phase prédictive, (c) phase d'acquisition, (d) phase d'estimation et de cartographie.*

Le problème SLAM peut se définir comme une question : « *Est-il possible pour un robot mobile placé à une position inconnue dans un environnement inexploré de construire une carte cohérente de cet environnement et simultanément déterminer sa position dans cette carte ?* »

La solution à ce problème implique la réalisation itérative de quatre étapes distinctes (cf. la figure 1.9) :

- *phase initiale (a)* : le système possède une estimation de sa position (robot de couleur rouge) dans une carte reconstruite de l'environnement (triangles rouges) à l'instant $(k-1)$. Tous ces éléments sont évalués avec une incertitude (ellipses rouges) due aux bruits sur les mesures proprioceptives¹⁴ et extéroceptives¹⁵. Leur position réelle est représentée en gris ;
- *phase prédictive (b)* : le véhicule se déplace. Il prédit sa nouvelle position ainsi que celles des amers en fonction des données fournies par ses capteurs proprioceptifs (éléments bleus). Comme ces mesures sont entachées de bruit, l'incertitude sur la position du véhicule augmente (ellipses bleues) ;

- *phase d'acquisition et d'association (c)* : une carte locale de l'environnement (éléments verts) est construite à partir des données extéroceptives. Après la mise en correspondance de cette carte mesurée à l'instant k et de la carte prédite (éléments bleus), le système calcule la valeur de la correction à apporter aux estimations. Il détermine également les amers mesurés non présents sur la carte estimée ;
- *phase d'estimation et de cartographie (d)* : la correction précédente est intégrée par le système afin de fournir une estimation plus précise de la position du véhicule et des amers. Cette mise à jour permet une réduction de l'incertitude sur les positions estimées des éléments de la carte. Les nouveaux éléments détectés lors de l'étape (c) sont ensuite ajoutés dans la carte estimée.

Depuis la définition de cet algorithme, de nombreuses méthodes ont été proposées [9] [58] [67] [149]. L'état de l'art présenté dans la suite de cette section n'est pas exhaustif mais répertorie les principales solutions proposées dans la littérature. Ces solutions sont classées en fonction du type de représentation de l'environnement employée. En effet, quelle que soit l'approche utilisée, résoudre le problème SLAM repose en premier lieu sur le choix de la représentation de l'environnement puis sur la définition d'une méthode d'estimation.

Approche SLAM avec une représentation topologique

Cette représentation est un graphe dont les nœuds correspondent à des acquisitions extéroceptives brutes, et les arcs à des informations sur le déplacement du véhicule entre deux acquisitions extéroceptives [42]. Elle possède l'avantage d'être économe en espace mémoire pour la sauvegarde de la carte. En outre, les données conservées sont brutes. Aucune approximation géométrique n'est réalisée. En revanche, cette représentation ne contient aucune information géométrique sur la position des mesures ou sur leurs incertitudes. La fermeture de la trajectoire du véhicule ne permet pas à la carte de converger. Les erreurs accumulées durant le parcours de la trajectoire du véhicule ne sont pas corrigées lorsque le système revoit des amers précédemment cartographiés.

Pour résoudre ce problème, cette fermeture de la trajectoire peut être détectée soit dans un processus annexe [15][137], soit en ajoutant des notions géométriques. Ces dernières méthodes, dites hybrides, représentent la carte par une approche métrique couplée à une approche topologique [94][154]. Par exemple, dans une approche multi-cartes, chaque nœud représente une carte locale et les arcs correspondent aux translations géométriques entre deux repères des sous-cartes considérées [21][60]. Ces représentations hybrides permettent l'emploi de données provenant de capteurs différents. Cole et al. [43] fusionne des informations provenant d'un télémètre laser et d'une caméra. Les données télémétriques sont traitées pour obtenir une carte métrique de l'environnement et les données visuelles sont employées pour détecter les endroits revisités par le véhicule. Une telle approche permet de cumuler les avantages des deux types de représentation. Cependant, rien ne garantit la détection des lieux déjà visités et donc la convergence de l'estimation.

Approche SLAM avec une grille d'occupation

Les grilles d'occupation correspondent à une représentation discrétisée de l'environnement. La carte est divisée en plusieurs cellules [59][139]. A l'origine, les cellules contenaient une information de présence ou non d'un objet (information de type binaire). Avec un tel système, la détection de la fermeture de la trajectoire du véhicule était impossible et la carte construite divergeait.

Pour permettre la convergence de la carte, l'information de présence, couplée à chaque cellule, fut représentée par une variable aléatoire distribuée selon une loi gaussienne [75][140]. Cette représentation fut également couplée à une estimation de probabilité maximale nommée EM¹⁸. Cet algorithme statistique recherche la carte la plus probable dans l'espace de toutes les cartes possibles [150]. Cette solution donne des résultats très précis en particulier pour le problème de correspondance des cartes [148].

Ce type de méthode est robuste, facile à implémenter et adaptable à des environnements dynamiques [149]. En revanche, une représentation précise de l'environnement impose des cellules avec des petites dimensions et donc une quantité de données très importante, en particulier, lorsque le véhicule parcourt de longs trajets.

Approche SLAM orientée trajectoire du véhicule

Cette approche est une variante de la représentation topologique. Le problème de SLAM est présenté comme l'estimation des différentes positions du véhicule le long de sa trajectoire. Les premiers travaux, introduits par Lu et Milios [106], posent le problème comme une minimisation des erreurs sur les positions du véhicule. A chacune des positions du véhicule est attachée une mesure extéroceptive. Dans cette approche, la carte de l'environnement n'est pas estimée mais correspond à la superposition de l'ensemble des balayages télémétriques. Ces méthodes, d'optimisation non-linéaire, sont très précises. En contrepartie, toutes les mesures issues des capteurs sont sauvegardées. Plus l'environnement est vaste, plus la quantité de données à traiter est conséquente, et plus les temps de calcul deviennent importants. Pour diminuer ces temps de calcul, deux nouvelles approches ont été proposées dans la littérature : des représentations graphiques et hiérarchisées des données telles que *Graphical SLAM* [63], *TreeMap* [66] et *multi-level relaxation* [68] ou des estimateurs des moindres carrés employant des méthodes de factorisation de matrices telles que l'approche SAM¹⁹ [49], reposant sur une méthode de Choleski. Pour appliquer ces méthodes, une carte de l'environnement est estimée en plus de la trajectoire du véhicule. Elle permet par exemple de mettre en correspondance les différentes cartes de l'approche *TreeMap*. Deux types de représentation de l'environnement sont employées dans ces approches : celle orientée trajectoire du véhicule et celle reposant sur une carte d'amers présentée dans le paragraphe suivant.

18. *Expectation and Maximization* est un algorithme composé de deux étapes : calcul des positions possibles du véhicule puis calcul de la carte la plus probable en fonction des prédictions précédentes.

19. *Smoothing and Mapping*, en anglais, est une méthode reposant sur un algorithme de Levenberg-Marquardt. Elle existe dans une version incrémentale, *iSAM*, [91] et dans une version s'appuyant sur une approche multi-cartes *tectonicSAM* [121].

Approche SLAM avec des cartes d'amers

Cette approche représente l'environnement à l'aide d'entités géométriques ou amers (cf. le paragraphe 1.2.2). Ces amers sont décrits par des variables aléatoires suivant une loi de probabilité, généralement gaussienne. Ce type de représentation est appelée une *carte stochastique* [39][141][142]. Le système estime à la fois la position courante du robot et celles des amers. Le filtre EKF (*Extended Kalman Filter*) [12] fut le premier estimateur présenté comme solution au problème SLAM [100]. La distribution gaussienne, suivie par les variables du système à estimer, est représentée sous une forme continue. Une telle représentation facilite l'implémentation du filtre (cf. les équations présentées dans le prochain chapitre). En outre, elle permet une convergence de toutes les estimations dès lors que le véhicule boucle sa trajectoire. Ceci est rendu possible par la présence des corrélations entre les éléments de la carte définies dans la matrice de variance-covariance [31].

Thrun et al. proposent une application du filtre dual de Kalman [12] nommé SEIF²⁰ [151]. Les probabilités sont écrites sous une forme canonique²¹. Dans ce type de représentation, de nombreux éléments dans la matrice informative (dans les parties triangles inférieures et supérieures) ont des valeurs très faibles. Ces éléments sont négligés et ne sont pas mis à jour. Ceci permet un gain de temps de calcul. Ce filtre est deux fois plus rapide que l'EKF et requiert quatre fois moins d'espace mémoire. Cependant, ses estimations sont moins précises en comparaison avec l'EKF [103] et souffrent d'inconsistance [61]. L'ajout de l'estimation des positions du véhicule le long de sa trajectoire améliore la précision et la consistance des résultats mais augmente de manière considérable le temps de calcul [62]. Cet ajout en fait une approche couplant à la fois une approche SLAM orientée trajectoire du véhicule et une approche carte d'amers. Cette méthode est donc semblable aux approches *Graphical SLAM* [63], *TreeMap* [66] et SAM [91].

Les filtres de Kalman et Informatif ont comme point commun de représenter que les bruits sur les mesures sont supposés suivre une loi gaussienne. Si ce n'est pas le cas, les estimations de ces filtres tendent à diverger (cf. l'annexe A). Pour répondre à ce problème, une méthode de filtrage à particules a été proposée dans la littérature. Dans cette approche, l'incertitude sur les estimations de la carte et des amers est représentée par des distributions discrétisées (particules pondérées) [56] et non gaussiennes. En effet, ce type de filtre correspond à des méthodes séquentielles de *Monte Carlo* [57]. Cette méthode est particulièrement robuste aux bruits non gaussiens mais souffre d'un problème d'épuisement des particules [75][143]. De surcroît, le nombre de particules et donc le temps de calcul deviennent rapidement trop grands lorsque le véhicule explore un environnement vaste. Une version du filtre à particules, *FastSLAM* [152], reposant sur une factorisation suivant le théorème de Rao-Blackwell [26], permet de diminuer le nombre de particules et d'alléger les temps de calcul. Seule la position estimée du véhicule est représentée par des particules et à chaque particule, correspond une carte d'amers supposés indépendants entre eux (chaque amer est représenté individuellement par une gaussienne). Au début, l'inexistence de corrélations entre les éléments des cartes rendait difficile la fermeture

20. *Sparse Extended Information Filter* en anglais.

21. La distribution de probabilité est représentée par un vecteur informatif et une matrice d'information. Le premier moment correspond au vecteur d'état du système pondéré par l'inverse de la matrice de covariance. Le second moment est égal à l'inverse de la matrice de covariance.

de la trajectoire du véhicule. Une nouvelle version de l'algorithme *FastSLAM2* utilisant une fonction d'importance optimale [55] a été réalisée pour permettre une convergence de la carte plus précise [16] [80] [113] [114] [144].

1.2.4 Approche SLAM retenue

Chaque représentation citée possède ses propres avantages et ses propres inconvénients. Chaque choix peut être contesté mais également défendu. Cependant, dans les approches SLAM représentant l'environnement soit par une approche topologique soit par une grille d'occupation, un post-traitement des données doit être réalisé pour obtenir une carte globale de l'environnement visité avec des formes géométriques. Pour implémenter notre solution de SLAM contraint par une carte connue (cf. le chapitre 4), une représentation stochastique de l'environnement avec des amers est retenue. En effet, dans notre approche, elle permet d'obtenir la position des amers de l'environnement simplement et de faciliter la mise en correspondance de la carte estimée et de la carte connue. Ensuite, pour estimer la position du véhicule et des amers, un filtrage de Kalman non-linéaire a été choisi. Cette approche possède les avantages suivants en comparaison aux autres techniques d'estimation :

- les distributions gaussiennes des variables sont représentées avec une moyenne et une matrice de covariance (cf. l'annexe A), à la différence du filtre Informatif où des calculs supplémentaires doivent être réalisés pour obtenir ces deux variables. Grâce à cette représentation, les mises en correspondance de ces variables avec les mesures recueillies sont réalisables facilement sous la condition que les estimations soient correctes ;
- son application est incrémentale. Seule la position courante du véhicule est estimée. Une fusion des données extéroceptives et proprioceptives est réalisée à chaque itération ;
- la fermeture de la trajectoire du véhicule est implicite. La représentation des probabilités, via la matrice de covariance, permet de maintenir les corrélations entre les différents amers de la carte. Ce phénomène permet une convergence et une correction de toute la carte lors de la revisite d'une partie de l'environnement déjà cartographiée [31].

Cependant, comme décrit dans les articles [58] et [67], l'emploi du filtre EKF-SLAM pose les problèmes suivants :

- dans l'approche classique, la complexité des calculs est égal à $O(n^2)$ pour chaque itération du filtre. Lorsque l'environnement visité est vaste, l'application en temps réel de l'approche est difficile ;
- le bruit sur les données est supposé gaussien. Dès lors que les mesures sont biaisées, les estimations obtenues par le filtre de Kalman divergent ;
- le filtre de Kalman est un filtre linéaire. Son application à des systèmes non-linéaires (comme dans un contexte SLAM) implique l'emploi de la méthode de linéarisation de Taylor. Une telle utilisation provoque des problèmes de consistance des estimations et de l'incertitude sur celles-ci [83] ;
- le **système SLAM, tel qu'il est modélisé, n'est pas observable**. En d'autres termes, même si la non-linéarité des équations du système pouvait être négligée, la convergence des estimations ne pourrait être garantie mathématiquement. Ce problème a été mis en évidence récemment par Lee et al. [99]. Il a été démontré que le système SLAM ne peut

être observable que s'il possède des informations absolues sur la position du véhicule ou sur l'environnement cartographié.

Comme le prouvent les méthodes présentées dans le paragraphe 1.3, le problème lié au temps de calcul a été largement étudié par la communauté scientifique ces dix dernières années au détriment des problèmes de consistance des estimations et d'observabilité du système. Pourtant, garantir l'observabilité et la cohérence des estimations est une étape essentielle dans la résolution du problème SLAM. Dans ce but, mes travaux proposent une étude approfondie de chacun de ces deux problèmes (cf. les chapitres 3 et 4). Ces études nous ont permis de fournir une approche SLAM permettant de répondre au problème d'observabilité tout en conservant des estimations consistantes. L'idée de notre méthode repose sur la contrainte des estimations d'une approche classique EKF-SLAM grâce à la connaissance d'une carte *a priori* de l'environnement.

Le prochain paragraphe présente un état de l'art des principaux travaux sur l'EKF-SLAM. Il met en relief les avantages et les inconvénients cités précédemment. Cette énumération permet également de justifier les études et l'approche finale présentées dans ce manuscrit.

1.3 Résoudre le problème du SLAM avec un filtre EKF

Les premiers travaux sur l'EKF-SLAM ont eu pour but d'améliorer la précision des estimations et de résoudre les problèmes d'application en temps réel. La première partie de ce paragraphe en présente les principales variantes en détaillant l'intérêt de chacune. Ces approches reposent sur des représentations différentes de la carte mais également sur une réécriture des équations du filtre de Kalman. Cependant, comme vous le constaterez à travers cet état de l'art, peu de travaux ont abordé les points suivants :

- la carte estimée de l'environnement est généralement constituée de points. Cependant, elle devrait être composée de plusieurs types géométriques d'amers pour apporter plus d'informations au système mais également à un opérateur éventuel ;
- la convergence des estimations ainsi que leurs consistances ont été peu étudiées depuis la formulation du problème SLAM ;
- le système, tel qu'il est abordé dans la plupart des articles, n'est pas observable. En d'autres termes, la convergence des estimations d'une approche EKF-SLAM, mais également des autres estimateurs employés pour résoudre ce problème, ne peut être démontrée théoriquement.

Dans la deuxième partie de cette section, ces différents problèmes sont décrits à travers un état de l'art. Les conclusions de chaque partie permettront d'introduire, mais également de justifier, les différentes études menées dans ce manuscrit.

1.3.1 Approches EKF-SLAM existantes

Les premiers travaux entrepris sur l'EKF-SLAM avaient pour objectif de réduire les temps de calcul en essayant de conserver la même précision sur les estimations du système. A chaque

itération du filtre, la complexité des calculs est équivalente à $O(n^2)$ où n est le nombre d'éléments de la carte. Pour diminuer cette valeur, différentes pistes ont été visitées : mise à jour des variables différée, mise à jour partielle de la matrice de covariance, représentations différentes de la carte. La suite de ce paragraphe présente les différentes méthodes proposées dans la littérature.

Compressed Extended Kalman Filter :

Dans le but de diminuer les temps de calcul, l'approche CEKF [78] réalise une mise à jour partielle du vecteur d'état et de la matrice de covariance. Seules les positions des amers à proximité de la position du véhicule sont estimées. La mise à jour globale de la carte est retardée. Cette approche permet de diminuer²² la charge de calcul à chaque itération du filtre et donc de pouvoir traiter les données en temps réel. Davison [48] propose une méthode similaire dans laquelle toutes les estimations du système ne sont pas mises à jour à chaque itération du filtre mais lors d'un post-traitement dans un processus annexe.

Covariance Intersection :

Julier et Uhlman proposent une solution alternative nommée CI²³ [87]. Cette méthode réalise une mise à jour partielle de la matrice de covariance. Les corrélations entre les amers de la carte et du véhicule ne sont pas toutes modifiées à chaque itération du filtre. Le problème est simplifié à $(n + 1)$ filtres indépendants. La méthode permet une réduction des temps de calcul et de l'espace de stockage nécessaire pour les variables du système. Cependant, cette solution n'est pas optimale. Des approximations importantes sont réalisées sur les estimations de la carte et de la position du véhicule.

Relative Map filter :

Ce filtre repose sur une représentation relative²⁴ de la carte. Dans cette approche, le vecteur d'état définit les distances entre les amers et non leurs positions. Cette représentation permet de réduire le nombre d'état du système et donc de gagner du temps de calcul lors de la mise à jour des états. Le problème avec ce type de représentation est qu'il ne donne aucune information sur la position des amers dans un repère commun. La carte estimée a tendance à se déformer, voire à diverger. Pour rendre cette approche plus consistante, certains chercheurs ont contraint la position des amers en repassant dans une représentation absolue de la carte [2][119]. Cette représentation a été couplée avec une approche multi-cartes [160] et une approche découplant la cartographie et la localisation [158].

22. La mise à jour partielle requiert une charge de calcul de l'ordre de $O(n_a^2)$ avec n_a le nombre d'amers à proximité du véhicule. La mise à jour globale de la carte est égale à $O(n^2)$.

23. Le premier algorithme *Covariance Intersection* [156] n'estimait aucune corrélation, pour accroître la précision des estimations, la nouvelle version *Split Covariance Intersection* [90] calcule les corrélations entre la position du véhicule et les amers. Cette méthode est plus précise mais requiert plus de temps de calcul en comparaison à l'ancienne.

24. Relations géométriques entre les amers (distance entre deux amers de types point par exemple).

Local map filters :

La carte globale de l'environnement est divisée en sous-cartes locales mises à jour individuellement. Les temps de calcul sont alors fonction du nombre n_l d'amers constituant la carte locale. Deux représentations de la carte globale sont possibles. Une première approche représente le niveau global par un graphe reliant plusieurs cartes locales [21][41][]. Cette représentation peut être définie comme topologique couplée avec une représentation métrique. La seconde approche correspond à une fusion de cartes locales, supposées indépendantes, dans une carte globale [132][146][160]. Cette méthode a pour avantage de réduire considérablement les temps de calcul mais également d'utiliser un filtre de Kalman optimal où toutes les corrélations sont mises à jour à chaque itération. De plus, la méthode a été largement étudiée pour déterminer la taille optimale des cartes locales [125] et une stratégie de fusion des cartes locales sur plusieurs "niveaux" hiérarchiques [124] permettant de diminuer les temps de calcul. Récemment, des travaux [128] ont prouvé que les cartes locales ne doivent pas être obligatoirement indépendantes d'un point de vue statistique. Cette nouvelle technique autorise une information partagée entre deux cartes corrélées. En d'autres termes, le suivi d'un objet mobile d'une carte locale vers la suivante devient donc possible.

Synthèse :

Premièrement, pour réduire le temps de calcul de l'approche EKF-SLAM, la solution la plus pertinente correspond à une application multi-cartes. Cette solution reste optimale tout en limitant le nombre d'amers à mettre à jour à chaque itération du filtre. Les approches comme le CEKF, SEIF et la méthode d'intersection de covariance sont toutes des solutions sous-optimales. En effet, pour diminuer les temps de calcul et l'espace mémoire requis, des approximations sont réalisées qui ont pour conséquence d'augmenter les erreurs commises sur les estimations. La méthode retenue, pour réduire les temps de calcul d'une application EKF-SLAM, est l'approche de cartes locales car c'est une solution optimale, aucune approximation n'est réalisée.

Deuxièmement, ces premiers travaux fondés sur l'EKF-SLAM s'intéressent uniquement à la réduction du temps de calcul, soit en ne mettant à jour qu'une partie des variables du système, soit en découpant la carte globale en plusieurs parties. Dans la littérature, les chercheurs n'ont commencé à répondre aux problèmes de consistance et d'observabilité que depuis quelques années comme le souligne le paragraphe suivant.

1.3.2 Problèmes observés avec une application EKF-SLAM

Dans cette partie, les quatre problèmes liés à l'application d'un EKF-SLAM et traités dans mes travaux sont présentés : la représentation de l'environnement en 2D, le bruit sur les mesures, la convergence des estimations et l'observabilité de l'approche SLAM.

Représentation de l'environnement :

Avec l'utilisation d'un filtre de Kalman, la représentation de l'environnement repose sur une carte stochastique. Cette représentation métrique utilise des primitives en deux dimensions dans notre cas d'étude. La majorité des travaux sur le SLAM utilisant ce filtre emploie des amers de type point [32][40][54][98][119]. La carte estimée est peu représentative de l'environnement exploré par le véhicule. Elle ne contient que des informations de présence ou non d'objets. Aucune description de leur forme géométrique n'est disponible.

Dans l'optique de représenter plus fidèlement des environnements intérieurs, des travaux emploient des amers de type droite [6][29][41][132][162]. Guivant *et al.* [78] utilisent des amers de type cercle pour représenter un environnement d'extérieur constitué essentiellement d'arbres. Peu de travaux reposent sur une représentation hétérogène de l'environnement [44][146][165]. Pourtant, cette approche apporte beaucoup plus d'informations sur l'environnement comme le prouvent les travaux présentés dans l'article [165].

Dans ce manuscrit, il est démontré la possibilité d'appliquer une représentation hétérogène de l'environnement sans altérer la précision des estimations du système. La principale difficulté d'une telle approche est de pouvoir différencier les amers et de pouvoir les associer le cas échéant. Pour résoudre ce problème, une étape d'association de données, permettant de mettre en correspondance deux amers avec des géométries différentes, est présentée.

Bruit sur les mesures supposé gaussien :

Lors de l'application d'un filtre de Kalman, le bruit sur les mesures extéroceptives et proprioceptives doit suivre une loi gaussienne. S'il existe un biais sur les mesures odométriques, l'estimation des paramètres risque de diverger très rapidement. Une calibration précise des capteurs et une détermination précise des variances sur les mesures sont primordiales lors de l'utilisation de ce filtre.

Dans la littérature, il existe des solutions réduisant ce problème. Par exemple, dans une méthode de localisation du véhicule, certains paramètres du véhicule (rayons des roues arrière, distance entre les deux essieux) sont estimés dans le vecteur d'état [23]. Des travaux similaires ont été réalisés dans un contexte SLAM. Dans les travaux de Martinelli *et al.* [110], deux filtres de Kalman sont employés. Le premier permet l'estimation de la position du véhicule et de la carte. Le second réalise un auto-calibrage des paramètres des capteurs odométriques. Une autre solution [40] réalise le couplage d'un réseau de neurones avec un filtre EKF dans une application SLAM.

Une dernière solution consiste à considérer le filtre de Kalman comme un système bouclé avec une correction proportionnelle. Depuis longtemps, il a été démontré qu'une telle correction ne permet pas une convergence du système lorsque les mesures sont biaisées. Pour converger, une action intégrale doit être ajoutée au correcteur. Partant de cette constatation, Kwon *et al.* ont proposé un système robotique de localisation reposant sur un filtre EKF couplé à un estimateur de bruit [95][97]. L'estimateur de bruit de cette approche correspond à l'action intégrale du filtre de Kalman [96]. Mes premières recherches me conduisirent à appliquer cet estimateur dans une approche SLAM. En présence de biais sur les mesures proprioceptives, le système ne divergeait pas sous certaines conditions [33]. Cependant, cette approche souffrait de problème

de convergence sur des trajectoires importantes.

Convergence et cohérence de l'EKF-SLAM :

Les estimations des positions des amers et du véhicule ainsi que leurs incertitudes associées doivent rester consistantes lors de la fermeture de trajectoire. La construction d'une carte dans une boucle fermée exige de corriger les erreurs de mesures qui s'accumulent, sous peine d'obtenir une carte incohérente. La reconnaissance des amers vus à la position de départ doit permettre une correction de cette accumulation d'erreurs et une convergence cohérente des estimations du système d'états (position du véhicule et carte de l'environnement)

La convergence des estimations d'un filtre de Kalman linéaire dans un contexte SLAM a été démontrée par Dissanayake [53] et Csorba [47]. Cependant, la majorité des applications SLAM est régie par des systèmes d'équations non-linéaires. Les théorèmes de convergence d'un système SLAM proposés dans l'article [53] ne sont plus valables comme le montrent les travaux de l'article [83]. En outre, l'emploi de méthodes de linéarisation telle que la méthode de Taylor entraîne des problèmes de consistance des estimations du filtre de Kalman [11] [30] [84] [89] [109] [133]. La linéarisation est réalisée par rapport à la valeur estimée des états et non la valeur réelle. Si l'erreur sur les estimations est importante, la linéarisation des équations ne sera pas viable [84] [89]. Les estimations du filtre deviendront incohérentes et la reconstruction de l'environnement divergera rapidement.

Depuis cinq ans, des méthodes employant un filtre de Kalman, sont proposées pour atténuer les problèmes de consistance des estimations. La méthode *RobotCentric* [30] propose un changement de référencement des états estimés dans le vecteur d'état du système. Tous les éléments de la carte sont définis par rapport à la position du véhicule. Ce changement de représentation réduit les erreurs introduites lors de la linéarisation des fonctions. Une autre solution correspond à l'application du filtre UKF²⁵ dans un contexte SLAM [4][111]. Ce filtre [88], introduit par Julier et al., propose une alternative à la méthode de linéarisation de Taylor. En effet, il approxime une distribution gaussienne par un ensemble de particules. Il repose sur le principe qu'il est plus simple d'approximer une distribution gaussienne avec un nombre de paramètres fixés que d'approximer arbitrairement une fonction non-linéaire. La dernière solution permettant d'atténuer les problèmes posés par la non-linéarité des équations correspond à une approche multi-cartes. L'environnement est représenté par une séquence de cartes locales [146]. En divisant l'environnement exploré par le véhicule, les erreurs sur les estimations restent faibles et les erreurs commises lors de la linéarisation des fonctions ne sont pas importantes. De nombreuses applications, reposant sur cette approche, soulignent son intérêt pour la consistance des estimations obtenues [112][132].

Mes travaux proposent l'application d'un nouveau filtre de Kalman [71] dans un contexte SLAM [37]. Cette nouvelle approche décrit un système non-linéaire avec des sommes de polynômes. Les premiers résultats de simulations [24] ont prouvé que ce filtre donnait des estimations plus précises que les approches UKF et EKF. Après avoir réalisé une synthèse des problèmes de consistance présentés précédemment, les résultats de cette application seront comparés aux approches classiques de Kalman (EKF et UKF). L'objectif sera d'obtenir une analyse

25. *Unscented Kalman Filter.*

du comportement de ce filtre dans un contexte SLAM et de répondre à la question : Est ce que l'application de ce filtre polynomial améliore les résultats en comparaison au filtre de Kalman classique ?

Observabilité de l'approche SLAM :

Ce critère reste peu traité dans les différentes approches SLAM proposée dans la littérature. Pourtant, cette notion est essentielle d'un point de vue mathématique. En effet, la description de l'observabilité d'un système permet de statuer sur la convergence ou non des estimations d'un système. Les études de l'observabilité de systèmes SLAM linéaire et non-linéaire furent présentés respectivement dans les articles [3] et [99]. Les conclusions de ces articles sont identiques : le système SLAM n'est pas observable. Une des solutions, permettant de rendre observable cette approche, correspond à la connaissance de positions exactes d'amers dans l'environnement. Cependant, il est rare de connaître parfaitement la position d'éléments dans la réalité. Mes travaux de recherche prouvent qu'un système EKF-SLAM reste observable lorsque la carte connue de l'environnement est imprécise [35]. De plus, l'étude de l'observabilité pose les conditions pour que le système soit observable. Fort de cette démonstration, une approche SLAM-contraint est proposée permettant de rendre le système SLAM localement observable et ainsi, d'améliorer la précision des estimations.

1.4 Description du manuscrit

1.4.1 Objectif et positionnement des travaux

Ce manuscrit propose une étude approfondie des différents problèmes relatifs à l'application EKF-SLAM. De surcroît, pour chaque problème étudié, une solution est proposée et critiquée à l'aide de résultats de simulations et d'expérimentations.

Le premier objectif est de prouver que l'emploi d'amers de géométries différentes, dans une même carte estimée, apporte plus d'informations dans l'interprétation de la carte estimée de l'environnement sans altérer la précision des estimations du filtre.

Le second objectif consiste à faire une synthèse des problèmes de consistance des estimations rencontrés lors de l'utilisation d'un filtre de Kalman classique ou UKF. A travers des scénarii simples de simulations, une étude analytique des raisons de l'inconsistance de l'EKF-SLAM est présentée et couplée à une étude comparative des résultats obtenus avec les filtres UKF et EKF. Après avoir établi les limites théoriques de convergence de l'EKF-SLAM, il est prouvé que l'application d'un filtre polynomial de Kalman permet de restreindre certains problèmes de consistance des estimations [34] [36].

Le troisième objectif est de proposer une solution SLAM permettant de rendre le système EKF-SLAM observable via la connaissance de mesures absolues : une carte connue a priori de l'environnement. De plus, les différentes études de la consistance des estimations permettent, d'une part, d'obtenir les précautions mathématiques nécessaires pour obtenir des estimations consistantes, d'autre part, de quantifier les résultats obtenus avec l'approche proposée. En effet, plusieurs contextes de simulations et d'expérimentations sont proposés pour quantifier l'apport

de données absolues sur la précision et la consistance des estimations. Une application de cette méthode de SLAM est présentée dans l'article [38].

1.4.2 Organisation du manuscrit

Après ce chapitre introduisant le contexte de l'étude réalisée, les autres chapitres traitent de :

Chapitre 2 : Dans ce chapitre, sont introduits les différents outils mathématiques employés dans mes travaux de recherche : les équations non-linéaires régissant le système, les équations du filtre EKF-SLAM et les méthodes d'extraction et de mise en correspondance d'amers avec des géométries différentes. L'ensemble de ces outils permet la représentation de l'environnement avec plusieurs amers de types géométriques différents. Pour conclure, une comparaison entre des représentations homogène et hétérogène de l'environnement est présentée sur des données expérimentales.

Chapitre 3 : Il présente les méthodes permettant de qualifier la consistance des estimations d'un filtre de Kalman. Les problèmes de consistance du filtre de Kalman sont présentés analytiquement puis décrits à l'aide de résultats de simulation. L'objectif est d'obtenir les limites de convergence du système mais également de prouver que les solutions proposées dans la littérature n'ont pas résolu ce problème de consistance. L'application d'un filtrage de Kalman fondé sur la description des équations non-linéaires par des sommes de polynômes, nommée **PEKF-SLAM**, est introduite. Les résultats obtenus avec cette approche sont comparés avec les résultats du chapitre précédent pour permettre de conclure sur l'apport de ce filtre dans une application SLAM.

Chapitre 4 : Une étude analytique de la matrice d'observabilité d'un système EKF-SLAM est présentée. Elle permet d'énoncer les conditions d'observabilité du SLAM. A partir des conclusions de cette étude, une approche est proposée : approche **SLAM-contraint**. Elle correspond à une application SLAM contraint par la connaissance d'une carte connue (complète ou partielle) de l'environnement. Les résultats expérimentaux soulignent l'intérêt d'une telle approche pour répondre au problème d'observabilité mais également au problème de consistance des estimations d'une approche EKF-SLAM.

Conclusion et discussion : Dans cette partie, une conclusion générale permet de mettre en relief les résultats présentés dans le manuscrit. Elle est suivie de perspectives envisageables de ces travaux. Dans un premier temps, sont présentés les problèmes non résolus par les approches proposées ainsi que les méthodes pouvant solutionner ces derniers problèmes. Dans un second temps, les applications possibles de nos approches sont proposées et permettent de conclure sur leur pertinence.

Pour permettre une meilleure compréhension des différents chapitres, six annexes sont fournies à la fin du manuscrit :

Annexe A : Cette annexe se compose de deux parties. Dans un premier temps, de brefs rappels sont faits sur les distributions gaussiennes. Ensuite, est proposé un rappel sur la théorie de Kalman.

Annexe B : Cette annexe présente les équations d'évolution des états du véhicule et les opérations dites de *composition* permettant de définir les équations d'observation du système.

Annexe C : Cette annexe présente les méthodes retenues pour extraire et paramétrer les deux types d'amers utilisés dans cette thèse : les cercles et les droites.

Annexe D : Cette annexe présente les différentes équations et démonstrations utiles, dans la compréhension des problèmes d'inconsistance des estimations mais également dans l'application du filtre PEKF-SLAM.

Annexe E : Ce manuscrit présente des résultats de simulation et d'expérimentation. Dans cette annexe, les deux simulateurs ainsi que les contextes d'expérimentation (véhicule et environnement) sont décrits.

Annexe F : Plusieurs approches ont été employées durant mes travaux de thèse : approche EKF-SLAM *WorldCentric* et *RobotCentric*, l'IEKF, le filtre sans parfum appliqué au SLAM et une approche multi-cartes. Cette annexe décrit les différentes équations (prédiction, observation et cartographie) utilisées pour leur implémentation.

Chapitre 2

EKF-SLAM avec une représentation hétérogène de l'environnement

2.1 Introduction

Ce chapitre présente une approche **EKF-SLAM** couplée à une représentation hétérogène de l'environnement. Dans cette application, la carte estimée est composée d'amers avec des géométries différentes : des cercles et des droites. Comme introduit dans le chapitre précédent, le filtre EKF-SLAM se résume en quatre étapes (cf. la figure 2.1) :

- **Proprioception ou prédiction** : modélisation du déplacement du véhicule et prédiction de sa position à partir des données proprioceptives recueillies (commandes) ;
- **Perception** : traitement des données extéroceptives et extractions des primitives géométriques ;
- **Association de données** : mise en correspondance des mesures et de la carte estimée par le système, cette étape de l'algorithme est incluse dans la phase de perception de l'environnement ;
- **Estimation** : filtrage utilisé pour estimer la carte de l'environnement et la position du robot dans celle-ci.

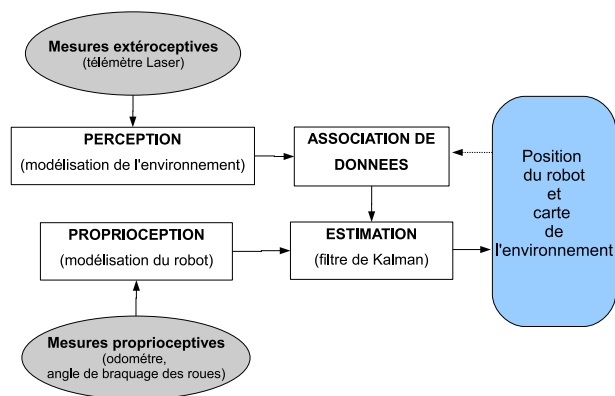


Figure 2.1: Les différentes étapes de l'algorithme EKF-SLAM.

Chacune de ces quatre parties est détaillée dans ce chapitre. En premier, les étapes spécifiques au filtre EKF sont présentées. Elles correspondent à la **prédiction** et à l'**estimation** des états du système. Les équations non-linéaires du système et du filtre estimateur sont introduites. Ensuite, sont présentées les phases de **perception** et d'**association de données**. Elles correspondent à la compréhension de l'environnement par le système. Elles sont primordiales pour l'obtention d'estimations précises. Enfin, pour justifier l'emploi d'un filtre EKF-SLAM couplé à une représentation hétérogène de l'environnement, des résultats sur des données réelles sont proposés dans une dernière partie.

2.2 Algorithme général de l'EKF-SLAM

L'algorithme du filtre de Kalman appliqué dans un contexte de localisation et de cartographie en simultané a été présenté par Dissanayake et al. [53]. Il correspond à l'étape **estimation** (cf. la figure 2.1). Ce filtre est composé de quatre étapes distinctes : la prédiction (proprioception), l'observation, la mise à jour des états et la cartographie des nouveaux amers. Chacune de ces étapes (exceptée l'étape de mise à jour) est régie par des équations non-linéaires.

L'objectif de ce paragraphe est d'introduire les variables modélisant le système mais également les équations décrivant le filtrage EKF-SLAM. Dans le but de ne pas surcharger le document, les différentes équations non-linéaires sont décrites dans l'annexe A.

2.2.1 Espace d'état

L'espace d'état correspond à la représentation d'un système par un ensemble de paramètres également appelés variables d'état. L'ensemble de ces n variables est écrit sous la forme d'un vecteur d'état :

$$\mathbf{X} = [x_1 \cdots x_i \cdots x_n]^T \quad (2.1)$$

Dans une application SLAM, ces paramètres x_i correspondent à des positions en deux dimensions d'un véhicule et d'amers constituant une carte de l'environnement dans lequel évolue le véhicule (cf. la figure 2.2). Le système d'état est défini par le vecteur suivant :

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_M \end{bmatrix} \quad (2.2)$$

Le vecteur \mathbf{X}_V décrit la position du véhicule dans un plan référencé par rapport à un repère W . Il se compose de trois états : (x_v, y_v) la position cartésienne du centre de l'essieu arrière du véhicule et θ_v , l'orientation de celui-ci (cf. la figure 2.2). \mathbf{X}_M se rapporte au vecteur d'état de la carte. Le nombre d'états de ce vecteur dépend du nombre d'amers détectés dans l'environnement et de la géométrie associée à chaque amer. Sur la figure 2.2, sont représentés des amers de type point. Les formes plus complexes liées à d'autres types d'amer seront introduites par la suite.

Pour employer un filtre EKF, le vecteur \mathbf{X} est modélisé par une variable aléatoire suivant une loi gaussienne (cf. l'annexe A). Le système (2.2) est représenté par deux moments définissant la loi normale ou gaussienne :

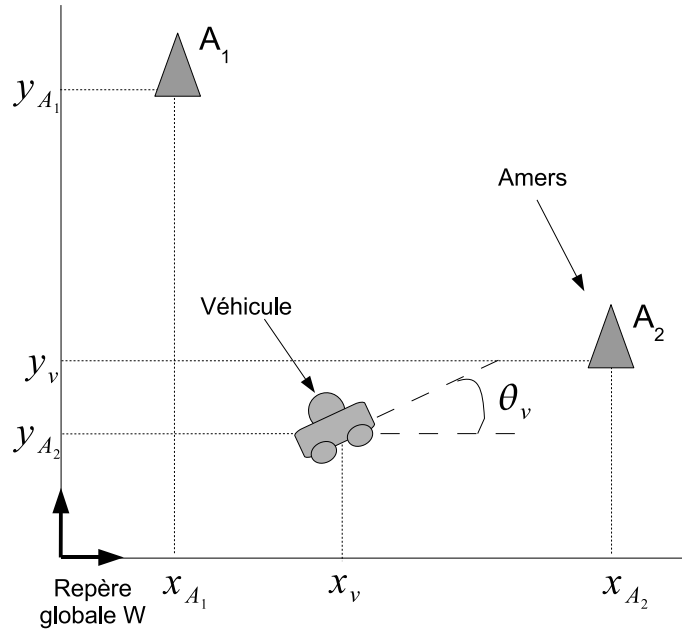


Figure 2.2: Schéma représentant les éléments estimés par le filtre EKF-SLAM : la position des amers de type point constituant la carte et la position du véhicule.

– une moyenne :

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{\mathbf{X}}_V \\ \hat{\mathbf{X}}_M \end{bmatrix} \triangleq \mathbb{E}[\mathbf{X}] \quad (2.3)$$

– une matrice de covariance :

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_V & \mathbf{P}_{VM} \\ \mathbf{P}_{VM}^T & \mathbf{P}_M \end{bmatrix} \triangleq \mathbb{E}[(\mathbf{X} - \hat{\mathbf{X}})(\mathbf{X} - \hat{\mathbf{X}})^T] \quad (2.4)$$

\mathbf{P}_V représente la matrice de covariance associée au vecteur \mathbf{X}_V et \mathbf{P}_M la matrice de covariance associée à \mathbf{X}_M . Les éléments de la matrice \mathbf{P}_{VM} représentent les corrélations entre les éléments de la carte et les états du véhicule. Cette modélisation de la carte et du véhicule est dite *stochastique* et a été introduite initialement dans l'article [141]. Dans la suite de cette section, sont introduits les différentes équations permettant d'initialiser, de faire évoluer et d'observer ces différentes variables.

2.2.2 Transition d'état

Les transitions d'état sont modélisées par la fonction non-linéaire \mathbf{f} . Cette équation d'évolution du système SLAM (le véhicule et la carte) entre deux instants k et $k + 1$ s'écrit :

$$\mathbf{X}(k + 1|k) = \mathbf{f}(\mathbf{X}(k|k), \mathbf{U}(k)) + \mathbf{w}(k) \quad (2.5)$$

$\mathbf{U}(k)$ correspond aux commandes appliquées au véhicule. $\mathbf{w}(k)$ représente le bruit gaussien sur les mesures proprioceptives et sur la modélisation du véhicule. Dans notre application, l'environnement est représenté par des éléments géométriques à 2D et 2D1/2 (le véhicule et les droites sont décrits par une orientation). En outre, ces différents éléments, appelés amers, sont considérés comme statiques. La carte n'évolue donc pas entre deux instants :

$$\mathbf{X}_M(k+1|k) = \mathbf{X}_M(k|k) \quad (2.6)$$

Le seul élément mobile correspond au véhicule. Son évolution est régie par l'équation non-linéaire \mathbf{f}_V :

$$\mathbf{X}_V(k+1|k) = \mathbf{f}_V(\mathbf{X}_V(k|k), \mathbf{U}(k)) + \mathbf{w}_V(k) \quad (2.7)$$

Le terme \mathbf{w}_V représente un bruit gaussien sur le déplacement du véhicule. Dans notre cas d'étude, il englobe deux types d'erreur : les erreurs sur la modélisation du véhicule et les erreurs sur les données proprioceptives \mathbf{U} . L'évolution des états du véhicule, fonction \mathbf{f}_V , correspond aux équations cinématiques dites du tricycle (cf. l'annexe B).

La forme matricielle de l'équation de prédiction des états (2.2) prenant en compte les évolutions du véhicule, \mathbf{f}_V , et l'évolution de la carte (2.6) s'écrit :

$$\mathbf{X}(k+1|k) = \begin{bmatrix} \mathbf{f}_V(\mathbf{X}_V(k|k), \mathbf{u}(k)) + \mathbf{w}_V(k) \\ \mathbf{X}_M(k|k) \end{bmatrix} \quad (2.8)$$

La matrice de covariance associée au vecteur d'état $\mathbf{X}(k+1|k)$ s'écrit :

$$\mathbf{P}(k+1|k) = \mathbf{F}_X \cdot \mathbf{P}(k|k) \cdot (\mathbf{F}_X)^T + \mathbf{F}_U \cdot \mathbf{Q}_U \cdot (\mathbf{F}_U)^T \quad (2.9)$$

où :

- \mathbf{Q}_U est la matrice de covariance associée aux bruits \mathbf{w}_V ;
- \mathbf{F}_X est la matrice jacobienne de la fonction \mathbf{f} par rapport à l'état $\mathbf{X}(k|k)$:

$$\mathbf{F}_X = \begin{bmatrix} \mathbf{F}_{X_V} & \mathbf{0}_{3 \times (n-3)} \\ \mathbf{0}_{(n-3) \times 3} & \mathbf{I}_{(n-3) \times (n-3)} \end{bmatrix} \quad (2.10)$$

n correspond aux nombres d'états du vecteur \mathbf{X}_M . \mathbf{F}_{X_V} représente la jacobienne de la fonction \mathbf{f}_V . Elle dépend de la modélisation du véhicule (cf. l'annexe B).

- \mathbf{F}_U est la matrice jacobienne de la fonction \mathbf{f} par rapport aux commandes \mathbf{U} :

$$\mathbf{F}_U = \begin{bmatrix} \mathbf{F}_{V_U} \\ \mathbf{0}_{(n-3) \times 2} \end{bmatrix} \quad (2.11)$$

où \mathbf{F}_{V_U} correspond à la matrice jacobienne de la fonction \mathbf{f}_V par rapport à la commande \mathbf{U} .

2.2.3 Fonction d'observation

Dans un contexte SLAM, les équations d'observation sont utilisées pour projeter la position estimée des amers dans le repère associé à la position du véhicule. Le but est de comparer, dans

un repère commun, la position des amers de la carte estimée avec les mesures recueillies. La fonction d'observation \mathbf{h} s'écrit :

$$\mathbf{Z}(k+1|k) = \mathbf{h}(\mathbf{X}(k+1|k)) + \mathbf{w}_m(t) \quad (2.12)$$

\mathbf{w}_m représente les bruits gaussiens à moyenne nulle des mesures extéroceptives et sur la modélisation des observations. La matrice de covariance associée à cette observation est donnée par :

$$\mathbf{P}_Z(k+1|k) = \mathbf{H}_X \cdot \mathbf{P}(k+1|k) \cdot \mathbf{H}_X^T \quad (2.13)$$

\mathbf{H}_X est la matrice jacobienne de la fonction \mathbf{h} par rapport à l'état $\mathbf{X}(k+1|k)$.

Ces équations d'observation correspondent à des fonctions de *composition* introduites dans l'article [141]. Leurs formulations dépendent de la forme géométrique de l'amer observé (cf. l'annexe B).

2.2.4 Fonction de cartographie

Une mesure extéroceptive est modélisée par le vecteur \mathbf{Z}_{mesure} et la matrice de covariance associée \mathbf{C}_{mesure} . Ce vecteur \mathbf{P}_{mesure} contient m positions d'éléments de l'environnement \mathbf{X}_{Ai} avec pour chacune, une matrice de covariance associée \mathbf{C}_{Ai} :

$$\mathbf{Z}_{mesure} = \begin{bmatrix} \mathbf{X}_{A1} \\ \vdots \\ \mathbf{X}_{Ai} \\ \vdots \\ \mathbf{X}_{Am} \end{bmatrix} \quad (2.14)$$

Comme ces positions d'amer \mathbf{X}_{Ai} sont considérées comme indépendantes les unes des autres, la matrice de covariance associée à la mesure s'écrit :

$$\mathbf{P}_{mesure} = \begin{bmatrix} \mathbf{C}_{A1} & \cdots & 0 \\ & \ddots & \\ \vdots & \mathbf{C}_{Ai} & \vdots \\ 0 & \cdots & \mathbf{C}_{Am} \end{bmatrix} \quad (2.15)$$

Lors de la phase d'association de données, la carte estimée $\hat{\mathbf{X}}_M$ est mise en correspondance avec la carte mesurée. Si un élément mesuré n'est pas associé avec un élément déjà cartographié, il est considéré comme un nouvel amer. Cet élément doit être inséré dans la carte estimée $\hat{\mathbf{X}}_M$. Il est noté $\mathbf{X}_{nouveau}$ avec comme matrice de covariance associée $\mathbf{C}_{nouveau}$.

Les mesures de l'environnement \mathbf{X}_{Ai} sont référencées par rapport à la position du véhicule \mathbf{X}_V . La fonction non-linéaire \mathbf{g} translate ces nouvelles positions dans le repère W de la carte \mathbf{X}_M :

$$\mathbf{X}_{nouveau}^W = \mathbf{g}(\mathbf{X}(k+1|k+1), \mathbf{X}_{nouveau}) + \mathbf{w}_m(k) \quad (2.16)$$

w_m représente les bruits gaussiens à moyenne nulle des mesures extéroceptives. Ensuite, le vecteur d'état du nouvel amer est inséré dans le vecteur d'état \mathbf{X} du système :

$$\mathbf{X}_{aug} = \begin{bmatrix} \mathbf{X}_v \\ \mathbf{X}_{nouveau}^W \end{bmatrix} \quad (2.17)$$

Le calcul de la matrice de covariance augmentée \mathbf{P}_{aug} doit prendre en compte les incertitudes sur la mesure de la position du nouvel amer $\mathbf{C}_{nouveau}$ et l'incertitude de la position du véhicule dans la carte estimée \mathbf{P}_v . Cette matrice est obtenue avec les calculs :

$$\mathbf{P}_{aug} = \mathbf{G} \cdot \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{nouveau} \end{bmatrix} \cdot \mathbf{G}^T \quad (2.18)$$

\mathbf{G} représente la jacobienne de la fonction \mathbf{g} par rapport aux vecteurs \mathbf{X} et $\mathbf{X}_{nouveau}$:

$$\mathbf{G} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{G}_{X_V} & \mathbf{G}_{X_{nouveau}} \end{bmatrix} \quad (2.19)$$

\mathbf{G}_{X_V} correspond à la matrice jacobienne de la fonction \mathbf{g} par rapport à \mathbf{X}_V et $\mathbf{G}_{X_{nouveau}}$ à la matrice jacobienne de la fonction \mathbf{g} par rapport au vecteur $\mathbf{X}_{nouveau}$.

2.2.5 Algorithme général de l'EKF-SLAM

A chaque itération, trois étapes de calcul sont réalisées. Tout d'abord, une phase de prédiction est réalisée en s'appuyant sur la fonction \mathbf{f} de l'évolution des états de \mathbf{X} :

$$\hat{\mathbf{X}}(k+1 | k) = \mathbf{f}(\hat{\mathbf{X}}(k | k), \hat{\mathbf{U}}(k)) \quad (2.20)$$

$$\mathbf{P}(k+1 | k) = \mathbf{F}_X \cdot \mathbf{P}(k | k) \cdot \mathbf{F}_X^T + \mathbf{F}_U \cdot \mathbf{Q}_U \cdot \mathbf{F}_U^T \quad (2.21)$$

Dans un second temps, les observations prédites (positions des objets composant l'environnement) sont calculées à l'aide de la fonction \mathbf{h} :

$$\hat{\mathbf{Z}}(k+1 | k) = \mathbf{h}(\hat{\mathbf{X}}(k+1 | k)) \quad (2.22)$$

Après la mise en correspondance des mesures et des éléments de la carte, une innovation est calculée. Cette innovation, v , correspond à la différence entre les positions estimées et mesurées d'une même entité de la carte. Elle est associée à une matrice de covariance \mathbf{S} .

$$v(k+1) = \mathbf{Z}_{mesure}(k+1) - \hat{\mathbf{Z}}(k+1 | k) \quad (2.23)$$

$$\mathbf{S} = \mathbf{H}_X \cdot \mathbf{P}(k+1 | k) \cdot (\mathbf{H}_X)^T + \mathbf{R} \quad (2.24)$$

Le vecteur $\mathbf{Z}_{mesure}(k+1)$ correspond aux mesures extéroceptives associées à une matrice de covariance \mathbf{R} .

Au final, une correction de l'estimation de $\hat{\mathbf{X}}$ est réalisée à partir de l'innovation v . Cette correction est pondérée par un gain \mathbf{W} permettant de prendre en compte les incertitudes sur les variables de $\hat{\mathbf{X}}$ et sur les mesures \mathbf{Z} .

$$\hat{\mathbf{X}}(k+1 | k+1) = \hat{\mathbf{X}}(k+1 | k) + \mathbf{W} \cdot v(k+1) \quad (2.25)$$

$$\mathbf{P}(k+1 | k+1) = \mathbf{P}(k+1 | k) - \mathbf{W.S.W}^T \quad (2.26)$$

Le gain du filtre de Kalman, \mathbf{W} , se calcule avec l'équation :

$$\mathbf{W} = \mathbf{P}(k+1 | k) \cdot (\mathbf{H}_X)^T \cdot \mathbf{S}^{-1} \quad (2.27)$$

Le récapitulatif des différentes équations du filtre de Kalman non-linéaire clôt la description de la partie **estimation** (cf. la figure 2.1). Des informations complémentaires sont données dans les annexes A et B. Ces deux annexes présentent respectivement la théorie du filtre de Kalman et les équations non-linéaires employées dans une approche EKF-SLAM.

La suite de ce document se focalise sur les deux autres grandes étapes du filtre EKF-SLAM : **perception** et **association de données**. L'objectif est de pouvoir représenter l'environnement par des amers de type cercle et droite à partir de données télémétriques.

2.3 Représentation 2D de l'environnement

Dans ce paragraphe, est décrite la partie correspondant à la **perception** de l'environnement de l'algorithme SLAM employé (cf. la figure 2.1). Cette partie est très importante. En effet, l'environnement doit être modélisé correctement pour que les mesures soient pertinentes. Ainsi, le filtre EKF-SLAM pourra réaliser une correction cohérente des estimations de la carte et de la position du véhicule.

Dans la littérature, plusieurs modélisations stochastiques de l'environnement en 2D sont proposées. Dans ce paragraphe, un état de l'art non exhaustif est proposé sur les méthodes de cartographie et de segmentation lorsque le capteur employé est un télémètre laser. Cet état de l'art permet de justifier l'approche retenue pour représenter l'environnement sous une forme hétérogène.

2.3.1 Principales représentations stochastiques de l'environnement

Il existe trois principaux types de représentation d'un environnement à partir de données télémétriques à 2 dimensions :

- carte de balayages télémétriques brutes ;
- carte basée sur des corrélations de balayages télémétriques ;
- carte constituée d'une ou plusieurs entités géométriques.

Carte de balayages télémétriques brutes :

Dans cette approche, le vecteur estimé est composé des différentes positions du véhicule le long de sa trajectoire. A chacune des positions du véhicule est associée un balayage télémétrique. La carte est obtenue en superposant tous les balayages télémétriques enregistrés et associés aux positions du véhicule dans l'environnement. Cette approche est utilisée pour la représentation en 2D de l'environnement [105] et pour la représentation en 3D [43] [161]. Elle traite l'intégralité des mesures extéroceptives. Cependant, un traitement supplémentaire de la carte obtenue doit être réalisé pour obtenir une description et une position précise des différents

éléments constituant l'environnement exploré.

Carte basée sur des corrélations de balayages télémétriques :

L'environnement est représenté par les fusions successives des données télémétriques. Une carte composée uniquement de points est mise à jour à chaque nouvelle mesure. Un exemple de cette représentation correspond à l'approche *Scan Correlation* décrite dans l'article [122]. Dans cette application, les données brutes sont modélisées par des sommes de gaussiennes. Ces sommes de gaussiennes sont mises à jour à chaque itération du filtre de Kalman. Chaque nouvelle mesure est ainsi fusionnée dans la carte. Cette méthode traite la totalité des données extéroceptives. Elle a l'avantage de n'enregistrer qu'une seule carte et non l'intégralité des mesures le long de la trajectoire du véhicule comme dans les méthodes précédentes.

Carte constituée d'une ou plusieurs entités géométriques :

La représentation stochastique de l'environnement par des entités géométriques est la plus utilisée dans les applications EKF-SLAM. Dans une cartographie en 2D, les formes géométriques utilisées sont :

- des points [32][40][54][98][119] ;
- des droites [6][29][41][132][162] ;
- des cercles [78] ;
- des coins formés par les murs [2][108].

Dans la majorité des travaux cités, un unique type d'amer est employé bien que les environnements, intérieurs ou extérieurs, soient composés de plusieurs formes géométriques. Une cartographie hétérogène de l'environnement, employant plusieurs types d'amers, reste peu utilisée dans une approche SLAM. Elle est abordée dans les articles [44][146][165]. Dans [44] et [146], l'environnement intérieur est représenté par des points, des droites et des coins. Dans l'article [165], l'environnement considéré est semi-structuré et représenté par des cercles et des droites. L'intérêt de ces cartes hétérogènes pour représenter l'environnement est évident. L'emploi d'amers plus complexes apporte plus d'informations sur l'environnement. L'interprétation de la carte est plus simple que lors de l'utilisation exclusive de points.

Représentation de l'environnement employée :

L'objectif du SLAM est de localiser un véhicule dans un environnement inconnu mais également de cartographier cet environnement. La dernière approche, à la différence des deux premières, ne traite pas directement les balayages télémétriques pour mettre à jour le système. L'étape d'extraction des formes géométriques ajoute des incertitudes à la mesure de l'environnement. Cette approche est donc moins précise que les deux autres. Cependant, elle apporte une cartographie plus représentative de l'environnement. Pour ces différentes raisons, la méthode retenue dans ces travaux est une cartographie hétérogène. Elle représente l'environnement par deux entités géométriques : des droites et des cercles.

Ces entités géométriques doivent être extraites des balayages télémétriques (cf. la figure 2.3). Cette extraction est réalisée en deux étapes successives : la segmentation de la mesure et l'estimation des paramètres des amers. Ces deux étapes sont détaillées dans la suite du chapitre.

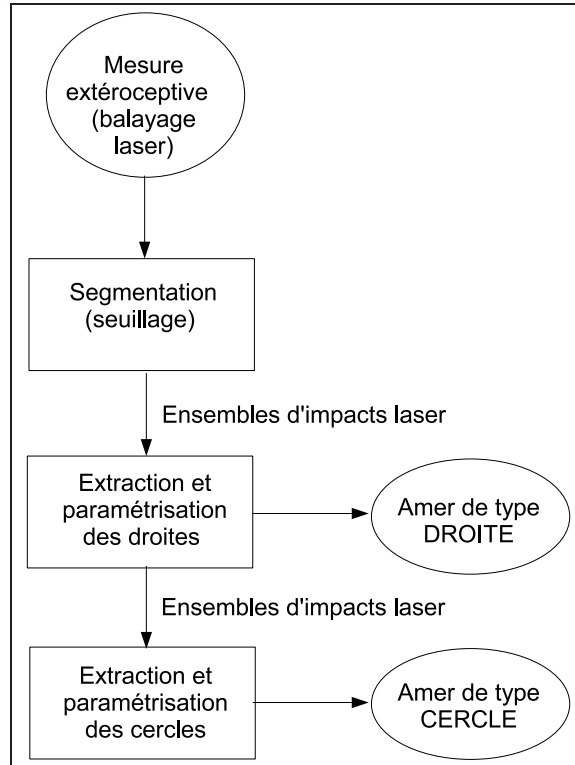


Figure 2.3: Schéma présentant les différentes étapes d'extraction des amers de type droite et cercle.

2.3.2 Segmentation d'un balayage télémétrique

L'information de base extraite d'un balayage télémétrique est un ensemble de points dans un espace en deux dimensions (cf. la figure 2.4). Dans les expérimentations et simulations proposées, cette mesure correspond à un balayage de 180 degrés avec une résolution de 0.5 degrés. Au final, chacune de ces mesures est donc composée de 361 points. Chaque impact est décrit par des coordonnées polaires. Le vecteur associé au $i^{\text{ème}}$ impact laser s'écrit :

$$\mathbf{X}_i = \begin{bmatrix} \rho_i \\ \beta_i \end{bmatrix} \text{ et la matrice de covariance associée : } \mathbf{R}_i = \begin{bmatrix} \sigma_{\rho_i}^2 & 0 \\ 0 & \sigma_{\beta_i}^2 \end{bmatrix} \quad (2.28)$$

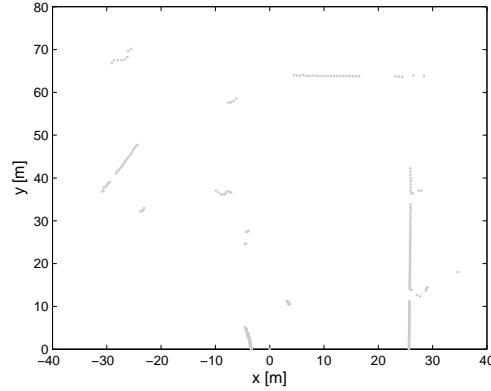


Figure 2.4: Exemple d'un balayage télémétrique brut représenté par les coordonnées cartésiennes des mesures.

Ces coordonnées polaires sont ensuite transformées¹ en coordonnées cartésiennes. Le vecteur associé au $i^{\text{ème}}$ impact laser se présente sous la forme :

$$\mathbf{X}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \text{ et matrice de covariance : } \mathbf{R}_i = \begin{bmatrix} \sigma_{x_i}^2 & \sigma_{x_i y_i} \\ \sigma_{x_i y_i} & \sigma_{y_i}^2 \end{bmatrix} \quad (2.29)$$

La segmentation consiste à diviser un balayage télémétrique (cf. la figure 2.4) en plusieurs amas de points. Ces ensembles d'impacts sont supposés appartenir au même élément de l'environnement. Il existe deux méthodes pour les obtenir [129].

La première est basée sur un filtre de Kalman (linéaire ou non-linéaire en fonction de la géométrie des amers). Les modèles des objets à extraire sont définis via les équations de prédiction et d'observation : des droites [134] et des cercles [165]. Ce sont des méthodes itératives traitant les impacts laser les uns après les autres. Elles permettent de réaliser simultanément la segmentation et l'extraction des paramètres des amers.

La seconde méthode de segmentation est basée sur le seuillage des distances entre chaque impact. Son implémentation est plus simple que la méthode reposant sur un filtrage de Kalman. De surcroît, les résultats sont équivalents [129]. C'est pour cette raison que cette méthode est retenue pour la segmentation des balayages télémétriques. Elle est présentée sur l'algorithme 2.1. Le seuil, D_{seuil} , permet de rejeter ou non un impact d'un ensemble A_k . Dans la littérature, il existe de nombreuses méthodes pour déterminer la valeur de ce seuil [20][50][136][129]. Suite à cette segmentation, les groupes de points sont seuillés en fonction de leur nombre d'impacts respectifs ($NB_{seuil} = 3$ dans l'algorithme 2.1). Si le nombre d'impacts de chaque ensemble est inférieur à ce seuil, les impacts composant l'ensemble rejeté sont éliminés.

1. Equations permettant la transformation de coordonnées polaires en coordonnées cartésiennes :

$$\begin{aligned} x_i &= \rho_i \cdot \cos \beta_i \\ y_i &= \rho_i \cdot \sin \beta_i \end{aligned}$$

A la fin, différents ensembles d'impacts laser sont obtenus. La prochaine étape est d'en extraire des formes géométriques (cercle et droite).

Algorithme 2.1 : Segmentation d'un balayage télémétrique

***Première partie* : Segmentation d'un balayage télémétrique fondée sur un seuillage**
 $n_{\text{impact}} \leftarrow$ nombre d'impacts laser

 $k \leftarrow 1$
 $i \leftarrow 1$
tant que ($i \leq n_{\text{impact}}$) **faire**

 Calcul de la distance Euclidienne entre deux impacts consécutifs : $D(X_i, X_{i+1})$

 Calcul de la distance seuil : D_{seuil}
si ($D(X_i, X_{i+1}) > D_{\text{seuil}}$) **alors**

Création d'un nouvel ensemble :

 $k \leftarrow k+1$
sinon

 Ajout de l'impacts dans l'ensemble courant A_k
 $A_k = [A_k, X_{i+1}]^T$
fin si
 $i \leftarrow i+1$
fin tant que
***Deuxième partie* : Trie des ensembles d'impacts laser**
 $n_{\text{ensemble}} \leftarrow$ nombre d'ensemble d'impacts laser

 $i \leftarrow 1$
tant que ($i \leq n_{\text{ensemble}}$) **faire**
 $n_i \leftarrow$ nombre d'impacts de l'ensemble A_i
si ($n_i > NB_{\text{seuil}}$) **alors**

Ensemble d'impacts laser retenu

sinon

Ensemble d'impacts laser éliminé

fin si
 $i \leftarrow i+1$
fin tant que

2.3.3 Extraction et paramétrisation des droites

2.3.3.1 Représentation des droites

Dans la littérature, une droite peut adopter deux types de représentation : polaire ou cartésienne (cf. la figure 2.5).

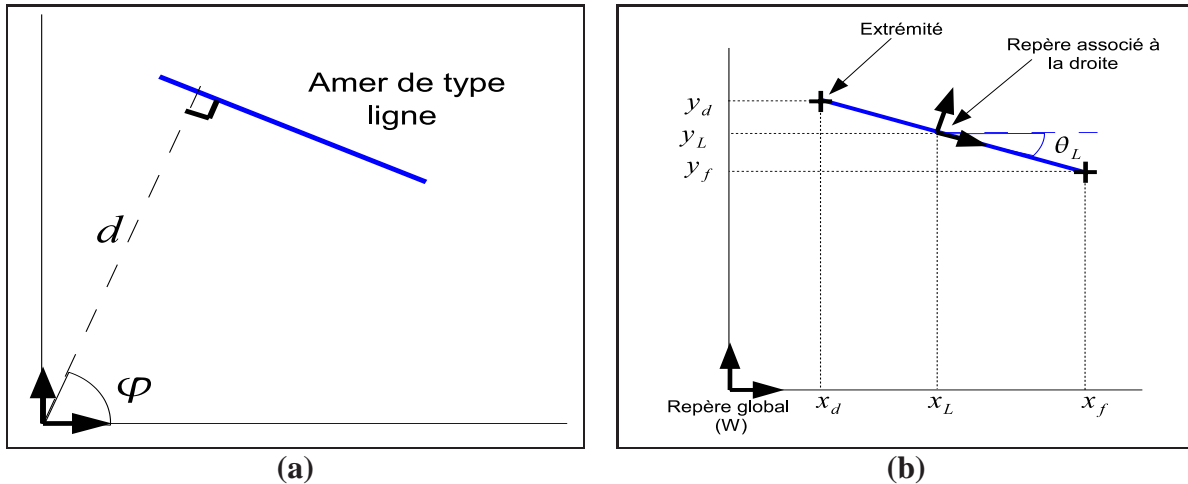


Figure 2.5: Type de représentation d'une droite : (a) représentation polaire d'une droite avec 2 états : $X_L = [d, \varphi]$, et (b) représentation cartésienne d'une droite avec 3 états : $X_L = [x_L, y_L, \theta_L]$.

Description des deux représentations :

La représentation polaire associe à une droite un couple de paramètres (d, φ) , respectivement la longueur de la normale de la droite passant par le centre du repère et l'angle formé par l'axe des abscisses et la normale à la droite (cf. la figure 2.5 (a)). Cette représentation est adoptée dans les travaux [5][85].

La représentation cartésienne associe un repère à une droite (cf. la figure 2.5 (b)). Elle s'appuie sur la théorie des symétries définie dans l'article [145]. Ce repère est décrit par ses coordonnées cartésiennes, x_L et y_L , et l'orientation de la droite dans le repère de la carte, θ_L . Cette modélisation est appliquée dans un contexte SLAM dans les travaux [130][44][163].

Avantages et inconvénients des deux représentations :

Chacune des représentations possède des avantages et des inconvénients. Pour la représentation polaire, la droite est décrite par seulement deux paramètres. En revanche, dans le cas de la représentation cartésienne, l'utilisation de la théorie des symétries [145] permet d'associer des amers de types différents comme le prouvent les travaux [44]. De surcroît, elle permet de résoudre des problèmes d'associations ambiguës (cf. la figure 2.6). Avec une représentation cartésienne, il est possible de faire la différence entre les deux droites dans le cas d'étude (b), chose impossible avec la représentation polaire.

Choix de la représentation :

L'approche SLAM présentée dans ce document s'appuie sur une représentation hétérogène de l'environnement. Pour pouvoir différencier facilement des amers de types différents, l'approche cartésienne est plus appropriée que la représentation polaire. De plus, elle permet une

meilleure mise en correspondance des amers.

Dans la suite de ce chapitre, la méthode, permettant de déterminer les paramètres des amers cartographiés, est présentée.

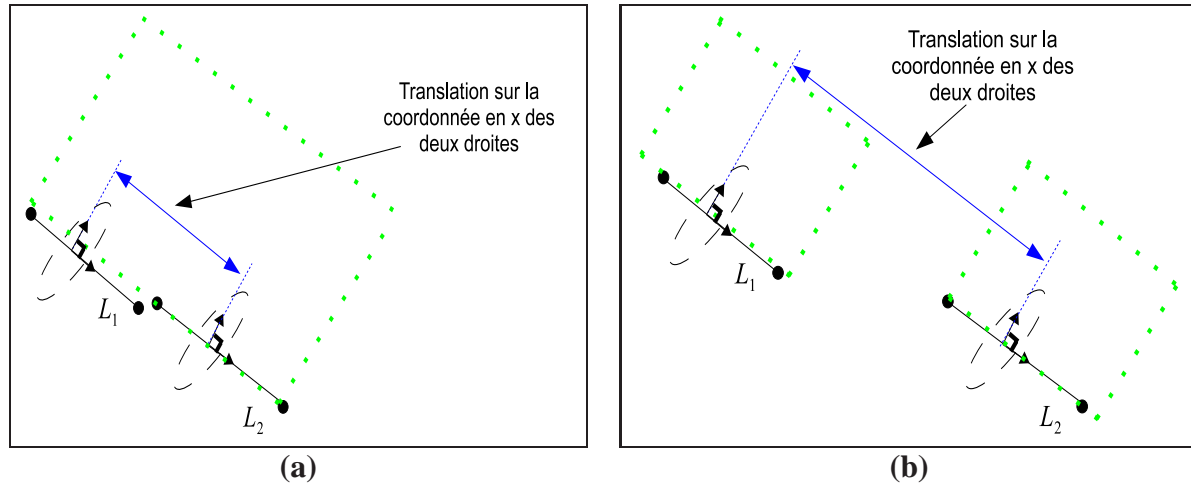


Figure 2.6: Emploi du troisième état d'une droite pour résoudre des associations ambiguës : (a) cas simple d'association de deux droites appartenant au même mur (pointillés verts) et (b) cas ambigu d'association de deux droites appartenant à des murs différents.

2.3.3.2 Extraction des droites

Il existe de nombreuses méthodes d'extraction et de paramétrisation de droites à partir de données télémétriques [120].

La méthode de Hough [74] est l'une des méthodes les plus usitées. Pour extraire les droites d'un balayage télémétrique, un changement d'espace des paramètres est effectué permettant de déterminer les impacts appartenant à la même droite. Cette méthode donne des résultats précis à la condition que la taille de la grille dans l'espace des paramètres des droites soit très petite. L'inconvénient est qu'une telle précision impose un temps de calcul conséquent. D'autres méthodes sont fondées sur des algorithmes de RANSAC² et des algorithmes EM³. Ces méthodes sont plus rapides que les méthodes basées sur la méthode de Hough. Cependant, elles réalisent des fausses détections comme le montrent les résultats de l'article [120].

La méthode de segmentation retenue s'appuie sur l'*algorithme de la corde*. A l'origine appliquée dans le domaine de la vision, cette méthode a été étudiée et appliquée dans de nombreux travaux utilisant un télémètre laser [20][27][7]. Cette méthode donne de très bons résultats avec ce type de capteur et, de surcroît, est peu coûteuse en temps de calcul.

L'algorithme de cette méthode se décompose en quatre étapes :

- extraction des points appartenant à la même droite ;
- détermination des paramètres de la droite avec la matrice de covariance associée ;

2. *Random Sample Consensus*

3. *Expectation and Maximization*

- mise en correspondance des droites extraites, détection et fusion des droites identiques ;
- détermination des coordonnées extrémités du segment porté par la droite.

Ces différentes étapes sont décrites dans l'annexe C. Une fois que les droites sont extraites du balayage télémétrique, les amers de type cercle sont extraits à partir des impacts restants (cf. la figure 2.3).

2.3.4 Extraction et paramétrisation des cercles

Après avoir déterminé les amers droites, les cercles sont extraits des ensembles d'impacts restants en deux étapes distinctes. Dans un premier temps, les paramètres de ces amers sont estimés à partir d'un algorithme de Levenberg-Marquardt [155]. Ensuite, un seuillage métrique sur la valeur du rayon est appliqué dans le but d'éliminer les fausses détections. Comme l'environnement considéré est semi-urbain, les cercles correspondent à des arbres ou à des poteaux. Leur rayon ne peut dépasser dans la majorité des cas le mètre.

2.3.4.1 Représentation des cercles

Un amer de type cercle est décrit basiquement par trois paramètres (cf. la figure 2.7) : les coordonnées de son centre, x_{ce} et y_{ce} , et la valeur de son rayon, r_{ce} . Cependant, des résultats préliminaires ont prouvé que l'estimation d'un rayon dans une approche EKF-SLAM donnait de mauvais résultats. La valeur de ce paramètre oscillait jusqu'à prendre des valeurs négatives. Pour éviter ce phénomène, un cercle est uniquement représenté par les coordonnées de son centre :

$$\mathbf{X}_{ce} = \begin{bmatrix} x_{ce} \\ y_{ce} \end{bmatrix} \quad (2.30)$$

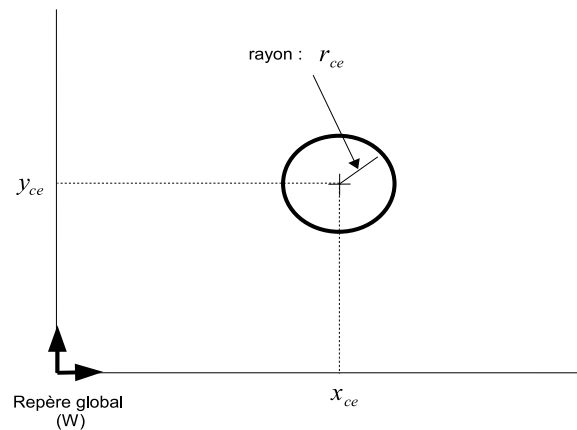


Figure 2.7: Représentation d'un cercle.

2.3.4.2 Extraction des cercles

Dans la littérature, il existe plusieurs méthodes pour extraire des cercles d'un balayage télémétrique. Elles sont comparées dans l'article [129]. Il en ressort que l'approche de Guivant

[8] est précise tout en étant rapide. Cette méthode est retenue pour extraire des cercles d'un ensemble de points. Cependant, dans le but d'optimiser la précision des paramètres estimés, cette méthode est comparée à deux autres algorithmes d'estimation : l'algorithme de Gauss-Newton [69] et celui de Levenberg-Marquardt [155].

La comparaison des trois algorithmes est présentée dans l'annexe C. Les résultats, sur des données simulées, prouvent que la méthode de Levenberg-Marquardt est plus précise que les deux autres méthodes. L'emploi d'un algorithme itératif de Levenberg-Marquardt, initialisé avec les valeurs obtenues avec la méthode de Guivant, est donc retenu pour l'extraction et l'estimation des paramètres d'un cercle.

2.3.5 Application de la méthode de segmentation sur des données réelles

Pour conclure cette partie, les méthodes de segmentation d'un balayage télémétrique et d'extraction d'amers sont appliquées sur des données réelles. La mesure extéroceptive utilisée est issue d'un environnement extérieur. Toutes les étapes, présentées dans les paragraphes précédents, sont décrites par différents graphiques (cf. la figure 2.8) :

- *Données extéroceptives issues du capteur télémétrique (a)* : tous les points gris correspondent aux impacts laser recueillis par le télémètre ;
- *Segmentation reposant sur un seuil métrique (b)* : le balayage télémétrique est découpée en différents ensembles de points ;
- *Seuillage en fonction du nombre de points (c)* : les ensembles retenus possèdent un nombre d'impacts suffisant pour permettre l'extraction d'une forme géométrique ;
- *Extraction des droites (d)* : chaque ensemble d'impacts est traité séparément pour extraire, si possible, une droite. Dans cet exemple, seulement deux ensembles ont permis une extraction. Ces droites représentées en bleues sont associées à un repère tracé en noir ;
- *Fusion des droites (e)* : les deux droites précédemment extraites correspondent visuellement à la même droite de l'environnement. Elles sont donc fusionnées ;
- *Extraction des cercles (f)* : des cercles sont extraits des derniers ensembles de points restants, ils sont représentés en vert sur l'exemple. Les ensembles de points restants, dont aucun amer n'a été extrait, sont éliminés.

La figure 2.9 représente la carte finale mesurée. Seuls les amers pertinents sont extraits du balayage télémétrique. Dans l'exemple étudié, la mesure finale se compose d'une droite et de quatre cercles. Les autres impacts laser sont éliminés. Ce résultat est en adéquation avec la philosophie présentée dans nos travaux : seules les informations les plus pertinentes et les plus sûres sont conservées pour construire la carte de l'environnement.

Cet exemple d'application clôt la partie sur la représentation de l'environnement à partir de données télémétriques. La suite de ce document présente les méthodes d'association de données permettant de mettre en correspondance ce type de mesure avec la carte estimée par le système.

2.4 Association de données

L'**association de données** correspond à une étape primordiale lors de l'application d'un filtre estimateur dans un contexte SLAM. Elle représente un des aspects les plus critiques de cet

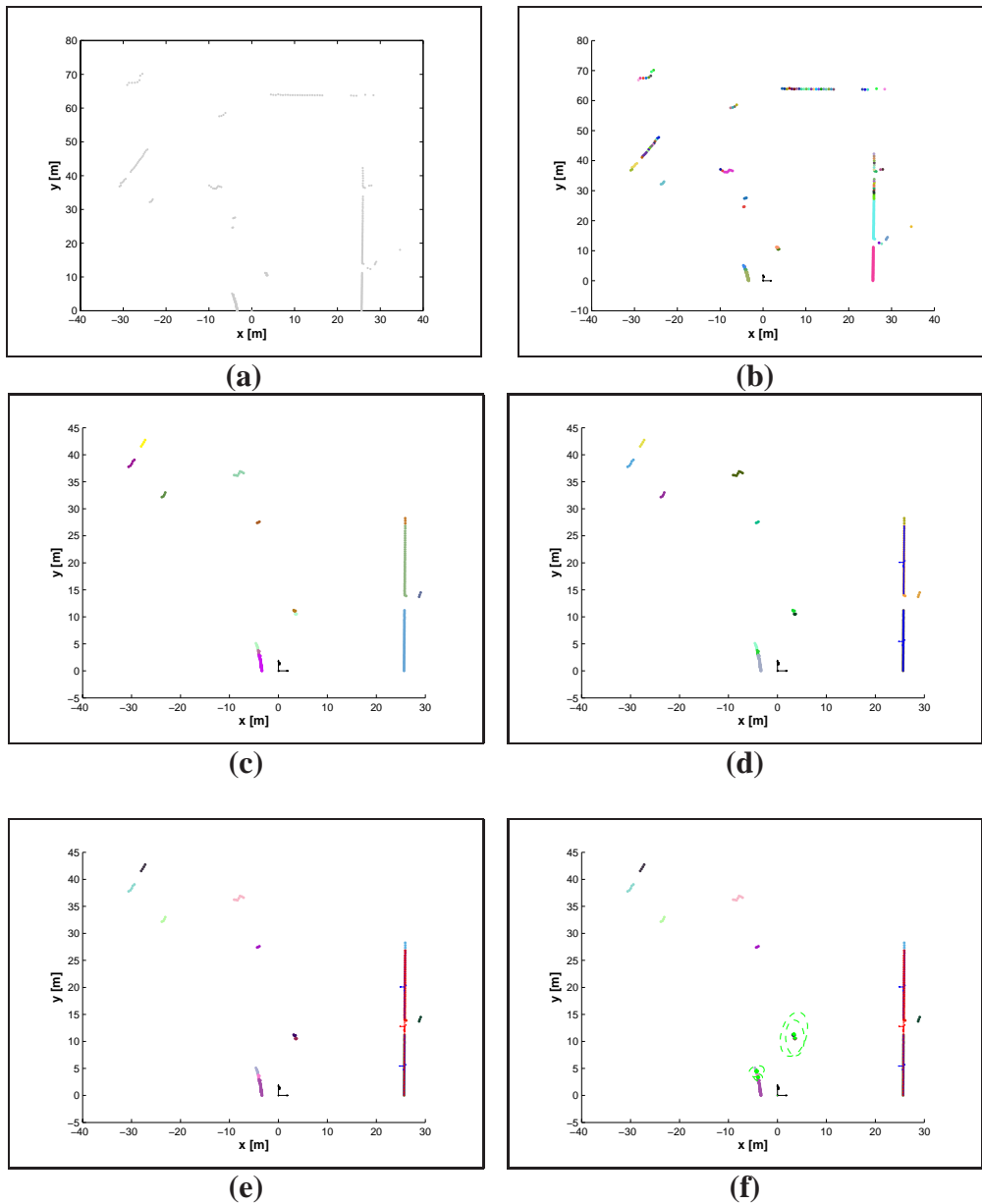


Figure 2.8: Application de la méthode de segmentation sur un balayage télémétrique : (a) mesure brute, (b) segmentation en fonction d'un seuil métrique, (c) ensemble d'impacts seuillés en fonction du nombre de points les constituant, (d) extraction des droites, (e) fusion des droites et (f) extraction des cercles.

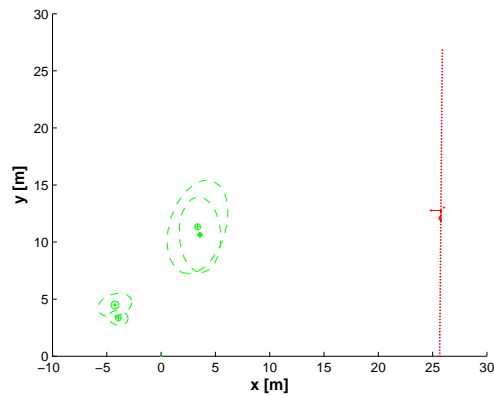


Figure 2.9: Carte d'amers obtenue avec les méthodes de segmentation et d'extraction d'amers de type droite et cercle.

algorithme. L'objectif est d'associer les mesures extéroceptives avec les éléments de la carte estimée. Cette mise en correspondance permet le calcul de la correction que l'estimateur bayésien doit apporter aux estimations de la position du véhicule et de la carte de l'environnement.

Avant de présenter les deux méthodes d'association de données utilisées, un état de l'art des méthodes de mise en correspondance de balayages télémétriques est présenté dans la suite de ce document.

2.4.1 Etat de l'art des méthodes existantes

De nombreuses méthodes de mise en correspondance existent dans la littérature scientifique. En 1995, Gutmann et al. [79] ont présenté un état de l'art des méthodes pionnières existantes pour les environnements intérieurs. Les algorithmes d'association de données sont classés en quatre groupes en fonction des représentations des mesures et de la carte estimée de l'environnement.

- mise en correspondance de balayages télémétriques ;
- mise en correspondance d'amers et d'impacts laser ;
- mise en correspondance fondée sur l'utilisation de corrélations d'histogrammes ;
- mise en correspondance d'amers extraits d'un balayage télémétrique avec une carte estimée d'amers.

Mise en correspondance de balayages télémétriques point par point :

L'objectif de cette mise en correspondance est de déterminer la translation et la rotation entre deux mesures laser pour permettre une mise à jour des estimations du système. La méthode la plus usitée correspond à l'algorithme ICP⁴ [17]. Pour se faire, cet algorithme recherche les distances les plus faibles entre les impacts associés de chaque balayage télémétrique. Dès

4. *Iterative Closest Point*

lors que ces distances sont trouvées, les paramètres de translation et de rotation entre les deux mesures sont déterminés. Pour améliorer le temps de calcul et la précision des résultats, deux algorithmes, basés sur la méthode ICP⁴, furent proposés par Lu et al. [106]. Le premier algorithme nommé IMRP⁵ propose une meilleure sélection des points mis en correspondance entre les deux mesures laser considérées. Le deuxième algorithme IDC⁶ combine les algorithmes ICP⁴ et IMRP⁵. Dans le but d'améliorer la qualité des mises en correspondance, la méthode PSM⁷ [52] emploie les coordonnées polaires des impacts pour les associer. D'autres méthodes, travaillant toujours avec des balayages télémétriques, s'inspirent des méthodes utilisées en vision telles que les méthodes SIFT⁸ [104] et SURF⁹ [14]. L'intérêt est d'optimiser la sélection des impacts. En effet, les points les plus pertinents sont retenus d'une manière similaire à la sélection des points d'intérêt dans des images.

Mise en correspondance d'amers et d'impacts laser :

La carte estimée constituée d'amers est comparée à un balayage télémétrique. Chaque impact de la mesure est associé aux modèles *a priori* connu de la carte estimée. Par exemple, dans les travaux [46], le modèle *a priori* d'amer connu correspond à des droites. L'objectif est de mettre en correspondance les impacts avec les différentes droites. Cet algorithme a été amélioré en ajoutant en amont un filtrage pour ne retenir que les impacts appartenant à une droite [79]. Cette sélection permet un gain en temps de calcul, comme le nombre de points mis en correspondance est moins important.

Mise en correspondance basée sur une représentation des histogrammes :

Dans les méthodes précédentes, des correspondances explicites sont utilisées pour déterminer la rotation et la translation entre deux mesures. Dans ce type d'approche, la mise en correspondance est basée sur des fonctions de corrélation croisée [159]. Les balayages télémétriques sont représentés par deux histogrammes. Le premier est un histogramme d'orientations des normales des impacts et le second contient les valeurs des projections pondérées des impacts sur ces normales. En recherchant les corrélations maximales entre les histogrammes de chacun des deux balayages, l'algorithme [159] détermine la rotation et la translation entre les deux mesures sans aucune connaissance *a priori*. Une méthode semblable est proposée par Bosse et al. [22]. Elle applique cet algorithme, basé sur des corrélations, pour détecter le recouvrement de cartes locales dans une application SLAM multi-cartes (travaux ATLAS [21]).

Mise en correspondance d'amers avec des amers :

Ces approches mettent en correspondance des cartes composées d'amers. La majorité de ces algorithmes est basée sur le calcul d'une distance, pondérée par des incertitudes [12], pour

5. *Iterative Matching Range Point*

6. *Iterative Dual Correspondence*

7. *Polar Scan Matching*

8. *Scale Invariant Feature Transform*

9. *Speeded Up Robust Features*

déterminer les mises en correspondance individuelles entre les amers. Ensuite, une méthode des plus proches voisins est appliquée. Cependant, lorsque l'environnement à cartographier est très dense, cette méthode ne suffit pas pour réaliser des associations correctes.

Pour améliorer la précision des mises en correspondance, la méthode [101] repose sur une minimisation des moindres carrés pour déterminer la translation entre les deux balayages télé-métriques. L'objectif est semblable aux méthodes de Cox [46] et de Lu (IDC⁵). Seuls des amers pertinents sont retenus pour l'association des deux cartes. Cela permet de diminuer les temps de calcul et d'appliquer cette méthode de mise en correspondance en temps réel. Ces méthodes sont utilisées lorsque le déplacement entre deux mesures n'est pas connu.

Dans notre application, le déplacement entre deux mesures laser est donné par les capteurs proprioceptifs. L'objectif est seulement d'optimiser la mise en correspondance des amers mesurés et estimés. Pour se faire, certaines méthodes d'association de données [126] [164] reformulent le problème de mise en correspondance. Les associations individuelles sont décrites par un 1 ou un 0 si un amer est respectivement associé ou non avec un autre amer. Ensuite, des algorithmes, nommés IHGR¹⁰ fondés sur des techniques de programmation linéaire, recherchent les associations optimales entre les différents amers. Cette technique améliore les résultats d'association en comparaison à la méthode des plus proches voisins tout en restant un algorithme applicable temps réel.

Cependant, toutes les méthodes précédentes d'association ne considèrent pas les corrélations existantes entre les amers présents sur la carte estimée. Ces informations ne peuvent être ignorées lors de la mise en correspondance avec les mesures. L'algorithme JCBB¹¹ [116] propose d'utiliser, en plus des associations individuelles, des probabilités jointes prenant en compte les corrélations entre les éléments cartographiés. Cet algorithme permet donc une mise en correspondance d'amers corrélés. En comparaison aux autres méthodes présentées dans cet état de l'art, cette méthode possède plusieurs avantages. Tout d'abord, elle réalise une mise en correspondance d'amers et non d'impacts laser. Elle s'applique donc à une représentation hétérogène de l'environnement. Elle explore toutes les possibilités de mise en correspondance pour ne retenir que la meilleure combinaison. Elle est directement applicable dans un filtre de Kalman qui repose sur la représentation des paramètres par des distributions gaussiennes. Malheureusement, le temps de calcul devient très vite conséquent dès lors que le nombre d'amers sur la carte devient important. Cet inconvénient est atténué en divisant la représentation de l'environnement par des cartes locales. Le nombre d'amers présents sur chaque carte reste faible. Ainsi, l'application de la méthode JCBB reste possible en temps réel.

2.4.1.1 Conclusion

Cet état de l'art nous montre qu'il existe de nombreuses méthodes de mise en correspondance de mesures provenant d'un télémètre laser. Cependant, seules deux d'entre elles traitent des cartes composées d'entités géométriques et non d'impacts laser : la méthode basique des plus proches voisins et l'algorithme JCBB [116]. Elles correspondent donc aux deux méthodes retenues pour la phase d'association de données. L'unique différence entre elles repose sur la considération ou non des probabilités jointes entre les amers de la carte.

10. *Iterative Heuristic Greedy Rounding*

11. *JointCompatibility Branch and Bound*

Ces deux approches s'appuient sur des associations individuelles d'amers. Comme l'environnement est représenté par des cercles et des droites, la suite de ce chapitre introduit :

- mise en correspondance de deux cercles ;
- mise en correspondance de deux droites ;
- mise en correspondance d'une droite et d'un cercle.

Une fois ces différentes associations individuelles présentées, les deux méthodes d'associations employées sont introduites.

2.4.2 Mise en correspondance des amers et méthode des plus proches voisins

2.4.2.1 Mise en correspondance individuelle des amers

Lors de l'utilisation d'un filtre de Kalman, la carte estimée par le système et les observations de l'environnement sont supposées suivre des lois gaussiennes. Il est alors possible de mettre en correspondance, de manière statistique, les amers composant la carte estimée et les mesures de ceux-ci.

Cette mise en correspondance est réalisée dans l'espace d'observation. La projection des amers de la carte estimée, $\hat{\mathbf{X}}_j$, dans le repère associé à la position du robot est nécessaire. Cette projection est modélisée par la fonction non-linéaire \mathbf{h} . La position prédite de l'amer s'écrit $\mathbf{h}(\hat{\mathbf{X}}_j)$.

Pour savoir si une entité estimée correspond à une mesure, les tests de validation s'appuient sur le calcul du carré de la distance de Mahalanobis¹². Étant donnée l'innovation $\nu_{ij} = \mathbf{Z}_i - \mathbf{h}(\hat{\mathbf{X}}_j)$ et sa matrice de covariance associée \mathbf{S}_{ij} , le carré de la distance de Mahalanobis¹² [12] est définie par :

$$d_{ij} = \nu_{ij}^T \cdot \mathbf{S}_{ij}^{-1} \nu_{ij} \quad (2.31)$$

Cette distance d_{ij} suit une distribution du χ^2 . Cette distribution permet de déterminer un seuil pour savoir si la mesure \mathbf{Z}_i correspond à l'amer de la carte estimée $\hat{\mathbf{X}}_j$. Pour prouver cette correspondance, l'inéquation suivante doit être vérifiée :

$$d_{ij} < \chi_{n,\alpha}^2 \quad (2.32)$$

Les paramètres de la distribution n et α correspondent respectivement au nombre de degrés de liberté des entités comparées et au seuil de signification.

Un exemple simple de l'application de la distance de Mahalanobis est donné sur la figure 2.10. La carte estimée est composée d'un amer $\hat{\mathbf{X}}_1$ et l'observation de l'environnement nous donne deux positions d'amer \mathbf{Z}_1 et \mathbf{Z}_2 . Les distances de Mahalanobis sont données par d_{11} et d_{21} et le seuil du χ^2 est représenté par les ellipses sur la figure 2.10. Dans cet exemple, seule la mesure \mathbf{Z}_1 correspond à l'amer $\hat{\mathbf{X}}_1$. La deuxième mesure, \mathbf{Z}_2 , sera considérée comme une nouvelle entité sur la carte estimée.

Maintenant que les notions de distances probabilistes et de seuils sont présentées, les prochains paragraphes vont présenter les différents types d'associations individuelles rencontrés lors de l'utilisation de cartes hétérogènes de l'environnement.

12. Cette distance est également appelée *Normalized Innovation Squared* (NIS)

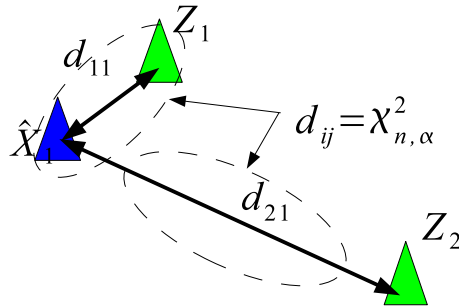


Figure 2.10: Application de la distance de Mahalanobis : les flèches noires représentent les distances de Mahalanobis entre les deux mesures et la position estimée de l'amer, les ellipses en pointillés correspondent au seuil du χ^2 employé pour la mise en correspondance des entités. Dans cet exemple, seule la mesure Z_1 est associée.

2.4.2.2 Mise en correspondance d'amers de type cercle

La mise en correspondance d'un amer de type cercle est identique à celle d'un amer point. Soit l'innovation ν correspondant à la différence entre le vecteur de la mesure d'un cercle, \mathbf{Z} , et le vecteur prédit, $\mathbf{h}(\hat{\mathbf{X}})$, d'un cercle de la carte estimée. Le carré de la distance de Mahalanobis, d , de ce couple d'amers s'écrit :

$$d = (\mathbf{Z} - \mathbf{h}(\hat{\mathbf{X}}))^T \cdot \mathbf{S}^{-1} \cdot (\mathbf{Z} - \mathbf{h}(\hat{\mathbf{X}})) \quad (2.33)$$

où la matrice \mathbf{S} correspond à la somme des covariances de la mesure et de l'amer prédit.

Ensuite, un test du χ^2 (inéquation (2.10)) est réalisé avec un nombre de degrés de liberté égal à deux et un nombre d'écart-types égal à deux. Le seuil, obtenu à partir de la distribution χ^2 s'écrit donc $\chi^2_{2,0.95}$.

2.4.2.3 Mise en correspondance de deux droites

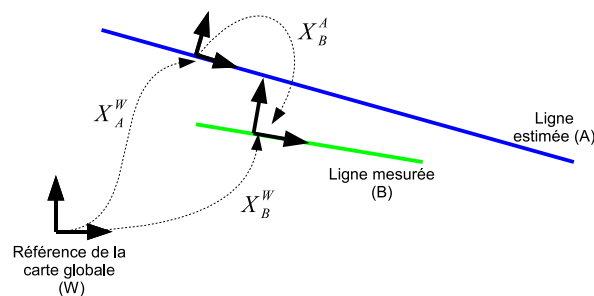


Figure 2.11: Mise en correspondance de deux droites.

Le calcul de l'innovation entre deux droites est différent de celui calculé avec des amers de type point ou de type cercle. L'orientation de la droite doit être prise en compte lors de l'association de deux droites.

Soient deux amers de type droite, \mathbf{X}_A^W et \mathbf{X}_B^W définis dans le repère global W avec comme matrice de covariance $\mathbf{C}_{X_A^W}$ et $\mathbf{C}_{X_B^W}$. L'innovation entre la droite estimée A et la droite mesurée B (figure 2.11), correspond à la transformation rigide entre les deux droites. Elle correspond à la variable \mathbf{X}_B^A associée à une matrice de covariance \mathbf{S}_{AB} . Le calcul de ce vecteur repose sur les équations de composition (cf. l'annexe B) :

$$\mathbf{X}_B^A = \ominus \mathbf{X}_A^W \oplus \mathbf{X}_B^W = \mathbf{X}_W^A \oplus \mathbf{X}_B^W \quad (2.34)$$

$$\mathbf{S}_{AB} = J_{1\oplus}(\mathbf{X}_W^A, \mathbf{X}_B^W) \cdot \mathbf{C}_{X_A^W} \cdot J_{1\oplus}^T(\mathbf{X}_W^A, \mathbf{X}_B^W) + J_{2\oplus}(\mathbf{X}_W^A, \mathbf{X}_B^W) \cdot \mathbf{C}_{X_B^W} \cdot J_{2\oplus}^T(\mathbf{X}_W^A, \mathbf{X}_B^W) \quad (2.35)$$

avec :

$$\mathbf{C}_{X_W^A} = J_{\ominus}(\mathbf{X}_A^W) \cdot \mathbf{C}_{X_A^W} \cdot J_{\ominus}^T(\mathbf{X}_A^W) \quad (2.36)$$

Si la transformation rigide \mathbf{X}_B^A est nulle, ou quasiment nulle à une incertitude près définie dans \mathbf{S}_{AB} , les amers A et B représentent la même entité. Pour statuer sur l'association de ces deux amers, une distance NIS (2.31) est calculée. Cependant, l'information utile apportée par une droite correspond à l'orientation et la translation latérale, respectivement θ_L et y_L . Pour n'utiliser que cette information, la théorie des symétries, définie dans l'article [145], est utilisée. La transformation rigide \mathbf{X}_B^A est multipliée par une matrice, dite de liaison, \mathbf{B}_{droite} définie par :

$$\mathbf{B}_{droite} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.37)$$

La distance NIS d_{AB} est obtenue par le calcul suivant :

$$d_{AB} = [\mathbf{B}_{droite} \cdot X_B^A]^T \cdot [\mathbf{B}_{droite} \cdot \mathbf{S}_{AB} \cdot \mathbf{B}_{droite}^T]^{-1} \cdot [\mathbf{B}_{droite} \cdot X_B^A] \quad (2.38)$$

Ensuite, le test du χ^2 est appliqué via l'inéquation (2.32). Le nombre de degrés de liberté de la distribution du χ^2 est défini par la matrice de liaison, dans le cas des droites, il est égal à deux.

2.4.2.4 Mise en correspondance d'une droite et d'un cercle

La représentation des droites utilisées [27] permet d'associer très simplement deux amers avec des géométries différentes, par exemple une droite et un point. L'amer de type point n'ayant pas d'information angulaire, la distance qui sera calculée correspondra au déplacement latéral entre le point et la droite (cf. la figure 2.12). Ce déplacement correspond tout simplement à la perpendiculaire à la droite passant par le point. Comme dans le cas de la mise en correspondance de deux droites, le calcul s'appuie sur la définition d'une matrice de liaison permettant de définir les symétries entre les deux amers :

$$\mathbf{B}_{droite-point} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.39)$$

Pour être cohérent avec les notations dans le calcul de la distance NIS, l'orientation de l'amer de type point sera nulle. La transformation rigide entre les deux amers est donnée par :

$$\mathbf{X}_C^A = \ominus \mathbf{X}_A^W \oplus \mathbf{X}_C^W \quad (2.40)$$

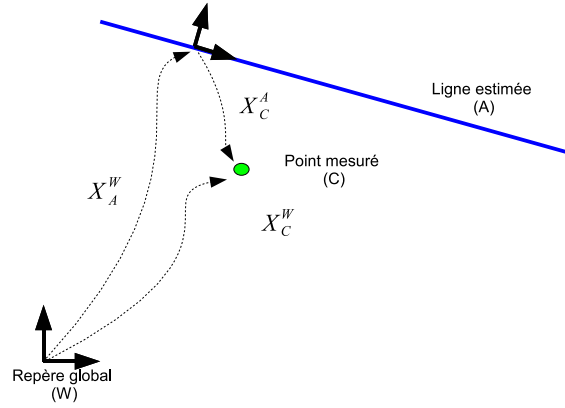


Figure 2.12: Mise en correspondance d'une droite et d'un point.

La distance de Mahalanobis s'obtient avec l'équation :

$$d_{AC} = [\mathbf{B}_{droite-point} \cdot \mathbf{X}_C^A]^T \cdot [\mathbf{B}_{droite-point} \cdot \mathbf{S}_C^A \cdot \mathbf{B}_{droite-point}^T]^{-1} \cdot [\mathbf{B}_{droite-point} \cdot \mathbf{X}_C^A] \quad (2.41)$$

Si la condition du test du χ^2 n'est pas remplie, la mesure du cercle est retenue. Si le test est validé, la mesure correspond à une fausse détection et sera donc supprimée.

2.4.3 Méthode des plus proches voisins

La mise en correspondance individuelle d'une mesure et d'une prédiction de position d'amer est simple. Dans une application SLAM, une carte, qu'elle soit mesurée à l'instant courant ou estimée, est composée de plusieurs amers. La mise en correspondance se révèle parfois délicate (cf. la figure 2.13). La position estimée de l'amer $\hat{\mathbf{X}}_1$ s'associe statistiquement avec les deux mesures \mathbf{Z}_1 et \mathbf{Z}_2 . Pour résoudre cette ambiguïté, la méthode la plus couramment utilisée est celle des plus proches voisins. Cette méthode correspond à une maximisation de la probabilité de l'association d'une entité i avec une autre entité j :

$$\Lambda_{ij} = \frac{1}{(2\pi)^{\frac{n}{2}} \cdot \sqrt{|\mathbf{S}_{ij}|}} \exp\left(-\left(\frac{1}{2}\right) \cdot \nu_{ij}^T \cdot \mathbf{S}_{ij}^{-1} \cdot \nu_{ij}\right) \quad (2.42)$$

La variable ν_{ij} , définie avec une matrice de covariance associée \mathbf{S}_{ij} , correspond à l'innovation entre la mesure i et l'estimation j . La maximisation de la probabilité Λ_{ij} revient à choisir la mesure \mathbf{Z}_i qui minimise la distance normalisée N_{ij} [12] :

$$N_{ij} = \nu_{ij}^T \cdot \mathbf{S}_{ij}^{-1} \cdot \nu_{ij} + \ln |\mathbf{S}_{ij}| \quad (2.43)$$

La méthode des plus proches voisins donne de bons résultats lorsque la carte estimée contient peu d'amers. Lorsque l'environnement est composé d'un grand nombre d'éléments, ses performances se détériorent rapidement. En pratique, à cause de mauvaises associations, la carte estimée diverge et la position du robot devient erronée. Plusieurs méthodes permettant d'atténuer ce problème existent dans la littérature.

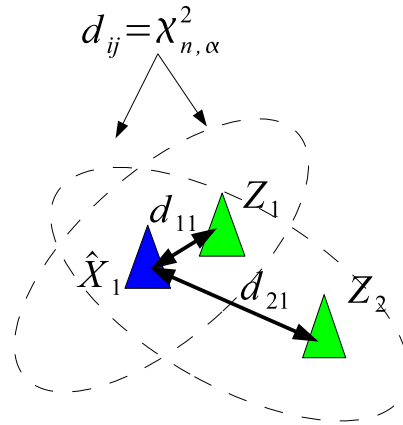


Figure 2.13: Méthode des plus proches voisins : exemple de mise en correspondance ambiguë.

La méthode d'association de données PDA¹³ [13] incorpore une incertitude induite par ces associations ambiguës. Cette méthode combine l'information provenant de toutes les associations candidates comme une moyenne pondérée de mises à jour individuelles. L'incertitude de cette mise à jour combinée est modifiée pour représenter l'incertitude de l'association. Une forme améliorée de cet algorithme nommée JPDA¹⁴ [65] permet un suivi de piétons à l'aide d'un télémètre laser.

L'approche de suivi d'hypothèses multiples, MHT¹⁵ [138], permet de résoudre en partie le problème. Cet algorithme retarde l'application d'assignation délicate entre une mesure et une prédiction en réalisant un suivi individuel pour chaque association ambiguë. Lorsque de nouvelles mesures sont réalisées, de nouvelles hypothèses sont créées et les anciennes sont mises à jour. L'apport de ces nouvelles informations permet de résoudre les associations ambiguës obtenues avec d'anciennes mesures. Le problème de cette méthode est qu'elle réduit le nombre d'associations à chaque itération bien qu'elle résolve le problème des ambiguïtés de façon efficace.

Les deux méthodes précédentes essaient de réduire les ambiguïtés d'association des amers soit en retardant la décision de mise en correspondance ou en prenant en compte statistiquement l'ambiguïté d'association. Ceci implique une réduction du nombre de possibilités d'associations et donc du nombre d'éléments de la carte mis à jour. Une solution plus pertinente correspond à l'algorithme JCBB¹¹ détaillé dans le chapitre suivant.

2.4.4 Algorithme de « Joint Compatibility Branch and Bound »

Neira et al. [116] proposent un algorithme d'association de données permettant de résoudre les ambiguïtés de mise en correspondance d'amers : *Joint Compatibility Branch and Bound* (JCBB). Cette méthode prend en considération les corrélations entre les amers de la carte esti-

13. *Probabilistic Data Association*

14. *Joint Probabilistic Data Association*

15. *Multiple Hypothesis Tracking*

mée. Elle permet d'obtenir des associations plus précises et donc des estimations de la carte et de la position du véhicule plus précises.

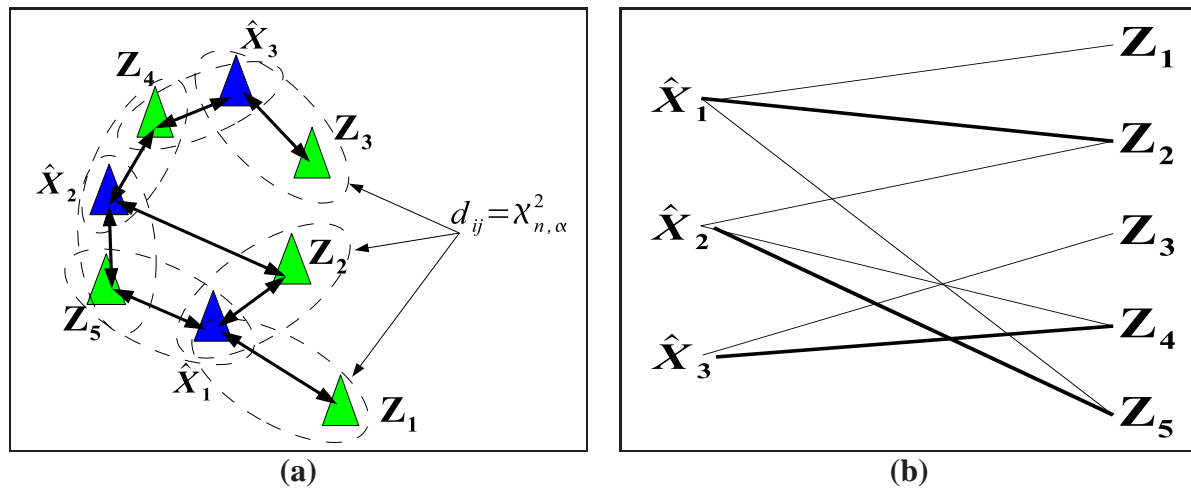


Figure 2.14: Exemple de mises en correspondance ambiguës : (a) représentation de la carte estimée et des mesures réalisées, (b) graphe des associations individuelles (les associations exactes sont représentées par des traits en gras.)

La carte présentée sur la figure 2.14 (a) est un exemple de mise en correspondance délicate. Les associations individuelles entre les estimations et les mesures sont données sur la figure 2.14 (b). Dans cet exemple, les valeurs des distances NIS entre les amers estimés et mesurés sont semblables. Par exemple, les mesures Z_3 à Z_5 sont toutes associées avec les estimations \hat{X}_1 à \hat{X}_3 . L'emploi de la méthode des plus proches voisins entraînerait des associations erronées.

Pour résoudre de telles ambiguïtés, l'algorithme JCBB combine deux outils : les probabilités jointes et le parcours intégral d'un arbre d'interprétation (cf. la figure 2.15).

2.4.4.1 Définition d'une probabilité jointe

La probabilité est dite «jointe» lorsque son calcul intègre les corrélations entre deux éléments. Désormais, nous n'allons plus calculer uniquement une probabilité de mise en correspondance entre un seul amer de la carte estimée et une mesure mais entre plusieurs amers avec plusieurs mesures. Un tel calcul permet donc de prendre en compte les corrélations existantes les positions estimées et définies dans la matrice de covariance du système, \mathbf{P} .

Dans la figure 2.14, les distances NIS individuelles ont prouvé que les mesures Z_1 et Z_2 s'associent respectivement avec les amers \hat{X}_1 et \hat{X}_2 . Il est possible de calculer si ces deux associations ont une compatibilité jointe. Les mesures Z_i sont supposées indépendantes entre elles. Le vecteur d'état et la matrice de covariance de ces mesures s'écrivent donc :

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \end{bmatrix} \text{ et la matrice de covariance : } \mathbf{R} = \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix} \quad (2.44)$$

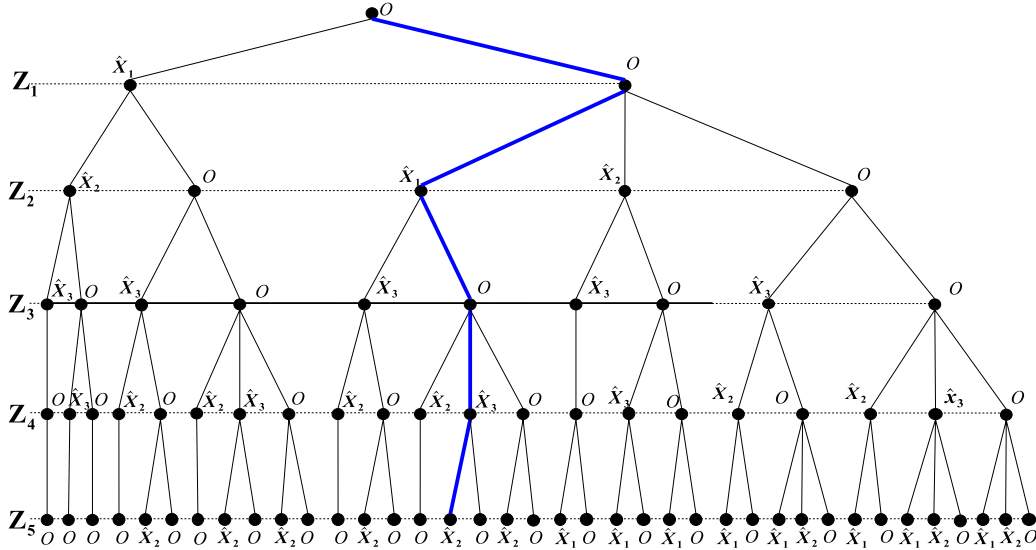


Figure 2.15: Arbre d'interprétation de l'exemple présenté sur la figure 2.14 avec les traits bleus représentant les mises en correspondance exactes et les traits noirs toutes les hypothèses d'association possibles.

Les observations des amers $\hat{\mathbf{X}}_1$ et $\hat{\mathbf{X}}_2$ sont prédites telle que :

$$\hat{\mathbf{Z}} = \begin{bmatrix} \mathbf{h}_1(\hat{\mathbf{X}}_1) \\ \mathbf{h}_2(\hat{\mathbf{X}}_2) \end{bmatrix} \text{ et la matrice de covariance : } \mathbf{P}_Z = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \cdot \mathbf{P} \cdot \begin{bmatrix} H_1 \\ H_2 \end{bmatrix}^T \quad (2.45)$$

La matrice \mathbf{P} correspond à la matrice de covariance du système. Les matrices H_1 et H_2 sont les jacobienes des fonctions d'observation \mathbf{h}_1 et \mathbf{h}_2 . A partir des variables définies par les équations (2.44) et (2.45), l'innovation ν entre ces deux ensembles peut être calculée avec une matrice de covariance \mathbf{S} :

$$\nu = \mathbf{Z} - \hat{\mathbf{Z}} \text{ et la matrice de covariance : } \mathbf{S} = \mathbf{R} + \mathbf{P}_Z \quad (2.46)$$

Ensuite, un test du χ^2 est appliqué sur la distance NIS (équation 2.31) tel que :

$$\nu^T \cdot \mathbf{S}^{-1} \nu < \chi_{4,0.95}^2 \quad (2.47)$$

Au final, ce test permet de conclure sur la compatibilité jointe (et non individuelle) des mises en correspondance des deux estimations avec les deux mesures.

2.4.4.2 Algorithme de type *Branch and Bound*

Cet algorithme parcourt dans son intégralité l'arbre de décision (cf. la figure 2.15) à la recherche de la meilleure combinaison d'associations, entre les observations prédites et les mesures, d'un point de vue statistique. Cette combinaison la plus probable est également appelée “*maximum clique*”. De surcroît, dans l'algorithme JCBB, le parcours de cet arbre est réalisé

en calculant les probabilités jointes entre les ensembles d'amers considérés. Le résultat obtenu prend donc en compte l'intégralité de la carte estimée avec les corrélations entre les différents amers.

Sur la figure 2.15, considérons la branche située à l'extrémité gauche du graphique. La mesure \mathbf{Z}_1 s'associe individuellement avec l'amer $\hat{\mathbf{X}}_1$. Avec cette association validée, la branche de l'arbre peut être "descendue". La mesure \mathbf{Z}_2 s'associe individuellement avec l'amer $\hat{\mathbf{X}}_2$. La compatibilité jointe entre ces deux associations est calculée. Si le test du χ^2 est correct pour cette compatibilité, la mesure \mathbf{Z}_3 est examinée. Si le test n'est pas bon, la branche est arrêtée. Une autre ramification de l'arbre d'interprétation est suivie depuis là où la mesure \mathbf{Z}_2 n'est pas associée. Dans un contexte SLAM, une mesure non associée est considérée comme un nouvel amer de l'environnement à ajouter sur la carte estimée.

Au final, l'hypothèse retenue est celle qui correspond au maximum d'associations entre amers et mesures. Graphiquement, elle correspond à la branche de l'arbre d'interprétation qui a été parcourue dans toute sa longueur avec la probabilité jointe la plus grande (branche représentée en bleu sur la figure 2.15).

2.4.5 Comparaison des algorithmes PPV et JCBB

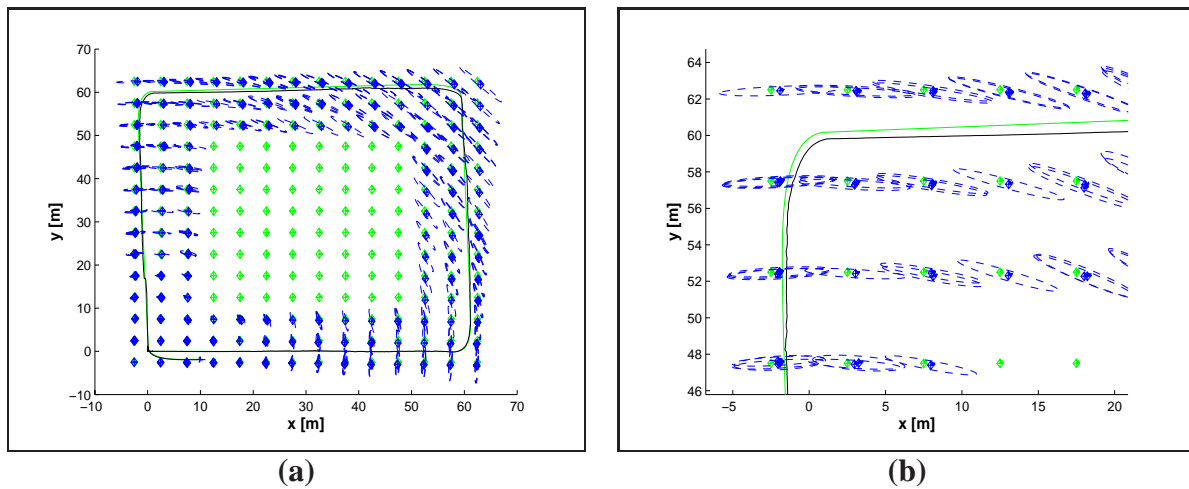


Figure 2.16: Résultats de simulation avec la méthode JCBB : carte estimée (bleu), trajectoire estimée du véhicule (noire) et vérité de terrain (vert) : (a) carte finale dans son ensemble et (b) zoom sur la partie en haut à gauche de la carte.

Une comparaison entre l'algorithme JCBB et celui des plus proches voisins est réalisée avec une simulation EKF-SLAM *Worldcentric* (cf. la figure 2.16). Le robot suit une trajectoire rectangulaire longue de 240 m à une vitesse constante de 1 m s^{-1} . L'environnement est représenté par une grille d'amers avec un maillage de 5 m. Les valeurs des bruits sur les données proprioceptives et extéroceptives sont présentées dans le tableau 2.1. La portée du télémètre est fixée à 10 m ce qui garantit que le robot ne reverra les premiers amers observés que lors de la fermeture de la trajectoire.

	Valeurs du bruit			
	Télémètre Laser		Commande	
	σ_ρ [m]	σ_β [rad]	σ_v [ms ⁻¹]	σ_φ [rad]
Simulation	0.1	$\pi/180$	0.3	$5.\pi/180$

TABLE 2.1: Valeurs des bruits sur les données proprioceptives et extéroceptives.

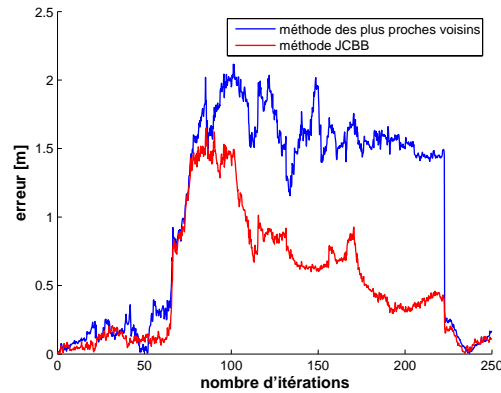


Figure 2.17: Comparaison des évolutions des erreurs sur la position du véhicule avec les différentes méthodes d'association de données : algorithme JCBB (rouge) et algorithme PPV (bleu).

La figure 2.17 présente l'évolution des erreurs sur la position du véhicule en fonction de la méthode d'association de données utilisée. Même si, pour les deux applications, le véhicule réalise une fermeture de la trajectoire, l'emploi de la méthode JCBB améliore la précision des résultats. Cependant, l'emploi de cette méthode possède un inconvénient. La carte estimée par le système (cf. la figure 2.16 (a)) contient plusieurs amers cartographiés correspondant à un unique amer sur la carte réelle (cf. la figure 2.16 (b)). En effet, lors du parcours de l'arbre d'interprétation, la meilleure solution avec le calcul des probabilités jointes peut correspondre à une branche où toutes les associations individuelles d'amer ne sont pas prises en compte. De nouveaux amers sont créés sur la carte bien que leurs mesures furent associées individuellement avec des amers de la carte estimée.

La faible erreur sur la position du véhicule confirme la précision des mises en correspondance d'amers avec la méthode JCBB. Cependant, pour obtenir une carte moins dense en amers, cette méthode est couplée avec une approche de gestion de carte présentée dans le paragraphe suivant.

2.4.6 Gestion de la carte

Les résultats de simulation (cf. la figure 2.17) soulignent que l'emploi de méthodes d'association de données évoluées ne permet pas d'éviter les cartographies incorrectes d'amers. Une méthode très simple mais efficace permet une diminution de ce phénomène. Elle repose sur une

gestion des amers de la carte estimée (cf. l'algorithme 2.2).

Cette section développe le processus permettant de retenir ou non les amers de la carte estimée en fonction de critères. Ensuite, le mécanisme de suppression d'un amer du vecteur d'état du système et de la matrice de covariance associée est introduit.

2.4.6.1 Présentation de la méthode

La méthode de gestion de la carte est similaire à celle présentée dans l'article [53]. Dès lors qu'un nouvel amer est cartographié, deux compteurs sont créés pour permettre sa gestion. Le premier, $C_{association}$, représente le nombre d'associations de l'amer à une mesure et le deuxième compteur, $C_{iteration}$, correspond au nombre d'itérations du filtre effectuées depuis la cartographie de celui-ci. Ensuite, pour déterminer si un amer doit être éliminé, les valeurs de ces compteurs sont comparées à des seuils prédéfinis :

- un minimum d'associations consécutives, $seuil_{ac}$;
- un maximum d'itérations à attendre, $seuil_{iteration}$;
- un minimum d'association à réaliser avant d'atteindre le seuil précédent, $seuil_{amin}$.

Si le premier seuil n'est pas atteint, l'amer est supprimé. Si le nombre d'associations, $C_{association}$, est plus faible que le nombre minimum d'associations ($seuil_{amin}$) et si le nombre maximum d'itérations ($seuil_{amin}$) est atteint, l'amer est supprimé. Dans tous les autres cas, l'amer est définitivement cartographié. Cette méthode est présentée par l'algorithme 2.2.

2.4.6.2 Elimination d'un amer du vecteur d'état et de la matrice de covariance

Dissanayake et al. prouvent dans l'article [54] qu'il est possible d'éliminer des amers de la carte estimée par le système. Cette opération n'affecte en rien la consistance des estimations. Elle consiste à enlever les informations relatives à l'amer éliminé dans le vecteur d'état et dans la matrice de covariance (cf. sur la figure 2.18). Elle correspond à la suppression de la ligne et de la colonne correspondante dans la matrice de covariance. Toutes les corrélations entre cet amer, la carte estimée et la position du véhicule, sont donc éliminées.

2.4.6.3 Résultats de simulation

La simulation, présentée dans le paragraphe 2.4.5, est utilisée pour comparer les résultats obtenus avec ou sans l'application de la méthode de gestion de la carte.

La comparaison des évolutions des erreurs est présentée sur la figure 2.19. Les deux courbes se superposent mise à part durant des courtes périodes. Par exemple, aux alentours de la centième itération, le système utilisant la gestion de la carte est moins précis. L'algorithme a dû supprimer des amers qui apportaient une information supplémentaire à la localisation du véhicule. Au contraire, avant la fermeture de la trajectoire du véhicule, la gestion de la carte a dû permettre la suppression d'informations redondantes sur la carte estimée. En conclusion, l'algorithme de gestion des amers de la carte estimée ne modifie que superficiellement les résultats en terme de précision de la position du robot.

En revanche, l'observation de la figure 2.20 démontre la pertinence de la gestion de la carte dans l'élimination des fausses détections. En comparaison avec la méthode utilisant uniquement

Algorithme 2.2 : Algorithme de gestion de la carte : Exemple avec un amer A_i

```

Amer  $A_i$  ajouté à la carte
 $c_{association} \leftarrow 0$ 
 $c_{iteration} \leftarrow 0$ 

tant que Amer non retenu faire
   $c_{iteration} \leftarrow c_{iteration} + 1$ 
  si (Amer  $A_i$  observé) alors
     $c_{association} \leftarrow c_{association} + 1$ 
    si ( $c_{association} \geq \text{seuil}_{iteration}$ ) alors
      Amer  $A_i$  retenu
    fin si
  sinon
    si ( $c_{iteration} \geq \text{seuil}_{iteration}$ ) alors
      si ( $c_{association} < \text{seuil}_{amin}$ ) alors
        Amer  $A_i$  éliminé de la carte
      fin si
    sinon si ( $c_{association} < \text{seuil}_{ac}$ ) alors
      Amer  $A_i$  éliminé de la carte
    sinon
      Amer en attente
    fin si
  fin si
fin tant que

```

l'association de données JCBB (cf. la figure 2.16), la carte obtenue ne comporte désormais que très peu d'amers doublés.

2.4.6.4 Conclusion

Cette méthode possède deux avantages. Premièrement, comme le nombre d'amers sur la carte estimée diminue, le temps de calcul diminue également (surtout pendant la phase de mise à jour du filtre de Kalman). Deuxièmement, comme le soulignent les résultats de simulation, cette méthode permet de réduire de manière efficace les fausses détections d'amers sans modifier les résultats sur l'estimation de la position du véhicule.

2.5 SLAM avec cartes hétérogènes : résultats expérimentaux

La méthode globale présentée dans ce chapitre est appliquée sur deux séries de données expérimentales. Les mesures ont été réalisées à l'aide d'un véhicule expérimental urbain (cf. l'annexe E).

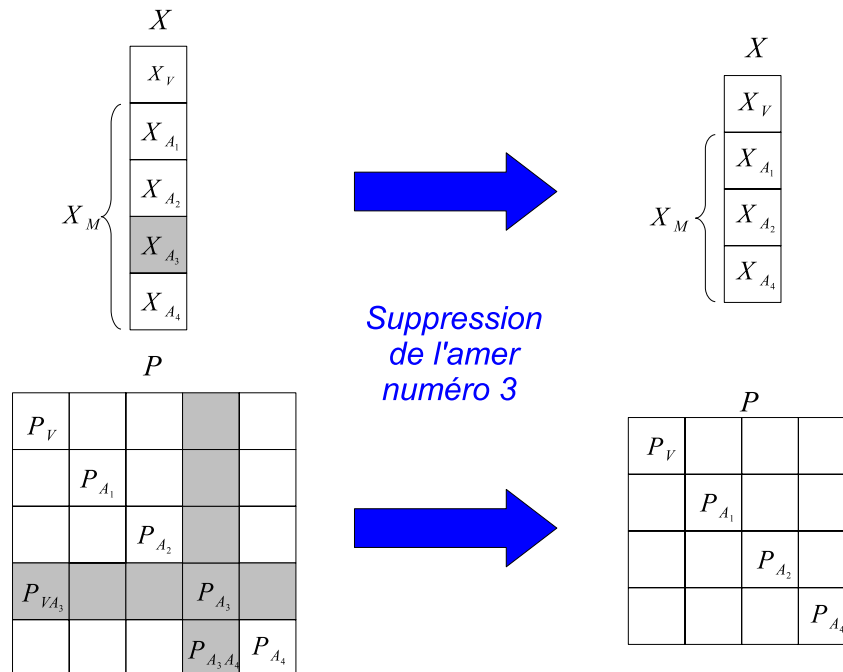


Figure 2.18: Schéma représentant l'élimination d'un amer d'un vecteur d'état X et de la matrice de covariance associée P .

La première série de mesures a été réalisée dans un environnement intérieur correspondant à un hall d'aéroport. La seconde série de mesures a été recueillie dans un parc privé composé de bâtiments et d'arbres. L'objectif est de valider l'utilisation jointe d'une cartographie hétérogène de l'environnement avec la méthode d'association JCBB.

2.5.1 Environnement intérieur : hall d'aéroport

Dans cette expérimentation, la seule vérité de terrain disponible correspond au plan de l'aéroport (cf. la figure 2.21). Il sera donc impossible de quantifier les résultats en termes de précision. Cependant, l'analyse de la forme de la carte estimée permettra de qualifier les résultats obtenus en comparant la carte estimée au plan de l'aéroport.

L'affichage des données télémétriques et proprioceptives sans aucun filtrage (cf. la figure 2.22 (a)) démontre la nécessité d'une application SLAM dans un tel contexte. La trajectoire du véhicule reconstruite avec les données odométriques ne boucle pas, alors qu'en réalité, le point de départ et d'arrivée du véhicule sont confondus. De plus, la comparaison de la carte obtenue avec la superposition des balayages télémétriques et le plan de l'aéroport n'ont aucun point commun. L'application de notre approche EKF-SLAM sur ce jeu de données permet de corriger ces deux problèmes (cf. la figure 2.22 (b)).

Premièrement, la comparaison de cette carte et du plan de l'aéroport permet de faire ressortir des points communs. Certains alignements de piliers ressortent très clairement dans la carte estimée ainsi que les murs permettant de discerner certaines entrées de l'aéroport. En outre,

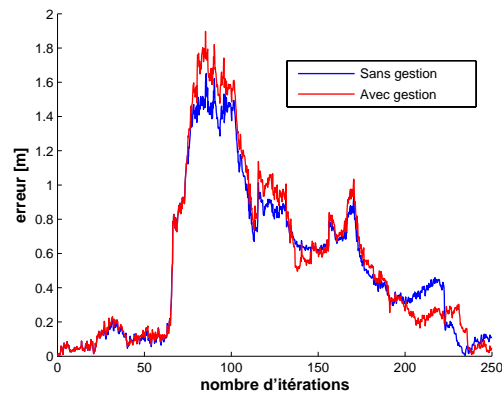


Figure 2.19: Courbes d'erreur sur la position du véhicule avec une méthode d'association de données JCBB avec ou sans gestion de la carte : algorithme avec la gestion (rouge) et algorithme sans gestion (bleu).

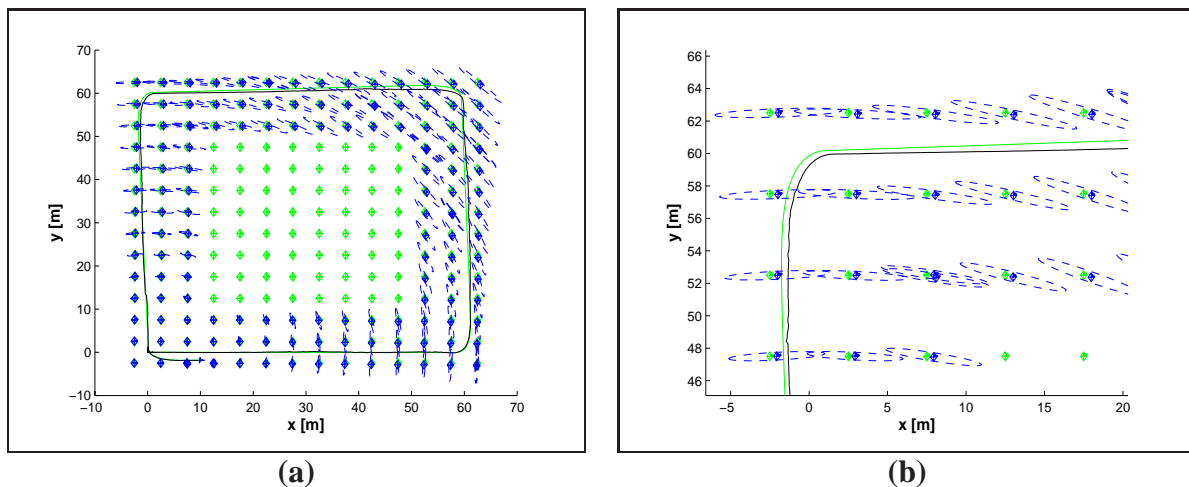


Figure 2.20: Résultats de simulation avec la méthode JCBB couplée à une approche de gestion de la carte : carte estimée (bleu), trajectoire estimée du véhicule (noire) et vérité de terrain (vert) : (a) carte finale dans son ensemble et (b) zoom sur la partie en haut à gauche de la carte.

d'autres éléments, non visibles sur le plan, sont détectés. Les aménagements de certaines parties de ce hall apparaissent clairement sur la carte estimée (cf. les photographies de l'annexe E).

Deuxièmement, la trajectoire du véhicule est bouclée et elle est similaire à celle réalisée réellement. Ce bouclage de la trajectoire du véhicule permet de mettre en évidence la convergence du système. La figure 2.23 présente l'évolution de l'incertitude estimée dans la matrice de covariance, \mathbf{P} , sur la position du véhicule. Deux diminutions sont remarquées. La première correspond au début du retour du véhicule lorsqu'il commence à réobserver des éléments déjà cartographiés de l'environnement. Puis, lorsque la trajectoire se boucle, l'incertitude redevient

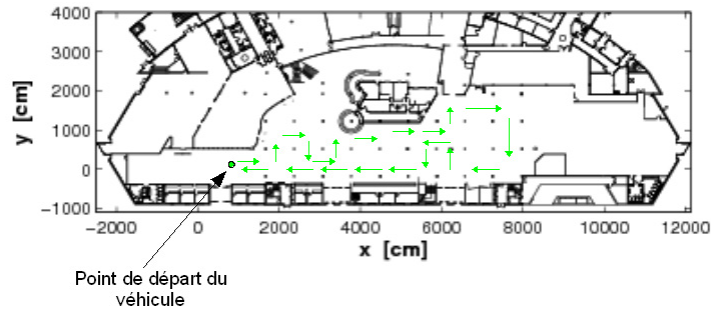


Figure 2.21: Plan du hall d'entrée de l'aéroport d'Aulnat et trajectoire empruntée par le véhicule représentée par des flèches de couleurs vertes.

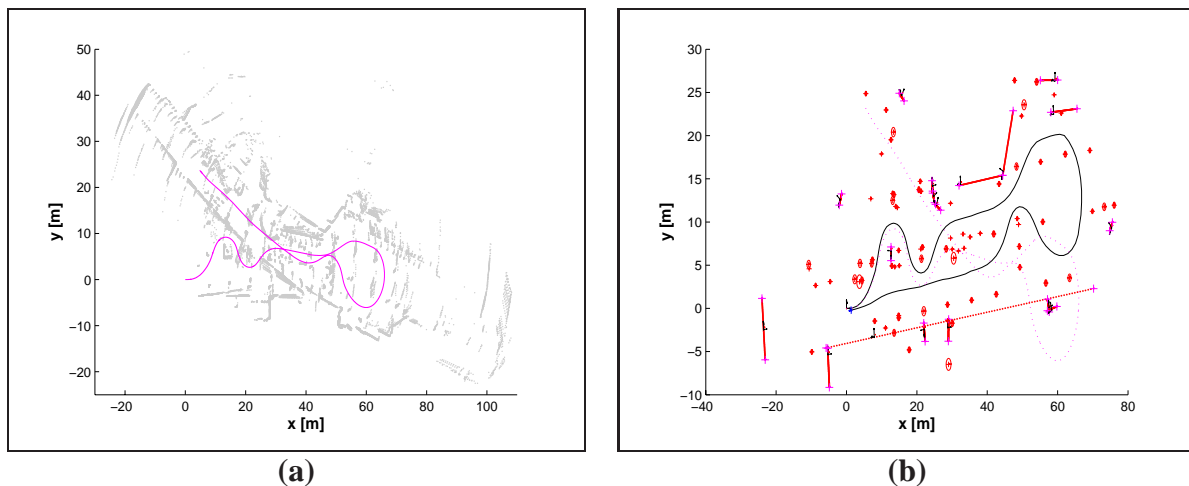


Figure 2.22: Expérimentation dans l'aéroport d'Aulnat : (a) données brutes sans aucun traitement et (b) carte estimée avec l'emploi d'un filtre EKF-SLAM couplée à une cartographie hétérogène. La trajectoire obtenue à partir des données odométriques est représentée en rose et la trajectoire reconstruite en noire.

quasiment nulle sur sa position.

Ce phénomène de convergence est étudié dans le prochain chapitre ainsi que les problèmes de consistance relatifs à l'emploi d'un filtre de Kalman étendu dans une application SLAM.

2.5.2 Environnement extérieur : Parc du Cemagref

Ces données ont été recueillies dans un parc privé du Cemagref. La vue aérienne et la trajectoire réelle du véhicule sont présentées sur la figure 2.24 (a). C'est un environnement semi-structuré composé de bâtiments, de clôtures et d'arbres. La vérité de terrain est fournie par un capteur DGPS embarqué sur le véhicule.

La difficulté avec ces données expérimentales correspond au relief du terrain. Même si cela est difficilement observable avec la vue aérienne, le terrain de l'expérimentation n'est pas plat.

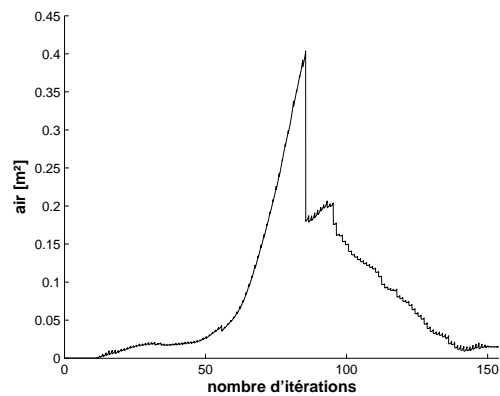


Figure 2.23: Evolution de l'air de l'ellipse associée à l'incertitude estimée sur la position du robot (x et y).

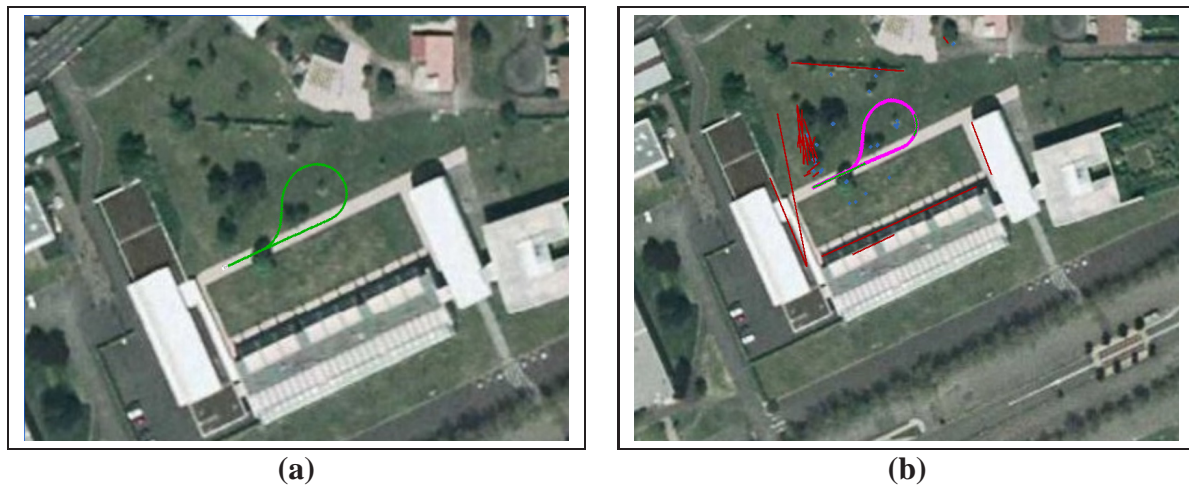


Figure 2.24: Vue aérienne de l'expérimentation réalisée au Cemagref : (a) vue de l'environnement expérimental avec la trajectoire réelle du véhicule en vert (DGPS) et (b) vue aérienne de la carte estimée avec les droites en rouge, les cercles en bleu et la trajectoire estimée du véhicule en rose.

Hors, le système présenté repose sur une cartographie en deux dimensions. Ce relief, avec l'emploi d'un télémètre une nappe, provoque de nombreuses fausses détections d'amers.

La carte estimée par le système est donnée sur la figure 2.25 (a) et superposée sur la vue aérienne de l'environnement sur la figure 2.24 (b). La trajectoire du véhicule est bouclée. Le système a corrigé les erreurs sur l'odométrie représentées par la trajectoire rose sur la figure 2.25 (a). L'évolution de l'erreur sur la position du véhicule est représentée sur la figure 2.25 (b). Cette erreur reste inférieure à un mètre. Cependant, la valeur de l'erreur atteint 70 cm avant la fermeture de la trajectoire. Ce phénomène représente la limite d'une cartographie en deux dimensions dans un environnement avec du relief. La multitude de droites détectées dans la partie

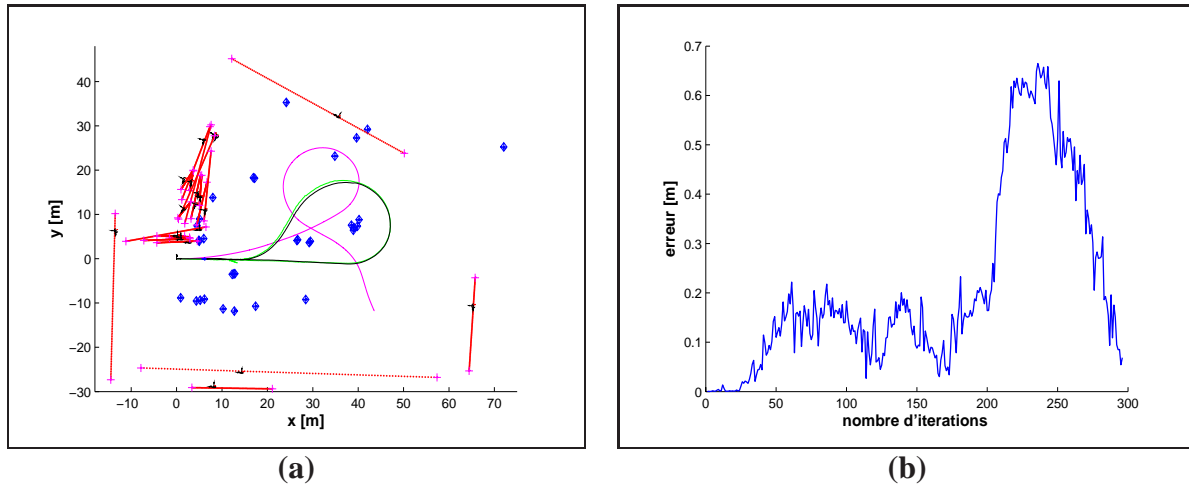


Figure 2.25: Résultats de la manipulation dans un environnement extérieur : (a) carte estimée par le système avec les droites représentées en rouge, les centres des cercles en bleu, la trajectoire du véhicule estimée en noir, non corrigée en rose et réelle en vert et (b) évolution de l'erreur sur la position du véhicule.

supérieure gauche de la carte (cf. les figures 2.25 (a) et 2.24 (b)) correspond au relief du terrain. Ces fausses détections d'amer traduisent l'augmentation de l'erreur sur la position du véhicule à la fin de la trajectoire. Cependant, le système ne diverge pas et ferme la trajectoire du véhicule grâce à l'emploi d'une méthode d'association de données performante. Ces résultats dans un contexte délicat prouvent la robustesse du système de perception présenté dans ce paragraphe.

2.6 Conclusion

Les résultats expérimentaux soulignent l'intérêt de l'emploi d'un filtre EKF-SLAM couplé à une représentation hétérogène de l'environnement. Les cartes estimées obtenues sont représentatives de l'environnement visité. D'un simple coup d'œil, dans le cadre de l'expérimentation dans le hall de l'aéroport, le lecteur peut discerner certains murs du bâtiment ainsi que les piliers le constituant.

La diminution de l'erreur sur la position du véhicule, lors de la fermeture de sa trajectoire, est notable. Cependant, est-elle cohérente avec l'incertitude définie dans la matrice de covariance ? Dans le chapitre 3, une étude de ce phénomène est développée. Elle permet de répondre à cette interrogation.

A la vue des résultats, cette erreur reste tout de même importante. Elle atteint une valeur de 0.4 m. Est ce possible d'améliorer la précision du système tout en conservant un filtrage EKF-SLAM ? A l'instar d'un individu, est-il possible d'employer des données existantes sur l'environnement comme des plans ou des cartes de l'environnement ? L'approche SLAM-contraint, présentée dans le chapitre 4, visera à répondre à cette autre interrogation.

Chapitre 3

Etude de la convergence et la consistance des estimations du filtre polynomial de Kalman

3.1 Introduction

Ce chapitre propose l'étude de la consistance des estimations de l'approche EKF-SLAM ainsi que l'application d'un filtre polynomial de Kalman dans un contexte SLAM.

Dans la littérature, des travaux présentant les problèmes d'estimation rencontrés lors de l'application du filtre EKF-SLAM existent mais peu d'articles, tels [11] et [83], proposent une explication précise de ces problèmes de cohérence des estimations. Dans ce chapitre, après une définition de la consistance d'un estimateur, une étude approfondie de ces phénomènes observés lors de l'application d'un EKF-SLAM est proposée. Pour se faire, une analyse détaillée des équations du filtre permet de décrire les limites de convergence de la matrice de covariance, P_M , associée à la carte estimée composée d'amer de type point ou droite. Ensuite, le comportement des estimations de l'EKF-SLAM est détaillé lors de la fermeture de la trajectoire du véhicule. Cette étude met en évidence le problème de consistance des estimations obtenues avec le filtre EKF-SLAM.

Les solutions existantes dans la littérature, fondées sur la théorie de Kalman, ne permettent pas de résoudre efficacement ce problème. Le filtre UKF [88] réalise une meilleure approximation des équations non-linéaires du système mais ne résout pas le problème de cohérence des estimations lorsque l'incertitude sur l'orientation du véhicule n'est plus négligeable. L'application d'une méthode de cartes locales [146] permet de résoudre en partie ce problème. En effet, dès que l'incertitude sur l'orientation du véhicule devient trop importante, une nouvelle carte est initialisée. En revanche, si l'incertitude initiale sur l'orientation est importante, le problème de consistance des estimations est toujours observé.

Pour répondre à ce problème de consistance des estimations, mes travaux proposent l'application d'une version polynomiale du filtre de Kalman proposée par Germani et al. [71] dans un contexte SLAM. Ce filtre, *Polynomial Extended Kalman Filter* (PEKF) en anglais, est employé pour l'estimation de paramètres d'un système stochastique non-linéaire. Les premières applications de ce filtre sur des systèmes non-linéaires [70] [107] ont permis d'améliorer la précision

des estimations en comparaison à une approche classique EKF. Ce chapitre essaie d'apporter une réponse à la question suivante : « Est ce que le filtre PEKF permet de résoudre les problèmes inhérents à l'application EKF-SLAM ? »

Pour répondre à cette question, une définition des différentes équations non-linéaires régissant le système accompagnée des différents concepts de base du filtre, nommée PEKF-SLAM, est proposée. Elle est accompagnée de résultats de simulations et d'expérimentations permettant de mettre en évidence, d'une part, les problèmes de consistance des estimations du filtre EKF et d'autre part, les améliorations apportées par l'approche PEKF-SLAM.

Finalement, cette étude de la consistance des estimations synthétise les différents problèmes rencontrés avec l'application EKF-SLAM. Elle fournit les outils pour quantifier les résultats obtenus avec l'approche de SLAM-contraint présentée dans le chapitre 4. Elle permet également une meilleure critique des résultats obtenus via les différents théorèmes proposés dans la suite de ce chapitre.

3.2 Consistance et convergence des estimations de l'EKF-SLAM

3.2.1 Définition de la consistance

Par définition, un estimateur d'état est dit consistant [12] si :

- les erreurs d'estimation sont à moyenne nulle, en d'autres termes, les estimations obtenues ne doivent pas être biaisées ;
- la variance des erreurs est inférieure ou égale à celle estimée par le filtre.

En résumé, les différences, entre les positions réelles et estimées du véhicule et des amers, doivent être égales ou inférieures aux valeurs estimées des incertitudes définies dans la matrice de covariance. La consistance est l'un des premiers critères pour évaluer la performance d'un estimateur. Si un estimateur ne répond pas à ce critère, il est dit inconsistant. La précision de ces estimations ne pourra être garantie et l'estimateur ne sera donc pas fiable. De surcroît, la convergence des estimations du filtre vers leurs valeurs réelles ne sera plus garantie également.

Pour évaluer la consistance des estimations des méthodes proposées, un test statistique reposant sur une erreur normalisée au carré est employé. De plus, trois théorèmes de convergence des estimations d'une approche SLAM fondée sur un filtrage de Kalman sont proposés. Ils permettent de conclure ou non sur la convergence des méthodes proposées dans ce document.

3.2.1.1 Définition du test statistique

Critère de consistance d'un estimateur :

Il existe plusieurs tests statistiques permettant d'évaluer la consistance d'un filtre estimateur [12]. Ils reposent tous sur le calcul de l'erreur entre une estimation et la valeur réelle du vecteur de paramètres. Cette erreur à l'instant k s'écrit :

$$\tilde{\mathbf{X}}(k|k) = \mathbf{X}(k) - \hat{\mathbf{X}}(k|k) \quad (3.1)$$

Dans cette expression, $\tilde{\mathbf{X}}(k|k)$ représente l'estimation du filtre avec une matrice de covariance associée $\mathbf{P}(k|k)$ et $\mathbf{X}(k)$ représente la valeur réelle du vecteur de paramètres. Pour que ce test soit fiable, cette valeur réelle doit être parfaitement connue, comme, par exemple, lors de l'utilisation de données simulées. Dans notre cadre expérimental, la vérité de terrain fournie par un capteur GPS est bruitée. Cette imprécision ne doit pas être négligée sinon le test de la consistance ne sera pas pertinent.

Le test statistique utilisé repose sur la définition d'une erreur normalisée d'estimation au carré¹ :

$$\epsilon(k) = (\tilde{\mathbf{X}}(k|k))^T \cdot \mathbf{P}(k|k)^{-1} \cdot (\tilde{\mathbf{X}}(k|k)) \quad (3.2)$$

Cette distance (3.2) est similaire au carré d'une distance de Mahalanobis entre la valeur réelle et la valeur estimée d'un paramètre. L'objectif de ce test est de déterminer si l'erreur $\tilde{\mathbf{X}}(k|k)$ correspond aux incertitudes définies dans la matrice de covariance $\mathbf{P}(k|k)$. En d'autres termes, si la définition de la consistance d'un filtre estimateur énoncée au début du paragraphe est vérifiée. Pour prouver ceci, un test du χ^2 est appliqué sur l'erreur NEES¹, $\epsilon(k)$:

$$\epsilon(k) \leq \chi_{n,\alpha}^2 \quad (3.3)$$

Le coefficient n représente le nombre de degrés de liberté du vecteur d'état \mathbf{X} ($n = \dim(\mathbf{X})$). Le chiffre α permet de définir le seuil d'acceptation de l'hypothèse (3.3). Si le test est positif, les estimations sont dites consistantes, ne signifiant pas obligatoirement précises. Dans le cas contraire, les estimations ne sont plus consistantes. Leur convergence et leur précision ne sont plus garanties.

Test statistique fondé sur une simulation de Monte Carlo :

Le test statistique (équations (3.2) et (3.3)) peut s'effectuer à partir d'une seule simulation. Cependant, un unique test statistique reposant sur la distance NEES¹ est une mesure inadéquate de la consistance d'un filtre. En effet, la séquence d'erreurs $\tilde{\mathbf{X}}$ est corrélée et ne suit pas une distribution du χ^2 . Un test de consistance sur une seule séquence de simulation peut être positif alors que, pour N séquences, le test sera négatif. Inversement, une seule séquence de simulation peut révéler une inconsistance des résultats d'un estimateur alors que N séquences peuvent prouver le contraire. Pour citer Bar-Shalom [12] : « *l'examen exhaustif d'un filtre non-linéaire nécessite de multiples séquences de Monte Carlo pour déterminer si ce filtre est consistant* ».

Des simulations de Monte Carlo fournissent N échantillons indépendants $\epsilon_i(k)$, $i = 1, \dots, N$ de la variable aléatoire $\epsilon(k)$. La moyenne de ces échantillons s'écrit donc :

$$\bar{\epsilon}(k) = \frac{1}{N} \sum_{i=1}^N \epsilon_i(k) \quad (3.4)$$

La variable $N \cdot \bar{\epsilon}(k)$ possède une densité de probabilité du χ^2 avec $(N \cdot n)$ degrés de liberté. Pour des séquences de Monte Carlo, le test du χ^2 , équation (3.3), s'écrit donc :

$$\bar{\epsilon}(k) \leq \frac{\chi_{N \cdot n, \alpha}^2}{N} \quad (3.5)$$

1. Normalised Estimation Error Squared (NEES).

Le test statistique, utilisé dans ce manuscrit, pour statuer sur la consistance des estimations vient d’être défini. Pour en comprendre l’intérêt, des simulations avec un scénario simple de SLAM sont proposées.

3.2.1.2 Application du test statistique sur un exemple simple

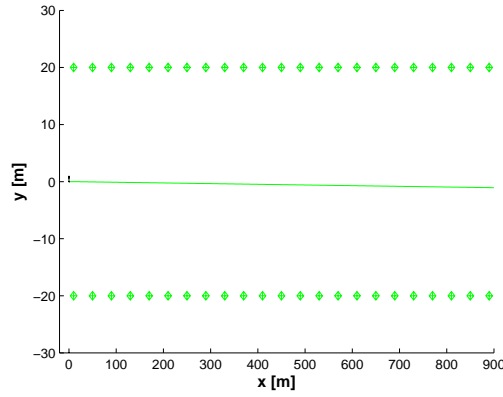


Figure 3.1: Carte et trajectoire simulées : le véhicule avance en ligne droite, à une vitesse de 3 m.s^{-1} , entre deux rangées d’amers espacés tous les 40 m.

La figure 3.1 présente un exemple simple d’application de l’EKF-SLAM *WorldCentric* (cf. l’annexe F). Plusieurs simulations sont réalisées en fonction des données extéroceptives :

- la portée du télémètre est nulle. Aucune correction n’est réalisée par le filtre ;
- la portée du télémètre est fixée à 30 m. Le véhicule n’observe que deux amers à chaque itération du filtre.

Les données proprioceptives et extéroceptives sont bruitées avec les valeurs présentées dans le tableau 3.1.

Simulations sans aucun filtrage

Dans ces simulations, le véhicule ne réalise aucune mesure extéroceptive. Sa position n’est donc pas corrigée par le filtre de Kalman. La figure 3.2 (a) représente l’évolution de la distance moyenne NEES, $\bar{\epsilon}$, sur la position du véhicule. Elle est calculée à partir de 20 simulations de

Valeurs des bruits				
Télémètre laser		données de contrôle		
	σ_ρ [m]	σ_β [rad]	σ_v [ms^{-1}]	σ_φ [rad]
Simulation	0.1	$\pi/180$	0.02	$3.\pi/180$

TABLE 3.1: Bruits sur les données proprioceptives et extéroceptives.

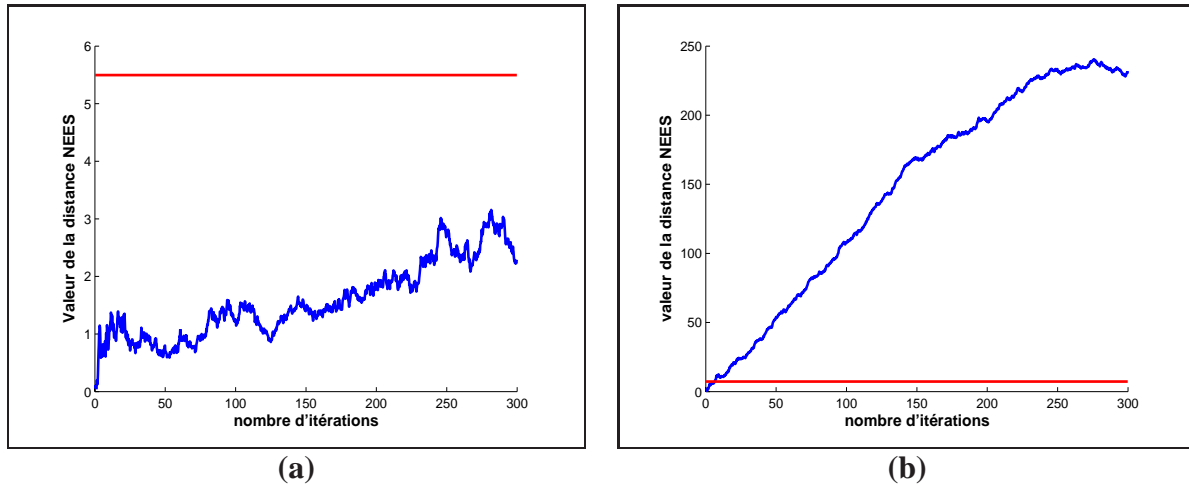


Figure 3.2: Evolution de la distance statistique NEES calculée par rapport à la position du véhicule. Ces résultats sont obtenus avec 20 séquences de Monte Carlo de la simulation présentée sur la figure 3.1. Le seuil de consistance est défini par la distribution du χ^2 et représenté en rouge. La figure (a) correspond à une simulation où un bruit gaussien est ajouté aux mesures proprioceptives et la figure (b) à une simulation où le bruit gaussien est ajouté à des mesures proprioceptives biaisées.

Monte Carlo lorsque les mesures proprioceptives sont seulement bruitées. La valeur de la distance reste inférieure au seuil du χ^2 pendant toute la simulation. Les résultats du filtre dans cette configuration sont consistants, même si ils ne sont pas précis. Lorsque un biais est ajouté sur la vitesse de déplacement du véhicule, la valeur de $\bar{\epsilon}$ dépasse très rapidement le seuil (cf. la figure 3.2 (b)). Les estimations du filtre sont inconsistantes car les mesures proprioceptives sont biaisées. Elles ne répondent pas aux critères d'utilisation d'un filtre de Kalman.

Simulations avec une correction du filtre

Cette simulation présente les résultats lorsque le robot réalise des observations de l'environnement. Seul un bruit gaussien (et non biaisé) est ajouté aux mesures extéroceptives et proprioceptives. Les résultats (cf. sur la figure 3.3) introduisent les problèmes rencontrés lors de l'utilisation d'un filtre EKF pour résoudre le problème du SLAM. Comme le robot ne réalise aucune boucle où il pourrait observer de nouveau certains amers, l'incertitude sur les états estimés du véhicule ne devrait cesser de croître. Or, l'évolution de l'incertitude sur l'orientation du véhicule (cf. la figure 3.3 (c)) augmente lors des 100 premières itérations du filtre puis la valeur reste plus ou moins constante. Ce phénomène n'est pas cohérent avec l'augmentation constante de l'erreur sur la position du véhicule (cf. la figure 3.3 (b)). Il se traduit par une valeur de la distance NEES devenant supérieure au seuil du χ^2 (cf. la figure 3.3 (a)). D'après ce résultat et les définitions proposées précédemment, les estimations du filtre EKF-SLAM deviennent incohérentes à partir de la centième itération.

Le problème de consistance observé sur la figure 3.3 a pour origine la linéarisation des équations d'un système SLAM évaluée à partir de valeurs estimées et non exactes (cf. l'annexe

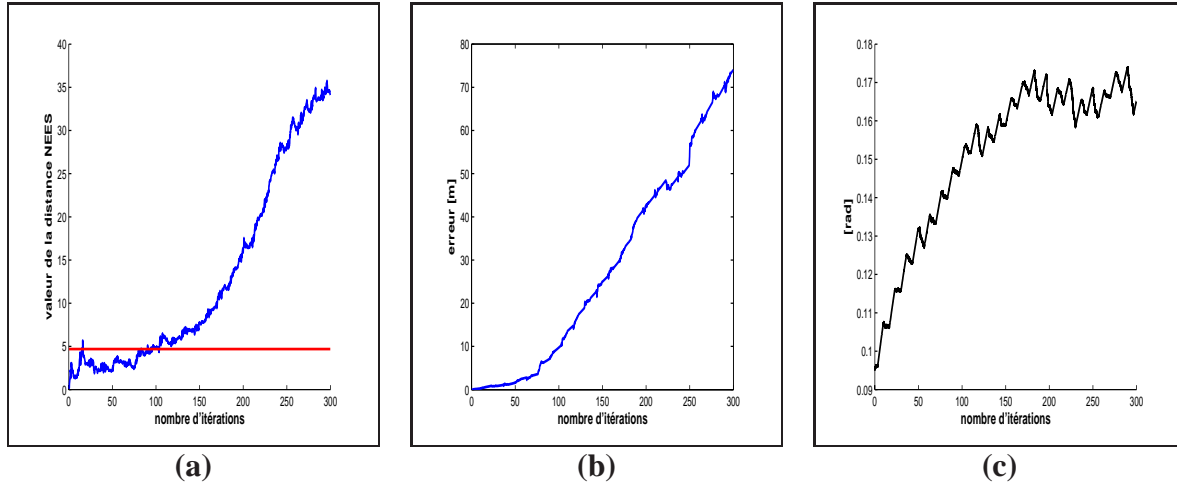


Figure 3.3: Présentation des résultats sur 20 séquences de Monte Carlo de la simulation (cf. sur la figure 3.1) : (a) évolution de la distance NEES, (b) évolution de l'erreur moyenne sur la position du véhicule et (c) évolution de l'incertitude à 3 écart-types sur l'orientation du véhicule.

A). Ce phénomène est détaillé dans la suite de ce chapitre.

3.2.2 Travaux existants

Les premiers travaux, mettant en évidence les problèmes d'inconsistance d'un filtre EKF-SLAM, ont été proposés par Julier et al. [89]. Pour étayer leurs explications, les auteurs proposent un théorème de la consistance fondée sur une analyse des équations de la méthode EKF-SLAM. Ils posent un théorème pour le cas où le véhicule reste stationnaire avec aucun bruit sur les mesures :

« Soit une position du véhicule initialisée avec une covariance non-nulle. Si le véhicule reste stationnaire, l'observation de positions d'amers, initialisées avec la fonction de cartographie, \mathbf{g} , ne doivent pas modifier la valeur des états associés au véhicule. Dans ce cas, la position du véhicule restera donc inchangée si et seulement si :

$$\mathbf{H}_{X_V} - \mathbf{H}_{X_a} \cdot \mathbf{G}_{X_V} = 0 \quad (3.6)$$

\mathbf{H}_{X_V} correspond à la matrice jacobienne de la fonction d'observation \mathbf{h} par rapport à l'état du véhicule \mathbf{X}_V , \mathbf{H}_{X_a} la matrice jacobienne de la fonction d'observation \mathbf{h} par rapport à l'état de l'amer \mathbf{X}_a et \mathbf{G}_{X_V} la matrice jacobienne de la fonction de cartographie d'un amer \mathbf{g} par rapport à \mathbf{X}_V . Cette démonstration s'appuie sur la décomposition du gain de Kalman \mathbf{W} en deux éléments : le gain relatif au véhicule \mathbf{W}_V et celui relatif à l'amer \mathbf{W}_a . Comme le véhicule est stationnaire, \mathbf{W}_V doit rester nul. Cette condition est traduite par l'équation (3.6). Or, lorsque le véhicule est stationnaire et réalise plusieurs observations relatives de la position d'un unique amer, le terme (3.6) n'est pas nul. Une diminution de la variance sur l'orientation du robot est observée. Cette diminution est "artificielle" et entraîne une inconsistance des résultats du filtre. Les mêmes constatations ont été décrites par Castellanos et al. [30]. Les auteurs soulignent

l'importance de la valeur initiale sur la variance de l'orientation du véhicule. Elle doit rester faible pour que les estimations du filtre soient consistantes.

Bailey et al. [11] traduisent les problèmes d'inconsistance de l'EKF-SLAM par un gain d'information erroné. Ce gain est défini par la différence entre les matrices de covariance réelle et estimée :

$$\mathbf{P}(k|k) - \mathbf{P}(k) \geq 0 \quad (3.7)$$

Cette quantité doit être définie semi-positive. Si ce n'est pas le cas, le filtre EKF-SLAM est optimiste. En utilisant cette définition, les auteurs mettent en évidence un phénomène d'inconsistance du filtre EKF-SLAM lors de la fermeture de la trajectoire du véhicule. Les résultats de simulations (50 séquences de Monte Carlo) prouvent que les estimations du filtre deviennent optimistes lors de la fermeture de trajectoire du véhicule. Ce phénomène est observé avec l'emploi du filtre de Kalman mais également avec le filtre UKF, le filtre informatif et le filtre itératif de Kalman.

La conclusion de l'ensemble de ces articles peut se résumer en quelques points :

- les matrices jacobiennes jouent un rôle primordial dans les problèmes de consistance des estimations car leurs calculs reposent sur des paramètres estimés et non exacts ;
- l'initialisation de la variance sur l'orientation du véhicule doit être très faible pour négliger les problèmes de consistance rencontrés avec l'utilisation d'un filtre EKF ;
- l'erreur sur l'orientation du véhicule et sur la variance associée doivent rester faibles pour que les résultats du système ne divergent pas.

Dans la suite de ce document, les différents points énumérés sont décrits à partir des équations du filtre EKF. Cette description permet l'écriture de différents théorèmes de convergence des estimations dans des cas d'études simples : le véhicule, stationnaire ou mobile, observe un ou plusieurs amers. Ces démonstrations s'appuient essentiellement sur les constatations et méthodes décrites dans les articles [11] et [83].

3.2.3 Etude des limites de convergence de l'EKF-SLAM

Dans ce paragraphe, un rappel des limites de convergence d'un filtre de Kalman appliqué à un contexte SLAM dans un cas linéaire puis dans un cas non-linéaire est fait. Aucun auteur, à l'exception de l'article [83], n'a réalisé cette étude. En reprenant les méthodes employées dans cet article, ce paragraphe propose une étude similaire pour des amers de type point mais également des amers de type droite. Cette étude s'appuie sur des simulations simples permettant une compréhension du problème posé par la non-linéarité du système. Les théorèmes de convergence des incertitudes pour un système SLAM linéaire seront numérotés $x.0$ et les théorèmes pour des systèmes SLAM non-linéaires $x.1$.

3.2.3.1 Théorèmes de convergence dans le cas linéaire

Dissanayake et al. [53] ont défini une solution au problème du SLAM. Cette solution s'appuyait sur un filtrage de Kalman dans un contexte linéaire. Les auteurs posèrent trois théorèmes de convergence de la matrice de covariance \mathbf{P}_M .

Théorème 1.0 : *Le déterminant de chaque matrice, associée à la position d'un amer et composant la matrice \mathbf{P}_M , décroît de manière monotone tant que des observations successives de l'amer sont réalisées.*

Lors de l'étape de prédiction, la matrice de covariance associée à la carte n'est pas modifiée. L'environnement, dans lequel évolue le véhicule, est considéré comme étant statique. Ceci implique que, lors de l'étape de mise à jour, l'ensemble de la matrice de covariance \mathbf{P} reste égale à la valeur prédite si aucune observation n'est réalisée. Sinon, cette valeur diminue lorsqu'une observation d'amer déjà cartographié est réalisée. Donc, le déterminant de la matrice \mathbf{P} n'augmente pas durant la phase de mise à jour du filtre (cf. l'annexe A) pour tout amer i entre les instants k et $k + 1$:

$$\det \mathbf{P}(k + 1|k + 1) = \det [\mathbf{P}(k + 1|k) - \mathbf{W}_i(k + 1) \cdot \mathbf{S}_i(k + 1) \cdot \mathbf{W}_i^T(k + 1)] \leq \det \mathbf{P}(k + 1|k) \quad (3.8)$$

Dans cette équation, le déterminant de la matrice de covariance est une mesure du volume de l'ellipse d'incertitude associée à l'estimation de l'état $\hat{\mathbf{X}}$. Cette équation est une traduction mathématique du **théorème 1.0** statuant que l'incertitude sur les estimations ne peut augmenter durant l'étape de mise à jour du filtre de Kalman.

Théorème 2.0 : *Lorsque le nombre d'itérations du filtre tend vers l'infini, la valeur minimale de la matrice de covariance associée à n'importe quelle position estimée d'amer est égale à la valeur initiale de la covariance du robot, \mathbf{P}_{V_0} , lorsque celui-ci a observé le premier amer de l'environnement.*

Plus précisément, la valeur de la covariance ne pourra être inférieure à la valeur limite calculée en fonction de \mathbf{P}_{V_0} , \mathbf{Q}_U , la matrice de covariance associée aux bruits sur les mesures odométriques et \mathbf{R} , la matrice de bruit associée aux bruits sur les mesures extéroceptives.

Le **théorème 2.0** n'est plus vrai lorsque une information telle que la connaissance d'une carte de l'environnement ou d'une position GPS est utilisée par le filtre de Kalman dans une application SLAM.

Théorème 3.0 : *Lorsque le nombre d'itérations du filtre tend vers l'infini, les estimations des positions des amers deviennent totalement corrélées.*

Toutes les observations des amers sont réalisées à partir de la position du véhicule. Toutes les estimations de la carte sont donc corrélées avec la position du véhicule. Lorsque le nombre d'observations tend vers l'infini, ces corrélations s'étendent à l'ensemble de la carte. De surcroît, une limite basse sera atteinte par la valeur de la matrice de covariance de la carte, \mathbf{P}_M , telle que :

$$\lim_{k \rightarrow \infty} [\mathbf{P}_M(k + 1|k + 1)] = \mathbf{P}_M(k|k) \quad (3.9)$$

Cette constatation a pour conséquence que la valeur limite du déterminant de la matrice de covariance d'une carte contenant au moins un amer tend vers zéro :

$$\lim_{k \rightarrow \infty} [\det \mathbf{P}_M(k|k)] = 0 \quad (3.10)$$

Toutes les relations géométriques entre les amers tendent à être connues parfaitement. Une telle connaissance lorsque le nombre d'itérations tend vers l'infini implique que les sous-matrices

de \mathbf{P}_M tendent à être identiques. En revanche, cette constatation n'indique pas que les déterminants associés à chaque position des amers tendent vers zéro. A l'infini, la position absolue des amers devrait rester incertaine. Le **théorème 2.0** répond à cette affirmation.

3.2.3.2 Théorèmes de convergence dans le cas non-linéaire

Les limites présentées dans le paragraphe précédent sont démontrées dans le cas linéaire. Pour un système non-linéaire, des modifications doivent être apportées à ces théorèmes de convergence. Pour une application EKF-SLAM en deux dimensions, les expressions générales pour l'évolution des matrices de covariance ne peuvent être obtenues. Par conséquent, deux scénarios simples de simulation sont utilisés pour analyser les limites de convergence :

- le robot est stationnaire et observe un amer de nombreuses fois ;
- le robot est stationnaire et observe deux amers de nombreuses fois.

Le raisonnement proposé dans ce paragraphe découle de l'article [83].

Théorème 1.1 : *Dans une application SLAM, le déterminant de chaque matrice, associée à la position d'un amer et composant la matrice \mathbf{P}_M , décroît de manière monotone tant que des observations successives de l'amer sont réalisées.*

Que le système soit linéaire ou non-linéaire, le **théorème 1** est toujours vrai. La matrice de covariance de la carte n'est pas modifiée lors de la phase de prédiction comme l'environnement est considéré comme statique. De surcroît, lors de l'étape de la mise à jour, le calcul de la matrice de covariance s'écrit :

$$\mathbf{P}(k+1 | k+1) = \mathbf{P}(k+1 | k) - \mathbf{W.S.W}^T \quad (3.11)$$

Seul change le calcul des matrices jacobiniennes pour obtenir les valeurs du gain de Kalman et de la matrice de covariance \mathbf{S} associée à l'innovation ν . Dans tous les cas, la quantité $\mathbf{P}(k+1 | k) - \mathbf{W.S.W}^T$ est supérieure ou égale à zéro. Cette observation confirme que l'équation (3.8) reste vraie dans un système EKF-SLAM.

Théorème 2.1 : Pour simplifier le raisonnement, un cas très simple est étudié. Le véhicule est stationnaire et observe un amer (cf. la figure 3.4 (a)). La position du véhicule est initialisée avec la valeur $\mathbf{X}_{V_0} = [10, 10, \pi]$ et une matrice de covariance associée \mathbf{P}_0 . Le robot reste à cette position et observe un amer situé à côté du repère de la carte.

Comme le véhicule est stationnaire, la matrice jacobienne \mathbf{F}_X reste nulle. L'incertitude du robot, initialisée avec la valeur \mathbf{P}_0 , ne change donc pas durant la phase de prédiction et reste égale à \mathbf{P}_0 . Cette matrice de covariance s'écrit :

$$\mathbf{P}_0 = \begin{bmatrix} P_{x_v y_v} & P_{x_v y_v \theta_v}^T \\ P_{x_v y_v \theta_v} & p_{\theta_v} \end{bmatrix} \quad (3.12)$$

$P_{x_v y_v}$ représente la matrice de covariance sur les coordonnées cartésiennes du véhicule, p_{θ_v} , la variance sur l'orientation du véhicule et $P_{x_v y_v \theta_v}$, la covariance entre les positions cartésiennes du véhicule et son orientation.

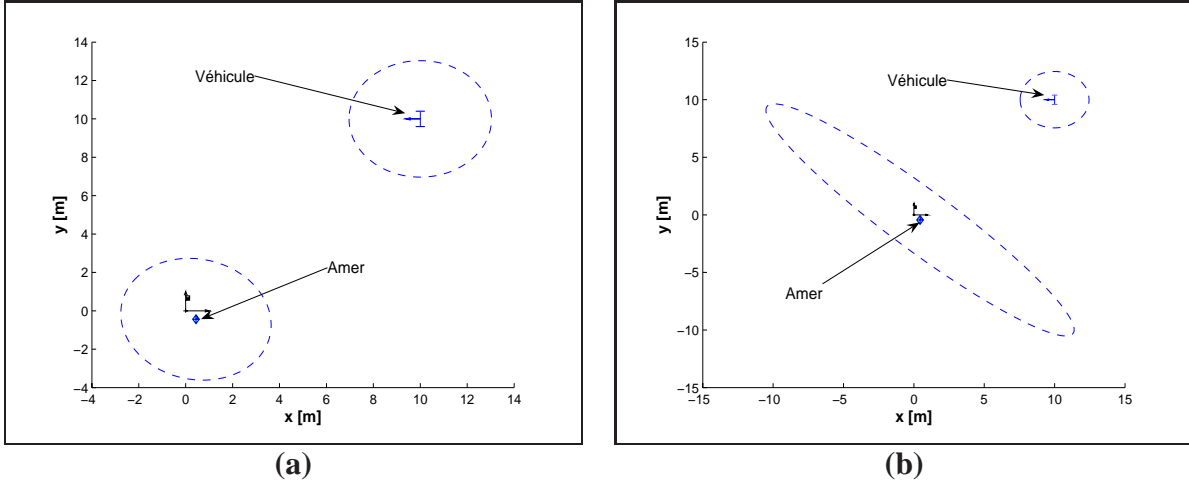


Figure 3.4: Véhicule stationnaire observant un amer. Représentation de la position et des ellipses d'incertitude : (a) $\mathbf{P}_0 = \text{diag}[1, 1, 0.001]$ et (b) $\mathbf{P}_0 = \text{diag}[1, 1, 0.03]$

Le véhicule observe un amer \mathbf{X}_A avec une matrice de covariance de bruit \mathbf{R}_A . L'information² apportée par une observation de l'amer correspond à :

$$\Omega_{new} = (\mathbf{H}_X)^T \cdot \mathbf{R}_A^{-1} \cdot \mathbf{H}_X \quad (3.15)$$

\mathbf{H}_X est la matrice jacobienne de la fonction d'observation. En reprenant les définitions des opérations de composition (cf. l'annexe B), cette matrice jacobienne s'écrit (le calcul de l'angle associé à l'opération de composition est omis pour un point) :

$$\begin{aligned} \mathbf{H}_X &= [\mathbf{H}_{X_V} \quad \mathbf{H}_{X_A}] \\ &= \begin{bmatrix} -\cos \theta_v & -\sin \theta_v & \sin \theta_v \cdot (x_v - x_A) + \cos \theta_v \cdot (y_v - y_A) & \cos \theta_v & \sin \theta_v \\ \sin \theta_v & -\cos \theta_v & \sin \theta_v \cdot (y_v - y_A) + \cos \theta_v \cdot (x_v - x_A) & -\sin \theta_v & \cos \theta_v \end{bmatrix} \\ &= [-\mathbf{H}_{X_A} \quad \mathbf{H}_{\theta_v} \quad \mathbf{H}_{X_A}] \end{aligned} \quad (3.16)$$

Si le véhicule réalise n observations de l'amer X_A , le gain d'information total sur le système s'écrit donc :

$$\begin{aligned} \Omega_n &= \begin{bmatrix} \Omega_0 & 0 \\ 0 & 0 \end{bmatrix} + n \cdot \Omega_{new} \\ &= \begin{bmatrix} \Omega_0 + n \cdot (\mathbf{H}_{X_V})^T \cdot \mathbf{R}_A^{-1} \cdot \mathbf{H}_{X_V} & n \cdot (\mathbf{H}_{X_V})^T \cdot \mathbf{R}_A^{-1} \cdot \mathbf{H}_{X_A} \\ n \cdot (\mathbf{H}_{X_A})^T \cdot \mathbf{R}_A^{-1} \cdot \mathbf{H}_{X_V} & n \cdot (\mathbf{H}_{X_A}) \cdot \mathbf{R}_A^{-1} \cdot \mathbf{H}_{X_A} \end{bmatrix} \end{aligned} \quad (3.17)$$

2. La forme informative [12] de la matrice de covariance est utilisée pour simplifier les calculs. Elle se définit simplement par :

$$\begin{aligned} \Omega(k+1|k) &= \mathbf{P}(k+1|k)^{-1} \\ \Omega(k+1|k+1) &= \Omega(k+1|k) + \Omega_{new} \\ \mathbf{P}(k+1|k+1) &= \Omega(k+1|k+1)^{-1} \end{aligned} \quad (3.13)$$

$$(3.14)$$

et la valeur initiale de la matrice d'information associée au véhicule s'écrit Ω_0 .

La matrice de covariance \mathbf{P}_n est obtenue en inversant³ la forme informative Ω_n :

$$\begin{aligned}\mathbf{P}_n &= \Omega_n^{-1} \\ &= \begin{bmatrix} \Omega_0^{-1} & -\Omega_0^{-1} \cdot (\mathbf{H}_{X_V})^T \cdot (\mathbf{H}_{X_A})^{-T} \\ -(\mathbf{H}_{X_A})^{-1} \cdot \mathbf{H}_{X_V} \cdot \Omega_0^{-1} & \mathbf{P}_{A_n} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{P}_0 & -\mathbf{P}_0 \cdot (\mathbf{H}_{X_V} \cdot (\mathbf{H}_{X_A}^{-1}))^T \\ -(\mathbf{H}_{X_A})^{-1} \cdot \mathbf{H}_{X_V} \cdot \mathbf{P}_0 & \mathbf{P}_{A_n} \end{bmatrix}\end{aligned}\quad (3.18)$$

avec :

$$\mathbf{P}_{A_n} = (\mathbf{H}_{X_A})^{-1} \cdot \mathbf{H}_{X_V} \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_V} \cdot (\mathbf{H}_{X_A}^{-1}))^T + \frac{(\mathbf{H}_{X_A})^{-1} \cdot \mathbf{R}_A \cdot (\mathbf{H}_{X_A})^{-T}}{n}\quad (3.19)$$

De surcroît, le terme avec les matrices jacobiennes peut être réécrit sous la forme :

$$\begin{aligned}\mathbf{H}_{X_A}^* &= -(\mathbf{H}_{X_A})^{-1} \cdot \mathbf{H}_{X_V} \\ &= -(\mathbf{H}_{X_A})^{-1} \cdot \begin{bmatrix} -\mathbf{H}_{X_A} & \mathbf{H}_{\theta_v} \end{bmatrix} \\ &= \begin{bmatrix} I & -(\mathbf{H}_{X_A})^{-1} \cdot \mathbf{H}_{\theta_v} \end{bmatrix}\end{aligned}\quad (3.20)$$

En conclusion, si le nombre d'observations n tend vers l'infini, la matrice de covariance devient :

$$\mathbf{P}_{A_\infty} = \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T\quad (3.21)$$

Ce résultat confirme le **théorème 2.0**. L'incertitude limite sur la position d'un amer de la carte est bien fonction de l'incertitude initiale sur la position du robot, \mathbf{P}_0 .

Cependant, quelques précisions doivent être ajoutées en analysant la forme de la matrice jacobienne $\mathbf{H}_{X_A}^*$. Premièrement, si l'incertitude sur l'orientation du robot est nulle alors l'incertitude sur la position de l'amer est égale à $P_{x_v y_v}$. La partie de la matrice, $-(\mathbf{H}_{X_A})^{-1} \cdot \mathbf{H}_{X_{\theta_v}}$, sera multipliée par une valeur très faible et n'apportera aucune information supplémentaire sur la valeur de l'incertitude. La figure 3.4(a) présente ce résultat. La variance sur l'orientation du

3. Étant donnée une matrice bloc M inversible tel que :

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

si les sous matrices A et D sont inversibles, l'inversion de M est obtenue par :

$$M^{-1} = \begin{bmatrix} X & Y \\ U & V \end{bmatrix}$$

avec

$$\begin{aligned}X &= (A - B \cdot D^{-1} \cdot C)^{-1} \\ Y &= -(A - B \cdot D^{-1} \cdot C)^{-1} \cdot B \cdot D^{-1} \\ U &= -D^{-1} \cdot C \cdot (A - B \cdot D^{-1} \cdot C)^{-1} \\ V &= -D^{-1} + D^{-1} \cdot C \cdot (A - B \cdot D^{-1} \cdot C)^{-1} \cdot B \cdot D^{-1}\end{aligned}$$

véhicule, p_{θ_v} , est initialisée avec une valeur faible de 10^{-3} rad^2 . L'ellipse d'incertitude sur la position de l'amer est égale à celle sur la position du véhicule. Cette remarque est en concordance avec le **théorème 2.0**.

En revanche, lorsque p_{θ_v} est initialisée avec une valeur importante, \mathbf{P}_{A_∞} est différent de \mathbf{P}_0 (cf. la figure 3.4(b)). La partie de la matrice jacobienne, $\mathbf{H}_{X_A}^*$, est multipliée par une valeur non nulle et apporte une information à l'incertitude sur la position de l'amer. Graphiquement, les aires des ellipses d'incertitude du robot et de l'amer sont différentes.

En conclusion, l'incertitude sur l'orientation du robot a un effet significatif sur l'incertitude de l'amer, \mathbf{P}_{A_∞} , lorsque le nombre d'observations de celui-ci tend vers l'infini. Même si la distance entre le robot et l'amer est connue, il restera impossible de déterminer une position exacte des deux éléments sans une connaissance parfaite de l'orientation du véhicule.

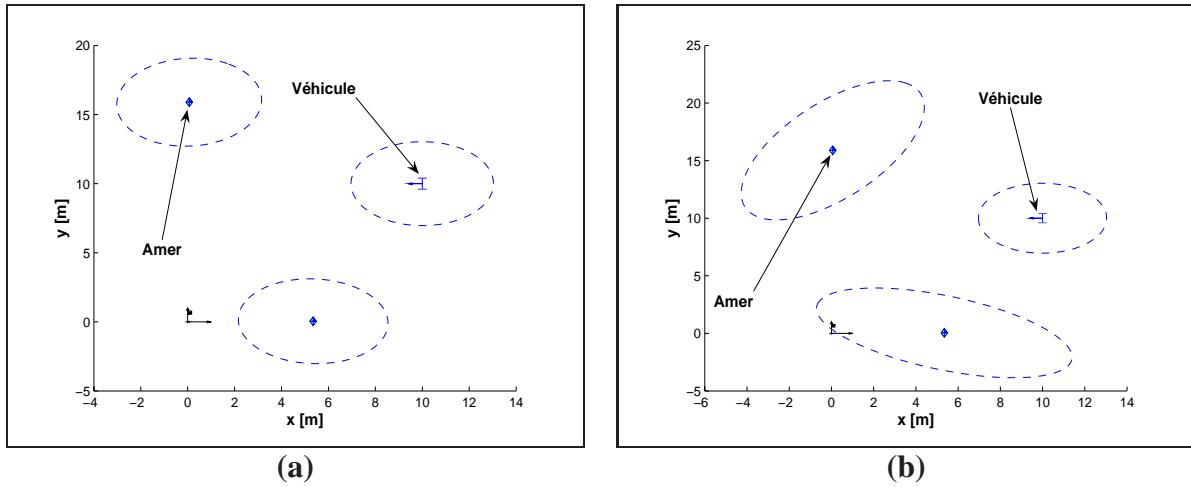


Figure 3.5: Véhicule stationnaire observant deux amers. Représentation de la position et des ellipses d'incertitude : (a) $\mathbf{P}_0 = \text{diag}[1, 1, 0.001]$ et (b) $\mathbf{P}_0 = \text{diag}[1, 1, 0.03]$.

Théorème 3.1 : Prenons une nouvelle fois un exemple simple où le véhicule observe deux amers \mathbf{X}_A et \mathbf{X}_B avec comme matrices de bruit de mesure respectives \mathbf{R}_A et \mathbf{R}_B (cf. la figure 3.5). Le même raisonnement, présenté précédemment, s'applique à ce cas d'étude. La matrice de covariance associée à la position du robot et des deux amers après n observations de ceux-ci s'écrit :

$$\mathbf{P}_n = \begin{bmatrix} \mathbf{P}_0 & \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T & \mathbf{P}_0 \cdot (\mathbf{H}_{X_B}^*)^T \\ \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 & \mathbf{P}_{A_n} & \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_B}^*)^T \\ \mathbf{H}_{X_B}^* \cdot \mathbf{P}_0 & \mathbf{H}_{X_B}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T & \mathbf{P}_{B_n} \end{bmatrix} \quad (3.22)$$

Les matrices de covariance associées aux positions des amers après n observations s'écrivent :

$$\begin{aligned} \mathbf{P}_{A_n} &= \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T + \frac{(\mathbf{H}_{X_A}^*)^{-1} \cdot \mathbf{R}_A \cdot (\mathbf{H}_{X_A}^*)^{-T}}{n} \\ \mathbf{P}_{B_n} &= \mathbf{H}_{X_B}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_B}^*)^T + \frac{(\mathbf{H}_{X_B}^*)^{-1} \cdot \mathbf{R}_B \cdot (\mathbf{H}_{X_B}^*)^{-T}}{n} \end{aligned} \quad (3.23)$$

Lorsque le nombre d'observations tend vers l'infini ($n \rightarrow \infty$), la matrice de covariance (3.22) s'écrit :

$$\mathbf{P}_\infty = \begin{bmatrix} \mathbf{P}_0 & \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T & \mathbf{P}_0 \cdot (\mathbf{H}_{X_B}^*)^T \\ \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 & \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T & \mathbf{H}_{X_A}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_B}^*)^T \\ \mathbf{H}_{X_B}^* \cdot \mathbf{P}_0 & \mathbf{H}_{X_B}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_A}^*)^T & \mathbf{H}_{X_B}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_B}^*)^T \end{bmatrix} \quad (3.24)$$

L'équation (3.24) met une nouvelle fois en évidence l'importance de la valeur initiale de l'incertitude sur l'orientation du véhicule. Comme le montre la figure 3.5 (a), si l'incertitude sur l'orientation du véhicule p_{θ_v} est nulle alors les limites des incertitudes sur les deux amers sont égales à P_{xy} . Dans ce cas particulier, le **théorème 2.0** est vérifié.

Cependant, lorsque l'incertitude sur l'orientation du véhicule n'est pas nulle (cf. la figure 3.5 (b)), les limites des incertitudes sur les positions des deux amers ne sont pas égales. Ceci est en contradiction avec le **théorème 2.0**. En effet, le calcul des jacobiennes défini par l'équation (3.16) donnera des valeurs différentes pour les matrices jacobiennes ($\mathbf{H}_{X_A}^* \neq \mathbf{H}_{X_B}^*$) comme les positions des amers \mathbf{X}_A et \mathbf{X}_B diffèrent. Les résultats de simulation présentés sur la figure 3.5 confirment ces constatations.

Au final, lorsque l'incertitude initiale sur l'orientation du véhicule est faible, les incertitudes sur les positions des amers et du véhicule sont équivalentes (ellipses bleues sur la figure 3.5 (a)). Ces résultats sont en accord avec le **théorème 2.0**. Cependant, lorsque l'incertitude initiale n'est pas faible, les incertitudes des trois éléments de la carte ne sont pas semblables (cf. la figure 3.5 (b)).

Conclusion : Dans ce paragraphe, une étude des valeurs limites de la matrice de covariance associée à une carte dans le cas linéaire puis dans le cas non-linéaire a été proposée. Ces limites de convergence, entre un système linéaire et non-linéaire, diffèrent avec le calcul des matrices jacobiennes permettant la linéarisation des équations dans le cas non-linéaire, en particulier pour le **théorème 2.0**. L'incertitude sur l'orientation du véhicule joue un rôle très important dans la définition de ces limites. Si cette orientation n'est pas connue parfaitement, le système est incapable d'estimer exactement les positions des éléments de l'environnement, seules les relations géométriques entre les amers peuvent être déterminées avec précision.

Seul un cas simple a été traité : le véhicule stationnaire. Lorsque le véhicule se déplace, le même raisonnement doit être appliqué en prenant en compte dans les calculs les matrices jacobiennes \mathbf{F}_X et \mathbf{F}_U introduisant l'incertitude liée au déplacement du véhicule. Dans la suite de ce document, ce cas de figure est traité sur des résultats de simulation. Cependant, les valeurs théoriques des estimations lors de la convergence ne sont pas calculées. Ces calculs sont laborieux et entraînent des conclusions similaires à celles présentées dans cette partie.

Les limites de convergence d'un amer de type point ont été présentées. Dans nos travaux, les mêmes démonstrations s'appliquent à un amer de type cercle comme seule sa position est estimée. Dans la suite, l'étude de la convergence d'un système SLAM avec une carte composée d'amer de type droite est proposée. Cette dernière démonstration permettra de clore cette partie sur les théorèmes de convergence d'un filtre EKF-SLAM.

3.2.3.3 Limite de convergence pour un amer de type droite

Les précédents théorèmes ont été démontrés en utilisant des amers de type point. Ce type d'amer ne comporte aucune information angulaire. Ce paragraphe présente l'analyse des limites de convergence lorsque l'amer employé est une droite.

A ce jour et à notre connaissance, aucun article n'a réellement abordé ce sujet à l'exception de l'article [133]. Dans cet article, des résultats de simulation apportent la preuve que l'orientation des droites augmente les problèmes d'inconsistance des estimations. Cependant, aucune analyse de la convergence de la matrice de covariance \mathbf{P}_M n'est réalisée. Dans cette section, une étude, similaire au paragraphe précédent et confirmant les conclusions obtenues dans l'article [133], est réalisée.

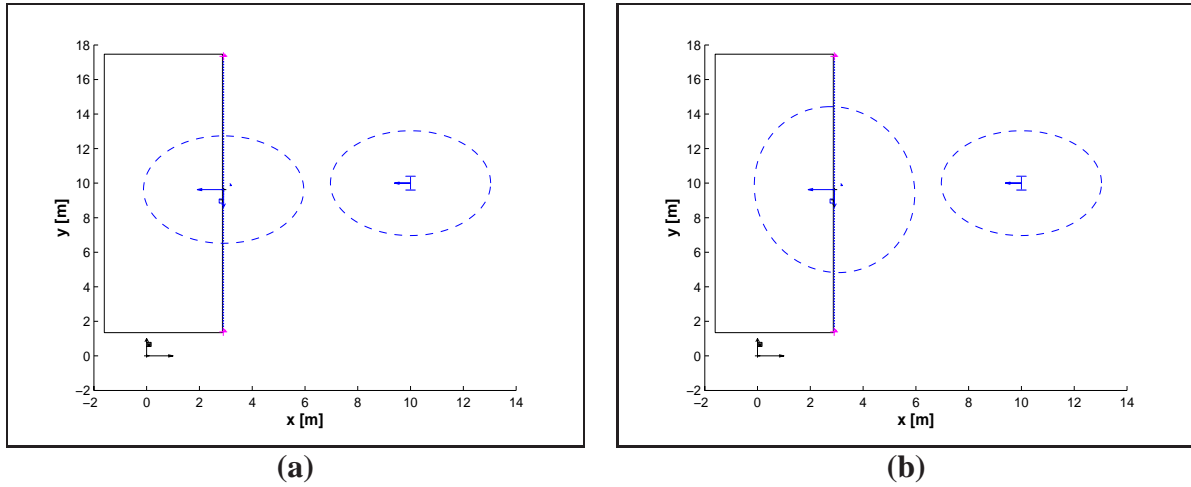


Figure 3.6: Véhicule stationnaire observant une droite. Représentation de la position et des ellipses d'incertitude : (a) $\mathbf{P}_0 = \text{diag}[1, 1, 0.001]$ et (b) $\mathbf{P}_0 = \text{diag}[1, 1, 0.03]$.

Description du système et de l'équation d'observation

Le robot est stationnaire et observe un unique amer de type droite $\mathbf{X}_L = [x_l \ y_l \ \theta_l]^T$ (cf. la figure 3.6). La mesure de cette droite s'écrit $\mathbf{X}_{Lm} = [x_m \ y_m \ \theta_m]^T$ avec comme matrice de covariance associée \mathbf{R}_{Lm} . La concaténation de ces vecteurs dans un unique vecteur s'écrit :

$$\mathbf{X}_{syst} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_L \\ \mathbf{X}_{Lm} \end{bmatrix} \quad (3.25)$$

La matrice de covariance associée à ce vecteur d'état s'écrit :

$$\mathbf{P}_{syst} = \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{R}_{Lm} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_V & \mathbf{P}_{VM} & 0 \\ \mathbf{P}_{VM} & \mathbf{P}_M & 0 \\ 0 & 0 & \mathbf{R}_{Lm} \end{bmatrix} \quad (3.26)$$

La fonction d'observation d'une droite, \mathbf{h} , s'écrit :

$$\mathbf{Z} = \mathbf{h}(\mathbf{X}_{syst}) = \ominus \mathbf{X}_L \oplus \mathbf{X}_V \oplus \mathbf{X}_{Lm} = 0 \quad (3.27)$$

La matrice jacobienne \mathbf{H} associée à ce calcul s'écrit :

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_X & \mathbf{H}_{X_{Lm}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{X_V} & \mathbf{H}_{X_L} & \mathbf{H}_{X_{Lm}} \end{bmatrix} \quad (3.28)$$

Les différentes matrices jacobiennes permettant l'écriture de la matrice \mathbf{H} sont décrites dans l'annexe D. A partir des équations (3.26) et (3.28), le calcul de la matrice de covariance, \mathbf{P}_Z , associée à l'observation \mathbf{Z} , se formule :

$$\mathbf{P}_Z = \mathbf{H} \cdot \mathbf{P}_{syst} \cdot \mathbf{H}^T = \mathbf{H}_X \cdot \mathbf{P} \cdot (\mathbf{H}_X)^T + \mathbf{R}_L \quad (3.29)$$

La matrice de covariance \mathbf{R}_L , associée à la mesure de la droite, s'écrit :

$$\mathbf{R}_L = \mathbf{H}_{X_{Lm}} \cdot \mathbf{R}_{Lm} \cdot (\mathbf{H}_{X_{Lm}})^T \quad (3.30)$$

En appliquant le même raisonnement que pour la démonstration du **théorème 2.1**, le gain d'information apporté par l'observation de l'amer \mathbf{X}_{Lm} s'écrit :

$$\Omega_{new} = (\mathbf{H}_X)^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_X \quad (3.31)$$

La matrice jacobienne, \mathbf{H}_X , est définie dans l'annexe D.

Si le véhicule réalise n observations de l'amer \mathbf{X}_L , le gain d'information total sur le système s'écrit :

$$\begin{aligned} \Omega_n &= \begin{bmatrix} \Omega_0 & 0 \\ 0 & 0 \end{bmatrix} + n \cdot \Omega_{new} \\ &= \begin{bmatrix} \Omega_0 + n \cdot (\mathbf{H}_{X_L})^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_V} & n \cdot (\mathbf{H}_{X_V})^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_L} \\ n \cdot (\mathbf{H}_{X_L})^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_V} & n \cdot (\mathbf{H}_{X_L}) \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_L} \end{bmatrix} \end{aligned} \quad (3.32)$$

La matrice de covariance \mathbf{P}_n est obtenue en inversant la matrice d'information Ω_n :

$$\mathbf{P}_n = \begin{bmatrix} \mathbf{P}_0 & -\mathbf{P}_0 \cdot (\mathbf{H}_{X_V} \cdot (\mathbf{H}_{X_L}^{-1}))^T \\ -(\mathbf{H}_{X_L})^{-1} \cdot \mathbf{H}_{X_V} \cdot \mathbf{P}_0 & \mathbf{P}_{L_n} \end{bmatrix} \quad (3.33)$$

avec :

$$\mathbf{P}_{L_n} = (\mathbf{H}_{X_L})^{-1} \cdot \mathbf{H}_{X_V} \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_V} \cdot (\mathbf{H}_{X_L}^{-1}))^T + \frac{(\mathbf{H}_{X_L})^{-1} \cdot \mathbf{R}_L \cdot (\mathbf{H}_{X_L})^{-T}}{n} \quad (3.34)$$

Après une infinité d'observations de l'amer \mathbf{X}_L , la limite de la matrice de covariance associée à la carte s'écrit :

$$\mathbf{P}_{L_\infty} = \begin{bmatrix} \mathbf{H}_{X_L}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_L}^*)^T \end{bmatrix} \quad (3.35)$$

Pour analyser la convergence de la matrice de covariance associée à la carte, \mathbf{P}_M , il faut étudier, comme dans le cas d'un amer de type point, la matrice $\mathbf{H}_{X_L}^*$. Cette matrice résulte de l'inversion de la matrice \mathbf{H}_{X_L} (cf. dans l'annexe D) :

$$\mathbf{H}_{X_L}^* = -(\mathbf{H}_{X_L})^{-1} \cdot \mathbf{H}_{X_V} = \begin{bmatrix} \mathbf{I} & -\mathbf{H}_{xlyl}^{-1} \cdot \mathbf{H}_{\theta_v} - \mathbf{H}_{\theta_l}^* \\ 0 & 1 \end{bmatrix} \quad (3.36)$$

L'observation de la forme de cette matrice jacobienne (équations (3.35) et (3.36)) permet de vérifier le **théorème 2.1**. En effet, si l'incertitude sur l'orientation du véhicule, p_{θ_v} , est nulle, alors l'incertitude sur la position de la droite sera égale à celle de la position du véhicule après une infinité d'observations. La matrice de covariance \mathbf{P}_M sera égale à \mathbf{P}_0 . En revanche, si cette incertitude initiale n'est pas nulle, alors ni la position du repère associée à la droite, ni son orientation, n'auront des incertitudes égales à celles associées aux états du véhicule.

Dans l'article [133], les auteurs soulignent que l'emploi d'un amer avec une information angulaire accentue les problèmes d'inconsistance dus à la non-linéarité des équations. L'observation de l'écriture de la matrice jacobienne $\mathbf{H}_{X_L}^*$ confirme cette observation. Le calcul de cette matrice est fonction des estimations de l'orientation du véhicule mais également de l'orientation de la droite considérée.

Observation du phénomène à partir d'une simulation

Le véhicule est stationnaire à la position $\mathbf{X}_V = [10 \ 10 \ \pi]^T$. Il observe une droite 300 fois consécutivement. La figure 3.6 présente les formes finales de l'incertitude sur ces deux éléments dans deux cas : lorsque l'incertitude sur l'orientation du véhicule, p_{θ_v} , est faible et lorsqu'elle ne l'est pas.

Lorsque l'incertitude sur l'orientation du véhicule est nulle, les aires des ellipses sur la position du repère associée à l'amer droite et celle associée à la position du véhicule sont égales (cf. la figure 3.6 (a)). Dans ces conditions, le système a donc un comportement linéaire. En revanche, lorsque la valeur initiale de l'incertitude sur l'orientation du véhicule n'est plus nulle, les deux aires d'incertitude ne sont plus égales (cf. la figure 3.6 (b)). Ce résultat de simulation confirme le **théorème 2.1** dans le cas où les amers utilisés sont des droites.

En outre, ce résultat valide également les conclusions décrites par le **théorème 3.1** lorsque plusieurs amers de type point sont observés. Comme la matrice \mathbf{H}_{X_L} , dans le cas d'une droite, est similaire à la matrice \mathbf{H}_{X_A} pour le cas d'un point (respectivement les équations (3.20) et (3.36)), le **théorème 3.1** s'applique également aux amers de type droite.

3.2.4 Conclusion sur les limites de convergence de l'EKF-SLAM

Dans cette section, différents outils permettant de quantifier la consistance des estimations du filtre de Kalman dans une application SLAM ont été introduits : une distance statistique nommée NEES et trois théorèmes de convergence.

Toutes les démonstrations proposées supposent que les matrices jacobiennes sont calculées avec la position exacte du véhicule et des amers. Les limites décrites sont les valeurs exactes ou idéales obtenues avec un filtre EKF-SLAM. Ces valeurs seront donc utilisées comme références pour quantifier les résultats obtenus avec les différentes méthodes testées. De surcroît, cette étude nous permet de conclure sur la convergence de la matrice de covariance associée à la carte. Que ce soit pour un amer de type point ou cercle, ou que ce soit pour un amer de type droite apportant une information angulaire, les mêmes théorèmes de convergence sont applicables.

De plus, l'incertitude sur l'orientation du véhicule joue un rôle très important dans la valeur finale de l'incertitude sur les éléments de la carte estimée. A travers des résultats de simulation (cf. la section 3.4), nous observerons qu'une trop grande valeur de celle-ci entraîne un

gain d'information erroné sur l'incertitude des états du véhicule et de la carte. L'origine de ces inconsistances correspond à la linéarisation des équations du système comme le prouvent les démonstrations proposées dans l'annexe D. Pour que le système ait un comportement linéaire, il est donc préférable d'initialiser cette incertitude avec une valeur nulle. Cependant, par définition, dans une application SLAM, le véhicule navigue dans un environnement inconnu. L'incertitude sur ces états, et donc sur son orientation, augmente. Un gain d'information erroné sur les estimations est donc inévitable. Ce phénomène entraîne donc un non respect des théorèmes non-linéaires cités précédemment. Ce problème est mis en relief via les résultats de simulation présentés dans la section 3.4.

Au final, la définition de ces théorèmes ont permis de mettre à jour la faiblesse de l'application EKF-SLAM. Elle réside dans la linéarisation des équations fonctions soit de l'orientation du véhicule soit de l'orientation d'amer de plus haut niveau comme une droite. Cette constatation a permis deux choses :

- décrire des conditions d'application des travaux finaux de cette thèse nommés SLAM-contraint via les théorèmes précédents et les différentes démonstrations présentées en annexe D ;
- appliquer une nouvelle approche du filtre de Kalman : le filtre **PEKF-SLAM** ayant pour but de diminuer les effets de la non-linéarité des équations d'un système SLAM.

3.3 Application du filtre polynomial de Kalman dans un contexte SLAM : PEKF-SLAM

Dans ce chapitre, il a été démontré que les estimations de l'EKF-SLAM sont inconsistantes lorsque la valeur de l'incertitude sur l'orientation du véhicule n'est pas négligeable. Pour répondre à ce problème de convergence, l'étude de l'application d'un filtre polynomial de Kalman dans un contexte SLAM, nommée **PEKF-SLAM**⁴, est proposée. Cette section en introduit les concepts de base.

3.3.1 Présentation du filtre polynomial de Kalman

3.3.1.1 Origine du filtre PEKF

Le filtre PEKF a été introduit pour la première fois par Carravetta et al. [25]. Comme son nom l'indique, ce filtrage s'appuie sur un algorithme récursif de Kalman. La différence, avec la version classique, repose sur la description des équations régissant le système. En effet, une transformation polynomiale des équations non-linéaires, fondée sur les puissances de Kronecker, est réalisée, permettant l'application de la théorie de Carleman [92].

Cette méthode d'estimation a été appliquée consécutivement sur un système linéaire [25], bilinéaire [24] et non-linéaire [70] [71]. Les résultats de simulation présentés dans chacun de ces articles prouvent que l'application du filtre PEKF améliore la précision des résultats obtenus

4. *Polynomial Extended Kalman Filter - Simultaneous Localization And Mapping*

avec le filtre de Kalman classique. De surcroît, une comparaison des résultats avec le filtre UKF [86] et le filtre itératif EKF [12] a souligné l'efficacité de ce nouveau filtre.

Dans le domaine de la robotique mobile, ce filtre a été récemment appliqué pour la localisation d'un véhicule dans un environnement simulé composé de droites [107]. Les résultats présentés de cette simulation soulignent également un meilleur comportement de ce filtre dans l'estimation de la position du véhicule lorsque les bruits sur les mesures proprioceptives et extéroceptives sont importants. La question est de savoir si l'application de ce filtre apporte la même amélioration des résultats dans un contexte SLAM.

Après un rappel sur les produits de Kronecker [25], la formulation générale du filtre polynomial de Kalman est introduite. Puis, dans la suite de ce chapitre, une application de ce filtre dans un contexte SLAM est présentée.

3.3.1.2 Produit de Kronecker

Ce produit a été introduit par le mathématicien allemand Léopold Kronecker (7 décembre 1823 - 29 décembre 1881). Soient une matrice A de taille $m \times n$ et une matrice B de taille $p \times q$. La matrice C , résultant du produit de Kronecker des matrices A et B , est de taille $(m.p) \times (n.q)$. Elle est définie par :

$$C = A \otimes B = \begin{bmatrix} a_{11}.B & \cdots & a_{1n}.B \\ \vdots & \ddots & \vdots \\ a_{m1}.B & \cdots & a_{mn}.B \end{bmatrix} \quad (3.37)$$

A partir de cette définition, la puissance de Kronecker nous permet de définir un vecteur d'état du système, \mathcal{X} , composé de monômes. Pour introduire cette nouvelle formulation, prenons un exemple simple. Soit un vecteur \mathbf{X} composé de deux variables quelconques, x_1 et x_2 :

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (3.38)$$

A partir de la formule (3.37), la puissance de Kronecker à l'ordre deux du vecteur \mathbf{X} s'écrit :

$$\mathbf{X}^{[2]} = \mathbf{X} \otimes \mathbf{X} = \begin{bmatrix} x_1.\mathbf{X} \\ x_2.\mathbf{X} \end{bmatrix} = \begin{bmatrix} (x_1)^2 \\ x_1x_2 \\ (x_2)^2 \\ x_2x_1 \end{bmatrix} \quad (3.39)$$

Avec les équations (3.38) et (3.39), le vecteur d'état \mathcal{X} , élevé à l'ordre deux, s'écrit :

$$\mathcal{X} = \begin{bmatrix} \mathbf{X} \\ \mathbf{X}^{[2]} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ (x_1)^2 \\ (x_2)^2 \\ x_1.x_2 \end{bmatrix} \quad (3.40)$$

Pour exemple, à l'ordre trois, le produit de Kronecker s'écrit donc :

$$\mathbf{X}^{[3]} = \mathbf{X} \otimes \mathbf{X}^{[2]} = \begin{bmatrix} x_1 \cdot \mathbf{X}^{[2]} \\ x_2 \cdot \mathbf{X}^{[2]} \end{bmatrix} = \begin{bmatrix} (x_1)^3 \\ x_1^2 \cdot x_2 \\ x_1 \cdot (x_2)^2 \\ (x_1)^2 \cdot x_2 \\ (x_1)^2 \cdot x_2 \\ x_1 \cdot (x_2)^2 \\ (x_2)^3 \\ x_1 \cdot (x_2)^2 \end{bmatrix} \quad (3.41)$$

Avec cette équation, le vecteur d'état du système à l'ordre trois s'écrit donc :

$$\mathcal{X} = \begin{bmatrix} \mathbf{X} \\ \mathbf{X}^{[2]} \\ \mathbf{X}^{[3]} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ (x_1)^2 \\ (x_2)^2 \\ (x_1)^2 \cdot x_2 \\ x_1 \cdot (x_2)^2 \\ (x_1)^3 \\ (x_2)^3 \end{bmatrix} \quad (3.42)$$

Le système peut être ainsi décrit par des vecteurs d'état obtenus avec une puissance de Kronecker à l'ordre μ du vecteur originel \mathbf{X} . L'objectif est de décrire les fonctions non-linéaires d'un système, uniquement à partir de ces nouveaux états (ou monômes).

3.3.1.3 Description générale du filtre polynomial

Un système non-linéaire est généralement décrit par deux équations, l'évolution des états et leurs observations (cf. l'annexe A) :

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{f}(\mathbf{X}(k), \mathbf{U}(k)) + w(k) \\ \mathbf{Z}(k+1) &= \mathbf{h}(\mathbf{X}(k+1)) + v(k) \end{aligned} \quad (3.43)$$

Les variables $w(k)$ et $v(k)$ représentent les bruits sur les mesures proprioceptives et extéroceptives. Le vecteur $\mathbf{U}(k)$ représente les données permettant l'évolution des états du système.

La particularité du filtre PEKF repose sur l'approximation du système par l'approche de Carleman [92]. Cette approximation est réalisée en définissant des séquences $\mathbf{X}^{[m]}$ et $\mathbf{Z}^{[m]}$ avec $m = [1 \cdots \mu]$ correspondant respectivement aux puissances de Kronecker des vecteurs associés

aux états et aux observations du système. Ces nouveaux vecteurs s'écrivent :

$$\begin{aligned} \mathcal{X} &= \begin{bmatrix} \mathbf{X}^{[1]} \\ \mathbf{X}^{[2]} \\ \vdots \\ \mathbf{X}^{[i]} \\ \vdots \\ \mathbf{X}^{[\mu]} \end{bmatrix} \in \mathbb{R}^{n_\mu} \quad \text{avec} \quad n_\mu = n + n^2 + \dots + n^\mu \\ \mathcal{Z} &= \begin{bmatrix} \mathbf{Z}^{[1]} \\ \mathbf{Z}^{[2]} \\ \vdots \\ \mathbf{Z}^{[i]} \\ \vdots \\ \mathbf{Z}^{[\mu]} \end{bmatrix} \in \mathbb{R}^{q_\mu} \quad \text{avec} \quad q_\mu = q + q^2 + \dots + q^\mu \end{aligned} \quad (3.44)$$

La variable $i = [1 \dots \mu]$ représente l'ordre de la puissance de Kronecker du vecteur. Avec cette définition, le système non-linéaire (3.43) s'écrit désormais :

$$\begin{aligned} \mathcal{X}(k+1) &= \mathcal{A}(k) \cdot \mathcal{X}(k) + \mathcal{U}(k) + \mathcal{W}(k) \\ \mathcal{Z}(k+1) &= \mathcal{C}(k) \cdot \mathcal{X}(k+1) + \mathcal{V}(k) \end{aligned} \quad (3.45)$$

La matrice \mathcal{A} est une matrice jacobienne fonction de la commande \mathcal{U} et du vecteur d'état \mathcal{X} . Elle permet l'évolution du vecteur d'état du système \mathcal{X} définie par la fonction non-linéaire \mathbf{f} . \mathcal{A} est décrite par :

$$\mathcal{A}(k+1) = \begin{bmatrix} \mathbf{A}_{1,1} & \cdots & \mathbf{A}_{1,\mu} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{\mu,1} & \cdots & \mathbf{A}_{\mu,\mu} \end{bmatrix} \quad (3.46)$$

Les différentes sous-matrices, $\mathbf{A}_{i,j}$, correspondent à des dérivées partielles :

$$\mathbf{A}_{i,j}(k) = \frac{\partial \mathbf{X}^{[i]}(k+1)}{\partial \mathbf{X}^{[j]}(k+1)} \quad (3.47)$$

La matrice \mathcal{C} est une matrice jacobienne permettant de modéliser la fonction d'observation \mathbf{h} :

$$\mathcal{C}(k+1) = \begin{bmatrix} \mathbf{C}_{1,1} & \cdots & \mathbf{C}_{1,\mu} \\ \vdots & \ddots & \vdots \\ \mathbf{C}_{\mu,1} & \cdots & \mathbf{C}_{\mu,\mu} \end{bmatrix} \quad (3.48)$$

Les différentes sous-matrices $\mathbf{C}_{i,j}$ correspondent à des dérivées partielles des observations du système par rapport aux états estimés :

$$\mathbf{C}_{i,j}(k) = \frac{\partial \mathbf{Z}^{[i]}(k+1)}{\partial \mathbf{X}^{[j]}(k+1)} \quad (3.49)$$

Les matrices $\mathcal{W}(k)$ et $\mathcal{V}(k)$ représentent les variables aléatoires à moyenne nulle associées respectivement à l'évolution du système d'état et de l'observation. Ces séquences de bruits sont non-corrélées.

$$\mathcal{W}(k) = \begin{bmatrix} w^{[1]} \\ w^{[2]} \\ \vdots \\ w^{[\mu]} \end{bmatrix} \quad et \quad \mathcal{V}(k) = \begin{bmatrix} v^{[1]} \\ v^{[2]} \\ \vdots \\ v^{[\mu]} \end{bmatrix} \quad (3.50)$$

D'un point de vue pratique, cette méthode consiste à définir un système d'équations dont, les processus d'état et de sorties correspondent aux puissances de Kronecker des processus originaux d'état et de sorties. Une description plus détaillée de ce filtre est donnée dans les articles [24], [70] et [71].

Maintenant que les principes de base du filtre PEKF ont été introduits, cette approche est appliquée dans un contexte SLAM. La suite de ce document se focalise sur la décomposition des équations non-linéaires d'un système SLAM en somme de polynômes.

3.3.2 Présentation du filtre PEKF-SLAM

Dans ce paragraphe, sont présentées les différentes étapes de l'implémentation du filtre PEKF-SLAM. Avant d'aborder les différentes équations de ce filtre, un travail préliminaire sur l'écriture du vecteur d'état associé au véhicule est réalisé. Ce travail est indispensable pour décrire les équations non-linéaires (prédiction, observation et cartographie) en sommes de polynômes. Ensuite, les nouveaux vecteurs d'état, découlant de l'application des puissances de Kronecker sur les vecteurs initiaux du filtre de Kalman, sont proposés. Puis, les décompositions en somme de polynômes des fonctions non-linéaires (prédiction, observation et cartographie) sont présentées.

Pour introduire l'implémentation du filtre PEKF-SLAM, nous allons nous focaliser sur un seul type d'amer, le point. Les équations liées aux amers de type droite sont présentées dans l'annexe D.

3.3.2.1 Application des puissances de Kronecker

L'approximation de Carleman peut se définir comme la décomposition des équations non-linéaires, \mathbf{f} , \mathbf{h} et \mathbf{g} , en somme de polynômes. Pour appliquer cette approximation, une reformulation du vecteur d'état associé au véhicule, \mathbf{X}_V , doit être réalisée. En effet, prenons comme exemple la fonction d'observation, \mathbf{h} , d'un amer point. Cette fonction projetée dans le repère du véhicule un point défini dans le monde en s'appuyant sur des opérations de composition (cf. l'annexe B) :

$$\mathbf{X}_{pi} = \mathbf{h}(\mathbf{X}) = \begin{bmatrix} x_i \cdot \cos(\theta_v) + y_i \cdot \sin(\theta_v) - x_v \cdot \cos(\theta_v) - y_v \cdot \sin(\theta_v) \\ y_i \cdot \cos(\theta_v) - x_i \cdot \sin(\theta_v) + x_v \cdot \sin(\theta_v) - y_v \cdot \cos(\theta_v) \end{bmatrix} \quad (3.51)$$

Pour décrire cette fonction non-linéaire en une somme de polynômes, il faut considérer comme états du véhicule le sinus et le cosinus de l'angle θ_v .

Extension du vecteur d'état associé au véhicule

Le vecteur d'état du véhicule est donc étendu pour inclure ces deux états supplémentaires tel que :

$$\mathbf{Z}_V = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix} = \begin{bmatrix} x_v \\ y_v \\ \theta_v \\ \sin(\theta_v) \\ \cos(\theta_v) \end{bmatrix} \quad (3.52)$$

Avec ce nouveau vecteur d'état, la fonction d'observation d'un amer point s'écrit :

$$\mathbf{X}_{pi} = \mathbf{h}(\mathbf{X}) = \begin{bmatrix} (x_i.z_5) + (y_i.z_4) - (z_1.z_5) - (z_2.z_4) \\ (y_i.z_5) - (x_i.z_4) + (z_1.z_4) - (z_2.z_5) \end{bmatrix} \quad (3.53)$$

Désormais, cette fonction correspond bien à une somme de polynômes d'états du système. Cependant, la modification du vecteur \mathbf{X}_V entraîne une réécriture des équations de son déplacement (cf. l'annexe D). De surcroît, pour réécrire les équations d'évolution du véhicule, un nouveau vecteur de commande, \mathbf{U} , comportant désormais quatre composantes, est créé :

$$\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} v.\Delta t \\ \frac{\Delta t.v.\tan(\varphi)}{L} \\ \sin\left(\frac{\Delta t.v.\tan(\varphi)}{L}\right) \\ \cos\left(\frac{\Delta t.v.\tan(\varphi)}{L}\right) \end{bmatrix} \quad (3.54)$$

La matrice de covariance, \mathbf{Q}_U , associée au vecteur \mathbf{U} , est proposée en annexe D. Au final, avec les nouveaux vecteurs d'état (3.52) et de commande (3.54), les équations d'évolution du véhicule s'écrivent :

$$\mathbf{Z}_V(k+1|k) = \begin{bmatrix} z_1(k|k) + u_1.z_5(k|k) \\ z_2(k|k) + u_1.z_4(k|k) \\ z_3(k|k) + u_2 \\ z_4(k|k).u_4 + z_5(k|k).u_3 \\ z_5(k|k).u_4 - z_4(k|k).u_3 \end{bmatrix} \quad (3.55)$$

Ces équations peuvent être réécrites sous la forme matricielle suivante :

$$\mathbf{Z}_V(k+1|k) = \mathbf{A}(k).\mathbf{Z}_V(k|k) + \mathbf{U}_c(k) \quad (3.56)$$

La matrice $\mathbf{A}(k)$ est égale à :

$$\mathbf{A}(k) = \begin{bmatrix} 1 & 0 & 0 & 0 & u_1 \\ 0 & 1 & 0 & u_1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & u_4 & u_3 \\ 0 & 0 & 0 & -u_3 & u_4 \end{bmatrix} \quad (3.57)$$

La matrice \mathbf{U}_c correspond à la commande sur l'orientation du véhicule :

$$\mathbf{U}_c(k) = [0 \ 0 \ u_2 \ 0 \ 0]^T \quad (3.58)$$

Le calcul de la matrice de covariance associée à l'état $\mathbf{Z}_V(k+1|k)$ correspond à :

$$\mathbf{P}_V(k+1|k) = \mathbf{A}(k) \cdot \mathbf{P}_V(k|k) \cdot \mathbf{A}(k)^T + \mathbf{F}_U \cdot \mathbf{Q}_U \cdot (\mathbf{F}_U)^T \quad (3.59)$$

\mathbf{F}_U est à la matrice jacobienne de l'équation (3.55) par rapport au vecteur \mathbf{U} :

$$\mathbf{F}_U = \begin{bmatrix} z_5(k|k) & 0 & 0 & 0 \\ z_4(k|k) & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & z_5(k|k) & z_4(k|k) \\ 0 & 0 & -z_4(k|k) & z_5(k|k) \end{bmatrix} \quad (3.60)$$

Les équations correspondent maintenant à des sommes de polynômes. La dernière étape du filtre PEKF est d'inclure ces polynômes dans le vecteur d'état du système. C'est dans ce but que les puissances de Kronecker sont employées. Elles permettent de décrire les différents polynômes des équations par des états présents dans le vecteur du système reformulé.

Application des puissances de Kronecker

Les composantes du vecteur d'état associé au véhicule (3.52) et aux amers de la carte sont élevées à l'ordre deux en employant les puissances de Kronecker. En toute rigueur, le calcul de la puissance de Kronecker de vecteur d'état \mathbf{X} du système s'écrit :

$$\mathcal{Z} = \begin{bmatrix} \mathbf{X} \\ \mathbf{X}^{[2]} \end{bmatrix} \quad (3.61)$$

Le deuxième membre du vecteur \mathcal{Z} se décompose en plusieurs variables :

$$\mathbf{X}^{[2]} = \mathbf{X} \otimes \mathbf{X} = \begin{bmatrix} \mathbf{Z}_V \\ \mathbf{X}_M \end{bmatrix} \otimes \begin{bmatrix} \mathbf{Z}_V \\ \mathbf{X}_M \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_V^{[2]} \\ \mathbf{Z}_V \otimes \mathbf{X}_M \\ \mathbf{X}_M^{[2]} \end{bmatrix} \quad (3.62)$$

Au final, le vecteur d'état \mathcal{Z} de cette nouvelle approche s'écrit :

$$\mathcal{Z} = \begin{bmatrix} \mathcal{Z}_V \\ \mathcal{Z}_M \end{bmatrix} \quad (3.63)$$

avec, les vecteurs \mathcal{Z}_V et \mathcal{Z}_M associés respectivement au véhicule et la carte estimée de l'environnement.

Ces nouveaux états permettent de décomposer en sommes de polynômes les fonctions non-linéaires. Cependant, seuls les états utiles à cette décomposition sont représentés dans les vecteurs \mathcal{Z}_V et \mathcal{Z}_M .

Puissance de Kronecker appliquée aux composantes du vecteur d'état du véhicule

En appliquant les puissances de Kronecker à l'ordre deux sur le vecteur, \mathbf{Z}_V , le nouveau vecteur associé à la position du véhicule s'écrit :

$$\mathbf{Z}_V = \begin{bmatrix} \mathbf{Z}_V \\ \mathbf{Z}_V^{[2]} \end{bmatrix} = \begin{bmatrix} x_v \\ y_v \\ \theta_v \\ \sin(\theta_v) \\ \cos(\theta_v) \\ \sin^2(\theta_v) \\ \cos^2(\theta_v) \\ x_v \cdot \sin(\theta_v) \\ x_v \cdot \cos(\theta_v) \\ y_v \cdot \sin(\theta_v) \\ y_v \cdot \cos(\theta_v) \\ \sin(\theta_v) \cdot \cos(\theta_v) \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \\ z_8 \\ z_9 \\ z_{10} \\ z_{11} \\ z_{12} \end{bmatrix} \quad (3.64)$$

Comme présenté dans l'introduction, ce vecteur \mathbf{Z}_V représente seulement une application « partielle » des puissances de Kronecker. Seuls les états nécessaires aux calculs ont été retenus. Le même raisonnement est appliqué sur les vecteurs étendus des différents types d'amer. Les nouvelles équations d'évolution de ces états sont décrites dans l'annexe D.

Puissance de Kronecker appliquée aux composantes du vecteur d'état d'un amer point

L'application des puissances de Kronecker sur le vecteur d'état d'un point \mathbf{X}_{pi} oblige de prendre en compte les états du véhicule (3.52). En effet, pour transformer les équations d'observation d'un amer point (3.51) en somme de polynômes, l'introduction de variables multipliant les états du véhicule à ceux de l'amer considéré est nécessaire. Ces nouvelles variables correspondent au terme $\mathbf{Z}_V \otimes \mathbf{X}_{pi}$. Le nouveau vecteur d'état d'un point s'écrit :

$$\mathbf{Z}_{pi} = \begin{bmatrix} \mathbf{X}_{pi} \\ \mathbf{Z}_V \otimes \mathbf{X}_{pi} \end{bmatrix} = \begin{bmatrix} x_{pi} \\ y_{pi} \\ \sin(\theta_v) \cdot x_{pi} \\ \cos(\theta_v) \cdot x_{pi} \\ \sin(\theta_v) \cdot y_{pi} \\ \cos(\theta_v) \cdot y_{pi} \end{bmatrix} = \begin{bmatrix} z_{pi1} \\ z_{pi2} \\ z_{pi3} \\ z_{pi4} \\ z_{pi5} \\ z_{pi6} \end{bmatrix} \quad (3.65)$$

Dans l'approche SLAM présenté, l'environnement est considéré comme statique. Seul le véhicule et donc, seuls les états définissant sa position dans la carte estimée, évoluent lors de l'étape de prédiction. Cependant, en redéfinissant les états des amers par le vecteur (3.65), les états des amers sont couplés aux états du véhicule (3.64), via le terme $\mathbf{Z}_V \otimes \mathbf{X}_{pi}$. Cette nouvelle écriture impose de faire évoluer le vecteur \mathbf{Z}_{pi} lors de la phase de prédiction du filtre de Kalman (cf. l'annexe D).

Dans cette section, seuls les amers de type point sont présentés. Les équations associées aux amers de type droite sont présentés en annexe D. Elles sont fondées sur le même raisonnement,

c'est à dire une réécriture de équations d'observation et de cartographie à partir d'un nouvelle état \mathcal{Z}_{li} . Cette réécriture impose également une réécriture des équations de prédiction des états d'une droite. En effet, à l'instar des points, le vecteur d'état \mathcal{Z}_{li} , définissant les droites, est également composé d'informations liées à la position du véhicule. Comme le véhicule évolue dans l'environnement, les nouveaux états d'une droite doivent également évoluer.

Au final, tous les vecteurs d'états, du véhicule et des amers sont désormais redéfinis. Les équations d'observation des amers vont pouvoir être linéarisées comme le prouvent les prochaines équations.

3.3.2.2 Equations d'observation

La définition de nouveaux états permet de reformuler les équations d'observation en des sommes de polynômes. Pour rappel, dans une application classique d'EKF-SLAM, les équations d'observation d'un amer point s'écrivent :

$$\mathbf{X}_{pi} = \mathbf{h}(\mathbf{X}) = \begin{bmatrix} x_i \cdot \cos(\theta_v) + y_i \cdot \sin(\theta_v) - x_v \cdot \cos(\theta_v) - y_v \cdot \sin(\theta_v) \\ y_i \cdot \cos(\theta_v) - x_i \cdot \sin(\theta_v) + x_v \cdot \sin(\theta_v) - y_v \cdot \cos(\theta_v) \end{bmatrix} \quad (3.66)$$

Ces équations sont clairement non-linéaires. En effet, la présence des sinus et des cosinus de l'angle θ_v en traduit la non-linéarité. A l'aide de la nouvelle définition des vecteurs d'états, \mathcal{Z}_V et \mathcal{Z}_{pi} , ces équations d'observation s'écrivent désormais :

$$\mathbf{X}_{pi} = \mathbf{H}_{X_{pi}} \cdot \mathcal{Z}(k+1|k) = \begin{bmatrix} z_{pi4} + z_{pi5} - z_9 - z_{10} \\ -z_{pi3} + z_{pi6} + z_8 - z_{11} \end{bmatrix} \quad (3.67)$$

Maintenant, l'équation d'observation (3.67) est composée par une somme de polynômes, tous décrits dans le vecteur d'état du système. La matrice $\mathbf{H}_{X_{pi}}$ s'écrit :

$$\mathbf{H}_{X_{pi}} = [\mathbf{H}_{\mathcal{Z}_V} \quad 0 \quad \cdots \quad 0 \quad \mathbf{H}_{\mathcal{Z}_{pi}} \quad 0 \quad \cdots \quad 0] \quad (3.68)$$

Cette matrice se décompose en deux jacobiennes, $\mathbf{H}_{\mathcal{Z}_V}$ et $\mathbf{H}_{\mathcal{Z}_{pi}}$. Ces matrices sont calculées respectivement par rapport au vecteur d'état du véhicule, $\mathcal{Z}_V(k+1|k)$, et par rapport au vecteur de l'amer considéré $\mathcal{Z}_{pi}(k+1|k)$. La première matrice jacobienne $\mathbf{H}_{\mathcal{Z}_V}$ s'écrit :

$$\mathbf{H}_{\mathcal{Z}_V} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix} \quad (3.69)$$

La deuxième matrice jacobienne $\mathbf{H}_{\mathcal{Z}_{pi}}$, dérivée de l'équation d'observation (3.71) par rapport au vecteur d'état associée au point, $\mathcal{Z}_{pi}(k+1|k)$, se formule :

$$\mathbf{H}_{\mathcal{Z}_{pi}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \quad (3.70)$$

La matrice de covariance associée à la prédiction de l'observation d'un point \mathbf{X}_{pi} s'écrit :

$$\mathbf{C}_{X_{pi}}(k+1|k) = \mathbf{H}_{X_{pi}} \cdot \mathbf{P}(k+1|k) \cdot (\mathbf{H}_{X_{pi}})^T \quad (3.71)$$

L'un des intérêts du filtre PEKF réside dans la formulation des états du système. Avec cette formulation, les équations d'observation (3.67) sont désormais linéaires par rapport au vecteur d'état du système \mathcal{Z} . Comme le prouveront les résultats de simulation, cette nouvelle écriture permet d'isoler les effets non-linéaires des équations du système et ainsi d'améliorer les estimations des incertitudes sur les positions des éléments de la carte.

Les équations de cartographie du PEKF-SLAM diffèrent de celles de l'approche classique EKF-SLAM (cf. le chapitre 2). Comme les vecteurs d'états sont agrandis, les équations de cartographie doivent être reformulées.

3.3.2.3 Equations de cartographie d'un amer point

La fonction de cartographie d'un amer correspond à la translation des coordonnées de l'amer mesuré de la position du véhicule vers le repère de la carte estimée. L'ajout d'un nouvel amer \mathbf{X}_{pn} , dans la carte estimée \mathcal{Z}_M , utilise donc la position du véhicule \mathcal{Z}_V . Les états définissant ce nouvel amer \mathcal{Z}_{pn} sont obtenus par les équations :

$$\mathcal{Z}_{pn} = \mathbf{g}(\mathcal{Z}, \mathbf{X}_{pn}) = \begin{bmatrix} x_{pn} \cdot z_5 - y_{pn} \cdot z_4 + z_1 \\ x_{pn} \cdot z_4 + y_{pn} \cdot z_5 + z_2 \\ -y_{pn} \cdot z_6 + x_{pn} \cdot z_{12} + z_8 \\ x_{pn} \cdot z_7 - y_{pn} \cdot z_{12} + z_9 \\ x_{pn} \cdot z_6 + y_{pn} \cdot z_{12} + z_{10} \\ y_{pn} \cdot z_7 + x_{pn} \cdot z_{12} + z_{11} \end{bmatrix} \quad (3.72)$$

La matrice de covariance du système, \mathbf{P}_{aug} , après l'insertion d'un nouvel amer point, est calculée à partir des matrices de covariance du système \mathbf{P} et de la mesure \mathbf{C}_{pn} :

$$\mathbf{P}_{aug} = \mathbf{G} \cdot \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{C}_{pn} \end{bmatrix} \cdot (\mathbf{G})^T \quad (3.73)$$

La matrice jacobienne \mathbf{G} correspond à :

$$\mathbf{G} = \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{G}_Z & \mathbf{G}_{pn} \end{bmatrix} \quad (3.74)$$

\mathbf{G}_Z est la matrice jacobienne de l'équation (3.72) par rapport au vecteur d'état \mathcal{Z} :

$$\mathbf{G}_Z = \begin{bmatrix} 1 & 0 & -y_{pn} & x_{pn} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & x_{pn} & y_{pn} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -y_{pn} & 0 & 1 & 0 & 0 & 0 & x_{pn} \\ 0 & 0 & 0 & 0 & 0 & x_{pn} & 0 & 1 & 0 & 0 & -y_{pn} \\ 0 & 0 & 0 & 0 & x_{pn} & 0 & 0 & 0 & 1 & 0 & y_{pn} \\ 0 & 0 & 0 & 0 & 0 & y_{pn} & 0 & 0 & 0 & 1 & x_{pn} \end{bmatrix} \quad (3.75)$$

\mathbf{G}_{pn} est la matrice jacobienne de l'équation (3.72) par rapport au vecteur \mathbf{X}_{pn} :

$$\mathbf{G}_{pn} = \begin{bmatrix} z_4 & -z_3 \\ z_3 & z_4 \\ z_{11} & -z_5 \\ z_6 & -z_{11} \\ z_5 & z_{11} \\ z_{11} & z_6 \end{bmatrix} \quad (3.76)$$

Cette étape de cartographie permet l'insertion d'un nouvel amer dans la carte estimée en agrandissant le vecteur \mathcal{Z} et la matrice de covariance \mathbf{P} . Lors de cette phase, les corrélations entre les différents états du système sont initialisées. Ces corrélations jouent un rôle essentiel lors de la mise à jour de la carte. Elles permettent la correction de l'ensemble de la carte à partir d'une seule mesure.

L'étude des équations de cartographie d'un point (3.72) et d'une droite (cf. l'annexe D) permet de souligner un fait important. Les états du véhicule, \mathcal{Z}_V , ne sont pas tous corrélés aux états de l'amer cartographié. Par exemple, les états z_1 à z_5 ne sont corrélés qu'avec les états z_{pi1} à z_{pi2} pour un point et z_{li1} à z_{li5} pour une droite. Des remarques similaires peuvent être formulées pour les autres états de ces vecteurs. Comme le prouvent les résultats des simulations, ces différentes corrélations permettent de réduire certains problèmes posés par la non-linéarité des équations d'un système SLAM.

Dans les paragraphes suivants, ce filtre est appliqué sur des données simulées puis réelles. Les résultats permettent de vérifier si cette nouvelle formulation des équations non-linéaires atténue les problèmes de consistance rencontrés avec l'approche classique EKF-SLAM.

3.4 Simulations : évaluation des propriétés du filtre PEKF-SLAM

Pour évaluer l'application PEKF-SLAM, les mêmes simulations sont réalisées pour des amers de type point et droite. Les résultats permettront de répondre aux questions suivantes :

- est-ce que les théorèmes non-linéaires sont respectés dans des cas d'études simples ?
- est-ce que l'application PEKF-SLAM améliore la précision et la consistance des estimations lors de la fermeture de la trajectoire du véhicule ?

Les réponses permettent de conclure sur les avantages et les inconvénients de cette nouvelle approche. Le dernier paragraphe de ce chapitre propose des résultats sur des données expérimentales dans le but de vérifier l'applicabilité du filtre dans un contexte réel.

3.4.1 Comportement des filtres sur des cas d'études simples

Dans la section précédente, trois théorèmes non-linéaires de convergence des estimations du filtre EKF-SLAM sont proposés. Ils soulignent l'importance de la valeur de l'incertitude dans le comportement des estimations du filtre. Le but de ces simulations est de quantifier l'action de la valeur initiale de ce paramètre sur la convergence des estimations des filtres EKF et PEKF.

Dans un premier temps, la valeur de l'incertitude sur l'orientation du véhicule sera faible. Elle permettra de vérifier le comportement quasi linéaire du filtre EKF-SLAM dans un tel contexte. Ensuite, cette valeur initiale sera augmentée dans le but de mettre en avant le problème de consistance des estimations de l'EKF-SLAM mais également les avantages du filtre PEKF-SLAM. Au final, une étude approfondie des matrices de corrélations obtenues, couplée à l'étude des équations du système (cf. l'annexe D) permet d'expliquer le pourquoi des améliorations apportées par le filtre PEKF.

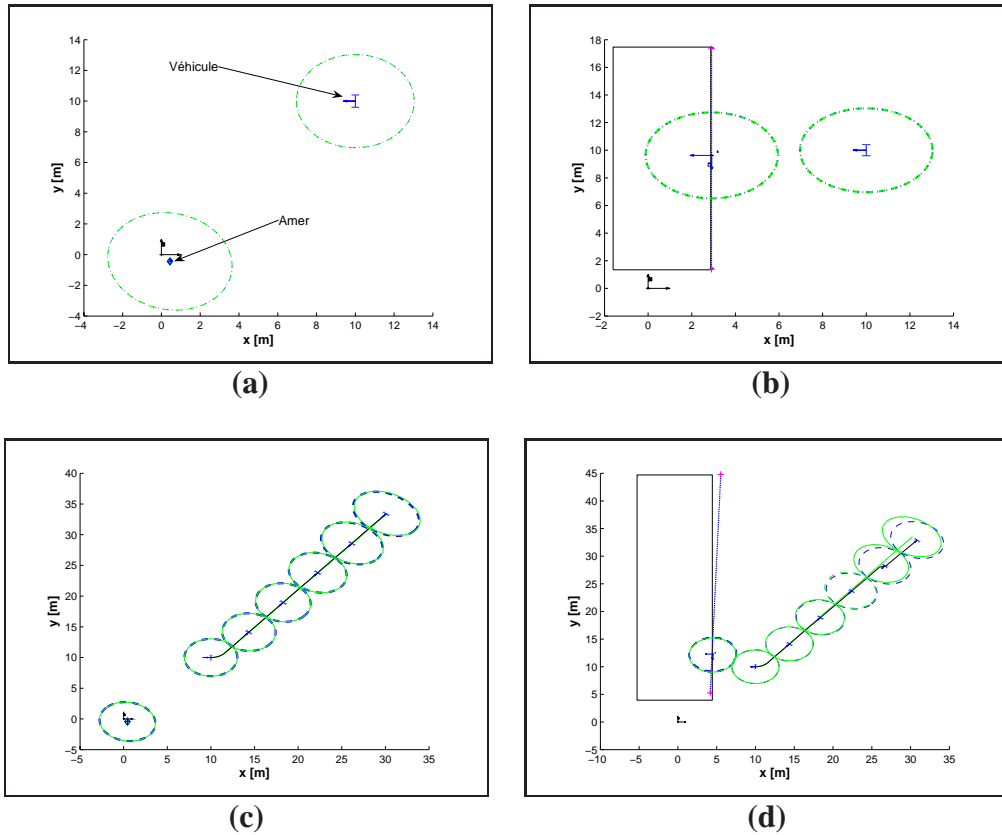


Figure 3.7: Représentation de la position et des ellipses d’incertitude lorsque véhicule est stationnaire et observe soit un point (a), soit une droite (b) et lorsque le véhicule se déplace d’un point A à un point B et observe soit un point (a), soit une droite (b). La matrice de covariance associée à la position du robot est initialisée avec la matrice $\mathbf{P}_0 = \text{diag} [1, 1, 0.001]$. Pour chaque figure, les valeurs des incertitudes calculées avec les positions réelles sont représentées en vert et celles calculées avec les valeurs estimées en bleu.

3.4.1.1 Comportement linéaire : incertitude sur l’orientation faible

Avant de comparer les résultats des filtres EKF-SLAM et PEKF-SLAM, nous allons vérifier si le filtre EKF a le comportement attendu lorsque l’incertitude sur l’orientation du véhicule est faible ($p_{\theta_v} = 0.001$). Nous allons donc vérifier si les théorèmes non-linéaires précédemment énoncés sont respectés.

Simulations lorsque le véhicule est statique :

Dans cette simulation, le véhicule observe toujours le même amer (un point ou une droite) en restant stationnaire. Les matrices jacobiniennes sont calculées avec les valeurs estimées des états. Les résultats sont présentés sur la figure 3.7. Les aires des incertitudes réelles (vertes) et estimées (bleues) sont semblables. Ce résultat confirme les observations réalisées dans les articles [11] [30] [89]. Les estimations du filtre de Kalman sont donc consistantes et respectent les théorèmes précédemment énoncés.

Simulations lorsque le véhicule se déplace :

Comme dans le cas où le véhicule est stationnaire, lorsque l'incertitude sur l'orientation du véhicule est faible, les phénomènes d'inconsistance peuvent être négligés que ce soit avec un amer de type point ou un amer de type droite (cf. la figure 3.7). Les aires des ellipses d'incertitudes idéales (vertes) et estimées (bleues) sont quasiment égales. Le **théorème 3.1** s'applique dans ce cas particulier.

Conclusion :

Que le véhicule soit mobile ou stationnaire et qu'il observe soit un point ou droite, lorsque l'incertitude sur l'orientation du véhicule est faible, le comportement du filtre EKF-SLAM est identique au comportement idéal (lorsque les jacobiennes sont idéales). La preuve en est apportée par les résultats précédents. De plus, une démonstration mathématique explique ces résultats (cf. l'annexe D). Elle repose sur l'étude de l'équation de l'innovation du filtre EKF-SLAM. Elle prouve qu'il y a un gain d'information non désiré sur les incertitudes du système. La valeur de ce gain est fonction de la valeur de l'incertitude sur l'orientation du véhicule. Si cette valeur est faible, comme dans les simulations précédentes, le gain d'information est négligeable.

3.4.2 Comportement non-linéaire : incertitude sur l'orientation importante

Dans les simulations suivantes, l'incertitude sur l'orientation du véhicule est désormais importante ($p_{\theta_v} = 0.03$). Les scénarii des simulations sont identiques aux précédents.

Simulations lorsque le véhicule est statique :

Dans cette étude, le phénomène d'inconsistance des estimations de l'EKF-SLAM est visible. Sur la figure 3.8, les ellipses d'incertitude obtenues à partir des valeurs estimées (ellipses bleues) sont largement inférieures aux valeurs théoriques (ellipses vertes). Cette constatation confirme les conclusions de l'étude proposée dans l'annexe D. Le gain d'information, provenant de la non-linéarité des équations du système n'est plus négligeable. Les résultats du filtre deviennent clairement inconsistants et prouvent que les **théorèmes 2.1** et **3.1** ne sont pas respectés. En revanche, les incertitudes sur les positions du véhicule et des amers (point ou droite), obtenues avec le filtre PEKF-SLAM (ellipses rouges), sont similaires à celles obtenues avec les jacobiennes idéales (ellipses vertes). Cette observation souligne que ce filtre répond aux **deux théorèmes 2.1** et **3.1** quand le véhicule est statique et observe soit un point, soit une droite.

Simulations lorsque le véhicule se déplace :

Lorsque le véhicule se déplace en observant un seul amer, le **théorème 2.1** n'est pas respecté (cf. la figure 3.8) avec l'application du filtre EKF-SLAM. Les incertitudes calculées à partir des estimations (ellipses bleues) sont inférieures aux valeurs réelles (ellipses vertes). Les estima-

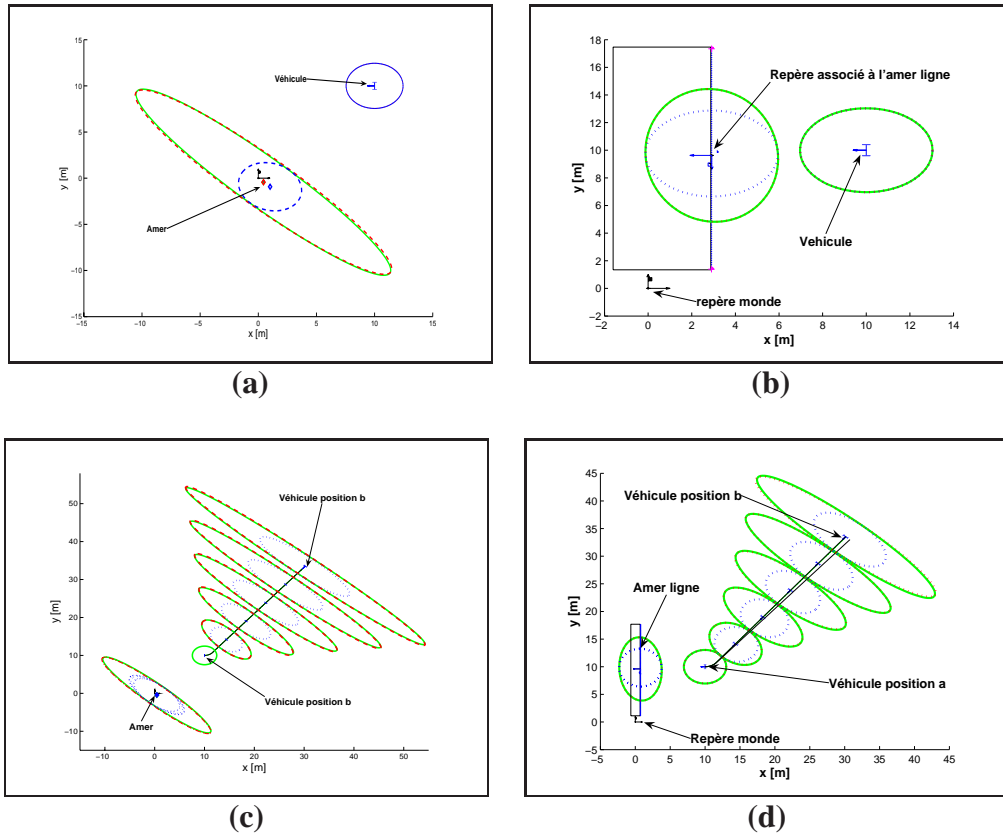


Figure 3.8: Représentation de la position et des ellipses d’incertitude pour deux cas d’étude différents. Le véhicule est stationnaire et observe un amer avec une incertitude initiale de $\mathbf{P}_0 = \text{diag} [1, 1, 0.03]$: (a) observation d’un point et (b) observation d’une droite. Le véhicule se déplace d’un point A à un point B et observe un seul amer : (c) observation d’un point et (d) observation d’une droite. Les ellipses vertes représentent l’incertitude calculée à partir des jacobiennes idéales, les ellipses bleues, celles calculées avec le filtre EKF-SLAM et les rouges correspondent à celles obtenues avec le filtre PEKF-SLAM.

tions obtenues par le filtre EKF sont inconsistantes. En revanche, les incertitudes sur la position du véhicule et de l’amer calculées par le filtre PEKF-SLAM (ellipses rouges) sont une nouvelle fois similaires aux valeurs obtenues avec les matrices jacobiennes idéales (ellipses vertes). Lorsque le véhicule se déplace, le filtre PEKF-SLAM répond également aux **deux théorèmes 2.1 et 3.1** définis précédemment.

Conclusion :

Dans les deux cas d’études, les résultats obtenus avec l’approche PEKF-SLAM sont meilleurs que ceux obtenus avec l’approche EKF-SLAM (ellipses bleues). Dans la suite de ce chapitre, une analyse des équations du filtre PEKF et des résultats apporte une explication à cette amélioration.

3.4.3 Analyses et études des corrélations

Pour le filtre EKF-SLAM, que le véhicule soit stationnaire ou mobile, dès que l'incertitude sur l'orientation du véhicule n'est plus négligeable, les estimations du filtre deviennent inconsistantes. En revanche, le filtre PEKF-SLAM répond aux théorèmes de convergence. Pourtant, cette approche calcule les matrices jacobienne à partir d'estimations de paramètres. Comme ces estimations sont corrigées à chaque mise à jour du filtre, les valeurs des jacobienne diffèrent d'une itération à l'autre du filtre. La linéarisation des équations est donc inexacte. Les mêmes phénomènes d'inconsistance rencontrés avec le filtre EKF devraient être observés. Mais, comme le montrent les résultats des simulations, le phénomène d'inconsistance des estimations n'est pas constaté. L'étude des matrices jacobienne d'observation des deux types d'amers couplée à une analyse des corrélations permettent d'apporter une explication. Prenons comme exemple les corrélations obtenues lorsque le véhicule se déplace et observe un seul amer. Les résultats du filtre EKF et du filtre PEKF sont présentés sur la figure 3.9.

Résultats avec l'approche EKF-SLAM :

Dans le cas de l'application du filtre EKF-SLAM (figures 3.9 (a) et (c)), une forte corrélation entre l'orientation du véhicule θ_v et les autres états de la carte estimée est constatée. Cette corrélation est la conséquence de la non-linéarité des équations du système (prédiction, observation et cartographie). Elle est logique d'un point de vue mathématique mais accentue l'inconsistance des estimations due à la non-linéarité du système. En effet, les erreurs réalisées lors de la linéarisation sont synonymes d'informations incohérentes. Elles sont modélisées par des valeurs de jacobienne différentes d'une itération à l'autre du filtre EKF. Lors de l'étape de mise à jour, ces informations erronées sur l'orientation du véhicule, θ_v , sont répercutées sur les estimations de toute la carte via les corrélations précédemment observées. Après seulement quelques itérations du filtre, les estimations de l'incertitude sur la position du véhicule et des amers deviennent trop optimistes.

Linéariser les équations du système à partir de valeurs exactes est la solution idéale. Cependant, cette hypothèse n'est pas concevable dans une application SLAM si aucune information sur l'environnement n'est disponible. Cependant, l'application PEKF-SLAM propose une alternative dans le but d'atténuer ce problème.

Résultats avec l'approche PEKF-SLAM :

Les matrices de corrélation obtenues avec le filtre PEKF-SLAM sont présentées sur les figures 3.9 (b) et (d). Dans les deux cas d'étude, certains états ne sont que très faiblement corrélés entre eux. Pour le véhicule, les états z_1 à z_5 , décrivant sa position dans la carte estimée, sont très faiblement corrélés avec les autres états z_6 à z_{12} qui décrivent la non-linéarité des équations du système. Comme pour le véhicule, les états z_{pi1} et z_{pi2} d'un amer point sont faiblement corrélés aux états, dits non-linéaires, z_{pi3} à z_{pi6} (figure 3.9 (b)). De même pour l'amer ligne, les états z_{li1} à z_{li5} sont faiblement corrélés aux états non-linéaires z_{li6} à z_{li17} (figure 3.9 (d)). Ces décorrélations entre ces différents états proviennent de la réécriture des équations du filtre EKF en sommes de polynômes. En effet, les équations d'observation sont uniquement fonc-

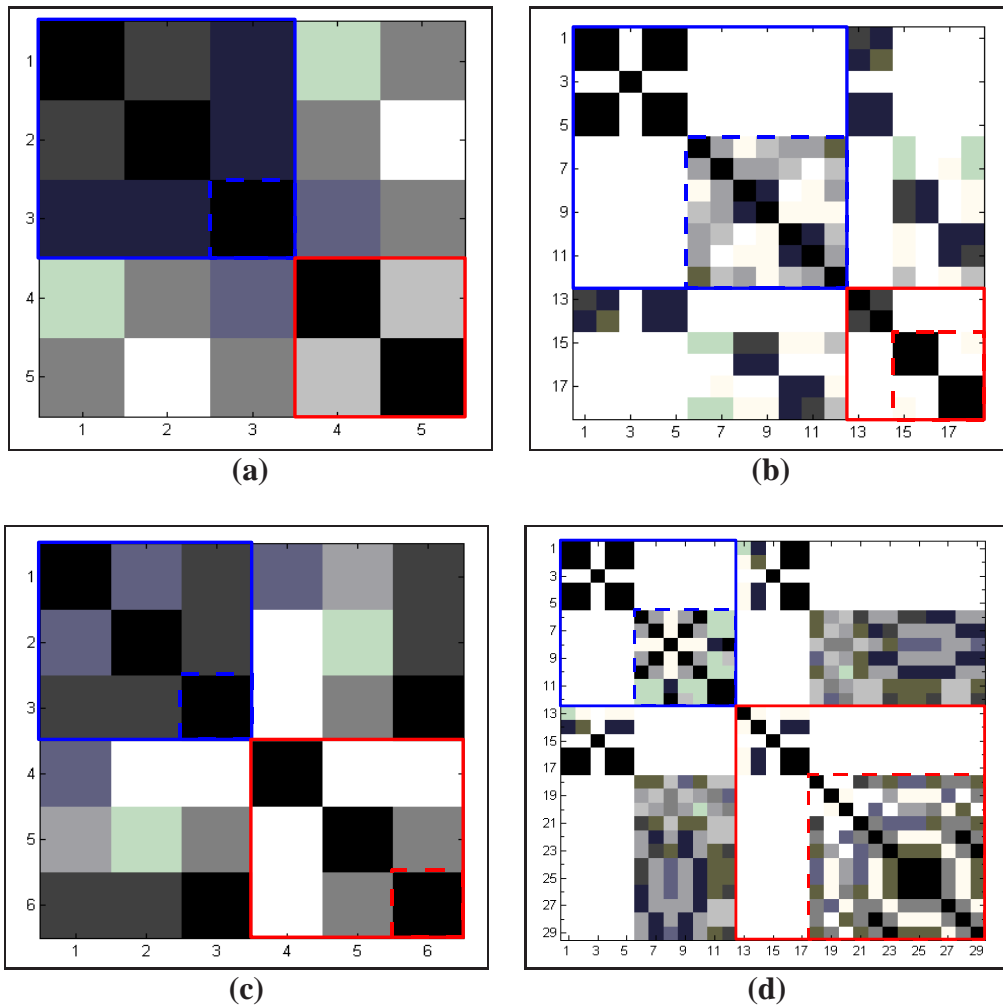


Figure 3.9: Matrice de corrélation pour le cas d'étude où le véhicule se déplace d'un point A à un point B et observe un point : (a) filtre EKF et (b) filtre PEKF. Matrice de corrélation pour le cas d'étude où le véhicule se déplace d'un point A à un point B et observe une droite : (c) filtre EKF et (d) filtre PEKF. Les cases de couleur foncée représentent une forte corrélation entre deux états et, inversement, une couleur claire représente une faible corrélation entre deux états.

tions des états z_6 à z_{12} pour le véhicule, z_{pi3} à z_{pi6} pour un amer point et z_{li6} à z_{li17} pour un amer ligne. Les erreurs introduites à cause de la linéarisation de ces équations n'affectent que les états non-linéaires. Les estimations des états décrivant la position du véhicule et des amers restent correctes comme le prouvent les résultats de simulation.

L'étude des fonctions d'évolution des états permet de compléter le raisonnement. Lors de cette étape, la matrice du système \mathbf{P} est modifiée avec l'ajout du bruit sur les commandes du système. Comme le déplacement du véhicule est modélisé par une fonction non-linéaire, des corrélations entre les états du système sont créées. En particulier, des corrélations entre les états linéaires et ceux non-linéaires. Ce sont ces corrélations qui permettront une correction de la carte estimée qui sera donc uniquement proportionnelle au bruit sur les commandes du véhicule. Comme dans notre exemple, les bruits sur les données proprioceptives sont faibles, les corrélations entre ces différents états sont faibles également.

Le filtre PEKF-SLAM garantit une estimation correcte des incertitudes décrites dans la matrice de covariance \mathbf{P} en comparaison à l'approche classique EKF-SLAM. Comme les incertitudes sur les états des éléments de la carte sont très proches des valeurs idéales, il est donc envisageable d'améliorer l'étape d'association de données du filtre (cf. l'annexe D).

Nous venons d'étudier des résultats de simulation sur des scenarii simples. Ceci nous a permis de mettre en lumière les comportements des filtres dans des configurations différentes. Maintenant, leur comportement lors de la fermeture de la trajectoire du véhicule va être étudié.

3.4.4 Analyse de la fermeture de la trajectoire du véhicule

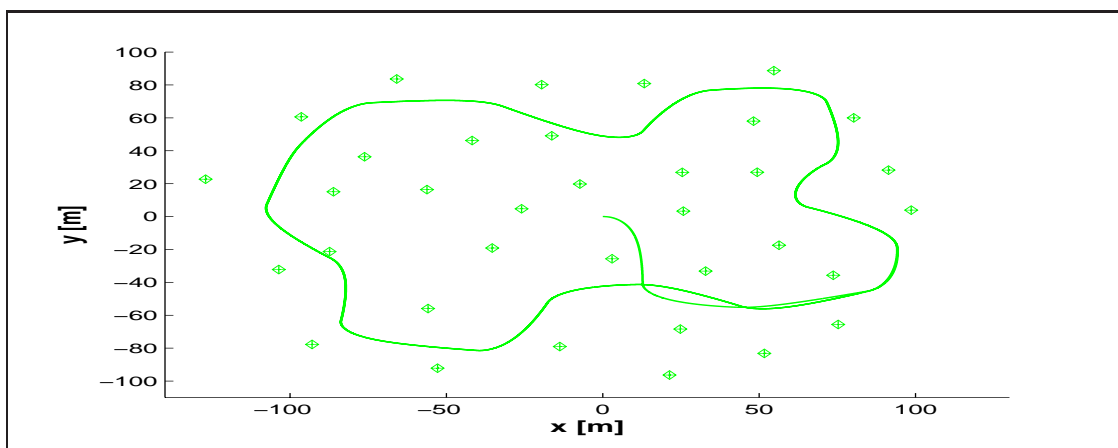


Figure 3.10: Trajectoire et carte simulées pour étudier l'évolution de la consistance des approches EKF-SLAM. La carte est constituée uniquement de points.

Une fermeture correcte de la trajectoire du véhicule est l'un des principaux objectifs d'une application SLAM. Lorsque le système est capable de ré-observer des amers cartographiés au début de la trajectoire du véhicule, les estimations convergent. Pour étudier les comportements des deux approches, la carte est composée uniquement d'amers de types points (cf. la figure 3.10).

3.4.5 Trajectoire du véhicule sans bouclage

Avant de présenter des résultats de simulation avec une fermeture de la trajectoire du véhicule, une simulation sans fermeture de la trajectoire est proposée pour souligner l'intérêt de l'emploi du filtre PEKF-SLAM.

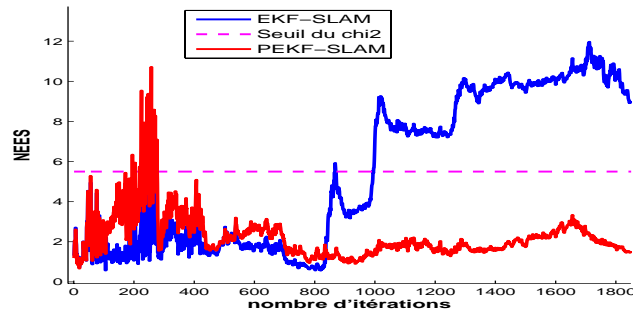


Figure 3.11: Evolution de la distance NEES lorsque le véhicule suit une trajectoire non bouclée dans une carte de points : la courbe bleue représente les résultats de l'approche EKF-SLAM et celle de couleur rouge ceux de l'approche PEKF-SLAM.

Le scénario de simulation est simple. Le véhicule avance tout droit dans un environnement avec des amers répartis aléatoirement autour de sa trajectoire. L'évolution de la distance NEES est présentée sur la figure 3.11. Dans un tel scénario, l'erreur sur la position du véhicule ne devrait cesser d'augmenter tout comme l'incertitude définie dans la matrice de covariance. Si tel est le cas, la distance NEES doit rester à des valeurs inférieures au seuil défini par la distribution du χ^2 (représenté par une ligne mauve).

En observant l'évolution de la valeur de la distance NEES pour l'approche EKF-SLAM, nous remarquons que les estimations deviennent inconsistantes à partir de 1000 itérations du filtre (courbe bleue). Ceci s'explique facilement en reprenant les conclusions du paragraphe précédent. En effet, si la valeur de l'incertitude sur l'orientation du véhicule n'est plus négligeable, un gain d'information erroné est observé sur les incertitudes du système. Dans une telle application, l'incertitude sur cet état croît jusqu'à ne plus être négligeable. Dès lors, les valeurs des estimations deviennent inconsistantes comme le prouvent les résultats. Ce phénomène n'est pas observé lors de l'emploi du filtre PEKF-SLAM. L'évolution de la distance NEES (courbe rouge) reste inférieure au seuil après de nombreuses itérations du filtre.

Ce résultat confirme les observations effectuées précédemment pour le filtre EKF-SLAM mais également les démonstrations présentées pour le filtre PEKF-SLAM. Maintenant, le comportement du filtre PEKF-SLAM est étudié dans un contexte de fermeture de trajectoire du véhicule.

3.4.5.1 Trajectoire avec bouclage : carte avec des amers de type point

Dans ce scénario de simulation, le véhicule réalise deux boucles de la trajectoire avec une vitesse de 3 ms^{-1} . L'incertitude initiale sur les états du véhicule est nulle. Avec ce réglage, les problèmes d'inconsistance du filtre EKF-SLAM, rencontrés aux paragraphes précédents,

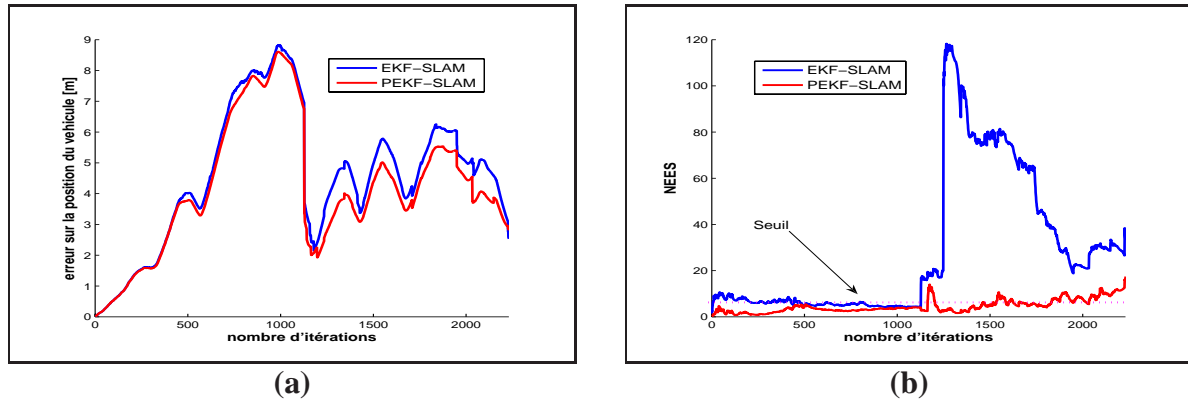


FIGURE 3.12: Simulation lorsque le véhicule réalise une trajectoire bouclée dans un environnement composé de points. L'évolution de l'erreur sur la position du véhicule est représentée sur la figure (a) et l'évolution de la distance NEES sur la figure (b). Les courbes bleues représentent les résultats de la méthode EKF-SLAM et les courbes rouges ceux de la méthode PEKF-SLAM.

	Valeurs des bruits				Incertitudes initiales		
	σ_ρ [m]	σ_β [rad]	σ_v [ms ⁻¹]	σ_φ [rad]	σ_x [m]	σ_y [m]	σ_θ [rad]
Simulations	1	$3.\pi/180$	0.6	$6.\pi/180$	10^{-5}	10^{-5}	10^{-5}

TABLE 3.2: bruits sur les données proprioceptives et extéroceptives

peuvent donc être négligés. Les valeurs des bruits sur les mesures extéroceptives et proprioceptives sont présentées dans le tableau 3.2. Une séquence de 20 simulations est réalisée pour mesurer la consistance des estimations.

Pour les deux approches, l'unique analyse de l'évolution de l'erreur permettrait de conclure que chaque système réalise une fermeture correcte de la trajectoire du véhicule. En effet, lorsque les systèmes ré-observent les amers cartographiés en début de trajectoire (itérations 230), l'erreur diminue. Puis, l'erreur sur la position du véhicule lors de la deuxième boucle reste inférieure à la première boucle. Pourtant, cette conclusion est erronée.

L'étude de l'évolution de la distance NEES permet d'améliorer ces conclusions (cf. la figure 3.12 (b)). Pour l'approche EKF-SLAM, l'évolution de cette valeur (courbe bleue) prouve clairement que les estimations sont inconsistantes lors de la première fermeture de la trajectoire du véhicule. En effet, lorsque le véhicule boucle sa trajectoire, la valeur de la distance NEES devient supérieure au seuil défini par la distribution du χ^2 (droite violette). La condition de consistance des estimations n'est plus respectée. Ce phénomène s'explique en reprenant l'exemple où le véhicule est stationnaire et observe un unique amer n fois. L'inconsistance des estimations était observée lorsque la valeur de l'incertitude sur l'orientation du véhicule n'était pas négligeable. La fermeture de la trajectoire correspond au même problème. Durant la visite de l'environnement, l'incertitude sur l'orientation augmente. Quand le système ré-observe des amers cartographiés, cette valeur est trop importante pour être négligée. Les **théorèmes 2.1** et **3.1** ne sont plus respectés. Les estimations sont donc incorrectes.

Les résultats obtenus avec le filtre PEKF-SLAM (courbe rouge) mettent en évidence plusieurs phénomènes. Premièrement, à l'instar du filtre EKF-SLAM, les estimations du filtre restent consistantes durant la première boucle du véhicule. La valeur de la distance NEES est même inférieure à celle du filtre EKF-SLAM. Ceci prouve que les états du véhicules et les états dits non-linéaires restent faiblement corrélés. En revanche, lors de la fermeture de la trajectoire du véhicule, la valeur de la distance NEES commence à osciller et à devenir supérieure au seuil. Les estimations du filtre PEK-SLAM deviennent donc inconsistantes. Les corrélations entre les différents états deviennent trop importantes. Cependant, les valeurs de la distance NEES sont largement inférieures à celles obtenues avec l'application EKF. De surcroit, elles sont très proches de la valeur du seuil. Cette amélioration a une influence dans la précision des estimations. L'erreur sur la position du véhicule est moins importante lors de la revisite de l'environnement. De plus, comme les valeurs de l'incertitude sont correctes, **les mises en correspondances des amers ainsi que la valeur de la correction calculée par le filtre de Kalman sont plus précises.**

L'application PEKF-SLAM améliore donc les résultats en termes de consistance des estimations mais également en termes de précision. Cependant, ce filtre ne résout pas totalement le problème d'inconsistance des estimations lors de la fermeture de la trajectoire du véhicule. Il ne fait que l'atténuer.

3.4.6 Conclusion sur les résultats de simulation

Le paragraphe 3.4 vient de présenter une étude de l'application du filtre EKF-SLAM dans différents scénarii. Les résultats de simulation prouvent que cette application n'est pas idéale et certains problèmes restent à résoudre. Premièrement, lorsque l'incertitude initiale sur l'orientation du véhicule n'est pas nulle, les estimations de l'incertitude sur les positions des éléments de la carte deviennent optimistes. Deuxièmement, lorsque le véhicule clôt sa trajectoire, après plusieurs itérations, les estimations deviennent là encore inconsistantes comme le prouvent l'évolution de la distance NEES comparée au seuil du χ^2 .

Dans la littérature, il existe des adaptations du filtre de Kalman permettant de diminuer les problèmes de non-linéarité telles que le filtre UKF pour *Unscented Kalman Filter* [88] et application itérative du filtre de Kalman, IEKF pour *Iterated Extended Kalman Filter* [12]. Cependant, leurs applications dans un contexte SLAM ne résolvent pas les problèmes présentés dans ce chapitre. Les résultats présentés dans notre article [34] le prouvent.

Les résultats de simulation prouvent que le filtre PEKF-SLAM est plus performant que le filtre EKF-SLAM classique. Lorsque l'incertitude initiale sur l'orientation du véhicule est importante, les estimations restent consistantes à la différence de celles du filtre EKF. Cependant, lors de la fermeture de la trajectoire du véhicule, nous remarquons que les estimations deviennent quand même inconsistantes. Cependant, cette inconsistance sur les estimations du filtre PEKF est moins importante que le filtre classique. Ceci permet d'obtenir de meilleures estimations de la position du véhicule. Dans le prochain paragraphe, les estimations de ces deux filtres sont comparées sur des données réelles. L'objectif est de démontrer que ce filtre est applicable sur des données réelles.

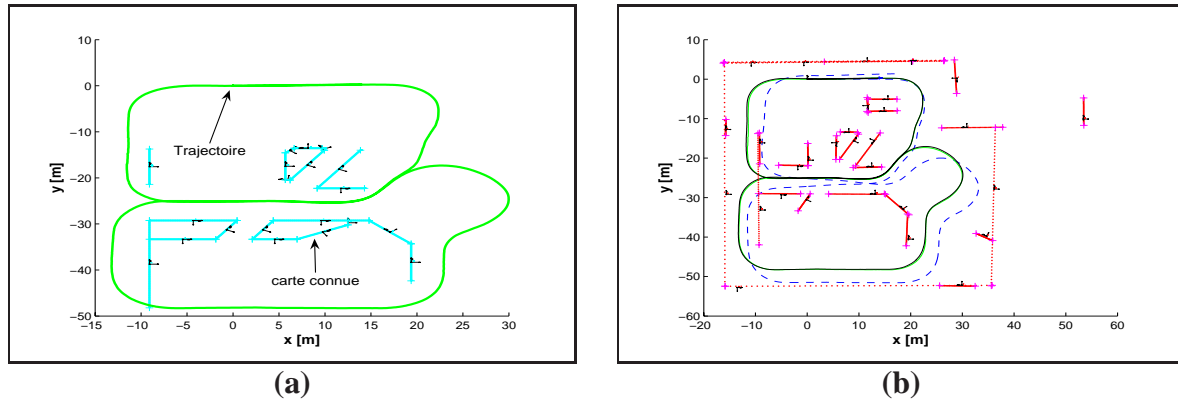


Figure 3.13: (a) Vérité de terrain de l'expérimentation réalisée sur site PAVIN et (b) carte estimée par la méthode PEKF-SLAM

3.5 Expérimentations

Les données expérimentales utilisées dans ce paragraphe ont été recueillies sur le site expérimental nommé PAVIN (cf. l'annexe E). Cet environnement se porte bien à l'application de l'algorithme de SLAM avec des cartes hétérogènes. Il est composé principalement de pans de murs représentant des petites maisonnettes. Ce site expérimental est plat et propice à une cartographie en deux dimensions. L'avantage sera donc de pouvoir se focaliser sur le comportement des estimations des filtres. L'ensemble des paramètres d'initialisation sont résumés dans l'annexe E.

Dans cette expérimentation, la vérité de terrain est fournie par un capteur DGPS (précision de 5 cm dans des conditions optimales) et par un plan d'architecte. La figure 3.13 présente la trajectoire empruntée par le véhicule (tracé vert) et une partie de l'environnement visitée (éléments de couleur bleue).

La carte estimée par le filtre PEKF-SLAM, ainsi que la trajectoire du véhicule, sont présentées sur la figure 3.13. La dérive des données odométriques (tirets bleus) est correctement corrigée. De plus, la trajectoire du véhicule estimée par le filtre PEKF est similaire à celle mesurée par le capteur DGPS. La comparaison avec la vérité de terrain prouve que les résultats sont cohérents et relativement proches de la réalité. Pour quantifier ces résultats, une analyse de l'évolution de l'erreur, sur la position du véhicule, est réalisée.

Les figures 3.14 (a) et (b) représentent respectivement les résultats obtenus avec le filtre EKF-SLAM et PEKF-SLAM. Pour les deux approches, les erreurs sur les coordonnées cartésiennes du véhicule restent bornées par les valeurs des incertitudes (2 écart-types) définies dans les matrices de covariance. Cependant, des différences sont visibles. Les erreurs obtenues avec le filtre EKF dépassent avec de plus grandes amplitudes les bornes d'incertitudes. Ce phénomène explique la différence de précision des deux filtres (cf. la figure 3.11). En effet, la comparaison de l'évolution des erreurs (cf. la figure 3.15) prouvent que le filtre PEKF réalise une fermeture de la trajectoire légèrement plus précise. Ces résultats expérimentaux sont en concordance avec les résultats obtenus avec les données simulées.

Ces résultats prouvent la validité de l'application PEKF-SLAM sur des données réelles. Cependant, le gain en précision est seulement de 1 cm de moyenne. A contrario, le nombre

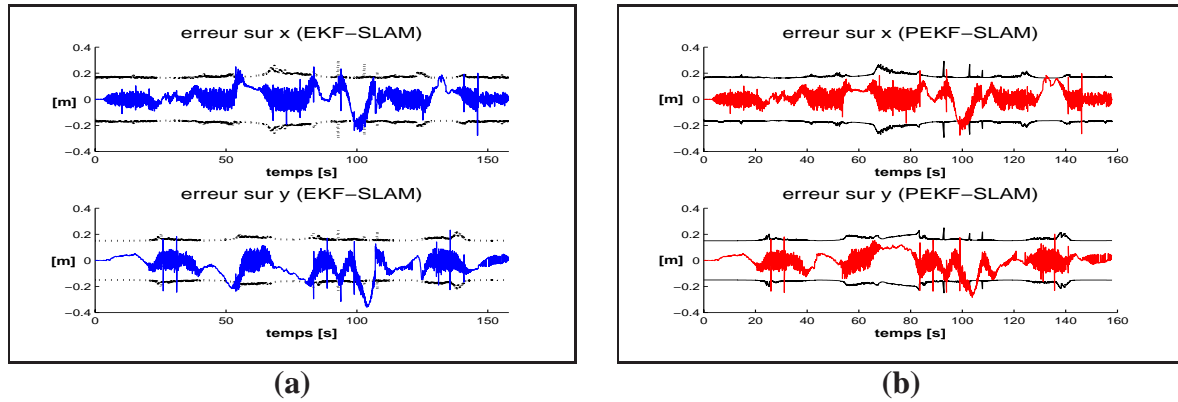


Figure 3.14: Evolution des erreurs sur les coordonnées cartésiennes du véhicule avec les bornes d'incertitude définies à 2 écart-types : (a) filtre EKF-SLAM et (b) filtre PEKF-SLAM

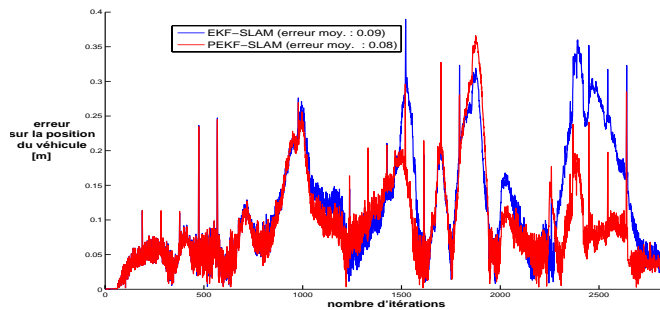


Figure 3.15: Evolution de l'erreur sur la position du véhicule : filtre EKF-SLAM (courbe bleue) et filtre PEKF-SLAM (courbe rouge)

d'états du système est multiplié par trois en comparaison aux nombres d'états d'une approche classique. Ceci implique une importante augmentation des temps de calcul. Cette remarque relativise donc le gain en précision dans un contexte réel.

3.6 Conclusion

Ce chapitre présente une application d'un filtre polynomial dans un contexte SLAM *World-centric* : le filtre **PEKF-SLAM**. Il repose sur une nouvelle écriture des vecteurs d'état du système et des équations non-linéaires du système.

Les résultats de simulation soulignent l'apport d'une telle application en termes de consistance des estimations. Cette approche répond aux différents théorèmes définis dans ce chapitre. Les résultats sur des trajectoires bouclées (simulées et expérimentales) prouvent que ce filtre est plus précis qu'un filtre EKF classique. Cependant, des inconvénients liés à son utilisation sont à souligner. Premièrement, cette solution reste une solution analytique encore difficilement applicable en temps réel aux vues du nombre d'états utilisés pour définir ce système. Une solution efficace pour réduire ces temps de calcul serait l'application d'une méthode de cartes locales

au système PEKF-SLAM. Deuxièmement, le problème de la fermeture de la trajectoire du véhicule pose encore des problèmes du point de vue de la consistance des estimations. Même si les tests statistiques prouvent que les estimations du filtre PEKF-SLAM sont plus consistantes que ceux de l'approche EKF-SLAM, des améliorations doivent être apportées. Une étude plus approfondie des phénomènes, engendrés lors du bouclage de la trajectoire, doit être réalisée à l'instar des travaux dans l'article [109] qui présentent des solutions permettant d'exploiter cette information.

Pour améliorer les résultats d'une approche EKF-SLAM et atténuer les problèmes de consistances, plusieurs solutions sont également envisageables :

- augmenter l'ordre de la linéarisation de Taylor dans le filtre PEKF ;
- employer une approche de type *RobotCentric* (cf. l'annexe F) ;
- appliquer une approche de cartes locales ;
- calculer les matrices jacobiennes d'observation à partir de la mesure utilisée lors de la cartographie de l'amer considéré [82].

La première solution correspondrait à une augmentation importante des temps de calcul. Une implémentation en temps réel ne serait plus envisageable. Cependant, comme le prouvent les résultats dans l'article [71], une amélioration des résultats devraient être observée.

Pour la deuxième solution, les résultats présentés dans l'article [112] prouvent l'efficacité de cette approche. Pour se faire, un changement de représentation de l'environnement est réalisé. L'avantage de cette représentation, centrée sur la position du robot, est de réduire les erreurs réalisées lors de la linéarisation des équations non-linéaires du système. Comme les linéarisations des équations sont plus exactes, les estimations sont consistantes et précises. Coupler cette représentation à l'approche PEKF-SLAM permettrait de résoudre les problèmes d'inconsistance lors de la fermeture de la trajectoire du véhicule.

La troisième solution permet, en quelque sorte, de « retarder » l'inconsistance des estimations. En divisant la carte estimée de l'environnement en plusieurs cartes de dimension inférieure, l'incertitude sur l'orientation du véhicule reste faible. Il faut souligner que cette approche souffre des mêmes problèmes que l'approche EKF-SLAM. En effet, si l'incertitude initiale sur l'orientation du robot est importante, les problèmes d'inconsistance seront toujours existants. Cependant, une approche de cartes locales avec le filtre PEKF-SLAM permettrait de réduire les temps de calcul et d'apporter au système une estimation plus consistante des paramètres.

La quatrième solution semble être la meilleure pour résoudre le problème observé. Les résultats présentés dans l'article [82] l'attestent. De surcroît, à la différence de l'augmentation de l'ordre du filtre PEKF, aucun calcul supplémentaire n'est réalisé. En effet, grâce à une démonstration fondée sur l'observabilité du système, les auteurs ont prouvé qu'il ne fallait pas employer les estimations courantes du système pour linéariser les équations du système. Pour obtenir des estimations semblables à une approche idéale (linéarisation faite avec les valeurs exactes), la solution consiste à employer les valeurs des positions des amers lors de leur cartographie dans la carte estimée. Grâce à ce principe, les estimations restent consistantes lorsque le véhicule parcourt une très longue distance, à l'instar de l'approche PEKF-SLAM, mais également, lorsque le véhicule boucle sa trajectoire et réobserve des amers.

Dans le but de répondre au problème observé, une autre approche est envisagée. Les travaux présentés dans le chapitre 4 proposent de contraindre les estimations de la carte par la connais-

114 Etude de la convergence et la consistance des estimations du filtre polynomial de Kalman

sance d'informations sur l'environnement. L'objectif est de minimiser les erreurs faites lors de la linéarisation des équations du système avec des estimations plus précises.

Chapitre 4

Méthode SLAM contrainte par une carte connue de l'environnement

4.1 Introduction

L'approche présentée dans ce chapitre propose d'employer des cartes de l'environnement visité par le véhicule. Ces cartes peuvent correspondre à plusieurs types de données : des relevés GPS de positions d'amers, des cartes SIG ou des plans d'architecte. Elles sont généralement imprécises et incomplètes.

Dans cette approche, le véhicule doit se localiser et cartographier l'environnement dans lequel il évolue, en connaissant une carte de celui-ci. Cette carte répertorie les positions, plus ou moins précises, de certains éléments de l'environnement. Cette approche, nommée **SLAM-contraint**, réalise donc des estimations contraintes par la connaissance d'une carte a priori de l'environnement. L'objectif est de proposer aux lecteurs une étude mathématique rigoureuse de cette méthode.

Tout d'abord, ce chapitre introduit les conditions d'observabilité d'une approche SLAM basée sur une représentation *WorldCentric* de l'environnement. Dans cette approche, tous les amers de la carte estimée sont représentés par rapport à la première position du véhicule (cf. l'annexe F). Après des rappels sur les conditions d'observabilité des systèmes, l'observabilité de cette approche SLAM est calculée. Son analyse conduit aux conclusions suivantes :

- un système SLAM basé sur une représentation *Worldcentric* n'est que partiellement observable ;
- un système SLAM doit employer des informations connues sur l'environnement ou sur la position du véhicule pour être observable.

Après un état de l'art sur les méthodes répondant à cette problématique, l'approche **SLAM-contraint** est décrite. Son étude théorique démontre que cette méthode est localement observable et des résultats de simulation en soulignent l'intérêt. De plus, une version multi-cartes est proposée garantissant l'observabilité du système sur l'ensemble de la carte de l'environnement.

A la fin de ce chapitre, l'analyse des résultats expérimentaux permet de conclure sur l'apport de cette nouvelle approche en termes de précision et de consistance des estimations en comparaison avec une approche classique.

4.2 Observabilité d'un système

Ce paragraphe décrit les outils mathématiques permettant de statuer sur l'observabilité d'un système. Les théorèmes employés dans ce chapitre sont démontrés dans les articles [12] et [81], respectivement pour les systèmes linéaires et non-linéaires.

4.2.1 Définition de l'observabilité

" Un système est dit observable si l'observation de ses entrées et de ses sorties pendant un intervalle de temps fini $[k; k + 1]$ permet de retrouver l'état initial $x(k)$. "

L'analyse de l'observabilité permet de conclure sur la convergence, ou non, des estimations d'un système. Une limite inférieure sur l'erreur ne sera donc garantie que si le vecteur d'état d'un système est complètement observable. Ce critère est un aspect important dans n'importe quel problème d'estimation.

4.2.2 Observabilité d'un système linéaire

Dans le cas linéaire, pour prouver qu'un système est observable, il suffit que la matrice d'observabilité, \mathbf{O} , soit de rang plein (critère de Kalman [12]). Soit un système linéaire invariant tel que :

$$\begin{cases} \dot{\mathbf{X}} = \mathbf{A}.\mathbf{X} + \mathbf{B}.\mathbf{U} \\ \mathbf{Z} = \mathbf{C}.\mathbf{X} \end{cases} \quad (4.1)$$

Le vecteur d'état \mathbf{X} est de dimension n . Le système (4.1) est observable si et seulement si la matrice d'observabilité, \mathbf{O} , définie par :

$$\mathbf{O} = [\mathbf{C} \quad \mathbf{C}.\mathbf{A} \quad \dots \quad \mathbf{C}.\mathbf{A}^{n-1}]^T \quad (4.2)$$

est de rang n .

Si tel est le cas, la valeur du vecteur d'état \mathbf{X} pourra être déterminée à tout instant t . L'observabilité est donc une propriété qui permet d'affirmer que l'état $x(t)$ peut être connu à tout instant compris dans un intervalle de temps quelconque allant de k à $k+1$.

4.2.3 Observabilité d'un système non-linéaire

Soit le système non-linéaire suivant :

$$\begin{cases} \dot{\mathbf{X}} = \mathbf{f}(\mathbf{X}, \mathbf{U}) \\ \mathbf{Z} = \mathbf{h}(\mathbf{X}) \end{cases} \quad (4.3)$$

Les fonctions non-linéaires \mathbf{f} et \mathbf{h} représentent, respectivement, l'évolution et l'observation du vecteur d'état du système. Ces fonctions peuvent être décomposées telles que :

$$\left\{ \begin{array}{l} \dot{\mathbf{X}} = \begin{bmatrix} f_1(\mathbf{X}, \mathbf{U}) \\ \vdots \\ f_i(\mathbf{X}, \mathbf{U}) \\ \vdots \\ f_n(\mathbf{X}, \mathbf{U}) \end{bmatrix} \text{ avec : } \mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_i \\ \vdots \\ \mathbf{X}_n \end{bmatrix} \in \mathbb{R}^n \\ \mathbf{Z} = \begin{bmatrix} h_1(\mathbf{X}) \\ \vdots \\ h_j(\mathbf{X}) \\ \vdots \\ h_m(\mathbf{X}) \end{bmatrix} \text{ avec : } \mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 \\ \vdots \\ \mathbf{Z}_j \\ \vdots \\ \mathbf{Z}_m \end{bmatrix} \in \mathbb{R}^m \end{array} \right. \quad (4.4)$$

Dans le cas d'un système non-linéaire, le calcul de la matrice d'observabilité doit prendre en compte la dépendance entre l'observation \mathbf{Z} et les commandes \mathbf{U} [81]. En effet, l'écriture de la fonction non-linéaire \mathbf{f} démontre la création d'un « lien » entre les commandes \mathbf{U} et les états \mathbf{X} du système. Ce n'est pas le cas pour un système linéaire comme le prouve l'équation (4.1). Pour prendre en considération cette dépendance, l'algèbre de Lie est employée dans les calculs de la matrice d'observabilité tel que :

$$\mathcal{O} = \begin{bmatrix} L_{\mathbf{f}}^0 dh_j(\mathbf{X}) \\ \vdots \\ L_{\mathbf{f}}^i dh_j(\mathbf{X}) \\ \vdots \\ L_{\mathbf{f}}^{n-1} dh_j(\mathbf{X}) \end{bmatrix} \text{ avec : } j \in [1 ; m] \quad (4.5)$$

Les éléments de la matrice \mathcal{O} , $L_{\mathbf{f}}^i dh_j(\mathbf{X})$, correspondent aux $i^{\text{èmes}}$ dérivées de Lie de la $j^{\text{ième}}$ composante de la dérivée de la fonction d'observation, $dh(\mathbf{X})$, par rapport à $\mathbf{f}(\mathbf{X}, \mathbf{U})$. Plus précisément, la dérivée de Lie à l'ordre 1 du scalaire $h_j(\mathbf{X})$ par rapport au vecteur $\mathbf{f}(\mathbf{X}, \mathbf{U})$ est un vecteur défini par :

$$L_{\mathbf{f}}^1 h_j(\mathbf{X}) = \frac{\partial h_j(\mathbf{X})}{\partial \mathbf{X}} \cdot \mathbf{f}(\mathbf{X}, \mathbf{U}) \quad (4.6)$$

Avec des calculs similaires, la dérivée de Lie de $dh_j(\mathbf{X})$, par rapport à $\mathbf{f}(\mathbf{X}, \mathbf{U})$, se définit par :

$$L_{\mathbf{f}}^1 dh_j(\mathbf{X}) = d(L_{\mathbf{f}}^1 h_j(\mathbf{X})) = \frac{\partial h_j(\mathbf{X})}{\partial \mathbf{X}} \cdot \frac{\partial \mathbf{f}(\mathbf{X}, \mathbf{U})}{\partial \mathbf{X}} + \left[\frac{\partial}{\partial \mathbf{X}} \cdot \left(\frac{\partial h_j(\mathbf{X})}{\partial \mathbf{X}} \right)^T \cdot \mathbf{f}(\mathbf{X}, \mathbf{U}) \right]^T \quad (4.7)$$

Cette équation permet d'écrire le calcul de ces dérivées sous une forme itérative tel que :

$$\begin{aligned} L_{\mathbf{f}}^0 dh_j(\mathbf{X}) &= \frac{\partial h_j(\mathbf{X})}{\partial \mathbf{X}} \\ L_{\mathbf{f}}^i dh_j(\mathbf{X}) &= L_{\mathbf{f}}^{i-1} dh_j(\mathbf{X}) \cdot \frac{\partial \mathbf{f}(\mathbf{X}, \mathbf{U})}{\partial \mathbf{X}} + \left[\frac{\partial}{\partial \mathbf{X}} \cdot \left(L_{\mathbf{f}}^{i-1} dh_j(\mathbf{X}) \right)^T \cdot \mathbf{f}(\mathbf{X}, \mathbf{U}) \right]^T \end{aligned} \quad (4.8)$$

Une fois la matrice \mathcal{O} définie, l'observabilité du système non-linéaire (4.3) peut être déterminée d'une manière rigoureuse. En effet, l'analyse de l'observabilité repose sur la condition de

rang (théorème 3.11 dans l'article [81]) appliquée à la matrice \mathcal{O} . Cette condition, permettant de statuer sur l'observabilité du système, stipule :

"Un système est localement observable si la condition de rang de l'observabilité est satisfaite."

Donc, un système est dit observable si la condition de rang est remplie par la matrice \mathcal{O} . En d'autres termes, le rang de cette matrice doit être égal ou supérieur au nombre d'états n du système (4.3). Cette condition est semblable à celle utilisée dans le cas linéaire. L'unique différence repose sur le calcul de la matrice d'observabilité prenant en compte les non-linéarités du système.

Désormais, la méthode d'analyse de l'observabilité d'un système non-linéaire a été définie. L'étude de cette propriété est proposée pour un système SLAM *WorldCentric* dans la suite de ce chapitre.

4.3 Observabilité d'une approche SLAM

Dans cette section, la preuve de la non-observabilité de l'approche SLAM *WorldCentric* est démontrée. A partir des équations précédentes, la matrice d'observabilité est calculée pour les deux types d'amers employés dans nos travaux. L'étude de son rang prouve que le système, ainsi formulé, n'est pas observable. Certaines approches SLAM répondent à ce problème d'observabilité grâce à la connaissance, a priori, d'informations sur l'environnement ou sur la position du véhicule. Un état de l'art en est proposé à la fin de cette section.

4.3.1 Observabilité d'un amer point

4.3.1.1 Système d'équations

En règle générale, la formulation d'une approche SLAM est de dimensions importantes et fortement non-linéaire. Pour étudier son observabilité, prenons un exemple simple. Soit un environnement composé d'un unique amer de type point avec pour coordonnées x_p et y_p . Le vecteur d'état du système s'écrit :

$$\mathbf{X} = \begin{bmatrix} x_v \\ y_v \\ \theta_v \\ x_p \\ y_p \end{bmatrix} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_M \end{bmatrix} \quad (4.9)$$

Dans la formulation d'un système SLAM *WorldCentric*, l'environnement est considéré comme statique. L'évolution du vecteur \mathbf{X} repose donc sur le déplacement du véhicule (cf. l'annexe B) :

$$\mathbf{X}(k+1) = \mathbf{f}(\mathbf{X}(k), \mathbf{U}) = \begin{bmatrix} x_v(k) + \Delta t \cdot v \cdot \cos(\theta_v(k)) \\ y_v(k) + \Delta t \cdot v \cdot \sin(\theta_v(k)) \\ \theta_v(k) + \frac{\Delta t \cdot v \cdot \tan(\varphi)}{L} \\ x_p(k) \\ y_p(k) \end{bmatrix} \quad (4.10)$$

Pour analyser l'observabilité du système non-linéaire avec les dérivées de Lie, cette modélisation (4.10) doit être reformulée en une fonction affine non-linéaire telle que :

$$\dot{\mathbf{X}} = f_0(\mathbf{X}) + \sum_{e=1}^E f_e(\mathbf{X}) \cdot u_e \quad (4.11)$$

Cette formulation, décrite dans l'article [45], est essentielle pour la construction de la matrice d'observabilité (4.5). A l'instar des calculs réalisés dans l'article [99], le système (4.10) est multiplié par la variable v^{-1} pour obtenir un système cinématique de la forme (4.11). Au final, l'évolution des états d'un système SLAM s'écrit sous une forme affine telle que :

$$\dot{\mathbf{X}} = f_0 + f_1 \cdot u_e = \mathbf{f}(\mathbf{X}, u_e) \quad (4.12)$$

Les vecteurs définis sous cette forme affine (4.12) s'écrivent :

$$f_0 = \begin{bmatrix} \cos(\theta_v(k)) \\ \sin(\theta_v(k)) \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{et} \quad f_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (4.13)$$

Avec cette écriture, le système SLAM ne possède qu'une seule commande :

$$u_e = \delta = \frac{\tan(\varphi)}{L} \quad (4.14)$$

Dans un système SLAM, la fonction d'observation \mathbf{h} s'écrit :

$$\mathbf{h}(\mathbf{X}) = \begin{bmatrix} h_1(\mathbf{X}) \\ h_2(\mathbf{X}) \end{bmatrix} = \begin{bmatrix} x_p \cdot \cos(\theta_v) + y_p \cdot \sin(\theta_v) - x_v \cdot \cos(\theta_v) - y_v \cdot \sin(\theta_v) \\ -x_p \cdot \sin(\theta_v) + y_p \cdot \cos(\theta_v) + x_v \cdot \sin(\theta_v) - y_v \cdot \cos(\theta_v) \end{bmatrix} \quad (4.15)$$

Dans ce système d'équations, ainsi que dans la suite de ce chapitre, les indices d'itération sont omis pour simplifier le raisonnement.

Toutes les variables et équations nécessaires au calcul de la matrice d'observabilité (4.5) ont été introduites. Il est désormais possible de l'étudier pour une approche SLAM représentant l'environnement par des points.

4.3.1.2 Etude de la matrice d'observabilité

La matrice d'observabilité du système (4.10), $\mathcal{O}_{SLAM-point}$, est calculée à partir des dérivées de Lie (équations (4.5) et (4.8)). Avec la formulation proposée, elle s'écrit :

$$\mathcal{O}_{SLAM-point} = \begin{bmatrix} L_{\mathbf{f}}^0 dh_1(\mathbf{X}) \\ L_{\mathbf{f}}^1 dh_1(\mathbf{X}) \\ L_{\mathbf{f}}^2 dh_1(\mathbf{X}) \\ L_{\mathbf{f}}^3 dh_1(\mathbf{X}) \\ L_{\mathbf{f}}^4 dh_1(\mathbf{X}) \\ L_{\mathbf{f}}^0 dh_2(\mathbf{X}) \\ L_{\mathbf{f}}^1 dh_2(\mathbf{X}) \\ L_{\mathbf{f}}^2 dh_2(\mathbf{X}) \\ L_{\mathbf{f}}^3 dh_2(\mathbf{X}) \\ L_{\mathbf{f}}^4 dh_2(\mathbf{X}) \end{bmatrix} \quad (4.16)$$

Les dérivées de Lie à l'ordre zéro, $L_{\mathbf{f}}^0 dh_1(\mathbf{X})$ et $L_{\mathbf{f}}^0 dh_2(\mathbf{X})$, correspondent, par définition, aux dérivées des fonctions h_1 et h_2 par rapport à \mathbf{X} :

$$\begin{aligned} \mathbf{L}_{p01} &= L_{\mathbf{f}}^0 dh_1(\mathbf{X}) = \begin{bmatrix} -\cos(\theta_v) \\ -\sin(\theta_v) \\ -\sin(\theta_v).(x_p - x_v) + \cos(\theta_v).(y_p - y_v) \\ \cos(\theta_v) \\ \sin(\theta_v) \end{bmatrix}^T \\ \mathbf{L}_{p02} &= L_{\mathbf{f}}^0 dh_2(\mathbf{X}) = \begin{bmatrix} \sin(\theta_v) \\ -\cos(\theta_v) \\ -\cos(\theta_v).(x_p - x_v) - \sin(\theta_v).(y_p - y_v) \\ -\sin(\theta_v) \\ \cos(\theta_v) \end{bmatrix}^T \end{aligned} \quad (4.17)$$

En utilisant les termes de Lie à l'ordre zéro, \mathbf{L}_{p01} et \mathbf{L}_{p02} , et la formule (4.6), les dérivées de Lie au premier ordre de h_1 et h_2 s'écrivent :

$$\begin{aligned} L_{\mathbf{f}}^1 h_1(\mathbf{X}) &= -1 - \sin(\theta_v).\delta.(x_p - x_v) + \cos(\theta_v).\delta.(y_p - y_v) \\ L_{\mathbf{f}}^1 h_2(\mathbf{X}) &= -\cos(\theta_v).\delta.(x_p - x_v) - \sin(\theta_v).\delta.(y_p - y_v) \end{aligned} \quad (4.18)$$

A partir des équations (4.8) et (4.18), les dérivées de Lie au premier ordre de dh_1 et dh_2 s'écrivent :

$$\begin{aligned} \mathbf{L}_{p11} &= L_{\mathbf{f}}^1 dh_1(\mathbf{X}) = \begin{bmatrix} \sin(\theta_v).\delta \\ -\cos(\theta_v).\delta \\ -\cos(\theta_v).(x_p - x_v).\delta - \sin(\theta_v).(y_p - y_v).\delta \\ -\sin(\theta_v).\delta \\ \cos(\theta_v).\delta \end{bmatrix}^T \\ \mathbf{L}_{p12} &= L_{\mathbf{f}}^1 dh_2(\mathbf{X}) = \begin{bmatrix} \cos(\theta_v).\delta \\ \sin(\theta_v).\delta \\ \sin(\theta_v).(x_p - x_v).\delta - \cos(\theta_v).(y_p - y_v).\delta \\ -\cos(\theta_v).\delta \\ -\sin(\theta_v).\delta \end{bmatrix}^T \end{aligned} \quad (4.19)$$

En observant les dérivées de Lie à l'ordre zéro (4.17) et à l'ordre un (4.19), l'égalité suivante peut être écrite :

$$\begin{aligned} \mathbf{L}_{p11} &= \delta.\mathbf{L}_{p02} \\ \mathbf{L}_{p12} &= -\delta.\mathbf{L}_{p01} \end{aligned} \quad (4.20)$$

Donc, les dérivées, \mathbf{L}_{11} et \mathbf{L}_{12} sont des combinaisons linéaires des dérivées à l'ordre zéro, \mathbf{L}_{01} et \mathbf{L}_{02} . La même remarque peut être faite pour les dérivées d'ordre supérieur. Au final, la matrice

d'observabilité s'écrit :

$$\mathcal{O}_{SLAM-point} = \begin{bmatrix} \mathbf{L}_{p01} \\ \mathbf{L}_{p02} \cdot \delta \\ -\mathbf{L}_{p01} \cdot \delta^2 \\ -\mathbf{L}_{p02} \cdot \delta^3 \\ \mathbf{L}_{p01} \cdot \delta^4 \\ \mathbf{L}_{p02} \\ -\mathbf{L}_{p01} \cdot \delta \\ -\mathbf{L}_{p02} \cdot \delta^2 \\ \mathbf{L}_{p01} \cdot \delta^3 \\ \mathbf{L}_{p02} \cdot \delta^4 \end{bmatrix} \quad (4.21)$$

Comme les lignes de la matrice $\mathcal{O}_{SLAM-point}$ sont des combinaisons linéaires des lignes \mathbf{L}_{p01} et \mathbf{L}_{p02} , le rang de cette matrice est de 2. La dimension du vecteur \mathbf{X} étant de 5, le système SLAM *WorldCentric* n'est donc pas observable selon les critères énoncés dans l'article [81]. Ces conclusions sont semblables à celles de l'article [99] avec une formulation différente des fonctions non-linéaires du système, \mathbf{f} et \mathbf{h} . Au final, l'observation d'un amer de type point (ou cercle) augmente de 2 le rang de la matrice d'observation. Donc, quelque soit le nombre d'amers de ce type observés, le système ne sera jamais observable avec cette formulation.

Le système présenté dans ces travaux utilise deux types d'amer : les cercles et les droites. Dans la suite de ce document, l'observabilité de l'approche SLAM employant des amers de type droite est réalisée. Cette étude permet de conclure sur l'observabilité de notre approche SLAM réalisant des cartes hétérogènes de l'environnement.

4.3.2 Observabilité d'un amer droite

4.3.2.1 Système d'équations

Pour simplifier le raisonnement, considérons un environnement composé d'une unique droite ayant pour coordonnées x_l et y_l avec une orientation θ_l . Le vecteur d'état du système s'écrit :

$$\mathbf{X} = \begin{bmatrix} x_v \\ y_v \\ \theta_v \\ x_l \\ y_l \\ \theta_l \end{bmatrix} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_M \end{bmatrix} \quad (4.22)$$

Comme l'environnement est considéré comme statique, l'évolution de ce système est identique à celle présentée dans le paragraphe 4.3.1.1. En revanche, comme le type d'amer est différent, les équations d'observation changent. Elles s'écrivent :

$$\mathbf{h}(\mathbf{X}) = \begin{bmatrix} h_1(\mathbf{X}) \\ h_2(\mathbf{X}) \\ h_3(\mathbf{X}) \end{bmatrix} = \begin{bmatrix} (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \cos(\theta_l) + \dots \\ \dots (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \cdot \sin(\theta_l) \\ - (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \sin(\theta_l) + \dots \\ \dots (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \cdot \cos(\theta_l) \\ \theta_v - \theta_l + \theta_m \end{bmatrix} \quad (4.23)$$

Les états x_m, y_m et θ_m représentent la mesure de la droite considérée.

4.3.2.2 Étude de la matrice d'observabilité

En appliquant la démarche présentée dans le paragraphe 4.3.1.2, les trois dérivées de Lie à l'ordre zéro, pour un amer de type droite, se formulent :

$$\mathbf{L}_{l01} = L_{\mathbf{f}}^0 dh_1(\mathbf{X}) = \begin{bmatrix} \cos(\theta_l) \\ \sin(\theta_l) \\ (-x_m \cdot \sin(\theta_v) - y_m \cdot \cos(\theta_v)) \cdot \cos(\theta_l) + (x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \sin(\theta_l) \\ -\cos(\theta_l) \\ -\sin(\theta_l) \\ (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \cdot \cos(\theta_l) - (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \sin(\theta_l) \end{bmatrix}^T \quad (4.24)$$

$$\mathbf{L}_{l02} = L_{\mathbf{f}}^0 dh_2(\mathbf{X}) = \begin{bmatrix} -\sin(\theta_l) \\ \cos(\theta_l) \\ (x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \cdot \sin(\theta_l) + (x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \cos(\theta_l) \\ -\cos(\theta_l) \\ -\sin(\theta_l) \\ -(x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \cos(\theta_l) + (-y_v + y_l - x_m \cdot \sin(\theta_v) - y_m \cdot \cos(\theta_v)) \cdot \sin(\theta_l) \end{bmatrix}^T \quad (4.25)$$

$$\mathbf{L}_{l03} = L_{\mathbf{f}}^0 dh_3(\mathbf{X}) = [0 \quad 0 \quad 1 \quad 0 \quad 0 \quad -1] \quad (4.26)$$

En utilisant les fonctions, \mathbf{f} , définies par le système (4.13), les dérivées de Lie au premier ordre se formulent :

$$\mathbf{L}_{l11} = L_{\mathbf{f}}^1 dh_1(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ (\cos(\theta_v) + y_m \cdot \delta \cdot \cos(\theta_v) - x_m \cdot \delta \cdot \sin(\theta_v)) \cdot \sin(\theta_l) - (x_m \cdot \delta \cdot \cos(\theta_v) - y_m \cdot \delta \cdot \sin(\theta_v) + \sin(\theta_v)) \cdot \sin(\theta_l) \\ 0 \\ 0 \\ (x_m \cdot \delta \cdot \sin(\theta_v) + y_m \cdot \delta \cdot \cos(\theta_v) - \cos(\theta_v)) \cdot \sin(\theta_l) + (x_m \cdot \delta \cdot \cos(\theta_v) - y_m \cdot \delta \cdot \sin(\theta_v) + \sin(\theta_v)) \cdot \cos(\theta_l) \end{bmatrix}^T \quad (4.27)$$

$$\mathbf{L}_{l12} = L_{\mathbf{f}}^1 dh_2(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ (\cos(\theta_v) - y_m \cdot \delta \cdot \cos(\theta_v) - x_m \cdot \delta \cdot \sin(\theta_v)) \cdot \cos(\theta_l) + (x_m \cdot \delta \cdot \cos(\theta_v) - y_m \cdot \delta \cdot \sin(\theta_v) + \cos(\theta_v)) \cdot \sin(\theta_l) \\ 0 \\ 0 \\ (x_m \cdot \delta \cdot \sin(\theta_v) + y_m \cdot \delta \cdot \cos(\theta_v) - \cos(\theta_v)) \cdot \cos(\theta_l) - (x_m \cdot \delta \cdot \cos(\theta_v) + y_m \cdot \delta \cdot \sin(\theta_v) + \sin(\theta_v)) \cdot \sin(\theta_l) \end{bmatrix}^T \quad (4.28)$$

$$\mathbf{L}_{l13} = L_f^1 dh_3(\mathbf{X}) = [\mathbf{0}_{1 \times 6}] \quad (4.29)$$

Les deux dérivées de Lie au premier ordre, (4.27) et (4.28), sont linéairement dépendantes avec la dérivée à l'ordre zéro \mathbf{L}_{03} . La matrice d'observabilité d'une droite s'écrit donc :

$$\mathcal{O}_{SLAM-droite} = \begin{bmatrix} \mathbf{L}_{l01} \\ \mathbf{L}_{l03} \cdot \Delta_1 \\ -\mathbf{L}_{l03} \cdot \Delta_2 \cdot \delta \\ \mathbf{L}_{l02} \\ \mathbf{L}_{l03} \cdot \Delta_2 \\ \mathbf{L}_{l03} \cdot \Delta_1 \cdot \delta \\ \mathbf{L}_{l03} \\ \mathbf{0}_{1 \times 6} \end{bmatrix} \quad (4.30)$$

Les coefficients Δ_i de la matrice d'observabilité (4.30) sont égaux à :

$$\begin{aligned} \Delta_1 &= (-x_m \cdot \delta \cdot \cos(\theta_v) + y_m \cdot \delta \cdot \sin(\theta_v) - \sin(\theta_v)) \cdot \cos(\theta_l) + \dots \\ &\dots (-x_m \cdot \delta \cdot \sin(\theta_v) - y_m \cdot \delta \cdot \cos(\theta_v) + \cos(\theta_v)) \cdot \sin(\theta_l) \\ \Delta_2 &= (-x_m \cdot \delta \cdot \sin(\theta_v) - y_m \cdot \delta \cdot \cos(\theta_v) + \cos(\theta_v)) \cdot \cos(\theta_l) + \dots \\ &\dots (x_m \cdot \delta \cdot \cos(\theta_v) - y_m \cdot \delta \cdot \sin(\theta_v) + \sin(\theta_v)) \cdot \sin(\theta_l) \end{aligned} \quad (4.31)$$

Le rang de la matrice $\mathcal{O}_{SLAM-droite}$ est donc égal à 3. Pour que le système (4.22) soit observable, le rang de cette matrice aurait dû être supérieur ou égal à 6. En conclusion, quelque soit le nombre de droites cartographiées, le système SLAM ne sera pas observable. De surcroît, la représentation des droites proposées dans cette démonstration ne tient pas compte de la théorie des symétries proposées dans le chapitre 2. L'observation d'une droite apporte uniquement deux informations au système (et non trois) [28]. Seules les informations sur la position latérale et sur l'orientation sont utilisées dans la mise à jour du système. Le rang de la matrice $\mathcal{O}_{SLAM-droite}$ est donc égal à 2. De plus, avec la théorie des symétries [28], un des états d'un amer droite est considéré comme non observable (état y_l). Donc, la condition de rang de la matrice $\mathcal{O}_{SLAM-droite}$ est égale à 5 (et non 6). Avec l'application de la théorie des symétries, le rang de la matrice $\mathcal{O}_{SLAM-droite}$ est toujours au seuil définissant l'observabilité du système. La conclusion sur l'observabilité du système est donc identique, que la théorie de symétrie soit prise en compte ou non dans le système.

En conclusion, un système SLAM, représentant l'environnement par des droites, n'est donc pas observable. Chaque droite observée augmente le rang de la matrice $\mathcal{O}_{SLAM-droite}$ de 2 mais également la condition d'observabilité de la même valeur. Donc quelque soit le nombre de droites cartographiées, ce système ne sera jamais observable. En prenant en compte la théorie des symétries des droites, comme dans nos travaux, la conclusion présentée est identique à celle formulée pour des amers de type point (ou cercle).

4.3.3 Conclusion et travaux existants

4.3.3.1 Conclusion sur l'observabilité du SLAM

Les démonstrations précédentes prouvent que l'approche SLAM *WorldCentric* n'est que partiellement observable quelque soit le type d'amer employé. Pour qu'un tel système devienne

localement observable, l'emploi d'informations absolues est obligatoire. Comme une approche SLAM correspond à la cartographie d'un environnement et à la localisation du véhicule dans celui-ci, deux types d'informations peuvent être employés :

- des informations sur la position du véhicule dans l'environnement comme des données GPS ;
- des informations sur les positions des amers constituant l'environnement comme des cartes SIG¹ (figure 4.1), des plans de bâtiments,

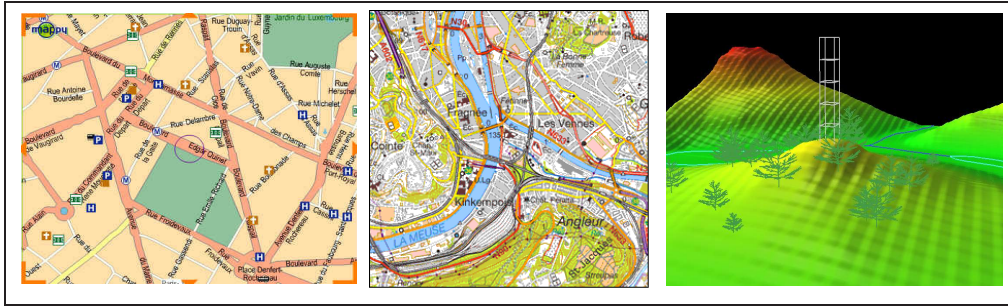


Figure 4.1: Exemples de cartes numériques.

Dans la suite, des approches SLAM employant l'un de ces deux types d'information sont introduites. Leurs présentations sont couplées à une analyse de leurs matrices d'observabilité. Ces études prouvent qu'un système SLAM devient observable lorsque de telles informations lui sont fournies.

4.3.3.2 Méthodes contraignant la position du véhicule

Un des capteurs les plus usités pour la localisation d'un véhicule est le GPS. Ses données peuvent facilement être employées dans une approche SLAM [76]. Prenons un exemple simple pour souligner l'apport de ce type de donnée sur l'observabilité du système. Soit le système (4.9) avec la fonction d'observation d'amer de type point (4.15). La position cartésienne du véhicule, x_v et y_v , est observée via des données GPS telle que :

$$\mathbf{h}_{GPS}(\mathbf{X}) = \begin{bmatrix} h_3(\mathbf{X}) \\ h_4(\mathbf{X}) \end{bmatrix} = \begin{bmatrix} x_v \\ y_v \end{bmatrix} \quad (4.32)$$

Ce nouveau système, nommé SLAM-GPS, réalise donc deux observations successives : la position de l'amer point et la position du véhicule. Pour calculer la matrice d'observabilité de ce système, les dérivées de Lie des fonctions (4.32) sont calculées jusqu'à l'ordre 1 avec les formules (4.8). Les dérivées de Lie à l'ordre zéro correspondent à :

$$\begin{aligned} \mathbf{L}_{g03} &= L_{\mathbf{f}}^0 dh_3(\mathbf{X}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} \\ \mathbf{L}_{g04} &= L_{\mathbf{f}}^0 dh_4(\mathbf{X}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \quad (4.33)$$

1. Système d'Information Géographique

En employant les fonctions d'évolution du système (4.13), les dérivées à l'ordre 1 des fonctions (4.32) s'écrivent :

$$\begin{aligned} \mathbf{L}_{g13} &= L_{\mathbf{f}}^1 dh_3(\mathbf{X}) = \begin{bmatrix} 0 & 0 & -\sin(\theta_v) & 0 & 0 \end{bmatrix} \\ \mathbf{L}_{g14} &= L_{\mathbf{f}}^1 dh_4(\mathbf{X}) = \begin{bmatrix} 0 & 0 & \cos(\theta_v) & 0 & 0 \end{bmatrix} \end{aligned} \quad (4.34)$$

Avec les équations (4.33) et (4.34) ainsi que la matrice $\mathcal{O}_{SLAM-point}$ (4.21), la matrice d'observation de ce système SLAM-GPS, $\mathcal{O}_{SLAM-GPS}$, s'écrit en utilisant les notations précédemment introduites :

$$\mathcal{O}_{SLAM-GPS} = \begin{bmatrix} \mathbf{L}_{p01} \\ \mathbf{L}_{p02} \cdot \delta \\ \mathbf{L}_{p02} \\ -\mathbf{L}_{p01} \cdot \delta \\ \mathbf{L}_{g03} \\ \mathbf{L}_{g13} \\ -\mathbf{L}_{g04} \cdot \delta \\ \mathbf{L}_{g04} \\ \mathbf{L}_{g14} \\ -\mathbf{L}_{g03} \cdot \delta \end{bmatrix} \quad (4.35)$$

La matrice d'observabilité du système SLAM-GPS est d'ordre 6. Ce système est donc localement observable d'après les critères énoncés dans l'article [81]. L'observation de la position du véhicule avec des données GPS répond au problème d'observabilité. En conclusion, si deux états du véhicule, \mathbf{X}_V , sont connus, un système SLAM *WorldCentric* est localement observable.

L'emploi de données GPS ne correspond pas à l'unique solution pour contraindre la position d'un véhicule dans une approche SLAM. Dans l'article [98], Lee et *al.* proposent une solution SLAM utilisant une carte digitale de la voirie. En supposant que le véhicule se déplace sur la route, il est possible de déterminer sa position précisément sur la carte avec une méthode de localisation [64]. Cette méthode estime la position latérale et l'orientation du véhicule. Au final, le couplage de la localisation du véhicule dans la carte digitale et de l'approche SLAM permet au système d'être observable [98].

4.3.3.3 Méthodes contraignant la position des amers

Pour répondre au problème d'observabilité, il est également possible de contraindre la carte estimée de l'environnement. A notre connaissance, il existe peu d'applications SLAM utilisant une carte existante de l'environnement pour contraindre la position de certains amers. L'article [99] propose une solution uniquement théorique. Les auteurs prouvent qu'une connaissance parfaite des positions de deux amers répond au problème décrit dans ce chapitre.

Cependant, dans une application réelle, il semble impossible de connaître exactement la position d'éléments de l'environnement. Les positions des amers sont toujours mesurées avec des instruments plus ou moins précis. Dans la continuité des travaux [99], l'approche **SLAM-contraint**, employant une carte existante de l'environnement, est proposée. La suite de ce document prouve, par l'étude théorique de la matrice d'observabilité et de résultats expérimentaux, que notre méthode répond au problème abordé et s'applique dans un contexte réel. Il est également prouvé, que lors de l'utilisation d'une carte « inexacte » de l'environnement, les conditions d'observabilité du système sont différentes de celles énoncées dans l'article [99].

4.4 Approche SLAM contrainte par une carte connue

4.4.1 Présentation de l'approche

Pour répondre au problème d'observabilité du SLAM, le système **SLAM-contraint** emploie une carte de l'environnement dans lequel évolue le véhicule (cf. la figure 4.2). La carte est connue par le système dès le début de l'application (représentée en vert sur la figure 4.2). Elle décrit certaines positions d'amers de l'environnement. Ces positions sont toutes référencées par rapport à un repère W_c . Ce repère peut être différent de celui de la carte estimée W_g . Pour mettre à jour la position des amers de la carte estimée avec ceux de la carte connue, un vecteur, $\mathbf{X}_{W_c}^{W_g}$, décrivant la transformation rigide entre les deux repères, est créé. Il correspond à la position du repère W_c dans la carte estimée référencée par rapport à W_g .

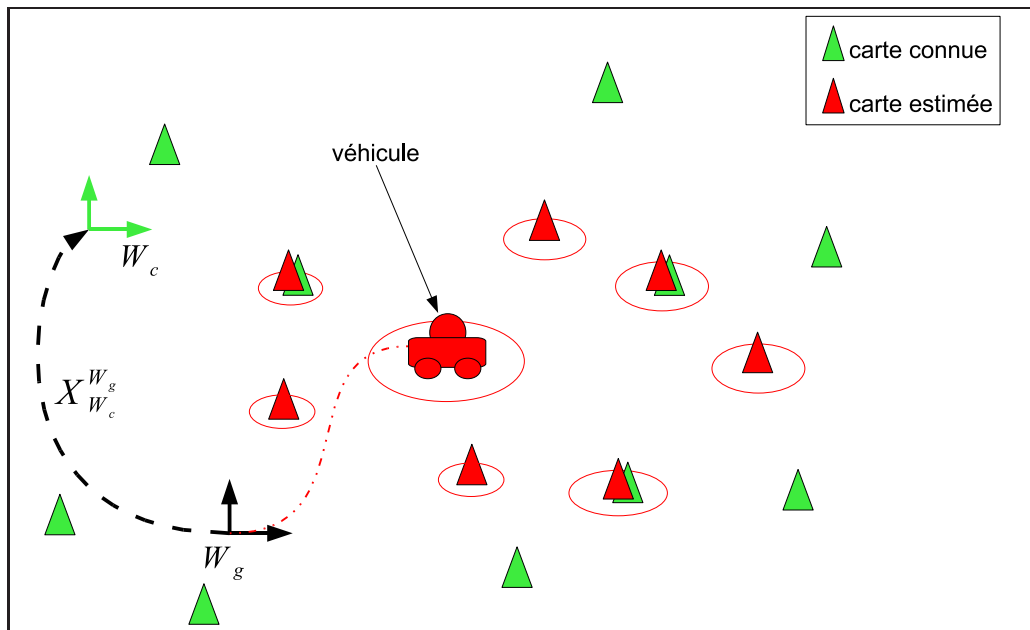


Figure 4.2: Schéma présentant l'approche SLAM-contrainte par une carte connue : les éléments rouges représentent les positions estimées et les éléments verts la carte connue de l'environnement.

Pour inclure l'information apportée par la carte connue, l'algorithme présenté dans le chapitre 2 est modifié (cf. la figure 4.3). Deux mises à jour consécutives sont réalisées à chaque itération du filtre **SLAM-contraint**. La première mise à jour correspond à une observation classique des amers. Les équations de mise à jour des amers sont présentées dans l'annexe F. La seconde mise à jour du système utilise la carte connue de l'environnement, \mathbf{X}_C . Cette étape contraint les positions estimées des amers de la carte \mathbf{X}_M . Avec cette seconde correction, l'approche **SLAM-contraint** répond au problème d'observabilité introduit précédemment.

Dans la suite de ce paragraphe, sont décrits les différents éléments et équations employés par notre méthode de **SLAM-contraint**.

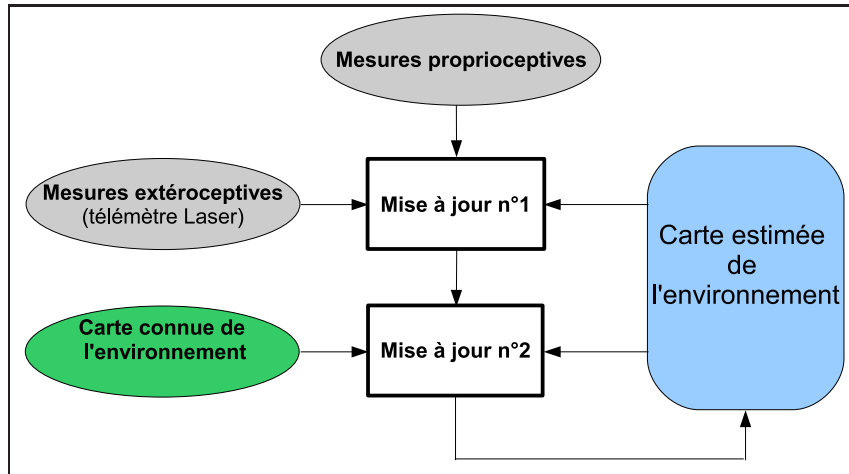


Figure 4.3: Schéma présentant l'algorithme SLAM-contraint.

4.4.2 Carte connue *a priori* de l'environnement

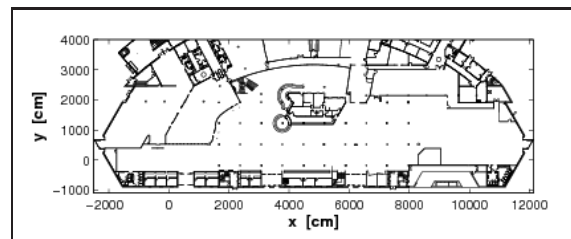
4.4.2.1 Description de la carte connue

Dans nos applications, la carte connue correspond soit à un plan de l'environnement (figure 4.4 (a)), soit à une cartographie réalisée avec un GPS (figure 4.4 (b)). Elle est composée d'amers de type cercle ou/et droite. Pour intégrer les incertitudes de mesures, cette carte est représentée par une variable aléatoire \mathbf{X}_C . Cette variable est supposée gaussienne telle que :

$$\mathbf{X}_C \sim \mathcal{N}(\hat{\mathbf{X}}_C, \mathbf{P}_C) \quad (4.36)$$



(a)



(b)

Figure 4.4: Exemples de cartes connues utilisées dans les applications : (a) carte GPS d'un parc (éléments représentés en rouge) et (b) plan de l'aéroport de Clermont Ferrand.

Cette carte connue est composée de l amers. Les deux moments définissant la variable \mathbf{X}_C (équation (4.36)) s'écrivent :

$$\hat{\mathbf{X}}_C = \begin{bmatrix} \hat{\mathbf{X}}_{E_1}^{W_c} \\ \vdots \\ \hat{\mathbf{X}}_{E_l}^{W_c} \end{bmatrix} \quad (4.37)$$

$$\mathbf{P}_C = \begin{bmatrix} \sigma^2 & & 0 \\ & \ddots & \\ 0 & & \sigma^2 \end{bmatrix} \quad (4.38)$$

Les états des amers constituant la carte \mathbf{X}_C sont supposés indépendants. Leurs variances, σ^2 , représentant les incertitudes de leurs mesures, sont définies telles que :

- σ_{x_l} , σ_{y_l} et σ_{θ_l} pour les amers de type droite,
- $\sigma_{x_{ce}}$ et $\sigma_{y_{ce}}$ pour les amers de type cercle.

Les valeurs affectées à ces variances dépendent du capteur employé pour la cartographie de l'environnement (figure 4.4 (a)) ou de la précision du plan du bâtiment (figure 4.4 (b)).

4.4.2.2 Transformation rigide entre la carte connue et la carte estimée

Le repère W_c de la carte connue est différent de celui de la carte estimée W_g (cf. sur la figure 4.2). Pour effectuer la seconde mise à jour reposant sur la carte connue (cf. la figure 4.3), le vecteur $\mathbf{X}_{W_c}^{W_g}$ est introduit. Ce vecteur représente la transformation rigide entre les deux cartes. Comme cette information n'est pas connue parfaitement, ce vecteur est défini comme une variable aléatoire suivant une loi gaussienne :

$$\mathbf{X}_{W_c}^{W_g} \sim \mathcal{N} \left(\hat{\mathbf{X}}_{W_c}^{W_g}, \mathbf{C}_{W_c}^{W_g} \right) \quad (4.39)$$

Ce vecteur est composé de trois états définissant la position de W_c par rapport au repère W_g :

$$\mathbf{X}_{W_c}^{W_g} = [x_c \quad y_c \quad \theta_c]^T \quad (4.40)$$

Les états x_c et y_c représentent les coordonnées cartésiennes du repère et θ_c , l'orientation du repère W_c .

Remarque sur la connaissance $\mathbf{X}_{W_c}^{W_g}$:

Dans l'application proposée dans ce document, la valeur du vecteur $\mathbf{X}_{W_c}^{W_g}$ est mesurée. Cependant, ces paramètres peuvent également être estimés en employant des méthodes d'association de données comme, par exemple, la méthode *Combined Constraint Data Association* (CCDA) [10]. Les résultats, présentés dans notre article [35], prouvent qu'une telle approche est réalisable.

Cependant, cette méthode nécessite l'estimation d'une partie de l'environnement avec une approche EKF-SLAM. Le système SLAM ne sera donc pas observable pendant le temps de la construction de cette partie de la carte. C'est seulement lorsque le vecteur $\mathbf{X}_{W_c}^{W_g}$ sera déterminé que le système deviendra localement observable.

4.4.3 Variables et équations du système SLAM-contraint

Dans cette section, l'initialisation des variables ainsi que la fonction d'observation de la carte connue sont introduites. Ensuite, grâce à ces équations, une étude de la matrice d'observabilité de l'approche SLAM-contraint est présentée.

4.4.3.1 Initialisation des variables

Pour permettre l'étude de l'observabilité de notre approche SLAM-contraint, le vecteur $\mathbf{X}_{W_c}^{W_g}$ est introduit dans le vecteur d'état, \mathbf{X} , lors de la phase d'initialisation. Bien qu'il ne soit pas observable directement dans l'approche proposée (voir les équations d'observation introduites dans le paragraphe 4.4.3.2), il représente une estimation de la projection du repère de la carte estimée vers la carte connue. L'initialisation du vecteur d'état s'écrit donc :

$$\mathbf{X}_0 = \begin{bmatrix} \mathbf{X}_{W_c}^{W_g} & \mathbf{X}_{V_0} \end{bmatrix}^T \quad (4.41)$$

Le vecteur \mathbf{X}_{V_0} représente la position initiale du véhicule. La matrice de covariance associée au vecteur d'état (4.41) se formule :

$$\mathbf{P}_0 = \begin{bmatrix} \mathbf{C}_{W_c}^{W_g} & 0 \\ 0 & \mathbf{P}_{V_0} \end{bmatrix} \quad (4.42)$$

La matrice \mathbf{P}_{V_0} correspond à la valeur de l'incertitude sur la position initiale du robot. La matrice $\mathbf{C}_{W_c}^{W_g}$ représente l'incertitude sur la mesure du vecteur $\mathbf{X}_{W_c}^{W_g}$.

4.4.3.2 Étape de mise à jour avec la carte connue

Pour contraindre les estimations du système SLAM, une seconde mise à jour du système est réalisée avec les informations de la carte connue (cf. la figure 4.3). Comme cette mise à jour est basée sur un filtrage de Kalman, seules les équations de mesure et d'observation sont décrites.

Pour simplifier l'écriture des différentes équations et matrices qui vont suivre, les vecteurs de la carte estimée et de la carte connue sont concaténés dans un seul vecteur s'écrivant :

$$\mathbf{X}_{syst} = \begin{bmatrix} \mathbf{X} & \mathbf{X}_C \end{bmatrix}^T \quad (4.43)$$

Comme la carte connue n'est pas corrélée avec la carte estimée, la matrice de covariance associée à ce nouveau vecteur s'écrit :

$$\mathbf{P}_{syst} = \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{P}_C \end{bmatrix} \quad (4.44)$$

Amer de type cercle (ou point) :

Soit un amer cercle (ou point) E_k de la carte connue associé avec un amer G_k de la carte estimée. Les vecteurs d'états associées à ces deux entités s'écrivent respectivement \mathbf{X}_{E_k} et \mathbf{X}_{G_k} . La fonction d'observation associée à la mise à jour de l'amer k s'écrit :

$$\mathbf{h}_{connu-point}(\mathbf{X}_{syst}) = \mathbf{X}_{G_k}^{W_g} \quad (4.45)$$

Le calcul de la matrice de covariance associée à cette observation s'écrit :

$$\mathbf{C}_{\mathbf{X}_{G_k}^{W_g}}(k+1|k) = \mathbf{H} \cdot \mathbf{P}_{\text{sys}t} \cdot \mathbf{H}^T \quad (4.46)$$

La matrice \mathbf{H} est la matrice jacobienne de la fonction $\mathbf{h}_{\text{connu-point}}$ calculée par rapport au vecteur $\mathbf{X}_{\text{sys}t}$.

L'équation de mesure associée à l'amer E_k correspond à un changement de repère. Elle translate les coordonnées de l'amer du repère de la carte connue, W_c , dans le repère de la carte estimée, W_g . Elle s'écrit :

$$\mathbf{X}_{E_k}^{W_g} = \mathbf{h}_{\text{mesure}}(\mathbf{X}_{\text{sys}t}) = \mathbf{X}_{W_c}^{W_g} \oplus \mathbf{X}_{E_k} \quad (4.47)$$

La matrice de covariance associée à la mesure $\mathbf{X}_{E_k}^{W_g}$ est obtenue à partir du calcul suivant :

$$\mathbf{C}_{\mathbf{X}_{E_k}^{W_g}}(k+1) = \mathbf{H}_{\text{mesure}} \cdot \mathbf{P}_{\text{sys}t} \cdot \mathbf{H}_{\text{mesure}}^T \quad (4.48)$$

La matrice $\mathbf{H}_{\text{mesure}}$ est la matrice jacobienne de la fonction $\mathbf{h}_{\text{mesure}}$ calculée par rapport aux vecteurs $\mathbf{X}_{\text{sys}t}$. Elle est constituée de deux matrices jacobienues, \mathbf{H}_{W_c} et \mathbf{H}_{E_k} , respectivement les dérivées de cette fonction par rapport au vecteur $\mathbf{X}_{W_c}^{W_g}$ et au vecteur \mathbf{X}_c . Cette matrice $\mathbf{H}_{\text{mesure}}$ s'écrit :

$$\mathbf{H}_{\text{mesure}} = \begin{bmatrix} \mathbf{H}_{W_c} & 0 & \cdots & 0 & \mathbf{H}_{E_k} & 0 & \cdots & 0 \end{bmatrix} \quad (4.49)$$

L'écriture de ces deux matrices jacobienues s'appuient sur les fonctions de composition décrites dans l'annexe B.

Le vecteur de transformation rigide, $\mathbf{X}_{W_c}^{W_g}$, est employé pour établir les mesures apportées par la carte connue. Les fonctions d'observation des amers de la carte estimée sont unitaires. Les observations correspondent donc aux positions des amers dans le repère de la carte estimée. Ces fonctions ont été écrites dans le but de respecter les théorèmes de convergence décrit dans le chapitre précédent. La dernière partie de ce paragraphe explique les raisons du choix de ces équations.

Amer de type droite :

La représentation des droites repose sur la méthode *SPmap* proposée dans l'article [27]. Pour considérer l'information angulaire apporté par ce type d'amer, les équations d'observation diffèrent de celles d'un amer cercle (cf. l'annexe F). Soit une droite E_k de la carte connue associé avec une droite G_k de la carte estimée. Les vecteurs d'états associés à ces deux entités s'écrivent respectivement \mathbf{X}_{E_k} et \mathbf{X}_{G_k} . La fonction d'observation de l'amer droite k s'écrit :

$$\mathbf{h}_{\text{connu-droite}}(\mathbf{X}_{\text{sys}t}) = \left(\mathbf{X}_{W_c}^{W_g} \oplus \mathbf{X}_{E_k} \right) \ominus \mathbf{X}_{G_k} = \mathbf{X}_{E_k}^{W_g} \ominus \mathbf{X}_{G_k} = 0 \quad (4.50)$$

Après la linéarisation de l'équation (4.50), le calcul de la matrice de covariance associée à cette observation s'écrit :

$$\mathbf{C}_{\text{connu-droite}} = \mathbf{H}_{\ominus} \cdot (\mathbf{H} \cdot \mathbf{P}_{\text{sys}t} \cdot \mathbf{H}^T) \cdot \mathbf{H}_{\ominus}^T \quad (4.51)$$

La matrice jacobienne \mathbf{H}_\ominus correspond à l'opération de composition effectuée entre le vecteur estimé \mathbf{X}_{G_k} et la position de l'amer E_k , $\mathbf{X}_{E_k}^{W_g}$, définie dans le repère de la carte estimée W_g . La matrice \mathbf{H} correspond à la matrice jacobienne calculée par rapport au vecteur \mathbf{X}_{syst} :

$$\mathbf{H} = [\mathbf{H}_c \ 0 \ \cdots \ 0 \ \mathbf{H}_{E_k} \ 0 \ \cdots \ 0] \quad (4.52)$$

Elle est constituée de deux matrices jacobienues, \mathbf{H}_{W_c} et \mathbf{H}_{E_k} , respectivement les dérivées de la fonction \mathbf{h}_{connu} par rapport au vecteur $\mathbf{X}_{W_c}^{W_g}$ et au vecteur \mathbf{X}_{E_k} .

Discussion sur le choix des fonctions d'observation :

Comme stipulé dans l'introduction, les équations du système ont été définies pour éviter toute divergence du système. Dans ce but, le vecteur de transformation rigide $\mathbf{X}_{W_c}^{W_g}$ n'est pas observé. En effet, ce vecteur n'apparaît que dans les équations permettant de définir les mesures et non les observations. En revanche, ce vecteur est ajouté dans le vecteur d'états du système pour permettre sa mise à jour via les corrélations existantes dans la matrice de covariance.

L'explication de cette écriture découle des conclusions obtenues dans le chapitre 3. Pour que les estimations du filtre EKF-SLAM soient consistantes, les incertitudes sur les estimations des orientations doivent rester négligeables. Dans les applications de l'approche SLAM-contraint, une valeur faible sur l'incertitude du vecteur $\mathbf{X}_{W_c}^{W_g}$ ne peut être garantie. La mesure de cette position est imprécise. D'après les observations faites dans le chapitre 3, une valeur non négligeable de cette incertitude entraîne une divergence des estimées du système. Pour s'assurer une convergence des estimations du système, le vecteur $\mathbf{X}_{W_c}^{W_g}$ n'est pas observable par le système. En revanche, si l'orientation du repère W_c peut être mesurée avec une très grande précision, le vecteur $\mathbf{X}_{W_c}^{W_g}$ pourra être observé. Une simple réécriture des équations d'observation du système devra être réalisée.

Dans cette section, les différentes équations d'observation de notre approche SLAM-contraint ont été présentées. Nous allons maintenant aborder l'étude de l'observabilité de l'approche et en prouver son intérêt.

4.5 Observabilité de l'approche SLAM-contraint

Pour étudier l'observabilité de notre système SLAM-contraint, deux cas ont été étudiés : la carte connue est composée d'un unique amer puis de trois amers. Les conclusions de ces deux études nous permettent de définir les conditions à satisfaire pour que notre système soit observable localement.

4.5.1 Première étude : 1 unique amer

Soit un environnement composé d'un unique amer de type point (ou cercle). Cet amer est cartographié sur la carte connue. Le vecteur d'état du système s'écrit :

$$\mathbf{X} = [x_c \ y_c \ \theta_c \ x_v \ y_v \ \theta_v \ x_p \ y_p]^T \quad (4.53)$$

Comme le vecteur de transformation rigide entre les deux cartes, $\mathbf{X}_{W_c}^{W_g}$, est inséré dans le vecteur d'état du système, les fonctions, modélisant l'évolution du système sous une forme affine (équation (4.12)), sont modifiées. Le vecteur $\mathbf{X}_{W_c}^{W_g}$ n'évoluant pas, ces fonctions se formulent :

$$f_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cos(\theta_v) \\ \sin(\theta_v) \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{et} \quad f_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (4.54)$$

Les équations d'observation de l'amer peuvent se détailler selon quatre composantes distinctes telles que :

$$\mathbf{h}(\mathbf{X}) = \begin{bmatrix} h_1(\mathbf{X}) \\ h_2(\mathbf{X}) \\ h_3(\mathbf{X}) \\ h_4(\mathbf{X}) \end{bmatrix} = \begin{bmatrix} x_p \cdot \cos(\theta_v) + y_p \cdot \sin(\theta_v) - x_v \cdot \cos(\theta_v) - y_v \cdot \sin(\theta_v) \\ -x_p \cdot \sin(\theta_v) + y_p \cdot \cos(\theta_v) + x_v \cdot \sin(\theta_v) - y_v \cdot \cos(\theta_v) \\ x_p \\ y_p \end{bmatrix} \quad (4.55)$$

Pour rappel, les fonctions h_1 et h_2 permettent la mise à jour des positions des amers avec les mesures extéroceptives. Les fonctions d'observation h_3 et h_4 permettent la mise à jour du système avec la position de l'amer de la carte connue.

Pour écrire la matrice d'observabilité du système (4.53), nommée $\mathcal{O}_{SLAM\text{-}contraint}$, les dérivées de Lie des quatre équations d'observation (4.55) sont formulées. Les dérivées de Lie à l'ordre zéro sont obtenues en dérivant les fonctions (4.55) par rapport au vecteur \mathbf{X} . Elles s'écrivent :

$$\mathbf{L}_{p01-c} = L_{\mathbf{f}}^0 dh_1(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\cos(\theta_v) \\ -\sin(\theta_v) \\ -\sin(\theta_v) \cdot (x_p - x_v) + \cos(\theta_v) \cdot (y_p - y_v) \\ \cos(\theta_v) \\ \sin(\theta_v) \end{bmatrix}^T \quad (4.56)$$

$$\mathbf{L}_{p02-c} = L_{\mathbf{f}}^0 dh_2(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\sin(\theta_v) \\ -\cos(\theta_v) \\ -\cos(\theta_v) \cdot (x_p - x_v) - \sin(\theta_v) \cdot (y_p - y_v) \\ \cos(\theta_v) \\ \sin(\theta_v) \end{bmatrix}^T \quad (4.57)$$

$$\mathbf{L}_{p03-c} = L_{\mathbf{f}}^0 dh_3(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}^T \quad (4.58)$$

$$\mathbf{L}_{p04-c} = L_{\mathbf{f}}^0 dh_4(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}^T \quad (4.59)$$

Pour compléter la matrice d'observabilité, les dérivées de Lie à l'ordre 1 s'écrivent :

$$\mathbf{L}_{p11-c} = L_{\mathbf{f}}^1 dh_1(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \sin(\theta_v) \\ -\cos(\theta_v) \\ -\cos(\theta_v).(x_p - x_v) - \sin(\theta_v).(y_p - y_v) \\ -\sin(\theta_v) \\ \cos(\theta_v) \end{bmatrix}^T = \mathbf{L}_{p02-c} \cdot \delta \quad (4.60)$$

$$\mathbf{L}_{p12-c} = L_{\mathbf{f}}^1 dh_2(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cos(\theta_v) \cdot \delta \\ \sin(\theta_v) \cdot \delta \\ \sin(\theta_v).(x_p - x_v) \cdot \delta - \cos(\theta_v).(y_p - y_v) \cdot \delta \\ -\cos(\theta_v) \cdot \delta \\ -\sin(\theta_v) \cdot \delta \end{bmatrix}^T = -\mathbf{L}_{p01-c} \cdot \delta \quad (4.61)$$

$$\mathbf{L}_{p13-c} = L_{\mathbf{f}}^1 dh_3(\mathbf{X}) = [\mathbf{0}_{1 \times 8}] \quad (4.62)$$

$$\mathbf{L}_{p14-c} = L_{\mathbf{f}}^1 dh_4(\mathbf{X}) = [\mathbf{0}_{1 \times 8}] \quad (4.63)$$

Les dérivées de Lie supérieures à l'ordre zéro des fonctions d'observation h_3 et h_4 sont nulles. De plus, les dérivées de Lie des fonctions h_1 et h_2 sont linéairement dépendantes aux dérivées de Lie à l'ordre zéro. Donc, toutes les dérivées de Lie d'ordres supérieures ne modifieront pas le rang de la matrice d'observabilité, $\mathcal{O}_{SLAM-contraint}$. Cette matrice d'observabilité

s'écrit au final :

$$\mathcal{O}_{SLAM-constraint} = \begin{bmatrix} \mathbf{L}_{p01-c} \\ \mathbf{L}_{p11-c} \\ \mathbf{L}_{p31-c} \\ \mathbf{L}_{p02-c} \\ \mathbf{L}_{p12-c} \\ \mathbf{L}_{p22-c} \\ \mathbf{L}_{p03-c} \\ \mathbf{L}_{p13-c} \\ \mathbf{L}_{p23-c} \\ \mathbf{L}_{p04-c} \\ \mathbf{L}_{p14-c} \\ \mathbf{L}_{p24-c} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{p01-c} \\ \mathbf{L}_{p02-c} \cdot \delta \\ -\mathbf{L}_{p01-c} \cdot \delta^2 \\ \mathbf{L}_{p02-c} \\ -\mathbf{L}_{p01-c} \cdot \delta \\ -\mathbf{L}_{p02-c} \cdot \delta^2 \\ \mathbf{L}_{p03-c} \\ \mathbf{0}_{1 \times 8} \\ \mathbf{0}_{1 \times 8} \\ \mathbf{L}_{p04-c} \\ \mathbf{0}_{1 \times 8} \\ \mathbf{0}_{1 \times 8} \end{bmatrix} \quad (4.64)$$

Les lignes de la matrice $\mathcal{O}_{SLAM-constraint}$ sont des combinaisons linéaires des lignes \mathbf{L}_{p01-c} , \mathbf{L}_{p02-c} , \mathbf{L}_{p03-c} et \mathbf{L}_{p04-c} . Le rang de $\mathcal{O}_{SLAM-constraint}$ est donc 4. Il est inférieur à 8, nombre d'états du vecteur \mathbf{X} (4.53). A l'instar des travaux réalisés dans la publication [99], le système n'est pas observable lorsqu'une seule position d'amer est connue, en se basant sur les critères énoncés dans l'article [81]. De surcroît, l'estimation du vecteur $\mathbf{X}_{W_c}^{W_g}$ agrandit le vecteur du système de trois états. La condition de rang de la matrice d'observabilité est donc supérieure à celle employée dans l'article [99]. Cependant, il faut souligner deux faits importants :

- l'observation d'un amer cercle (ou point) sur la carte connue (fonctions h_3 et h_4) augmente le rang de la matrice d'observabilité de 2 mais n'augmente pas la condition de rang pour déterminer l'observabilité du système ;
- l'observation d'un amer cercle (ou point) sur la carte estimée (fonctions h_1 et h_2) augmente de 2 le rang de la matrice d'observabilité mais augmente également la condition de rang de deux.

Pour que le système SLAM-contraint soit observable, les observations des amers de la carte connue doivent augmenter le rang de la matrice d'observabilité $\mathcal{O}_{SLAM-constraint}$ d'au moins 6, et ceci, quel que soit le nombre d'amers observés (fonctions h_1 et h_2) à l'instant t . En prenant en compte les deux constatations précédentes, notre système doit observer, à chaque instant t , au minimum trois amers présents sur la carte connue pour que le système soit observable. L'observation d'uniquement deux amers connus ne suffirait pas. Le rang de la matrice d'observabilité serait inférieur à la condition de rang d'Hermann. Le paragraphe suivant présente une démonstration prouvant la véracité de cette conclusion.

4.5.2 Deuxième étude : 3 amers présents sur la carte connue

Soit le cas où trois amers, $a1$, $a2$ et $a3$, sont cartographiés sur la carte estimée \mathbf{X}_M et présents sur la carte connue \mathbf{X}_c (cf. la figure 4.5). Les positions de ces amers sont mesurées à chaque instant de temps t . Dans cet exemple, le vecteur d'état s'écrit :

$$\mathbf{X} = \begin{bmatrix} x_c & y_c & \theta_c & x_v & y_v & \theta_v & x_{p1} & y_{p1} & x_{p2} & y_{p2} & x_{p3} & y_{p3} \end{bmatrix}^T \quad (4.65)$$

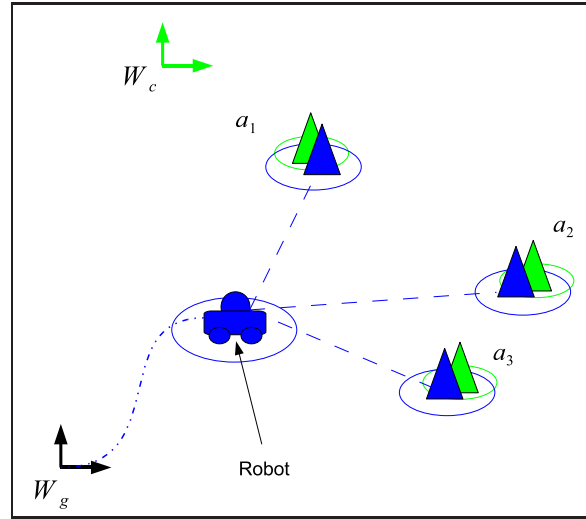


Figure 4.5: Exemple où le système SLAM-contraint est observable : les éléments verts représentent la carte connue et les éléments bleus la carte estimée par le système.

Ce vecteur contient 12 états distincts. Ceci signifie que la condition de rang d'observabilité est égale à 12. Pour être observable, la matrice d'observabilité du système doit avoir un rang égal ou supérieur à 12.

Les équations d'observation des amers ont été présentées dans le paragraphe 4.5.1. Les dérivées de Lie à l'ordre zéro du système (4.65) s'écrivent donc :

$$\mathbf{L}_{p01i-c} = L_{\mathbf{f}}^0 dh_{1i}(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\cos(\theta_v) \\ -\sin(\theta_v) \\ -\sin(\theta_v) \cdot (x_{pi} - x_v) + \cos(\theta_v) \cdot (y_{pi} - y_v) \\ \cos(\theta_v) \\ \sin(\theta_v) \end{bmatrix}^T \quad \text{avec : } i \in [1 ; 3] \quad (4.66)$$

$$\mathbf{L}_{p02i-c} = L_{\mathbf{f}}^0 dh_{2i}(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\sin(\theta_v) \\ -\cos(\theta_v) \\ -\cos(\theta_v) \cdot (x_{pi} - x_v) - \sin(\theta_v) \cdot (y_{pi} - y_v) \\ \cos(\theta_v) \\ \sin(\theta_v) \end{bmatrix}^T \quad \text{avec : } i \in [1 ; 3] \quad (4.67)$$

$$\mathbf{L}_{p03i-c} = L_{\mathbf{f}}^0 dh_{3i}(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}^T \quad \text{avec : } i \in [1 ; 3] \quad (4.68)$$

$$\mathbf{L}_{p04i-c} = L_{\mathbf{f}}^0 dh_{4i}(\mathbf{X}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}^T \quad \text{avec : } i \in [1 ; 3] \quad (4.69)$$

En se basant sur les calculs réalisés précédemment, les dérivées de Lie du premier ordre s'écrivent :

$$\mathbf{L}_{p11i-c} = L_{\mathbf{f}}^1 dh_{1i}(\mathbf{X}) = \mathbf{L}_{p02i-c} \cdot \delta \quad \text{avec : } i \in [1 ; 3] \quad (4.70)$$

$$\mathbf{L}_{p12i-c} = L_{\mathbf{f}}^1 dh_{2i}(\mathbf{X}) = -\mathbf{L}_{p01i-c} \cdot \delta \quad \text{avec : } i \in [1 ; 3] \quad (4.71)$$

$$\mathbf{L}_{p13i-c} = L_{\mathbf{f}}^1 dh_{3i}(\mathbf{X}) = [\mathbf{0}_{1 \times 8}] \quad \text{avec : } i \in [1 ; 3] \quad (4.72)$$

$$\mathbf{L}_{p14i-c} = L_{\mathbf{f}}^1 dh_{4i}(\mathbf{X}) = [\mathbf{0}_{1 \times 8}] \quad \text{avec : } i \in [1 ; 3] \quad (4.73)$$

Comme il a été démontré dans le paragraphe 4.3.1, le développement des dérivées de Lie à des ordres supérieurs est inutile. En effet, ces dérivées sont linéaires aux dérivées à l'ordre zéro. En n'écrivant que les dérivées de Lie non linéaires entre elles, la matrice d'observabilité $\mathcal{O}_{SLAM\text{-}constraint}$ s'écrit donc :

$$\mathcal{O}_{SLAM\text{-}constraint} = \begin{bmatrix} \mathbf{L}_{p01i-c} \\ \mathbf{L}_{p02i-c} \\ \mathbf{L}_{p03i-c} \\ \mathbf{L}_{p04i-c} \end{bmatrix} \quad \text{avec : } i \in [1 ; 3] \quad (4.74)$$

La matrice $\mathcal{O}_{SLAM\text{-}constraint}$ est de rang 12. Or la condition de rang d'observabilité de ce système (4.65) est égale à 12. D'après les conditions énoncées dans l'article [81], ce système est localement observable.

4.5.3 Conclusions

Comme démontré dans le paragraphe 4.3.2 de ce chapitre, les droites présentent les mêmes caractéristiques d'observabilité que des amers de type point (ou cercle). A l'instar des amers de type cercle, la carte connue doit également contenir au moins 3 droites mesurées (et associées)

à l'instant t . Les mêmes conclusions s'appliquent sur les amers de type point et sur les amers de type droite.

Le système SLAM-contraint est localement observable à la condition qu'au moins 3 amers de la carte connue puissent être observés à chaque itération du filtre. Cependant, il semble évident que le système ne puisse pas être observable localement à chaque itération du filtre. Pour être localement observable, il faudrait que la carte connue de l'environnement contienne de très nombreuses positions d'amers. Alors, se poserait la question suivante : est-ce que le système réalise une application de SLAM-contraint ou simplement une localisation d'un véhicule dans un environnement ?

Dans le but de rendre le système localement observable, cette méthode SLAM-contraint est appliquée dans une approche multi-cartes.

4.6 Méthode SLAM-contraint dans une approche multi-cartes

Dans cette partie, nous proposons de coupler la méthode SLAM-contraint à une approche multi-cartes. L'objectif est de garantir l'observabilité de notre système lors des mises à jour de la carte globale à défaut de pouvoir le garantir lors des mises à jour des cartes locales.

4.6.1 Présentation de l'approche

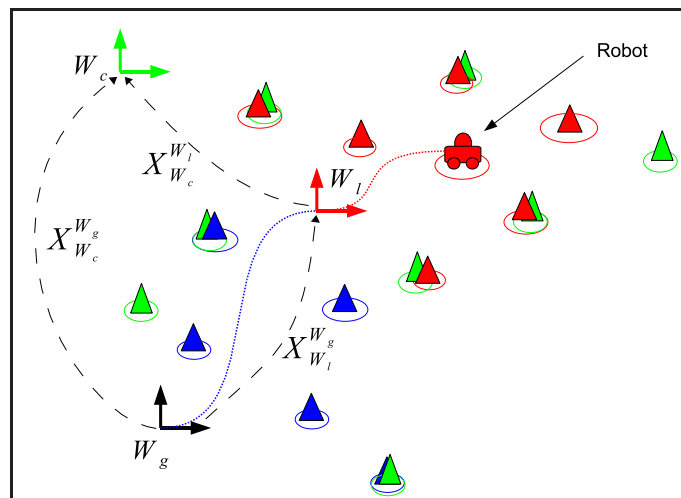


Figure 4.6: Méthode SLAM-contraint couplée à une approche multi-cartes : les éléments en bleu représentent la carte globale, les éléments en rouge représentent la carte locale et les éléments verts la carte connue de l'environnement.

Une approche multi-cartes représente l'environnement par plusieurs cartes locales toutes référencées dans un même repère global W_g (cf. la figure 4.6). Elle se décompose en deux étapes distinctes : construction d'une carte locale puis mise à jour de la carte globale avec cette

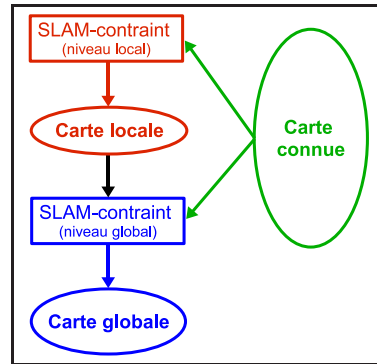


Figure 4.7: Schéma présentant les différentes étapes de l'approche SLAM-contraint multi-cartes.

carte locale. Pour chacune de ces deux étapes, la méthode SLAM-contraint est appliquée aux estimations des cartes de l'environnement (cf. la figure 4.7).

La construction de la carte locale est couplée à l'approche SLAM-contraint dans l'optique d'augmenter la précision des estimations. En effet, au niveau local, l'observabilité du système ne peut être garantie à chaque mise à jour du filtre. Cependant, le fait d'employer des données connues de l'environnement permet d'obtenir des estimations plus précises. Les équations des deux mises à jour successives sont présentées respectivement dans l'annexe F pour la mise à jour classique de la carte locale et dans le paragraphe 4.4 pour la mise à jour de l'approche contrainte. Le vecteur $\mathbf{X}_{W_c}^{W_l}$ est introduit pour permettre la mise à jour de chaque carte locale avec la carte connue. Il permet de traduire la transformation rigide entre le repère de la carte locale W_l , et celui de la carte globale, W_g (cf. la figure 4.6).

La mise à jour de la carte globale suit le même principe que celles des cartes locales. Deux mises à jour consécutives sont réalisées. La première mise à jour est effectuée grâce au vecteur $\mathbf{X}_{W_l}^{W_g}$ représentant les coordonnées du repère de la carte locale W_l dans la carte globale W_g (cf. la figure 4.6). La seconde mise à jour est identique à celle réalisée avec les cartes locales. Seule la mesure employée est différente. Elle correspond à une carte locale et non à une mesure télémétrique. Comme cette carte locale est assez vaste, elle contient au moins trois amers référencés sur la carte connue. Donc d'un point de vue global, le système SLAM-contraint multi-cartes est localement observable à chaque itération du filtre EKF si chaque carte locale recense au moins 3 amers de la carte connue. Ceci correspond à l'intérêt principal de cette approche.

4.6.2 Initialisation du vecteur des cartes locales

Dans l'approche SLAM-contraint, la position du repère de la carte connue doit être présente dans le vecteur de la carte à contraindre. Pour une carte locale, cette information correspond au vecteur d'état $\mathbf{X}_{W_c}^{W_l}$.

Soit le vecteur d'état de la carte globale \mathbf{X} et sa matrice de covariance \mathbf{P} . La position du robot, \mathbf{X}_V , dans le repère de la carte globale, correspond au vecteur $\mathbf{X}_{W_g}^{W_l}$ dans le vecteur associé

à une carte locale. Le calcul du vecteur $\mathbf{X}_{W_c}^{W_l}$ s'écrit :

$$\mathbf{X}_{W_c}^{W_l} = \mathbf{g}(\mathbf{X}) = \ominus \mathbf{X}_{W_l}^{W_g} \oplus \mathbf{X}_{W_c}^{W_g} = \mathbf{X}_{W_g}^{W_l} \oplus \mathbf{X}_{W_c}^{W_g} \quad (4.75)$$

Le calcul de la matrice de covariance, $\mathbf{C}_{W_c}^{W_l}$, s'écrit :

$$\mathbf{C}_{W_c}^{W_l} = \mathbf{G} \cdot \mathbf{P} \cdot (\mathbf{G})^T \quad (4.76)$$

La matrice \mathbf{G} correspond à la jacobienne de la fonction \mathbf{g} par rapport au vecteur \mathbf{X} . Cette matrice s'écrit donc :

$$\mathbf{G} = \begin{bmatrix} \mathbf{H}_{W_c}^{W_g} & \mathbf{H}_{X_V} & 0 & \cdots & 0 \end{bmatrix} \quad (4.77)$$

La matrice $\mathbf{H}_{W_c}^{W_g}$ est la jacobienne de la fonction \mathbf{g} par rapport au vecteur $\mathbf{X}_{W_c}^{W_g}$ et la matrice \mathbf{H}_{X_V} , la jacobienne de la fonction \mathbf{g} par rapport au vecteur \mathbf{X}_V . Comme dans l'approche globale, il est supposé que la position du repère de la carte connue est indépendante de celui de la carte estimée et contrainte. L'insertion du vecteur $\mathbf{X}_{W_c}^{W_l}$ dans le vecteur de la carte locale, \mathbf{X}^l , correspond à :

$$\mathbf{X}^l = \begin{bmatrix} \mathbf{X}_{W_c}^{W_l} \\ \mathbf{0} \end{bmatrix} \text{ avec } \mathbf{C}^l = \begin{bmatrix} \mathbf{C}_{W_c}^{W_l} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (4.78)$$

Il faut souligner un fait important découlant de l'initialisation du vecteur de la carte locale. En insérant le vecteur $\mathbf{X}_{W_c}^{W_l}$, une carte locale est désormais liée statistiquement à la carte globale. En effet, le vecteur du véhicule \mathbf{X}_V est utilisée pour calculer le vecteur $\mathbf{X}_{W_c}^{W_l}$. Cependant, dans les travaux de [128], il est prouvé que les cartes locales peuvent être liées par un élément commun, de manière statistique. De plus, deux précautions ont été prises dans les équations pour ne pas créer de corrélation. Premièrement, les vecteurs $\mathbf{X}_{W_c}^{W_g}$ et $\mathbf{X}_{W_c}^{W_l}$ ne sont pas observables (voir les équations de mise à jour de la carte connue). Deuxièmement, lors de l'initialisation d'une carte locale, les corrélations entre les vecteurs $\mathbf{X}_{W_c}^{W_l}$ et le vecteur du véhicule sont initialisées avec des valeurs nulles. L'addition de ces deux faits implique qu'aucune corrélation entre ces vecteurs et tout autre élément ne peut être créée. Ce vecteur ne sera donc pas mis à jour dans la carte locale mais il n'entraînera aucune divergence du système. L'application SLAM-contraint multi-cartes respectera donc les théorèmes non-linéaires de convergence proposée dans le chapitre 3.

4.7 Résultats de simulation

Deux séries de simulations sont présentées dans cette partie. L'objectif est de valider d'une part, les conclusions sur l'observabilité du système SLAM-contraint, d'autre part la version multi-cartes de cette approche. La première série est basée sur une application très simple. Les résultats vérifieront les conclusions obtenues à partir de l'étude de l'observabilité de l'approche. La seconde série repose sur une carte hétérogène composée de droites et cercles. Les résultats valident l'approche proposée dans un contexte proche de la réalité.

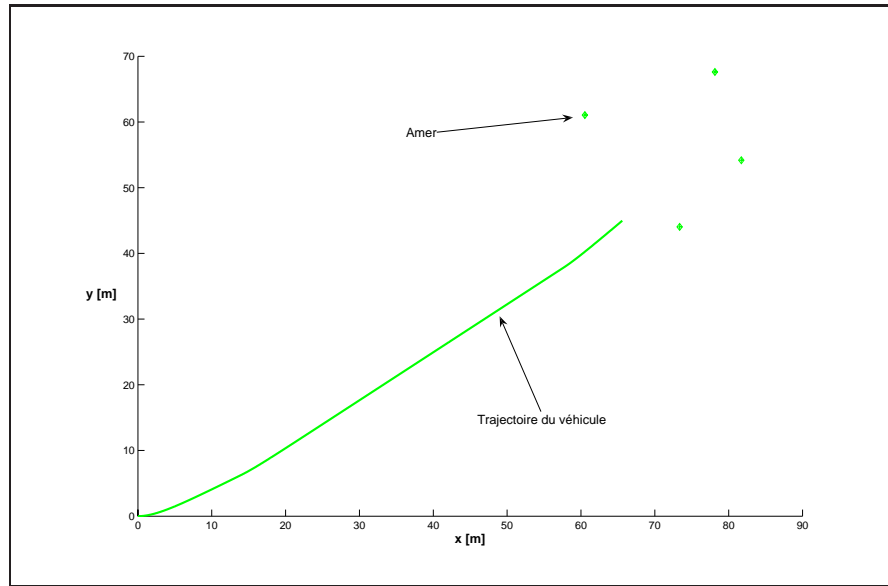


Figure 4.8: Scénario des deux séries de simulation : le véhicule suit une trajectoire rectiligne et observe, à chaque itération du filtre, 4 amers.

4.7.1 Séries de simulations simples

Pour valider les conclusions de l'étude de l'observabilité du système SLAM-contraint, deux séries de simulations sont présentées. Le scénario est simple. Le robot réalise une trajectoire rectiligne et observe continuellement 4 amers (cf. figure 4.8). Le nombre d'amers de la carte connue varie de trois à zéro. Pour vérifier l'observabilité de notre approche, un biais sur la position initiale du véhicule est ajouté. Il est égal à 1 m sur les coordonnées cartésiennes et de 3 degrés sur son orientation.

Deux contextes expérimentaux sont simulés, soit le vecteur $\mathbf{X}_{W_c}^{W_g}$ est connu parfaitement, soit il est imprécis avec une incertitude sur sa position.

4.7.1.1 Position exacte du repère de la carte connue

Dans cette série de simulation, les états du vecteur $\mathbf{X}_{W_c}^{W_g}$ ont des valeurs exactes. Les positions des amers décrites dans la carte connue et fournies au système, sont parfaites. Les résultats proposés sont obtenus à partir d'une séquence de 20 simulations de Monte Carlo (cf. la figure 4.9).

Lorsque le système ne connaît aucun amer de l'environnement, les estimations de la position du véhicule divergent. L'erreur sur cette position ne cesse de croître (courbes bleues sur la figure 4.9 (a)). Lorsque le système connaît la position exacte d'un unique amer, les estimations semblent convergées. En effet, l'erreur sur la position du véhicule tend vers zéro. Cependant,

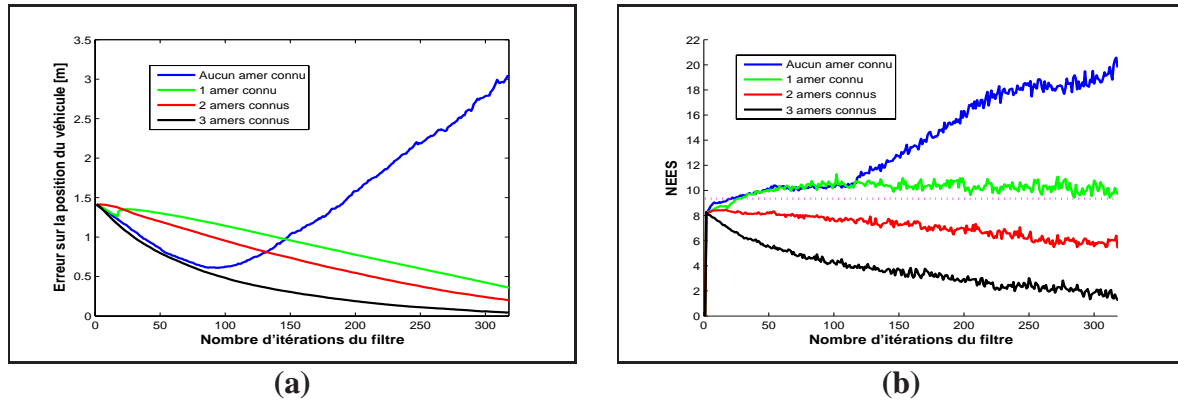


Figure 4.9: Résultats d'une séquence de 20 simulations de Monte Carlo lorsque la position du repère de la carte connue est exacte : (a) évolution de l'erreur sur la position du véhicule et (b) évolution de la distance $NEES^2$.

la valeur de la distance $NEES^2$ est supérieure au seuil définissant la consistance (courbe verte sur figure 4.9 (b)). Les estimations de ce système ne sont donc pas observables lorsqu'une seule position d'amer est connue parfaitement. Lorsque le système connaît la position d'au moins deux amers de la carte, les estimations du système convergent. De plus, elles sont également consistantes car les valeurs de la distance $NEES$ restent inférieures au seuil du χ^2 (courbes noires et rouges sur la figure 4.9). Dans ces deux cas d'études, le système est donc observable. Ces résultats sont en concordance avec les résultats présentés dans la publication [99]. Lorsque les positions décrites par le vecteur $\mathbf{X}_{W_c}^{W_g}$ sont exactes, l'approche SLAM-contraint doit observer au moins deux amers connus pour que le système soit observable.

4.7.1.2 Position imprécise du repère de la carte connue

Une erreur de mesure sur la connaissance du vecteur $\mathbf{X}_{W_c}^{W_g}$ est ajoutée. Elle correspond à des erreurs de 10 et 20 cm sur les coordonnées cartésiennes de ce vecteur. Les résultats sont présentés sur la figure 4.10.

Comme dans le cas d'étude précédent, lorsque le système ne possède aucune connaissance sur son environnement, il n'est pas observable. L'erreur sur la position du véhicule ne cesse de croître et la distance $NEES^2$ est largement supérieure au seuil (courbes bleues sur la figure 4.10). Lorsque le système connaît la position d'un ou deux amers de l'environnement, l'erreur sur la position du véhicule converge vers la valeur de l'erreur sur la position du vecteur $\mathbf{X}_{W_c}^{W_g}$. Cependant, les valeurs de la distance $NEES^2$ sont supérieures au seuil du χ^2 dans les deux cas. Donc, dans ces deux configurations, les estimations ne sont pas consistantes. Comme le prouvent les résultats (courbe noire sur la figure 4.10 (b)), le système est observable uniquement lorsqu'au moins 3 positions d'amers de l'environnement sont connues. L'erreur sur la position du véhicule tend vers zéro et les valeurs de la distance $NEES$ restent inférieures au seuil défini par la distribution du χ^2 .

2. Erreur normalisée d'estimation au carré (Normalised Estimation Error Squared) : cette distance correspond au carrée d'une distance de Mahalanobis entre la valeur réelle d'un paramètre et sa valeur estimée. Elle est décrite dans la section 3.2.

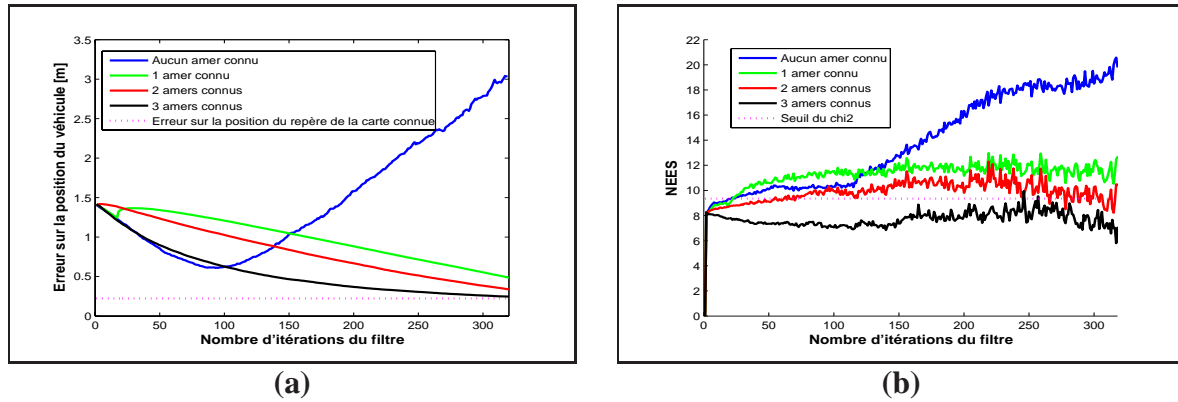


Figure 4.10: Résultats d'une séquence de 20 simulations de Monte Carlo lorsque la position du repère de la carte connue est imprécise : (a) évolution de l'erreur sur la position du véhicule et (b) évolution de la distance $NEES^2$.

Ces résultats confirment les conclusions obtenues avec l'étude de l'observabilité de l'approche SLAM-contraint : lorsque le vecteur $\mathbf{X}_{W_c}^{W_g}$ n'est pas connu parfaitement, le système doit observer au moins trois amers de la carte connue de l'environnement pour que le système soit localement observable.

4.7.2 Simulation sur une trajectoire bouclée

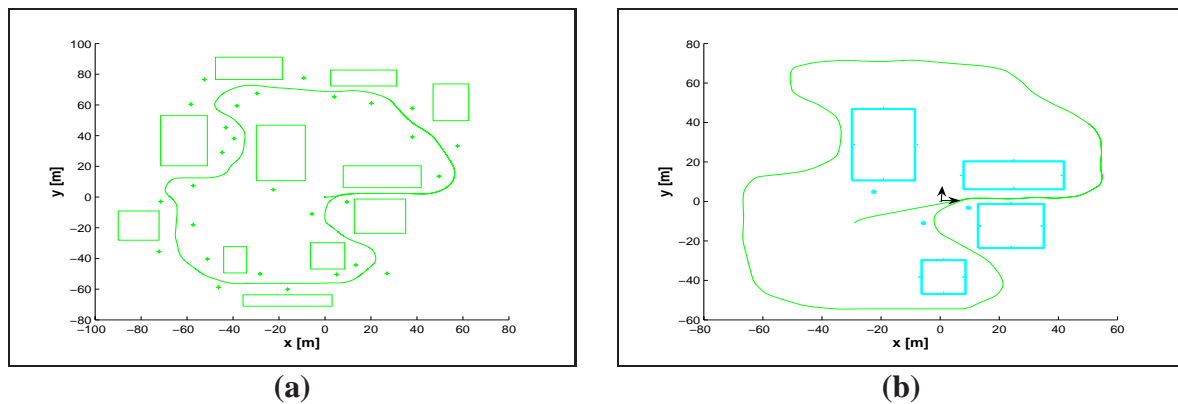


Figure 4.11: Simulation de l'approche SLAM-contraint : la carte simulée de l'environnement, composée de cercles et de rectangle, est présentée sur la figure (a) et la carte connue par le système en bleu sur la figure (b).

Les résultats précédents ont confirmé les conclusions de l'étude de l'observabilité de notre approche SLAM-contraint sur un scénario simple de simulation. Nous allons maintenant tester la fiabilité de notre approche sur un environnement hétérogène et une trajectoire fermée (cf. la figure 4.11 (a)). Le véhicule parcourt cette trajectoire dans son intégralité. La valeur des bruits sur les données extéroceptives et proprioceptives est donnée en annexe E. Pour l'approche SLAM-contraint multi-cartes reposant sur un filtrage de Kalman, la carte connue par le système

est présentée sur la figure 4.11 (b). Une erreur d'un mètre est ajoutée sur les coordonnées cartésiennes du repère de cette carte connue.

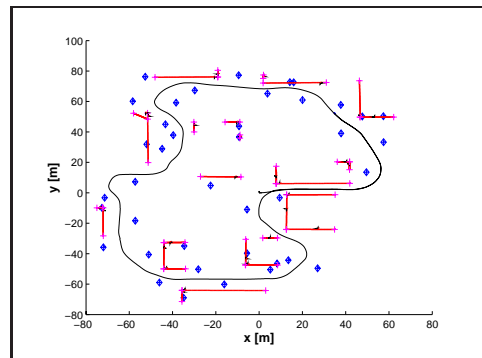


Figure 4.12: Carte estimée par le système SLAM-contraint avec les données simulées d'un environnement hétérogène : les amers de type droite sont représentés en rouge, les amers de type point en bleu et la trajectoire estimée du véhicule en noir.

La carte estimée par le système SLAM-contraint (cf. la figure 4.12) est cohérente avec la vérité de terrain de l'expérimentation (cf. la figure 4.11 (a)). Tous les éléments présents dans l'environnement simulé ont été cartographiés. Cependant, quelques fausses détections d'amers sont réalisées. Des cercles ont été cartographiés à proximité de certaines droites. Les étapes d'association de données et de gestion de la carte n'ont donc pas éliminés la totalité des fausses détections. Cependant, la localisation du véhicule est précise avec une erreur moyenne de 24 cm (cf. la figure 4.13 (a)).

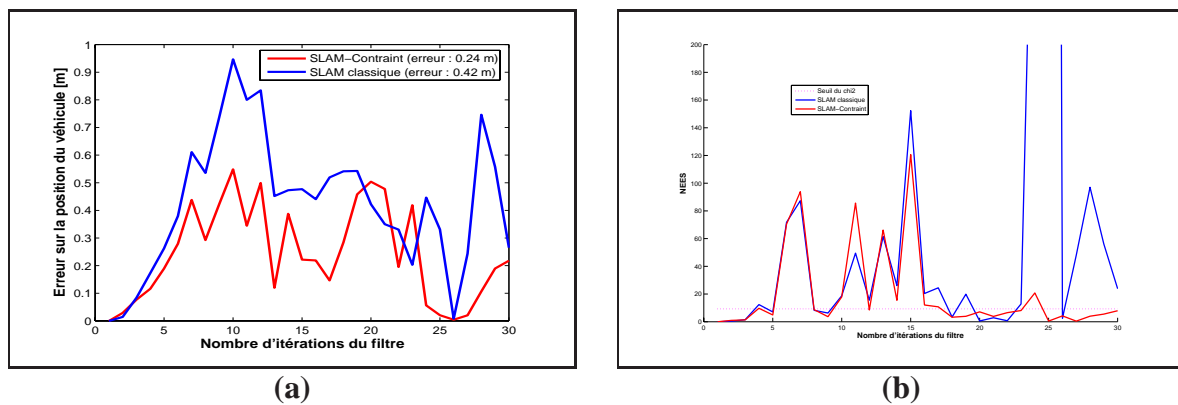


Figure 4.13: Résultats avec une trajectoire du véhicule bouclée dans un environnement hétérogène (le repère de la carte connue n'est pas exact) : (a) évolution de l'erreur et (b) évolution de la distance NEES.

Pour souligner l'intérêt de notre approche SLAM-contraint multi-cartes, ses résultats ont été comparés à ceux d'une approche classique multi-cartes. L'erreur sur la position du véhicule ainsi que l'évolution de la distance NEES ont été étudiées. L'analyse de l'évolution de l'erreur confirme l'intérêt de l'approche SLAM-contraint (cf. la figure 4.13 (a)). Cette approche donne

des résultats plus précis que l'approche classique. Ce résultat est logique. Comme le système possède des informations sur l'environnement, les estimations sur la position du véhicule sont plus précises. De plus, l'évolution de la distance NEES (cf. la figure 4.13 (a)) prouve que les estimations de l'approche SLAM-contraint sont plus consistantes. En effet, les valeurs de la distance NEES restent faibles, en particulier lors de la fermeture de la trajectoire, en comparaison avec l'approche sans contrainte. Il faut tout même remarquer que ces résultats sont inconsistants durant une partie de l'expérimentation (de l'itération 5 à 15). Ces phénomènes s'expliquent en observant la carte connue et la trajectoire réalisée par le véhicule. En effet, cette carte est composée de peu d'amers qui sont, de plus, regroupés au début de la trajectoire du véhicule. Au début, le système peut observer au moins trois amers de la carte connue. Ces estimations sont donc consistantes. Puis, lorsque le véhicule se trouve à gauche de la carte, il ne peut plus observer trois amers de la carte connue. La condition d'observabilité n'est plus respectée et les estimations ne peuvent plus être garanties. L'incertitude sur l'orientation du véhicule augmente durant ce parcours et les problèmes dus à la non linéarité des équations sont de plus en plus visibles (cf. le chapitre 3). En revanche, grâce à la contrainte apportée par la carte connue, lors de la fermeture de la trajectoire (itérations numéro 25), les estimations redeviennent consistantes. En effet, le système peut de nouveau observer trois amers de la carte connue.

Au final, l'approche SLAM-contraint multi-cartes s'appliquent à des environnements hétérogènes composés de cercles et de droites. L'emploi d'une carte connue de l'environnement permet d'améliorer la précision d'un système EKF-SLAM. De surcroît, cette amélioration entraîne une fermeture correcte de la trajectoire du véhicule en termes de consistance des estimations. Maintenant, l'approche SLAM-contraint multi-cartes est testée sur des données réelles.

4.8 Résultats expérimentaux

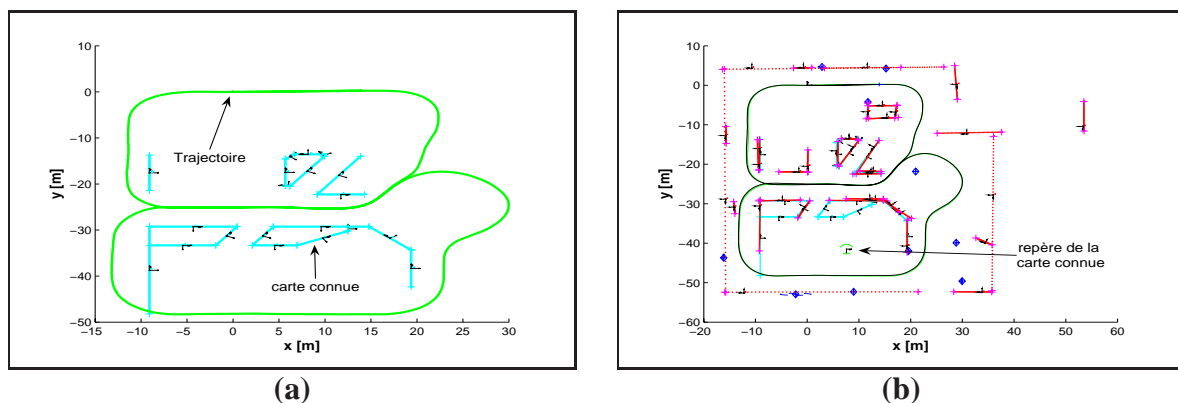


Figure 4.14: Présentation du contexte expérimental de l'application de l'approche SLAM-contraint multi-cartes : (a) Vérité de terrain de l'expérimentation réalisée sur site PAVIN et (b) carte estimée par la méthode SLAM-contraint.

Les données ont été recueillies sur le site expérimental de PAVIN (cf. l'annexe E). La carte connue de cet environnement a été obtenue à partir des plans électroniques de l'architecte (cf. la figure 4.14 (a)). La vérité de terrain de la trajectoire du véhicule a été mesurée à partir d'un

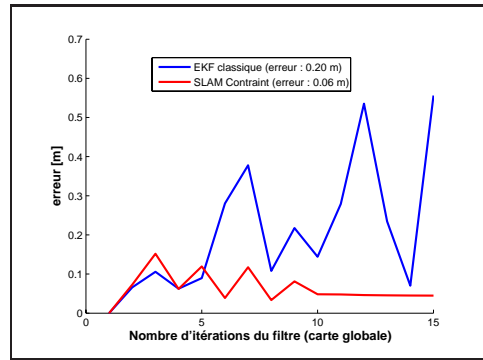


Figure 4.15: Evolution de l'erreur sur la position du véhicule.

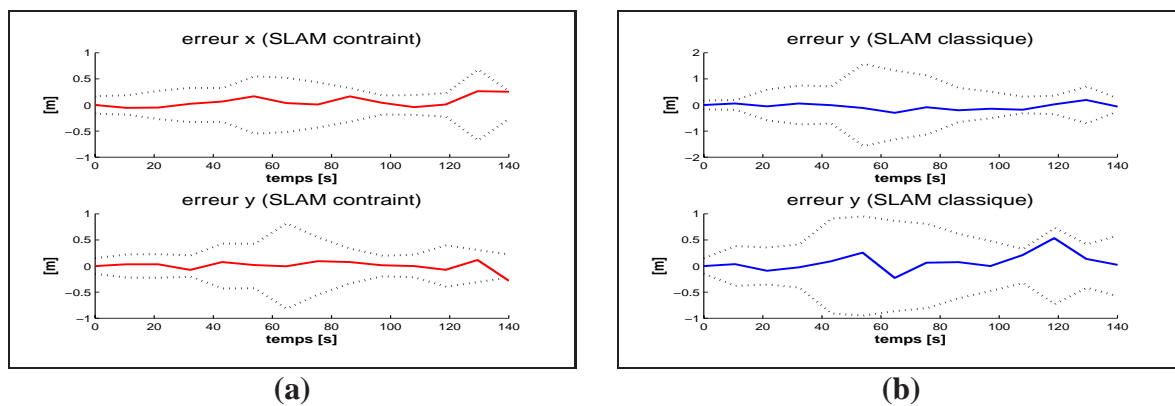


Figure 4.16: Evolution de l'erreur sur les coordonnées du véhicule avec les bornes d'incertitudes à 2 écart-types : (a) SLAM-contraint et (b) Approche classique de cartes locales.

capteur D-GPS d'une précision de 5 cm. Pour prouver la robustesse de l'approche SLAM-contraint, la valeur du vecteur $\mathbf{X}_{W_c}^{W_g}$ a été déterminée approximativement en cochant « à la main » la position sur le plan d'architecte (cf. la figure 4.14 (b)). Aucun instrument de mesure n'a été employé. Comme la mesure n'est pas précise, la matrice de covariance $\mathbf{C}_{W_c}^{W_g}$ a été initialisée avec des valeurs d'incertitude importantes : 0.5 m sur les coordonnées cartésiennes, x_c et y_c , et 6 degrés (0.1 rad) sur l'orientation, θ_c .

L'évolution de l'erreur sur la position du véhicule confirme les résultats de simulation (cf. la figure 4.15). L'approche SLAM-contraint, grâce aux informations incluses dans la carte connue, améliore la précision et la convergence des estimations, en particulier lors de la fermeture de la trajectoire. De surcroît, les estimations restent consistantes durant l'ensemble de l'expérimentation pour les deux approches (cf. la figure 4.16). En effet, les erreurs sur les coordonnées du véhicule restent bornées par les valeurs d'incertitude définies dans la matrice de covariance du système.

Il faut également remarquer que les valeurs des écart-types obtenues avec l'approche SLAM-contraint sont plus faibles qu'avec l'approche classique. Elles ne dépassent pas 1 m avec l'approche SLAM-contraint alors que celles estimées par le filtre EKF atteignent 2 m. Comme les incertitudes sont plus faibles, les problèmes dus à la non-linéarité du système sont minimisés

(voir les démonstrations présentées dans le chapitre 3). Le système réalise ainsi une fermeture de sa trajectoire plus précise.

Au final, les estimations de la position du véhicule, avec l'approche SLAM-contraint multi-cartes, sont plus précises mais également plus sûres. Cette constatation prouve la pertinence de l'approche. Cependant, bien que les résultats soient précis, une mesure plus correcte de la position du repère de la carte connue aurait permis l'obtention de meilleurs résultats. En particulier, au début de l'expérimentation, lorsque la valeur de l'erreur sur la position du véhicule ne cesse d'osciller (cf. la figure 4.15).

4.9 Conclusions

Dans ce chapitre, différents points importants viennent d'être abordés :

- le calcul ainsi que l'étude de l'observabilité des estimations d'un système non-linéaire ;
- la non-observabilité d'une approche SLAM ;
- une méthode SLAM-contraint permettant de répondre au problème d'observabilité ;
- une application multi-cartes de la méthode SLAM-contraint basée sur un filtre EKF-SLAM.

Non-observabilité d'une approche SLAM :

La notion d'observabilité est simple : si les estimations d'un système ne sont pas observables, alors leur convergence ne peut être garantie. Pour statuer sur ce critère dans le cas d'un système non-linéaire, le rang de la matrice d'observabilité est étudié. Dans la littérature, peu d'articles, traitant d'approche SLAM, font référence à ce critère. Pourtant, l'étude proposée dans ce document prouve que la matrice d'observabilité d'un système SLAM, tel qu'il est défini, ne remplit pas la condition de rang énoncée dans l'article [81]. Elle a été réalisée pour deux types d'amers différents : des points et des droites. Les conclusions sont identiques : **les estimations d'un système SLAM ne sont pas observables**. Donc, si, par exemple, un biais existe sur la position initiale du véhicule ou si, de fausses détections et de fausses associations de données sont réalisées, les estimations du système peuvent diverger.

Méthode de SLAM-contraint multi-cartes :

Pour répondre au problème de la non-observabilité du SLAM, une méthode, nommée **SLAM-contraint**, est proposée. Dans cette approche, des informations sur la position de certains amers composant l'environnement, sont fournies au système. L'objectif est de contraindre les positions d'éléments de la carte estimée pour que le système devienne observable.

Dans notre méthode, les informations fournies correspondent à une carte connue de l'environnement. Elles peuvent être entachées d'erreurs en fonction de l'instrument de mesure employé. Dans le but de modéliser cette imprécision, un vecteur, décrivant la transformation rigide entre le repère de la carte estimée et celui de la carte connue, est ajouté dans le vecteur d'état du système. L'étude de l'observabilité démontre que ce nouveau système est **localement obser-**

vable à la condition qu’au moins trois amers, présents sur la carte connue, soient observés à chaque itération du filtre.

Pour garantir cette condition, la méthode SLAM-contraint fut couplée à une approche multi-cartes. En effet, à part si la carte connue contient la plupart des amers présents dans l’environnement, il est quasiment impossible que le système observe au moins trois amers à chaque itération. Pour garantir cette condition sans avoir à employer une carte décrivant une multitude de positions d’amers, la méthode SLAM-contraint est couplée à une approche multi-cartes [29]. En effet, dans une approche multi-cartes, les cartes locales sont plus vastes que les mesures télémétriques. Il est donc fort probable que ces cartes locales contiennent au moins trois amers de la carte connue. Ainsi, la **méthode SLAM-contraint multicartes, présentée dans mes travaux, permet de garantir l’observabilité des estimations du système** lors des mises à jour de la carte globale de l’environnement.

Applications de la méthode SLAM-contraint multi-cartes :

La première application de la méthode SLAM-contraint multi-cartes est proposée dans ce chapitre. Elle repose sur une représentation *WorldCentric* de l’environnement et utilise un filtre EKF (cf. l’annexe F). Pour diminuer les problèmes de non-linéarité inhérents à l’application EKF-SLAM (cf. le chapitre 3), deux précautions ont été adoptées. Premièrement, les vecteurs définissant la transformation rigide entre la carte connue et les autres cartes du système ne sont pas observables. Deuxièmement, lors de l’initialisation des vecteurs des cartes estimées (locales ou globale), ces vecteurs ne sont pas corrélés avec les autres états du système. Ceci permet de garder l’indépendance entre les différentes cartes du système. Les résultats obtenus avec des données simulées et réelles confirment l’intérêt de la méthode. Les estimations du système sont plus précises en comparaison d’une approche classique. De plus, comme les estimations sont plus justes, la méthode permet une fermeture de la trajectoire du véhicule avec des estimations consistantes.

La deuxième application fut présentée dans l’article [38]. Deux modifications ont été apportées par rapport à l’approche précédente. Premièrement, pour que la carte globale et les locales soient indépendantes statistiquement, le vecteur de transformation rigide, entre la carte connue et les autres cartes, est mesuré à l’aide de la méthode CCDA³ [10]. Deuxièmement, pour améliorer la consistance des estimations, une représentation *RobotCentric* de l’environnement a été adoptée. Avec ces deux modifications, les équations du système ont été reformulées pour permettre l’observation des vecteurs de transformation rigide entre les différentes cartes. Les résultats obtenus sont de nouveau plus précis qu’une approche classique. Cependant, l’estimation du vecteur de transformation rigide diverge lorsque des fausses associations de données se produisent. Ce problème entraîne la divergence de l’ensemble du système. De plus, l’emploi de la représentation *RobotCentric* ne résout pas totalement les problèmes de consistance des estimations du filtre de Kalman. C’est pour cela que seule la première application a été présentée dans ce manuscrit.

3. *Combined Constraint Data Association*

Perspectives :

Le système décrit par la méthode SLAM-contraint multi-cartes est observable. Cependant, le filtre EKF ne peut garantir des estimations consistantes. Comme le prouvent les résultats obtenues avec les deux précédentes applications, les équations du système sont écrites en prenant soin de minimiser les problèmes dus à la non-linéarité des équations. L'emploi d'un filtre plus robuste tel que le PEKF-SLAM (cf. le chapitre 3), un filtre à particules [152] ou une approche SAM [49] pourrait permettre d'améliorer la consistance des estimations tout en conservant la propriété d'observabilité de la méthode.

Au final, lorsque toutes les précautions sont prises via les équations du système, les estimations ne divergent pas. La carte estimée par l'approche SLAM-contraint multi-cartes peut donc être utilisée comme nouvelle carte connue par le système lorsque le véhicule se déplace dans le même environnement. Comme la carte connue est décorrélée des autres cartes, les problèmes de consistance du filtre EKF peuvent être négligés. Une telle technique possède deux avantages. Premièrement, à chaque nouvelle expérimentation, les cartes de l'environnement deviennent de plus en plus précises. Deuxièmement, la comparaison des deux cartes de l'environnement obtenues permet de détecter les changements survenus dans l'environnement et ainsi de pouvoir mettre à jour les plans existants des lieux visités.

Chapitre 5

Conclusions et perspectives

5.1 Bilan général

L'objectif des travaux présentés dans ce mémoire était d'étudier trois aspects différents d'une application EKF-SLAM : une représentation hétérogène de l'environnement, la consistance des estimations d'un filtre EKF-SLAM et l'observabilité d'un tel système. Ces études ont permis de réaliser deux applications originales de SLAM : une application de SLAM reposant sur un filtre polynomial de Kalman et une approche EKF-SLAM contrainte par la connaissance a priori d'une carte de l'environnement.

Représentation hétérogène de l'environnement :

Dans l'optique de représenter plus fidèlement les environnements visités par le véhicule, deux types d'amers ont été employés dans la cartographie de l'environnement : les droites et les cercles. Pour adopter cette représentation hétérogène dans notre approche SLAM, une méthode de mise en correspondance d'amers de géométries différentes a été employée. Elle s'appuie sur la théorie des symétries définie dans l'article [145]. Grâce à cette méthode d'association de données, le système identifie des amers représentant la même entité et une étape de gestion de la carte estimée élimine les détections erronées.

Les premiers résultats expérimentaux valident l'emploi d'un filtre EKF-SLAM reposant sur une représentation hétérogène de l'environnement. Les cartes obtenues sont représentatives de l'environnement visité. En effet, un opérateur consultant cette carte peut facilement discerner par exemple des murs, des arbres ou des piliers. Cependant, ces premiers résultats soulèvent deux interrogations. Tout d'abord, la convergence des estimations du système n'est pas cohérente. En effet, les valeurs des incertitudes, en particulier celle associée à l'orientation du véhicule, n'augmentent plus alors que le véhicule ne réalise pas une fermeture de sa trajectoire. Deuxièmement, lorsque le système réalise des fausses détections, ses estimations divergent.

Pour répondre à ces problèmes, deux études ont été réalisées et détaillées dans ce manuscrit. La première porte sur la consistance des estimations du filtre EKF-SLAM. La seconde traite de la notion d'observabilité du système SLAM.

Etude de la consistance des estimations du filtre EKF-SLAM :

Dans la littérature, des travaux présentant les problèmes d'estimation rencontrés lors de l'application du filtre EKF-SLAM existent mais peu d'articles, si ce n'est [11] et [83], en proposent une explication précise. Dans mes travaux, une étude approfondie, reposant sur ces deux articles, est réalisée dans le but d'identifier et de quantifier les problèmes de consistance des estimations d'un filtre EKF-SLAM. Pour se faire, les trois théorèmes de convergence définis dans [83] sont proposés lorsque la carte est composée de cercles et de droites. En effet, à la différence des articles [11] et [83], l'étude est étendue aux droites. Cette étude souligne que les mêmes théorèmes de convergence des estimations s'appliquent aux deux types d'amers. Cependant, comme le prouvent les démonstrations présentées dans ce manuscrit, la convergence de l'incertitude associée aux droites ne dépend pas uniquement des états du véhicule comme les amers de type point, mais également de son orientation. L'application de ces théorèmes sur des résultats expérimentaux souligne que les estimations souffrent effectivement d'inconsistance. Cette étude a permis de conclure sur l'incohérence des estimations obtenues avec l'approche EKF-SLAM, en particulier lorsque l'incertitude initiale sur l'orientation du véhicule est importante [34] mais également lorsqu'un amer comportant une information angulaire est employé.

Fondée sur l'identification de ces problèmes, une approche, redéfinissant les équations du système, est proposée dans notre cadre applicatif : le **filtre PEKF-SLAM**. Elle repose sur une nouvelle écriture du vecteur d'état permettant de décomposer les équations du système en sommes de polynômes. L'étude des résultats, sur des scénarii simples de simulations, prouvent que cette approche réalise, pour une entité de la carte estimée, une "décorrélation" des états décrivant sa position dans la carte (position cartésienne et orientation) des états définis par les puissances de Kronecker et permettant l'écriture des équations de prédiction et d'observation du système (cf. le paragraphe 3). Grâce à ce phénomène, les incertitudes estimées par le filtre PEKF-SLAM sont plus consistantes et plus précises que celles obtenues avec l'approche EKF-SLAM classique ou des approches fondées sur le filtre UKF [34]. Les résultats obtenus avec des données simulées et expérimentales et présentés dans ce manuscrit le soulignent. Cependant, des inconvénients liés à son utilisation sont à souligner. Premièrement, cette solution reste une solution analytique encore difficilement applicable en temps réel à la vue du nombre d'états utilisés pour définir le système. Deuxièmement, le problème de la fermeture de la trajectoire du véhicule pose encore des difficultés du point de vue de la cohérence des estimations. Même si les tests statistiques prouvent que les estimations du filtre PEKF-SLAM sont plus consistantes que celles de l'approche EKF-SLAM, des améliorations sont à apporter.

En conclusion, cette étude nous a permis de comprendre le problème d'inconsistance des estimations lié à l'emploi d'un filtre EKF-SLAM. Pour que les résultats de l'approche soient cohérents, l'erreur et l'incertitude sur les orientations, du véhicule et des amers, doivent rester faibles. Cette constatation nous a amené à réfléchir à une nouvelle approche contrainte par la connaissance d'informations sur l'environnement. L'objectif est de minimiser les erreurs faites lors de la linéarisation des équations avec des estimations plus précises.

Observabilité du système :

En étudiant les premiers résultats expérimentaux, il a été remarqué que si le système réalise de fausses associations ou si des mesures sont biaisées, les estimations divergent. Pour comprendre pourquoi les estimations ne peuvent que diverger dans de telles situations, une étude de l'observabilité du système a été réalisée pour deux types d'amers différents : des points (ou cercles) et des droites. Les conclusions sont identiques : **les estimations d'un système SLAM ne sont pas observables**. Ce problème a été identifié initialement dans l'article [99]. Dans la continuité de ces travaux, nous proposons une méthode générale nommée **SLAM-contraint**. Cette approche repose sur l'utilisation d'informations disponibles sur l'environnement : des cartes ou des plans. Cependant, cette carte connue par le système est imprécise et non exacte comme celle employée dans les travaux [99].

L'étude théorique de la matrice d'observabilité prouve que cette méthode répond au problème d'observabilité. Plus précisément, elle démontre que ce nouveau système est **localement observable à la condition qu'au moins trois amers, présents sur la carte connue, soient observés à chaque itération du filtre**. Pour garantir l'observation de ces trois amers à chaque mise à jour de la carte, la méthode SLAM-contraint a été couplée à une approche multi-cartes [29]. Grâce à cette nouvelle représentation de l'environnement, les estimations de cette nouvelle approche, nommée **SLAM-contraint multi-cartes**, sont observables à chaque itération du filtre.

Finalement, une application de la méthode SLAM-contraint multi-cartes, utilisant un filtre EKF, est proposée avec les résultats suivants :

- les positions estimées dans la carte reconstruite sont plus précises et plus consistantes qu'une approche sans contrainte (si la condition d'observabilité est respectée) ;
- lors de la fermeture de la trajectoire du véhicule, les estimations de l'approche SLAM-contraint restent consistantes (si la condition d'observabilité est respectée).

De plus, comme toutes les précautions sont prises via les équations du système, les estimations ne divergent pas. La carte estimée par l'approche SLAM-contraint multicartes peut donc être utilisée comme carte connue par le système lorsque le véhicule se déplace de nouveau dans le même environnement. Comme la carte connue est décorrélée des autres cartes, les problèmes de consistance du filtre EKF sont négligés. Une telle technique possède deux avantages :

- à chaque nouvelle expérimentation, les cartes de l'environnement deviendraient de plus en plus précises ;
- la comparaison des deux cartes de l'environnement obtenues permettrait de détecter les changements survenus dans l'environnement et ainsi de pouvoir mettre à jour les plans existants du lieu visité.

5.2 Perspectives

Les travaux de thèse présentés ici ont permis de mettre en évidence deux problèmes majeurs d'une approche SLAM : la **consistance des estimations** et l'**observabilité du système**. Pour solutionner ces problèmes, deux approches ont été proposées : le filtre **PEKF-SLAM** et la méthode **SLAM-contraint**. Pour chacune de ces deux méthodes, des améliorations sont envisageables. Elles ont été présentées respectivement dans les conclusions des chapitres 3 et 4.

Cependant, deux aspects de l'approche EKF-SLAM peuvent encore être améliorés : une représentation plus adaptée à l'environnement visité par le véhicule et une correction des biais potentiels sur les données proprioceptives. Dès que ces problèmes auront été solutionnés, la méthode de SLAM-contraint multi-cartes pourra être introduite dans des approches plus complexes telles que de la navigation de véhicule autonome ou une expérimentation de SLAM dans un environnement dynamique. L'objet de cette section est d'exposer les limitations des approches présentées, d'en proposer des solutions envisageables et d'en présenter les applications potentielles.

Représentation de l'environnement

Même si des précautions ont été prises via une représentation hétérogène de l'environnement, une cartographie en deux dimensions de l'environnement reste inadaptée au monde réel. Les résultats expérimentaux, proposés dans le chapitre 2, en attestent. En effet, lors de la cartographie de cet environnement extérieur, le relief du terrain a entraîné des fausses détections et des cartographies erronées d'amers. Par exemple, un monticule de terre a été représenté avec plusieurs cercles et plusieurs droites. Même si une étape de gestion de la carte estimée permet de supprimer certains amers (cf. le chapitre 2), de telles erreurs peuvent entraîner une divergence des estimations du système. Pour éviter de tels scénarii, il est primordial que le système puisse avoir à disposition des informations en trois dimensions de l'environnement.

L'emploi d'un télémètre avec plusieurs nappes est une solution. Les travaux présentés dans l'article [43] répondent à cette problématique. Cependant, la carte de l'environnement correspond à un nuage de points. Pour suivre la philosophie de mes travaux, il faudrait définir des amers plus évolués tels que des plans, des cylindres et d'autres formes géométriques à trois dimensions. Une autre approche [118] correspond à la fusion d'informations visuelles avec des informations télémétriques. Là encore, l'extraction de formes géométriques des informations issues de la fusion de données représenterait une réelle difficulté.

Une dernière solution correspond à l'emploi de capteurs plus évolués tels que le radar [72] ou une caméra panoramique [73]. Ces capteurs permettent de cartographier l'environnement sur 360 degrés. Grâce à une telle vision de l'environnement, le véhicule serait capable d'observer dans une même mesure des amers situés derrière lui, observés antérieurement de nombreuses fois et dont l'incertitude sur leurs positions serait donc faible, et des nouveaux amers situés devant lui. A chaque itération du filtre, l'incertitude peu importante de certains amers serait donc propagée aux amers ajoutés à la carte estimée. Ainsi, l'incertitude sur les positions des amers et du véhicule resterait faible. L'emploi de tels capteurs permettrait de limiter les problèmes de non-linéarité décrits précédemment. Ainsi, la carte estimée par le système serait consistante donc plus précise, et surtout, plus représentative de l'environnement.

L'emploi de capteurs fournissant des informations en trois dimensions, fourniront au système des données extéroceptives permettant une représentation de l'environnement plus pertinente. Cependant, pour qu'une application SLAM telle que celle présentée dans mes travaux donne de meilleurs résultats, il faut également garantir des données proprioceptives sans aucun biais. Dans le paragraphe suivant, sont introduites une définition des biais sur des mesures relatives au déplacement du véhicule et des solutions envisageables pour les corriger.

Biais sur les données proprioceptives du véhicule

L'approche SLAM-contraint multi-cartes pourrait être une solution à ce problème. Cependant, les estimations sont observables à l'unique condition qu'au moins trois amers présents sur la carte connue soient observés. Il faut donc que le système soit capable d'associer les positions estimées des amers avec celles référencées sur la carte connue. Un biais sur les données proprioceptives (un glissement des roues par exemple) ne garantirait plus l'observation de ces trois amers. Les estimations du système divergeraient.

Comme le détaille l'article [19], la modélisation du déplacement d'un véhicule est loin d'être exacte. Il existe des erreurs systématiques qui peuvent être corrigées avec une calibration et une modélisation précise. Il existe également des erreurs dites non systématiques telles que des glissements. Ces erreurs sont difficilement corrigeables. De plus, leur modélisation est délicate car ces erreurs se produisent de manière ponctuelle. Des filtres à particules ou des approches de type Monte-Carlo peuvent représenter des solutions à ce problème. En revanche, le filtre de Kalman ne peut être une solution. Ce type de filtre corrige uniquement les bruits blancs et gaussiens.

Durant nos travaux, nous avons essayé de résoudre ce problème en comparant le filtre de Kalman à un système de régulation avec une boucle de retour. Par analogie, le filtre de Kalman correspond à une correction proportionnelle reposant sur le calcul d'incertitudes. Une solution corrigeant les biais correspond à l'ajout d'une action de type intégrale dans ce correcteur. En reprenant la philosophie présentée dans l'article [97], nous avons proposé une approche EKF-SLAM couplée avec un estimateur du biais sur les données proprioceptives (cf. la figure 5.1).

Les premiers résultats sur des simulations de localisation GPS avec des données proprioceptives biaisées sont proposés sur la figure 5.2. Cette simulation de localisation repose sur la trajectoire présentée dans l'annexe E. Les résultats démontrent deux choses. Premièrement, le filtre de Kalman est un estimateur non-biaisé. Dès lors qu'un biais est ajouté sur les mesures proprioceptives, l'erreur sur la position du véhicule augmente rapidement et les estimées deviennent inconsistantes (la valeur de la distance NEES dépasse le seuil). L'ajout de l'estimateur du biais permet de garder les estimations consistantes lorsqu'un biais est ajouté sur les données. Cependant, lorsqu'aucun biais est ajouté, le filtre EKF classique donne des résultats plus précis (cf. la figure 5.2).

Après ces premières observations, cette approche a été appliquée dans un contexte SLAM. La méthode est présentée dans notre article [33]. Pour obtenir ces premiers résultats, il a été supposé que les variables d'état \mathbf{X} et l'estimation du biais \mathbf{w} étaient indépendantes. Cette approximation permettait de rendre le système stable. La prochaine étape est de considérer les corrélations entre ces deux variables dans le calcul de la matrice de covariance associée au biais \mathbf{w} . Une telle approche pourrait être une solution innovante pour corriger un biais avec un filtre

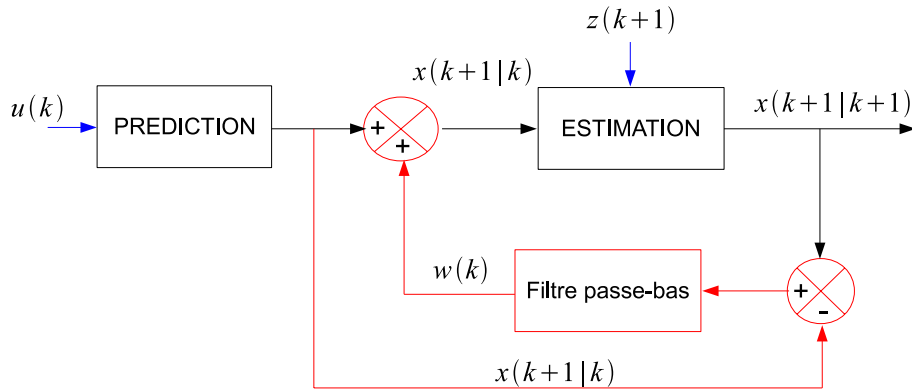


Figure 5.1: Schéma présentant l'approche EKF couplé à un estimateur de perturbations sur les données proprioceptives : les éléments de couleur noire représentent l'approche classique et l'estimateur de perturbations (action intégrale) est représenté en rouge. La variable $w(k)$ représente l'estimation du biais sur les mesures. Cette estimation est filtrée dans le but d'atténuer ces variations. Par analogie à la régulation, les paramètres de ce filtre correspondent au gain sur l'action intégrale du correcteur.

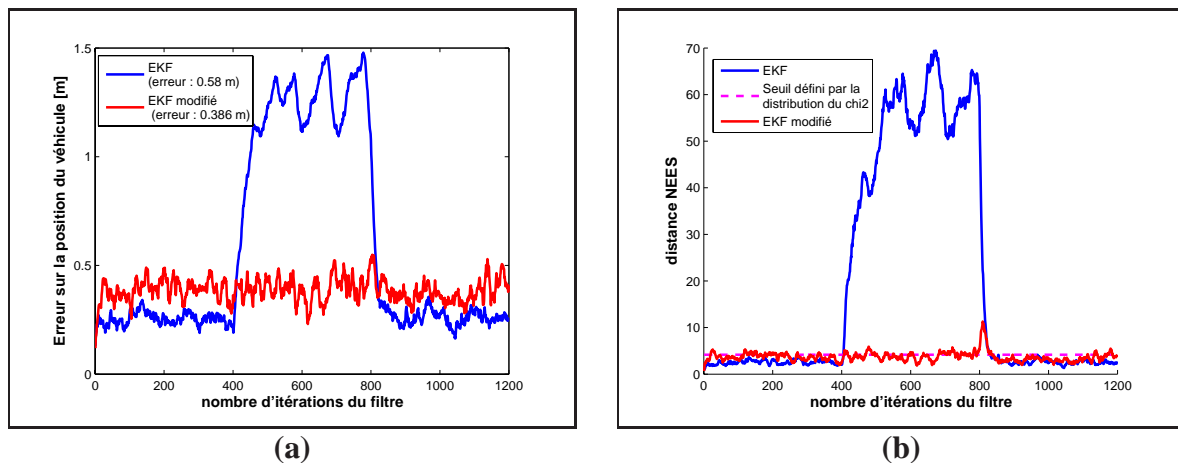


Figure 5.2: Résultats d'une simulation de localisation d'un véhicule avec des mesures GPS bruitées (variance de 1 mètre). Des biais constants de -0.2 ms^{-1} sur la vitesse et 1.5 degrés sur l'angle de braquage des roues, sont ajoutés sur les données proprioceptives entre les itérations 400 et 800 du filtre. La figure (a) représente l'évolution de l'erreur moyenne sur la position du véhicule et la figure (b) l'évolution de la distance NEES sur 20 simulations de Monte Carlo.

EKF.

Applications envisageables de l'approche SLAM-contraint

Dès lors que la modélisation de l'environnement sera plus appropriée et que les biais sur les données odométriques seront corrigés, soit par l'apport de capteurs plus performants, soit par

l'emploi de filtres plus robustes, la méthode SLAM-contraint peut permettre de résoudre des problèmes bien plus complexes.

Les systèmes de navigation autonome de véhicule sont issus de la mise en commun d'une approche de localisation couplée à un algorithme de commande. Pour que les commandes envoyées au véhicule soient le plus justes possibles, il faut que le système de localisation fournisse une estimation de la position du véhicule précise mais également consistante. L'incertitude doit être cohérente avec l'erreur sur l'estimation des paramètres du système. Dans les travaux de Tessier et al. [147], ce module de localisation correspond à la fusion d'informations provenant de plusieurs capteurs et de cartes de l'environnement. L'ajout de notre méthode de SLAM-contraint dans ce système de localisation en améliorerait deux aspects essentiels. Premièrement, le système de localisation du véhicule serait plus précis en employant une approche SLAM-contraint. Deuxièmement, une détection des nouveaux amers de l'environnement permettrait de mettre à jour la carte connue mais également de localiser les éventuels obstacles présents sur la trajectoire du véhicule.

Cependant, avant de réaliser soit le système précédent, soit une application dans un environnement dynamique [157], il faut s'assurer que les problèmes d'inconsistance des estimations et de biais sur les mesures proprioceptives soient solutionnés et que la représentation de l'environnement soit améliorée.

Annexe A

Loi gaussienne et description du filtre de Kalman

La loi gaussienne, appelée également loi normale, est une des distributions de probabilité les plus usitées pour modéliser des paramètres à estimer. Elle fut introduite par le mathématicien Abraham de Moivre en 1733. Elle fut employée pour approcher des probabilités associées à des variables aléatoires binomiales possédant un nombre de paramètres, n , très grand.

Ensuite, cette loi de probabilité fut mise en évidence par Gauss au XIXe siècle, d'où son nom de loi gaussienne. Elle permit de modéliser de nombreuses études biométriques. Sa densité de probabilité dessine une courbe dite "courbe en cloche" ou courbe de Gauss (définition issue de Wikipedia).

A.0.1 Loi gaussienne à une dimension

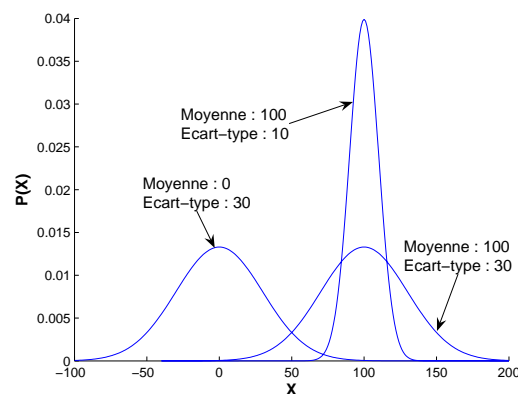


Figure A.1: Distribution gaussienne avec des paramètres différents

Une variable aléatoire continue X est distribuée selon une loi normale ou gaussienne (figure

A.1) si sa densité de probabilité, $p(X)$, est exprimée par :

$$p(X) = \frac{1}{\sqrt{2\pi \cdot \sigma^2}} \exp \left\{ -\frac{1}{2} \left(\frac{X - \mu}{\sigma} \right)^2 \right\} \quad (\text{A.1})$$

L'observation de l'équation (A.1) permet de définir la densité de probabilité, $p(X)$, par deux moments :

- une moyenne : $\mu = E(X)$
- une variance : $\sigma^2 = E(X^2) - [E(X)]^2$

A.0.2 Loi gaussienne à n dimensions

La variable aléatoire correspond maintenant à un vecteur de n paramètres :

$$\mathbf{X} = [x_1 \quad \cdots \quad x_n]^T \quad (\text{A.2})$$

En reprenant la définition d'une densité de probabilité (A.1), cette variable aléatoire est distribuée selon une loi gaussienne si sa densité de probabilité s'écrit :

$$p(\mathbf{X}) = \frac{1}{\sqrt{2\pi \cdot \mathbf{P}}} \exp \left\{ -\frac{1}{2} (\mathbf{X} - \hat{\mathbf{X}})^T \cdot \mathbf{P}^{-1} \cdot (\mathbf{X} - \hat{\mathbf{X}}) \right\} \quad (\text{A.3})$$

Les deux moments statistiques permettant de définir $p(X)$ sont :

- un vecteur d'état moyen : $\hat{\mathbf{X}} = E(\mathbf{X})$
- une matrice de variance-covariance : $\mathbf{P} = E(\mathbf{X}^2) - [E(\mathbf{X})]^2$ représentant les variances sur les paramètres \mathbf{X} (éléments sur la diagonale de la matrice \mathbf{P}) et les covariances entre ces mêmes paramètres (éléments des parties triangles de la matrice \mathbf{P}).

A.1 Estimateurs probabilistes

Dans ce paragraphe, les principales méthodes d'estimation vont être décrites pour permettre l'introduction du filtre de Kalman.

A.1.1 Pourquoi utiliser un estimateur ?

Dans notre cadre d'étude, une estimation peut être définie comme le calcul approximatif d'une position d'un véhicule et d'une carte de l'environnement par référence à des données plus ou moins sûres. En d'autres termes, le but est d'obtenir la meilleure estimation, notée $\hat{\mathbf{X}}$, d'un paramètre \mathbf{X} étant donné un ensemble de k mesures noté \mathbf{Z}_k entachées de bruit :

$$\mathbf{Z}_k \triangleq \{z_0, z_1, z_2 \cdots, z_k\} \quad (\text{A.4})$$

Il faut remarquer que si les mesures \mathbf{Z}_k n'étaient pas bruitées, une simple solution algébrique suffirait pour déterminer les paramètres X du système. Or les mesures \mathbf{Z}_k provenant des capteurs extéroceptifs sont supposées ne pas être précises et donc corrompues par du bruit. La

$i^{\text{ème}}$ mesure, z_i , du paramètre \mathbf{X} à l'instant i correspond donc à l'estimation $\hat{X}(i)$ avec une incertitude associée. Pour définir cette fonction de vraisemblance entre ces deux variables, une distribution gaussienne est définie :

$$p(\mathbf{Z}|\mathbf{X}) = \frac{1}{C} \cdot \exp\left\{-\frac{1}{2}(\mathbf{Z} - \mathbf{X})^T \mathbf{P}^{-1}(\mathbf{Z} - \mathbf{X})\right\} \quad (\text{A.5})$$

$p(\mathbf{Z}|\mathbf{X})$ se lit comme la probabilité conditionnelle d'obtenir une mesure \mathbf{Z} étant donnée une valeur particulière de \mathbf{X} . Dans l'équation (A.5), C est une constante de normalisation indépendante de \mathbf{X} et \mathbf{P} la matrice de covariance associée au paramètre estimé $\hat{\mathbf{X}}$.

L'équation (A.5) implique que les variables aléatoires modélisant le vecteur de paramètres \mathbf{X} et les mesures \mathbf{Z} suivent des lois gaussiennes :

$$\begin{aligned} p(\mathbf{X}) &\sim \mathcal{N}(\hat{\mathbf{X}}, \mathbf{P}) \\ p(\mathbf{Z}) &\sim \mathcal{N}(\hat{\mathbf{Z}}, \mathbf{R}) \end{aligned} \quad (\text{A.6})$$

Après avoir défini les différentes variables aléatoires permettant la détermination de paramètres d'un système, les principales méthodes d'estimation statistique vont être introduites.

A.1.2 Méthodes d'estimation d'une variable aléatoire

La distribution $p(\mathbf{X})$ est supposée connue et les observations \mathbf{Z} sont modélisées par la probabilité $p(\mathbf{Z}|\mathbf{X})$; avec toutes les définitions précédentes, les trois principales méthodes d'estimation paramétrique peuvent s'écrire :

- **Estimateur des moindres carrés pondérées :**

$$\hat{\mathbf{X}} = \arg \min_X \|\mathbf{R}^{-1}(\mathbf{H}\mathbf{X} - \mathbf{Z})\|^2 \quad (\text{A.7})$$

La matrice \mathbf{H} correspond à la matrice jacobienne de la fonction d'observation \mathbf{h} et la matrice \mathbf{R} correspond à la matrice de covariance associée aux mesures \mathbf{Z} :

$$\mathbf{R} = \begin{bmatrix} \sigma_{z_1}^2 & 0 & \cdots & 0 \\ 0 & \sigma_{z_2}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \sigma_{z_k}^2 \end{bmatrix} \quad (\text{A.8})$$

Dans cette méthode, le but est de minimiser l'erreur pondérée par l'incertitude sur les mesures au carrée. La valeur de l'estimation de \mathbf{X} , solution de l'équation (A.7), s'écrit :

$$\hat{\mathbf{X}} = (\mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H})^{-1} \cdot \mathbf{H} \cdot \mathbf{R}^{-1} \cdot \mathbf{Z} \quad (\text{A.9})$$

- **Estimateur à maximum de vraisemblance :**

$$\hat{\mathbf{X}} = \arg \max_X p(\mathbf{X}|\mathbf{Z}) \stackrel{\text{règle de Bayes}}{=} \arg \max_X p(\mathbf{Z}|\mathbf{X}) \cdot p(\mathbf{X}) \quad (\text{A.10})$$

Connaissant les probabilités $p(\mathbf{Z}|\mathbf{X})$ et $p(\mathbf{X})$, le but est de maximiser la probabilité $p(\mathbf{X}|\mathbf{Z})$. Dès que la probabilité $p(\mathbf{X}|\mathbf{Z})$ atteint une valeur maximale, l'estimation $\hat{\mathbf{X}}$ correspond à la valeur la plus précise du paramètre à estimer en fonction de la précision des mesures.

– **Estimateur à minimum d'erreur en moyenne quadratique :**

$$\hat{\mathbf{X}} = \underset{\hat{\mathbf{X}}}{\operatorname{arg\,min}} \mathbb{E}[(\hat{\mathbf{X}} - \mathbf{X})^2 | \mathbf{Z}] \quad (\text{A.11})$$

Le principe de cet estimateur est de minimiser la variance sur l'estimation $\hat{\mathbf{X}}$. La valeur optimale de $\hat{\mathbf{X}}$ est obtenue lorsque la variance atteint sa valeur minimale. Cette solution est une moyenne conditionnelle de \mathbf{X} :

$$\hat{\mathbf{X}} = \mathbb{E}[\mathbf{X} | \mathbf{Z}] \triangleq \int_{-\infty}^{\infty} \mathbf{X} \cdot p(\mathbf{X} | \mathbf{Z}) \cdot dX \quad (\text{A.12})$$

Le filtre de Kalman, employé dans ces travaux, est basé sur cette dernière approche d'estimation.

A.1.3 Filtre de Kalman linéaire : phase d'estimation

Les probabilités $p(\mathbf{X})$ et $p(\mathbf{Z} | \mathbf{X})$ sont toujours supposées gaussiennes. Désormais, les mesures \mathbf{Z} sont modélisées par une fonction linéaire de \mathbf{X} corrompue par du bruit \mathbf{w} :

$$\mathbf{Z} = \mathbf{H} \cdot \mathbf{X} + \mathbf{w} \quad (\text{A.13})$$

\mathbf{w} est un bruit gaussien à moyenne nulle avec une matrice de covariance associée \mathbf{R} tel que :

$$p(\mathbf{w}) = \det(2\pi \cdot \mathbf{R})^{-\frac{1}{2}} \times \exp\left\{-\frac{1}{2} \mathbf{w}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{w}\right\} \quad (\text{A.14})$$

Maintenant, le paramètre à estimer possède n_X états et le vecteur d'observation n_Z mesures. La fonction de probabilité de la variable $p(\mathbf{Z} | \mathbf{X})$ associée à l'observation définie par l'équation (A.13) se formule :

$$p(\mathbf{Z} | \mathbf{X}) = \det(2\pi \cdot \mathbf{R})^{-\frac{1}{2}} \times \exp\left\{-\frac{1}{2} (\mathbf{Z} - \mathbf{H} \cdot \mathbf{X})^T \cdot \mathbf{R}^{-1} \cdot (\mathbf{Z} - \mathbf{H} \cdot \mathbf{X})\right\} \quad (\text{A.15})$$

La croyance a priori de \mathbf{X} est décrite par une loi gaussienne à plusieurs dimensions avec une moyenne \mathbf{X}_\ominus et une matrice de covariance \mathbf{P}_\ominus :

$$p(\mathbf{X}_\ominus) = \det(2\pi \cdot \mathbf{P}_\ominus)^{-\frac{1}{2}} \times \exp\left\{-\frac{1}{2} (\mathbf{X} - \mathbf{X}_\ominus)^T \cdot \mathbf{P}_\ominus^{-1} \cdot (\mathbf{X} - \mathbf{X}_\ominus)\right\} \quad (\text{A.16})$$

Le but est de calculer la probabilité a posteriori $p(\mathbf{X} | \mathbf{Z})$ à partir de la probabilité a priori $p(\mathbf{X}_\ominus)$ et de la probabilité conditionnelle $p(\mathbf{Z} | \mathbf{X})$. Pour obtenir $p(\mathbf{X} | \mathbf{Z})$, la loi de Bayes est appliquée tel que :

$$p(\mathbf{X} | \mathbf{Z}) = \frac{p(\mathbf{Z} | \mathbf{X}) \cdot p(\mathbf{X})}{p(\mathbf{Z})} \stackrel{\text{théorème de la probabilité totale}}{=} \frac{p(\mathbf{Z} | \mathbf{X}) \cdot p(\mathbf{X})}{\int_{-\infty}^{\infty} p(\mathbf{Z} | \mathbf{X}) \cdot p(\mathbf{X}) \cdot dX} \quad (\text{A.17})$$

En utilisant les expressions (A.15) et (A.16), l'équation (A.17) s'écrit :

$$p(\mathbf{X} | \mathbf{Z}) = \frac{\det(2\pi \cdot \mathbf{R})^{-\frac{1}{2}} \cdot \exp\left\{-\frac{1}{2} (\mathbf{Z} - \mathbf{H} \cdot \mathbf{X})^T \cdot \mathbf{R}^{-1} \cdot (\mathbf{Z} - \mathbf{H} \cdot \mathbf{X})\right\} \cdot \det(2\pi \cdot \mathbf{P}_\ominus)^{-\frac{1}{2}} \cdot \exp\left\{-\frac{1}{2} (\mathbf{X} - \mathbf{X}_\ominus)^T \cdot \mathbf{P}_\ominus^{-1} \cdot (\mathbf{X} - \mathbf{X}_\ominus)\right\}}{C(\mathbf{Z})} \quad (\text{A.18})$$

Pour simplifier l'écriture, la probabilité a posteriori $p(\mathbf{X}|\mathbf{Z})$ sera notée $p(\mathbf{X}_\oplus)$ et la matrice de covariance associée \mathbf{P}_\oplus . En ne tenant pas compte du facteur de normalisation, la probabilité $p(\mathbf{X}_\oplus)$ s'écrit donc :

$$p(\mathbf{X}_\oplus) = \det(2\pi \cdot \mathbf{P}_\oplus)^{-\frac{1}{2}} \cdot \exp\left\{-\frac{1}{2}(\mathbf{X} - \mathbf{X}_\oplus)^T \cdot \mathbf{P}_\oplus^{-1} \cdot (\mathbf{X} - \mathbf{X}_\oplus)\right\} \quad (\text{A.19})$$

avec les équations (A.18) et (A.19), l'égalité suivante est obtenue :

$$(\mathbf{X} - \mathbf{X}_\oplus)^T \cdot \mathbf{P}_\oplus^{-1} \cdot (\mathbf{X} - \mathbf{X}_\oplus) = (\mathbf{Z} - \mathbf{H} \cdot \mathbf{X})^T \cdot \mathbf{R}^{-1} \cdot (\mathbf{Z} - \mathbf{H} \cdot \mathbf{X}) + (\mathbf{X} - \mathbf{X}_\ominus)^T \cdot \mathbf{P}_\ominus^{-1} \cdot (\mathbf{X} - \mathbf{X}_\ominus) \quad (\text{A.20})$$

En développant l'égalité (A.20) et en procédant par équivalence, les équations de l'estimation, \mathbf{X}_\ominus , et de la matrice de covariance associée, \mathbf{P}_\ominus , permettant de définir la distribution de probabilité $p(\mathbf{X}_\oplus)$ s'écrivent :

$$\mathbf{X}_\ominus = (\mathbf{P}_\ominus^{-1} + \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H})^{-1} \cdot (\mathbf{P}_\ominus^{-1} \cdot \mathbf{X}_\ominus^{-1} + \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{Z}) \quad (\text{A.21})$$

$$\mathbf{P}_\oplus = (\mathbf{P}_\ominus^{-1} + \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H})^{-1} \quad (\text{A.22})$$

Un dernier développement¹ des équations (A.21) et (A.22) permet d'obtenir les équations finales du filtre de Kalman linéaire :

– l'estimation \mathbf{X}_\oplus du paramètre \mathbf{X} :

$$\mathbf{X}_\oplus = \mathbf{X}_\ominus + \mathbf{W} \cdot \nu \quad (\text{A.24})$$

– la matrice de covariance \mathbf{P}_\oplus associée à l'estimation \mathbf{X}_\oplus :

$$\mathbf{P}_\oplus = \mathbf{P}_\ominus - \mathbf{W} \cdot \mathbf{S} \cdot \mathbf{W}^T \quad (\text{A.25})$$

Les variables de calculs intermédiaires correspondant à \mathbf{W} , ν et \mathbf{S} , se nomment et s'écrivent :

– l'innovation ν , apportée par la mesure \mathbf{Z} :

$$\nu = \mathbf{Z} - \mathbf{H} \cdot \mathbf{X}_\ominus \quad (\text{A.26})$$

– la matrice de covariance \mathbf{S} associée à l'innovation ν :

$$\mathbf{S} = \mathbf{H} \cdot \mathbf{P}_\ominus \cdot \mathbf{H}^T + \mathbf{R} \quad (\text{A.27})$$

– le gain du filtre de Kalman linéaire \mathbf{W} :

$$\mathbf{W} = \mathbf{P}_\ominus \cdot \mathbf{H}^T \cdot \mathbf{S}^{-1} \quad (\text{A.28})$$

1.

$$(\mathbf{A} + \mathbf{B} \cdot \mathbf{C} \cdot \mathbf{B}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \cdot \mathbf{B} \cdot (\mathbf{C}^{-1} + \mathbf{B}^T \cdot \mathbf{A}^{-1} \cdot \mathbf{B})^{-1} \cdot \mathbf{B}^T \cdot \mathbf{A}^{-1} \quad (\text{A.23})$$

A.1.3.1 Remarque :

Le filtre de Kalman linéaire, à la vue des équations précédentes, peut être décrit simplement comme un filtre bayésien récursif qui minimise l'erreur moyenne au carré faite sur l'estimation. Cette forme récursive lui permet de fusionner les probabilités a priori estimées sur les paramètres, $p(\mathbf{X}_\ominus)$, et les mesures courantes, \mathbf{Z} , pour obtenir une estimation des deux moments définissant la probabilité a posteriori, $p(\mathbf{X}_\oplus)$, à chaque itération du filtre. C'est un des avantages de ce filtre de traiter les mesures à chaque fois qu'elles sont réalisées. De surcroît, la sauvegarde de l'historique des mesures n'est pas nécessaire car le calcul de l'estimation fusionne à chaque itération la mesure courante.

A.1.4 Filtre de Kalman linéaire : phase de prédiction

Dans le paragraphe précédent, les formules de l'étape estimation d'un filtre de Kalman linéaire ont été démontrées pour un système de paramètres qui n'évolue pas. Dans le cadre de notre étude par exemple, le véhicule se déplace dans l'environnement. Les paramètres de ce système évoluent donc dans le temps. Ils imposent une modélisation dynamique du système. Cette évolution est modélisée par des équations linéaires (ou non-linéaires) dites équations de transition d'état entre deux instants de temps, k et $k + 1$. Cette phase, également appelée **prédiction** est décrite par l'équation suivante :

$$\mathbf{X}(k + 1|k) = F.\mathbf{X}(k|k) + B.\mathbf{U}(k) + G.\mathbf{w}(k) \quad (\text{A.29})$$

Le terme $\mathbf{U}(k)$ représente le vecteur de commande appliqué au système à l'instant k , $\mathbf{w}(k)$ est un bruit gaussien à moyenne nulle correspondant à l'incertitude sur le modélisation de l'évolution et sur les commandes $\mathbf{U}(k)$. $\mathbf{X}(k + 1|k)$ correspond à la prédiction du paramètre \mathbf{X} à l'instant $k + 1$ connaissant les mesures jusqu'à l'instant k . En reprenant la définition (A.11), $\mathbf{X}(k + 1|k)$ s'écrit :

$$\hat{\mathbf{X}}(k + 1|k) = \mathbb{E}[\mathbf{X}(k + 1)|Z_k] \quad (\text{A.30})$$

La covariance associée à l'état prédit $\hat{\mathbf{X}}(k + 1|k)$ s'écrit :

$$\mathbf{P}(k + 1|k) = \mathbb{E}[(\mathbf{X}(k + 1) - \hat{\mathbf{X}}(k + 1|k))(\mathbf{X}(k + 1) - \hat{\mathbf{X}}(k + 1|k))^T|Z_k] \quad (\text{A.31})$$

La définition des moments (A.30) avec les équations de transition d'état (A.29) permettent d'écrire le calcul de la moyenne lors de la phase de prédiction :

$$\hat{\mathbf{X}}(k + 1|k) = \mathbb{E}[\mathbf{F}.\mathbf{X}(k) + \mathbf{B}.\mathbf{U}(k) + \mathbf{G}.\mathbf{w}(k)|Z_k] = \mathbf{F}.\hat{\mathbf{X}}(k|k) + \mathbf{B}.\mathbf{U}(k) \quad (\text{A.32})$$

La matrice de covariance associée à la prédiction, $\hat{\mathbf{X}}(k + 1|k)$, s'écrit en employant l'équation (A.31) :

$$\mathbf{P}(k + 1|k) = \mathbf{F}.\mathbf{P}(k|k).\mathbf{F}^T + \mathbf{G}.\mathbf{Q}_U.\mathbf{G}^T \quad (\text{A.33})$$

La matrice \mathbf{Q}_U est la matrice de covariance associée au bruit sur les commandes \mathbf{U} .

Cette dernière équation ponctue la présentation d'un filtre de Kalman linéaire. Une itération de ce filtre se compose donc de deux phases : une prédiction puis une estimation. Dans la suite de ce paragraphe, l'application de ce filtre sur un système non-linéaire va être introduite.

A.1.5 Filtre de Kalman non-linéaire

Fondamentalement, le filtre de Kalman estime les paramètres d'un système linéaire. Son application sur un système régi par des équations non-linéaires nécessite de réaliser une transformation de ces fonctions.

A.1.5.1 Approximation d'une fonction non-linéaire

La linéarisation d'une fonction non-linéaire \mathbf{f} repose sur le théorème de Taylor. En analyse, ce théorème permet "l'approximation d'une fonction plusieurs fois dérivable au voisinage d'un point par une fonction polynôme dont les coefficients dépendent uniquement des dérivées de la fonction en ce point". La formule de Taylor s'écrit :

$$\mathbf{f}(x) = \mathbf{f}(a) + \frac{\mathbf{f}'(a)}{1!} \cdot (x - a) + R(x) \quad (\text{A.34})$$

$R(x)$ correspond à un reste dépendant des variables x et a . En effet, plus la valeur de a est proche de x , plus la valeur de $R(x)$ est faible et inversement, plus la valeur de a est éloignée de celle de x , plus le résidu est grand.

Lorsque x est représenté par un vecteur de n paramètres, \mathbf{X} , l'extension en série de Taylor s'écrit :

$$\mathbf{f}(\mathbf{X}) = \mathbf{f}(\mathbf{X}_a) + \mathbf{F}_{\mathbf{X}_a} \cdot (\mathbf{X} - \mathbf{X}_a) + R(\mathbf{X}) \quad (\text{A.35})$$

Le terme $\mathbf{F}_{\mathbf{X}_a}$ correspond à la jacobienne de la fonction \mathbf{f} par rapport au vecteur \mathbf{X}_a et s'écrit :

$$\nabla \mathbf{F}_{\mathbf{X}_a} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad (\text{A.36})$$

Dans l'application d'un filtre de Kalman non-linéaire, l'approximation de Taylor est calculée par rapport à la valeur de l'estimation $\hat{\mathbf{X}}$. L'extension en séries de Taylor (A.35) s'écrit donc :

$$\mathbf{f}(\mathbf{X}) = \mathbf{f}(\hat{\mathbf{X}}) + \mathbf{F}_{\hat{\mathbf{X}}} \cdot (\mathbf{X} - \hat{\mathbf{X}}) \quad (\text{A.37})$$

La matrice $\mathbf{F}_{\hat{\mathbf{X}}}$ correspond à la jacobienne de la fonction non-linéaire f calculée par rapport à la valeur de l'estimation $\hat{\mathbf{X}}$. Le terme $(\mathbf{X} - \hat{\mathbf{X}})$ représente l'erreur commise sur les estimations. Il faut remarquer que le résidu $R(\mathbf{X})$ est négligé dans l'approximation des fonction non-linéaires. Il ne peut pas être calculé car il dépend du paramètre \mathbf{X} dont la valeur n'est pas connu par le système. Cette approximation joue un rôle principal dans les problèmes rencontrés avec l'application EKF SLAM. En effet, plus l'estimation $\hat{\mathbf{X}}$ aura une valeur éloignée de \mathbf{X} (cela signifie une erreur d'estimation importante), plus $R(\mathbf{X})$ sera grand et plus l'approximation sera importante.

A.1.5.2 Linéarisation de la transition des états

L'équation non-linéaire de transition d'états, \mathbf{f} , dans un filtre de Kalman non-linéaire s'écrit :

$$\mathbf{X}(k+1|k) = \mathbf{f}(\mathbf{X}(k|k), \mathbf{U}(k)) + \mathbf{w}(k) \quad (\text{A.38})$$

L'approximation de la fonction \mathbf{f} est réalisée autour de l'estimation $\mathbf{f}(k|k)$:

$$\mathbf{f}(k+1|k) \approx \mathbf{f}(\mathbf{f}(k|k), \mathbf{U}(k)) \quad (\text{A.39})$$

Le calcul de la matrice de covariance associée est obtenue par une transformation non-linéaire avec des jacobienues évaluées au point $\hat{\mathbf{X}}(k|k)$:

$$\mathbf{P}(k+1|k) \approx \mathbf{F}_X \cdot \mathbf{P}(k|k) \cdot \mathbf{F}_X^T + \mathbf{G}_v \cdot Q_U \cdot \mathbf{G}_v^T \quad (\text{A.40})$$

\mathbf{F}_X correspond à la matrice jacobienne de la fonction \mathbf{f} par rapport à l'estimation $\hat{\mathbf{X}}(k+1|k)$.

A.1.5.3 Linéarisation de la fonction d'observation

La fonction non-linéaire d'observation, \mathbf{h} , s'écrit :

$$\mathbf{Z}(k+1|k) = \mathbf{h}(\mathbf{X}(k+1|k)) + \mathbf{v}(k) \quad (\text{A.41})$$

L'approximation de la fonction \mathbf{h} est réalisée autour de l'estimation $\hat{\mathbf{X}}(k+1|k)$. La valeur de la prédiction de l'observation est obtenu à partir de la formule :

$$\hat{\mathbf{Z}}(k+1|k) \approx \mathbf{h}(\hat{\mathbf{X}}(k+1|k)) \quad (\text{A.42})$$

Le calcul de la covariance associée est calculé par une transformation non-linéaire avec des jacobienues évaluées au point $\hat{\mathbf{X}}(k+1|k)$:

$$\mathbf{P}_z(k+1|k) \approx \mathbf{H}_X \cdot \mathbf{P}(k+1|k) \cdot \mathbf{H}_X^T \quad (\text{A.43})$$

\mathbf{H}_X correspond à la matrice jacobienne de la fonction \mathbf{h} par rapport à l'estimation $\hat{\mathbf{X}}(k+1|k)$.

A.1.5.4 Formulation du filtre de Kalman Etendue

La linéarisation des fonctions non-linéaires réalisée, les équations du filtre de Kalman linéaire (A.24-A.28) sont appliquées au système. Dans un filtre EKF, l'estimation du vecteur d'état et de la matrice de covariance correspond à :

$$\hat{\mathbf{X}}(k+1 | k+1) = \hat{\mathbf{X}}(k+1|k) + \mathbf{W} \cdot v(k+1) \quad (\text{A.44})$$

$$\mathbf{P}(k+1 | k+1) = \mathbf{P}(k+1 | k) - \mathbf{W} \cdot \mathbf{S} \cdot \mathbf{W}^T \quad (\text{A.45})$$

$v(k+1)$ représente l'innovation et \mathbf{S} , la matrice de covariance associée à celle-ci :

$$v(k+1) = \mathbf{Z}(k+1) - \hat{\mathbf{Z}}(k+1|k) \quad (\text{A.46})$$

$$\mathbf{S} = \mathbf{H}_X \cdot \mathbf{P}(k+1 | k) \cdot \mathbf{H}_X^T + \mathbf{R} \quad (\text{A.47})$$

\mathbf{W} représente le gain calculé par le filtre de Kalman :

$$\mathbf{W} = \mathbf{P}(k+1 | k) \cdot \mathbf{H}_X^T \cdot \mathbf{S}^{-1} \quad (\text{A.48})$$

Annexe B

Modélisation mathématique du véhicule et des observations

Cette annexe présente successivement les équations non-linéaires employées lors des phases de prédiction et d'observation de l'approche EKF-SLAM. Dans un premier temps, le système d'équations modélisant le déplacement du véhicule à partir de données odométriques est présenté. Puis, les opérations spatiales, correspondant à des changements de repère dans un monde 2D, sont rappelées.

B.1 Modélisation du déplacement du véhicule

B.1.1 Système d'équations

L'évolution d'un véhicule dans notre système est régie par l'équation non-linéaire \mathbf{f}_v :

$$\mathbf{X}_V(k+1|k) = \mathbf{f}_V(\mathbf{X}_V(k|k), \mathbf{U}(k)) + \mathbf{w}_V(k) \quad (\text{B.1})$$

L'objectif de cette annexe est de définir les équations constituant la fonction non-linéaire \mathbf{f}_V mais également de définir le vecteur de commande représenté par le vecteur \mathbf{U} et les matrices jacobiennes associées aux différents calculs. Le terme \mathbf{w}_V représente un bruit gaussien sur le déplacement du véhicule. Dans notre cas d'étude, il englobe deux types d'erreur : les erreurs sur la modélisation du véhicule et les erreurs sur les données proprioceptives $\mathbf{U}(k)$.

La modélisation retenue dans ces travaux de recherche est celle du tricycle ou modèle d'Ackerman (cf. la figure B.1). La fonction non-linéaire \mathbf{f}_V s'écrit alors :

$$\begin{cases} x_v(k+1|k) = x_v(k|k) + \Delta t \cdot v(k) \cdot \cos(\theta_v(k|k)) \\ y_v(k+1|k) = y_v(k|k) + \Delta t \cdot v(k) \cdot \sin(\theta_v(k|k)) \\ \theta_v(k+1|k) = \theta_v(k|k) + \frac{\Delta t \cdot v(k) \cdot \tan(\varphi(k))}{L} \end{cases} \quad (\text{B.2})$$

avec :

- Δt , le temps en secondes entre les instants k et $k+1$,
- $v(k)$ et $\varphi(k)$, les deux données du vecteur de commande $\mathbf{U}(k) = [v(k), \varphi(k)]^T$,
- x_v et y_v , coordonnées cartésiennes du robot et θ_v , orientation du véhicule,

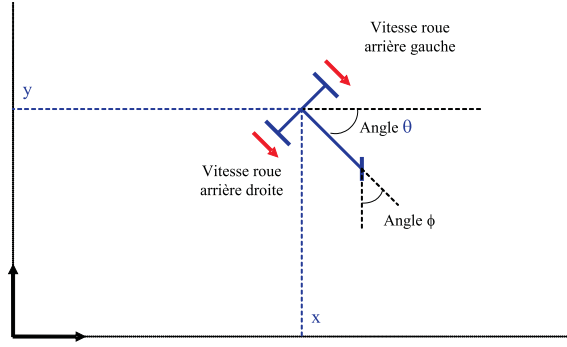


Figure B.1: *Modèle de type tricycle*

– $\mathbf{X}_V = [x_v, y_v, \theta_v]^T$, vecteur d'état du véhicule.

Dans nos applications, les données odométriques disponibles sont les vitesses des roues arrières ainsi que l'angle de braquage des roues avant. A partir de ces données deux modélisations sont envisageables, l'une employant les deux types de mesure et l'autre n'utilisant que les vitesses des roues arrières qui permettent d'estimer la valeur de l'angle de braquage des roues. Dans la suite de cette annexe, ces deux modélisations sont développées ainsi que le calcul de l'angle de braquage des roues avant à partir des vitesses des roues arrières.

B.1.2 Vecteur de commande avec l'angle de braquage des roues avant

Dans cette modélisation, le vecteur de commande de l'équation non-linéaire (B.1) s'écrit :

$$\mathbf{U} = \begin{bmatrix} v_d \\ v_g \\ \varphi \end{bmatrix} \quad (\text{B.3})$$

La vitesse du centre de l'essieu des roues arrières est obtenue en réalisant la moyenne entre les deux vitesses des roues :

$$v(k) = \frac{v_d(k) + v_g(k)}{2} \quad (\text{B.4})$$

Le calcul de la matrice de covariance associée à la prédiction de l'état $\mathbf{X}(k+1|k)$ (B.2) s'écrit :

$$\mathbf{P}_V(k+1|k) = \mathbf{F}_{X_V} \cdot \mathbf{P}_V(k|k) \cdot \mathbf{F}_{X_V}^T + \mathbf{F}_U \cdot \mathbf{Q}_U \cdot \mathbf{F}_U^T \quad (\text{B.5})$$

La jacobienne du processus, dérivée partielle de l'équation (B.2) par rapport au vecteur X_V , se calcule par :

$$\mathbf{P}_V = \frac{\partial \mathbf{f}_V(\mathbf{X}_V(k|k), U_k)}{\partial \mathbf{X}_V(k|k)} \quad (\text{B.6})$$

$$\mathbf{F}_{X_V} = \begin{bmatrix} 1 & 0 & -\frac{1}{2} \cdot \Delta t \cdot (v_d(k) + v_g(k)) \cdot \sin(\theta_v(k|k)) \\ 0 & 1 & \frac{1}{2} \cdot \Delta t \cdot (v_d(k) + v_g(k)) \cdot \cos(\theta_v(k|k)) \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.7})$$

La dérivée de la fonction (B.2) par rapport au vecteur $\mathbf{U}(k)$ (B.3) donne la jacobienne de commande :

$$\mathbf{F}_U = \frac{\partial f_v(\mathbf{X}_V(k|k), \mathbf{U}_k)}{\partial \mathbf{U}(k)} \quad (\text{B.8})$$

$$\mathbf{F}_U = \begin{bmatrix} \frac{\Delta t \cdot \cos(\theta)}{2} & \frac{\Delta t \cdot \cos(\theta)}{2} & 0 \\ \frac{\Delta t \cdot \sin(\theta)}{2} & \frac{\Delta t \cdot \cos(\theta)}{2} & 0 \\ \frac{\Delta t \cdot \tan(\varphi)}{2L} & \frac{\Delta t \cdot \tan(\varphi)}{2L} & \frac{\Delta t \cdot (v_d(k) + v_g(k))}{2 \cdot L \cdot \cos^2(\theta)} \end{bmatrix} \quad (\text{B.9})$$

B.1.3 Vecteur de commande uniquement composé des vitesses des roues arrières

Dans cette modélisation, le vecteur de commande de l'équation non-linéaire (B.1) s'écrit :

$$\mathbf{U} = \begin{bmatrix} v_d \\ v_g \end{bmatrix} \quad (\text{B.10})$$

Le calcul de la vitesse du centre de l'essieu des roues arrières, $v(k)$, est obtenue avec l'équation (B.4).

Obtention de l'angle de braquage des roues avant φ :

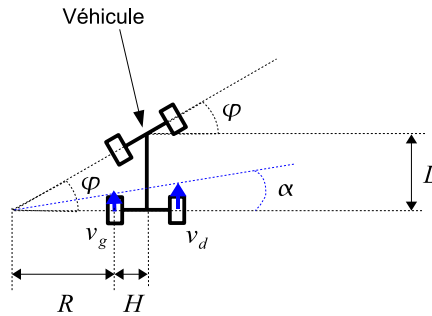


Figure B.2: Calcul de l'angle de braquage des roues avant à partir des vitesses des roues arrière du véhicule.

Le calcul de l'angle de braquage des roues avant est expliqué sur un exemple simple. La figure B.2 présente le véhicule lorsqu'il effectue un virage sur la gauche. En observant le schéma, il apparaît clairement que la tangente de l'angle de braquage des roues avant s'écrit :

$$\tan \varphi = \frac{L}{H+R} \quad (\text{B.11})$$

H et L correspondent respectivement à la longueur de l'essieu arrière du véhicule et à la distance entre les deux essieux du véhicule. R est le rayon de courbure de la trajectoire du véhicule. Sa valeur est obtenue à partir des vitesses des roues via le système suivant :

$$\begin{cases} v_g \cdot \Delta t = R \cdot \alpha \\ v_d \cdot \Delta t = (R + 2 \cdot H) \cdot \alpha \end{cases} \quad (\text{B.12})$$

En remplaçant la variable Δt calculé avec la première ligne dans la seconde, la valeur du rayon de courbure de la trajectoire s'écrit :

$$R = \frac{2.H.v_g}{v_d - v_g} \quad (\text{B.13})$$

En reprenant l'équation (B.11) avec la valeur de R (B.13), la tangente de l'angle de braquages des roues avant s'écrit :

$$\tan \varphi = \frac{L}{H + \frac{2.H.v_g}{v_d - v_g}} = \frac{L.(v_d - v_g)}{H.(v_d + v_g)} \quad (\text{B.14})$$

L'angle de braquage des roues avant du véhicule, $\varphi(k)$, en fonction des vitesses des deux roues arrières s'écrit donc :

$$\varphi(k) = \frac{L.(v_d(k) - v_g(k))}{H.(v_d(k) + v_g(k))} \quad (\text{B.15})$$

Modélisation du déplacement du véhicule :

En reprenant le système d'équation du modèle tricycle (B.2) avec les valeurs de la vitesse, v (B.4), et de l'angle de braquage, φ (B.15), le déplacement du véhicule, entre deux instants, en fonction des vitesses des roues arrières s'écrit :

$$\begin{cases} x_v(k+1|k) = x_v(k|k) + \frac{1}{2}.\Delta t.(v_d(k) + v_g(k)).\cos(\theta_v(k|k)) \\ y_v(k+1|k) = y_v(k|k) + \frac{1}{2}.\Delta t.(v_d(k) + v_g(k)).\sin(\theta_v(k|k)) \\ \theta_v(k+1|k) = \theta_v(k|k) + \frac{\Delta t.(v_d(k) - v_g(k))}{2.H} \end{cases} \quad (\text{B.16})$$

La jacobienne du processus, dérivée partielle de l'équation (B.2) par rapport au vecteur X_v , se calcule par :

$$\mathbf{F}_{X_v} = \frac{\partial \mathbf{f}_v(\mathbf{X}_v(k|k), \mathbf{U}_k)}{\partial \mathbf{X}_v(k|k)} \quad (\text{B.17})$$

$$\mathbf{F}_{X_v} = \begin{bmatrix} 1 & 0 & -\frac{1}{2}.\Delta t.(v_d(k) + v_g(k)).\sin(\theta_v(k|k)) \\ 0 & 1 & \frac{1}{2}.\Delta t.(v_d(k) + v_g(k)).\cos(\theta_v(k|k)) \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.18})$$

La dérivée de la fonction (B.2) par rapport au vecteur \mathbf{U} (B.10) nous donne la jacobienne de commande :

$$\mathbf{F}_U = \frac{\partial \mathbf{f}_v(\mathbf{X}_v(k|k), \mathbf{U}_k)}{\partial \mathbf{U}(k)} \quad (\text{B.19})$$

$$\mathbf{F}_U = \begin{bmatrix} \frac{\Delta t.\cos(\theta_v(k|k))}{2} & \frac{\Delta t.\cos(\theta_v(k|k))}{2} \\ \frac{\Delta t.\sin(\theta_v(k|k))}{2} & \frac{\Delta t.\sin(\theta_v(k|k))}{2} \\ \frac{\Delta t}{2H} & -\frac{\Delta t}{2H} \end{bmatrix} \quad (\text{B.20})$$

B.2 Opération de composition entre deux entités géométriques dans un plan 2D

Dans cette seconde partie, les relations spatiales entre les différentes entités géométriques dans un plan 2D sont détaillées. La description de ces opérateurs de changement de repère (entre la carte d'amers et le véhicule) permet de définir les fonctions d'observation de l'approche EKF-SLAM.

B.2.1 Opérateur « \oplus »

Soient deux relations spatiales $\mathbf{X}_{ij} = [x_{ij} \ y_{ij} \ \theta_{ij}]^T$ et $\mathbf{X}_{jk} = [x_{jk} \ y_{jk} \ \theta_{jk}]^T$, permettant de passer respectivement du repère i à j et du repère j à k . Le calcul de la relation résultante \mathbf{X}_{ik} est défini par :

$$\mathbf{X}_{ik} \triangleq \mathbf{X}_{ij} \oplus \mathbf{X}_{jk} = \begin{bmatrix} x_{jk} \cdot \cos(\theta_{ij}) - y_{jk} \cdot \sin(\theta_{ij}) + x_{ij} \\ x_{jk} \cdot \sin(\theta_{ij}) + y_{jk} \cdot \cos(\theta_{ij}) + y_{ij} \\ \theta_{ij} + \theta_{jk} \end{bmatrix} \quad (\text{B.21})$$

En approximant la fonction non-linéaire (B.21) par un développement en séries de Taylor au premier ordre, l'estimation de la moyenne $\hat{\mathbf{X}}_{ik}$ s'écrit :

$$\hat{\mathbf{X}}_{ik} \approx \hat{\mathbf{X}}_{ij} \oplus \hat{\mathbf{X}}_{jk} \quad (\text{B.22})$$

La matrice de covariance associée au vecteur $\hat{\mathbf{X}}_{ik}$ est obtenue avec :

$$\mathbf{C}_{X_{ik}} \approx \mathbf{J}_{\oplus} \cdot \begin{bmatrix} \mathbf{C}_{X_{ij}} & \mathbf{C}_{X_{ij}, X_{jk}} \\ \mathbf{C}_{X_{ij}, X_{jk}}^T & \mathbf{C}_{X_{jk}} \end{bmatrix} \cdot \mathbf{J}_{\oplus}^T \quad (\text{B.23})$$

$\mathbf{C}_{X_{ij}}$ et $\mathbf{C}_{X_{jk}}$ correspondent aux matrices de covariance associées respectivement aux vecteurs $\hat{\mathbf{X}}_{ij}$ et $\hat{\mathbf{X}}_{jk}$. La matrice de covariance croisée, $\mathbf{C}_{X_{ij}, X_{jk}}$, représente les corrélations existantes entre les deux vecteurs. La matrice \mathbf{J}_{\oplus} correspond à la matrice jacobienne de l'opération définie par l'équation (B.22). Elle se décompose en deux matrices jacobienes distinctes :

$$\mathbf{J}_{\oplus} \triangleq \frac{\partial \mathbf{X}_{ik}}{\partial (\mathbf{X}_{ij}, \mathbf{X}_{jk})} = [\mathbf{J}_{1\oplus} \quad \mathbf{J}_{2\oplus}] \quad (\text{B.24})$$

Les jacobienes $\mathbf{J}_{1\oplus}$ et $\mathbf{J}_{2\oplus}$ s'écrivent :

$$\mathbf{J}_{1\oplus} \triangleq \frac{\partial \mathbf{X}_{ik}}{\partial (\mathbf{X}_{ij})} = \begin{bmatrix} 1 & 0 & -x_{jk} \cdot \sin(\theta_{ij}) - y_{jk} \cdot \cos(\theta_{ij}) \\ 0 & 0 & x_{jk} \cdot \cos(\theta_{ij}) - y_{jk} \cdot \sin(\theta_{ij}) \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.25})$$

$$\mathbf{J}_{2\oplus} \triangleq \frac{\partial \mathbf{X}_{ik}}{\partial \mathbf{X}_{jk}} = \begin{bmatrix} \cos(\theta_{ij}) & -\sin(\theta_{ij}) & 0 \\ \sin(\theta_{ij}) & \cos(\theta_{ij}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.26})$$

B.2.2 Opérateur « \ominus »

Etant donnée une relation spatiale $\mathbf{X}_{ij} = [x_{ij} \ y_{ij} \ \theta_{ij}]^T$, le calcul de la relation inverse \mathbf{X}_{ji} est défini par :

$$\mathbf{X}_{ji} \triangleq \ominus \mathbf{X}_{ij} = \begin{bmatrix} -x_{ij} \cdot \cos(\theta_{ij}) - y_{ij} \cdot \sin(\theta_{ij}) \\ x_{ij} \cdot \sin(\theta_{ij}) - y_{ij} \cdot \cos(\theta_{ij}) \\ -\theta_{ij} \end{bmatrix} \quad (\text{B.27})$$

En approximant la fonction non-linéaire (B.27) par un développement des séries de Taylor au premier ordre, l'estimation de la moyenne $\hat{\mathbf{X}}_{ji}$ s'écrit :

$$\hat{\mathbf{X}}_{ji} \approx \ominus \hat{\mathbf{X}}_{ij} \quad (\text{B.28})$$

La matrice de covariance associée au vecteur $\hat{\mathbf{X}}_{ji}$ est obtenue avec :

$$\mathbf{C}_{X_{ji}} \approx \mathbf{J}_{\ominus} \cdot [\mathbf{C}_{X_{ij}}] \cdot \mathbf{J}_{\ominus}^T \quad (\text{B.29})$$

La matrice $\mathbf{C}_{X_{ij}}$ correspond à la matrices de covariance associée au vecteur $\hat{\mathbf{X}}_{ij}$. La matrice \mathbf{J}_{\ominus} correspond à la matrice jacobienne de l'opération définie par l'équation (B.28) et s'écrit :

$$\mathbf{J}_{\ominus} \triangleq \frac{\partial \mathbf{X}_{ji}}{\partial \mathbf{X}_{ij}} = \begin{bmatrix} -\cos(\theta_{ij}) & -\sin(\theta_{ij}) & x_{ij} \cdot \sin(\theta_{ij}) - y_{ij} \cdot \cos(\theta_{ij}) \\ \sin(\theta_{ij}) & -\cos(\theta_{ij}) & x_{ij} \cdot \cos(\theta_{ij}) + y_{ij} \cdot \sin(\theta_{ij}) \\ 0 & 0 & -1 \end{bmatrix} \quad (\text{B.30})$$

Avec les deux opérations présentées précédemment, la résultante de n'importe quelle séquence de relations peut être calculée. Il faut noter que ces opérations sont associatives mais pas commutatives. Par exemple, il faut écrire :

$$\mathbf{X}_{ij} \ominus \mathbf{X}_{kj} \triangleq \mathbf{X}_{ij} \oplus (\ominus \mathbf{X}_{kj}) \quad (\text{B.31})$$

Annexe C

Description des méthodes d'extraction des amers ligne et cercle

C.1 Introduction

Cette annexe décrit les méthodes permettant d'extraire et de paramétrer les amers de type droite et cercle d'un balayage télémétrique. Sont ainsi décrits les différents calculs permettant d'obtenir les paramètres de ces amers ainsi que les matrices de covariance associées. Sont présentées successivement les méthodes pour une droite puis pour un cercle. Certaines de ces descriptions sont accompagnées de résultats de simulation permettant d'appuyer le choix de la méthode d'extraction.

C.2 Paramétrisation d'une droite

Une droite est représentée par un repère défini par trois états : x_L , y_L et θ_L , respectivement les coordonnées cartésiennes et son orientation dans le repère de la carte estimée (cf. la figure C.1).

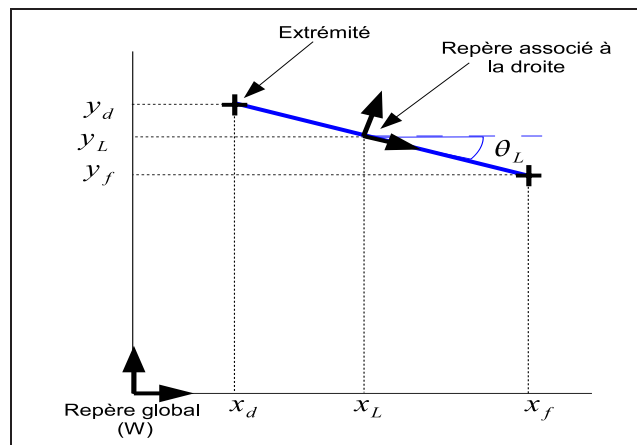


Figure C.1: Représentation cartésienne d'une droite avec 3 états : $X_L = [y_L, y_L, \theta_L]$.

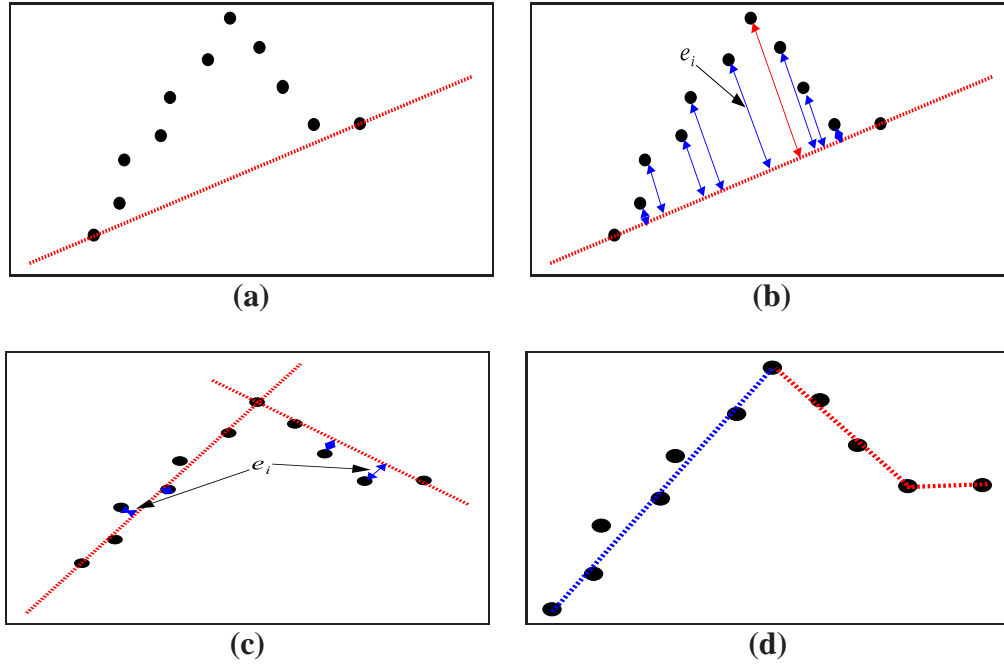


Figure C.2: Description de l'algorithme de découpage récursif pour extraire les droites.

A partir d'un balayage télémétrique, l'objectif est d'extraire une droite mais également d'en déterminer les trois paramètres. L'algorithme employé dans ce but se décompose en quatre étapes distinctes :

- extraction des points appartenant à la même droite ;
- détermination des trois paramètres de la droite avec la matrice de covariance associée ;
- mise en correspondance des droites extraites, détection et fusion des droites identiques ;
- détermination des coordonnées extrémités du segment portée par la droite.

Chacune de ces étapes est décrite dans la suite de ce document.

C.2.1 Algorithme de découpage récursif

La figure C.2 présente une application de l'algorithme de découpage récursif à un ensemble d'impacts extrait lors de la phase de segmentation. Dans un premier temps, une droite est tracée entre le premier et le dernier impact laser de l'ensemble, respectivement \mathbf{X}_{debut} et \mathbf{X}_{fin} (cf. la figure C.2(a)). Les paramètres de cette droite sont déterminés en coordonnées polaires avec les équations suivantes :

$$\varphi = \tan^{-1} \left(\frac{\left(\frac{\rho_{debut}}{\rho_{fin}} \right) \cdot \cos(\beta_{debut}) - \cos(\beta_{fin})}{\sin(\beta_{fin}) - \left(\frac{\rho_{debut}}{\rho_{fin}} \right) \cdot \sin(\beta_{debut})} \right) \quad (C.1)$$

$$d = \rho_{fin} \cdot \cos(\varphi - \beta_{debut}) \quad (C.2)$$

A l'aide de ces paramètres, toutes les distances e_i des impacts laser X_i par rapport à la droite sont calculées (cf. la figure C.2 (b)) :

$$e_i = |\rho_i \cdot \cos(\varphi - \beta_i) - d| \quad (C.3)$$

La plus grande distance e_i notée e_{max} (en rouge sur la figure C.2**(b)**) est comparée à un seuil métrique (10 centimètres correspondant à la précision du télémètre laser). Si la distance e_{max} est inférieure au seuil, tous les points sont supposés appartenir à la même droite. Sinon, l'ensemble est découpé en deux à partir de l'impact correspondant à la distance e_{max} . Les mêmes traitements sont recommencés de part et d'autre de la position de ce point (cf. la figure C.2**(c)**). Au final, plusieurs segments sont extraits de l'ensemble d'impacts laser traité (cf. la figure C.2**(d)**). Dans une dernière étape, seuls les segments étant constitués d'un minimum de point sont retenus. Dans l'exemple présenté, seul le segment tracé en bleu est retenu.

C.2.2 Estimation des paramètres de la droite

L'objectif est d'estimer les paramètres de la droite passant au mieux par les points extraits par l'algorithme de découpage, ainsi que l'incertitude réalisée sur la détermination de ces paramètres. Il existe deux principales méthodes d'estimation :

- une minimisation au sens des moindres carrés [5][85][127],
- une régression linéaire [51][123].

Dans mes travaux, une méthode de minimisation des moindres carrés [102] est utilisée. L'estimation des paramètres d'une droite est basée sur une modélisation cartésienne :

$$a.x + b.y + 1 = 0 \quad (\text{C.4})$$

Les paramètres a et b de l'équation (C.4) sont obtenus par les équations :

$$a = \frac{\sum_{i=1}^n y_i \cdot \sum_{i=1}^n (x_i y_i) - \sum_{i=1}^n x_i \cdot \sum_{i=1}^n y_i^2}{\sum_{i=1}^n x_i^2 \cdot \sum_{i=1}^n x_i^2 - \sum_{i=1}^n (x_i y_i)^2} \quad (\text{C.5})$$

$$b = \frac{\sum_{i=1}^n x_i \cdot \sum_{i=1}^n (x_i y_i) - \sum_{i=1}^n y_i \cdot \sum_{i=1}^n x_i^2}{\sum_{i=1}^n x_i^2 \cdot \sum_{i=1}^n x_i^2 - \sum_{i=1}^n (x_i y_i)^2} \quad (\text{C.6})$$

Les variances associées à ces deux paramètres, σ_a^2 et σ_b^2 , sont obtenues par les calculs :

$$\sigma_a^2 = \sum_{i=1}^n \left(\sigma_{x_i}^2 \cdot \left(\frac{\partial a}{\partial x_i} \right)^2 \right) + \sum_{i=1}^n \left(\sigma_{y_i}^2 \cdot \left(\frac{\partial a}{\partial y_i} \right)^2 \right) \quad (\text{C.7})$$

$$\sigma_b^2 = \sum_{i=1}^n \left(\sigma_{x_i}^2 \cdot \left(\frac{\partial b}{\partial x_i} \right)^2 \right) + \sum_{i=1}^n \left(\sigma_{y_i}^2 \cdot \left(\frac{\partial b}{\partial y_i} \right)^2 \right) \quad (\text{C.8})$$

Les covariances entre les paramètres a et b sont négligées. Les matrices jacobiniennes des calculs des variances, (C.7) et (C.8), sont données par :

$$\frac{\partial a}{\partial x_i} = \frac{y_i \cdot \sum_{j=1}^n (y_j) - \sum_{j=1}^n (y_j^2)}{4 \cdot x_i \cdot \sum_{j=1}^n (x_j^2) - 2 \cdot x_i \cdot y_i^2} \quad (\text{C.9})$$

$$\frac{\partial a}{\partial y_i} = \frac{\sum_{j=1}^n (x_j \cdot y_j) + x_i \cdot \sum_{j=1}^n (y_j) - 2 \cdot y_i \cdot \sum_{j=1}^n (x_j)}{-2 \cdot y_i \cdot x_i^2} \quad (\text{C.10})$$

$$\frac{\partial b}{\partial x_i} = \frac{\sum_{j=1}^n (x_j \cdot y_j) + y_i \cdot \sum_{j=1}^n (x_j) - 2 \cdot x_i \cdot \sum_{j=1}^n (y_j)}{4 \cdot x_i \cdot \sum_{j=1}^n (x_j^2) - 2 \cdot x_i \cdot y_i^2} \quad (\text{C.11})$$

$$\frac{\partial b}{\partial y_i} = \frac{x_i \cdot \sum_{j=1}^n (x_j) - \sum_{j=1}^n (x_j^2)}{-2 \cdot y_i \cdot x_i^2} \quad (\text{C.12})$$

Dès lors que les paramètres de la droite couplés avec leurs incertitudes respectives sont déterminés, il est possible de déterminer l'orientation de la droite θ_L telle que :

$$\theta_L = \begin{cases} \tan^{-1}\left(\frac{b}{a}\right) & \text{si } a \neq 0 \\ 0 & \text{si } a = 0 \end{cases} \quad (\text{C.13})$$

avec comme incertitude associée σ_{θ_L} à ce paramètre :

$$\sigma_{\theta_L}^2 = \frac{a^2 \cdot \sigma_a^2 + b^2 \cdot \sigma_b^2}{(a^2 + b^2)^2} \quad (\text{C.14})$$

Les deux derniers paramètres de la droite, x_L et y_L , correspondent aux moyennes des abscisses et des ordonnées respectivement de tous les impacts composant le segment porté par la droite :

$$x_L = \frac{\sum_{i=1}^n (x_i)}{n} \quad \text{avec :} \quad \sigma_{x_L}^2 = \frac{\sum_{i=1}^n (\sigma_{x_i}^2)}{n^2} \quad (\text{C.15})$$

$$y_L = \frac{\sum_{i=1}^n (y_i)}{n} \quad \text{avec :} \quad \sigma_{y_L}^2 = \frac{\sum_{i=1}^n (\sigma_{y_i}^2)}{n^2} \quad (\text{C.16})$$

C.2.3 Fusion des droites identiques

Dans certains cas d'études comme sur la figure C.3, des segments extraits du même balayage télémétrique doivent être fusionnés car ils appartiennent à la même droite. Cette mesure représente, par exemple, un mur avec un objet devant celui-ci (cf. la figure C.3(a)). A cause de la présence de cette occlusion, deux droites différentes sont extraites (cf. la figure C.3(b)). Pour résoudre ce problème, un test statistique est réalisé sur toutes les droites extraites de la

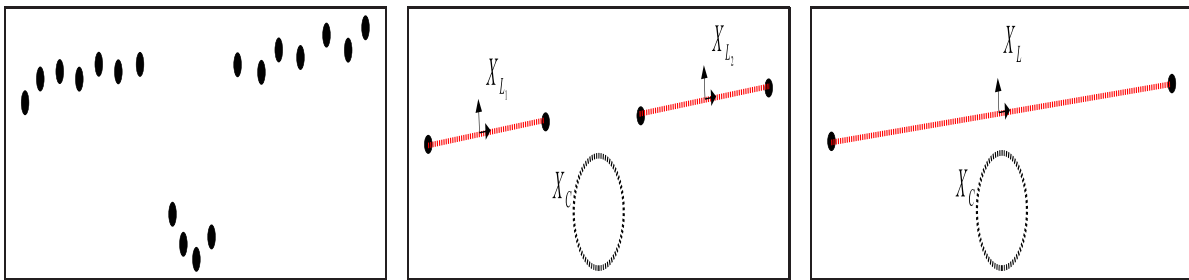


Figure C.3: Fusion de deux droites dans le même balayage télémétrique : (a) mesure avec une occlusion, (b) extraction des droites et (c) fusion des deux droites extraites.

même mesure. Si deux droites sont statistiquement associées (cf. la figure C.3(a)), les deux états de cette droite seront fusionnés ainsi que leurs matrices de covariance. L'état \mathbf{X}_L (cf. la figure C.3(c)) est le résultat de la fusion entre les deux droites \mathbf{X}_{L_1} et \mathbf{X}_{L_2} :

$$\mathbf{X}_L = \mathbf{P}_L \cdot ((\mathbf{P}_{L_1})^{-1} \cdot \mathbf{X}_{L_1} + (\mathbf{P}_{L_2})^{-1} \cdot \mathbf{X}_{L_2}) \quad (\text{C.17})$$

avec comme matrice de covariance associée au vecteur de la droite :

$$\mathbf{P}_L = ((\mathbf{P}_{L_1})^{-1} + (\mathbf{P}_{L_2})^{-1})^{-1} \quad (\text{C.18})$$

C.2.4 Gestion des extrémités des segments

Dans l'approche proposée, les extrémités des segments représentées par les points \mathbf{X}_{debut} et \mathbf{X}_{fin} (cf. la figure C.1) sont utilisées pour l'affichage des segments de la carte et sont sauvegardées dans un processus annexe. Elles ne sont pas estimées dans le vecteur d'état du système car leurs mises à jour peuvent entraîner de fortes variations sur l'estimation de la carte de l'environnement. Les positions de ces extrémités sont mises à jour en employant une moyenne pondérée (équations (C.17) et (C.18)) de leurs mesures (impacts situés de part et d'autre du segment extrait).

C.2.5 Discussions sur la méthode de détection des droites

L'approche présentée ne traite pas de tous les problèmes envisageables dans une application réelle. Par exemple, aucune différence n'est faite entre un objet devant un mur (cf. la figure C.3) et une entrée comme par exemple un porche. Pour traiter ce type de problème, des améliorations peuvent être apportées aux étapes de détection des droites et de gestion des extrémités des segments (cf. les paragraphes C.2.3 et C.2.4). Premièrement, la détection des droites doit être plus intelligente. Si deux segments extraits d'un même balayage télémétrique appartiennent à la même droite, l'algorithme doit être capable de déterminer si c'est dû à la présence d'une occlusion ou non. Deuxièmement, la différenciation de ces deux cas d'études soulève un autre problème constaté dans nos applications. Si deux droites cartographiées correspondent au même pan de mur par exemple, plusieurs états vont représenter la même entité. Cette redondance d'information entraînera des conflits lors de la phase d'association de données. Chaque mesure s'associant avec une des droites sera forcément associée avec l'autre. Pour répondre à cette problématique, une carte contenant uniquement les extrémités des segments peut être mise en place. Elle gèrera les fusions des différents segments appartenant à une même droite.

C.3 Paramétrisation d'un amer cercle

La détermination des paramètres du cercle entraîne l'utilisation de plusieurs algorithmes dans le but d'obtenir les meilleures estimations possibles des paramètres du cercle ainsi que de la matrice de covariance associée.

Un amer de type cercle est décrit par trois paramètres (cf. la figure C.4) tel que :

$$\mathbf{X}_{ce} = \begin{bmatrix} x_{ce} \\ y_{ce} \\ r_{ce} \end{bmatrix} \quad (\text{C.19})$$

Sous une forme générale, le calcul des paramètres d'un cercle peut se résumer par la fonction :

$$\mathbf{X}_{ce} = \mathbf{f}(\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_i, \dots, \mathbf{X}_n) \quad (\text{C.20})$$

Les vecteurs \mathbf{X}_i correspondent aux coordonnées des impacts laser mesurés et la fonction \mathbf{f} à la fonction non-linéaire utilisée pour obtenir les paramètres du cercle \mathbf{X}_{ce} . Dans les applications d'estimation telle que le filtre de Kalman, il faut également déterminer une matrice de covariance $\mathbf{C}_{X_{ce}}$ associée au vecteur \mathbf{X}_{ce} à partir des incertitudes \mathbf{R}_i associées à chaque impacts laser.

Pour extraire ces trois paramètres, trois algorithmes différents ont été comparés : la méthode de Guivant [8], l'algorithme de Gauss-Newton [69] et celui de Levenberg-Marquardt [155]. Avant de présenter les résultats des simulations, ces trois approches sont présentées.

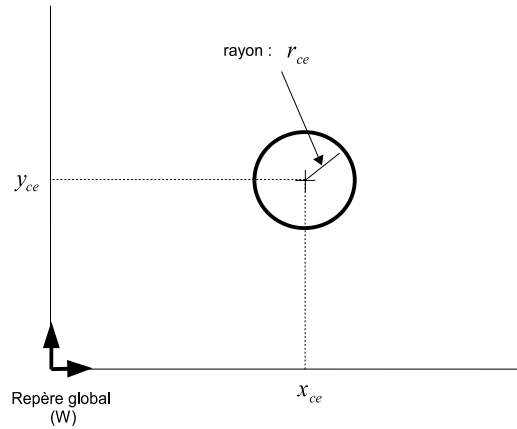


Figure C.4: Représentation d'un amer de type cercle.

C.3.1 Méthode de Guivant

Soit n le nombre de points de mesure considérés. Le calcul des paramètres d'un cercle par la méthode de Guivant [8] est donné par les équations :

$$\mathbf{X}_{ce} = \begin{bmatrix} x_{ce} \\ y_{ce} \\ r_{ce} \end{bmatrix} = \begin{bmatrix} (\rho_{min} + r_{ce}) \times \cos(\beta_{min}) \\ (\rho_{min} + r_{ce}) \times \sin(\beta_{min}) \\ \frac{1}{2n} \times \sum_{i=1}^n \rho_i \times \sum_{i=1}^{n-1} (\beta_{i+1} - \beta_i) \end{bmatrix} \quad (\text{C.21})$$

Le point $\mathbf{X}_{min} = [\rho_{min} \ \beta_{min}]^T$ est déterminé en prenant le plus petit ρ dans l'ensemble des n points de mesure (cf. la figure C.5).

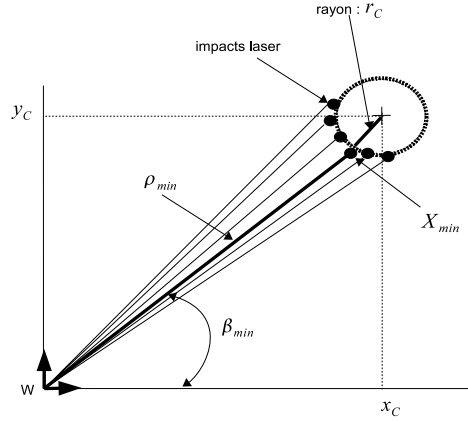


Figure C.5: Méthode de Guivant pour déterminer les paramètres d'un cercle à partir de mesures télémétriques.

La matrice de covariance associée au vecteur \mathbf{X}_{ce} s'écrit :

$$\mathbf{C}_{X_{ce}} = \begin{bmatrix} \sigma_{x_{ce}}^2 & \sigma_{x_{ce}y_{ce}} & \sigma_{x_{ce}r_{ce}} \\ \sigma_{x_{ce}y_{ce}} & \sigma_{y_{ce}}^2 & \sigma_{y_{ce}r_{ce}} \\ \sigma_{x_{ce}r_{ce}} & \sigma_{y_{ce}r_{ce}} & \sigma_{r_{ce}}^2 \end{bmatrix} \quad (\text{C.22})$$

Les calculs associés aux différentes variances et covariances composant la matrice $\mathbf{C}_{X_{ce}}$ sont décrits un par un dans la suite de cette annexe.

C.3.1.1 Calcul des variances

Variance sur le paramètre x_{ce} :

$$\sigma_{x_{ce}}^2 = \sum_{i=1}^n \left(\left(\frac{\partial x_{ce}}{\partial \rho_i} \right)^2 \times \sigma_{\rho}^2 \right) + \sum_{i=1}^{n-1} \left(\left(\frac{\partial x_{ce}}{\partial \Delta \beta_i} \right)^2 \times \sigma_{\Delta \beta}^2 \right) \quad (\text{C.23})$$

$$\sigma_{x_{ce}}^2 = (\sigma_{\rho_{min}} \times \cos(\beta_{min}))^2 + \sum_{i=1}^n \left(\left(\frac{\partial r_{ce}}{\partial \rho_i} \right) \times \sigma_{\rho_i} \times \cos(\beta_{min}) \right)^2 + \dots \\ \sum_{i=1}^{n-1} \left(\left(\frac{\partial r_{ce}}{\partial \Delta \beta} \right) \times \sigma_{\Delta \beta} \times \cos(\beta_{min}) \right)^2 + ((\rho_{min} + r_{ce}) \times \sin(\beta_{min}) \times \sigma_{\beta_{min}})^2 \quad (\text{C.24})$$

Variance sur le paramètre y_{ce} :

$$\sigma_{y_{ce}}^2 = \sum_{i=1}^n \left(\left(\frac{\partial y_{ce}}{\partial \rho_i} \right)^2 \times \sigma_{\rho}^2 \right) + \sum_{i=1}^{n-1} \left(\left(\frac{\partial y_{ce}}{\partial \Delta \beta_i} \right)^2 \times \sigma_{\Delta \beta}^2 \right) \quad (\text{C.25})$$

$$\sigma_{y_{ce}}^2 = (\sigma_{\rho_{min}} \times \sin(\beta_{min}))^2 + \sum_{i=1}^n \left(\left(\frac{\partial r_{ce}}{\partial \rho_i} \right) \times \sigma_{\rho_i} \times \sin(\beta_{min}) \right)^2 + \dots \\ \sum_{i=1}^{n-1} \left(\left(\frac{\partial r_{ce}}{\partial \Delta \beta} \right) \times \sigma_{\Delta \beta} \times \sin(\beta_{min}) \right)^2 + ((\rho_{min} + r_{ce}) \times \cos(\beta_{min}) \times \sigma_{\beta_{min}})^2 \quad (\text{C.26})$$

Variance sur le paramètre r_{ce} :

$$\sigma_{r_{ce}}^2 = \sum_{i=1}^n \left(\left(\frac{\partial r_{ce}}{\partial \rho_i} \right)^2 \times \sigma_{\rho}^2 \right) + \sum_{i=1}^{n-1} \left(\left(\frac{\partial r_{ce}}{\partial \Delta \beta_i} \right)^2 \times \sigma_{\Delta \beta}^2 \right) \quad (\text{C.27})$$

$$\sigma_{r_{ce}}^2 = \sum_{i=1}^n \left(\frac{\sum_{i=1}^{n-1} \Delta \beta}{2n} \times \sigma_{\rho_i} \right)^2 + \sum_{i=1}^{n-1} \left(\frac{\sum_{i=1}^n \rho_{-i}}{2n} \times \sigma_{\Delta \beta} \right)^2 \quad (\text{C.28})$$

C.3.1.2 Calcul des covariances

Covariance entre les paramètres x_{ce} et y_{ce} :

$$\sigma_{x_{ce}y_{ce}} = \sum_{i=1}^n \left(\left(\frac{\partial y_{ce}}{\partial \rho_i} \right) \times \sigma_{\rho_i}^2 \times \left(\frac{\partial x_{ce}}{\partial \rho_i} \right) \right) + \sum_{i=1}^{n-1} \left(\left(\frac{\partial y_{ce}}{\partial \Delta\beta} \right) \times \sigma_{\Delta\beta}^2 \times \left(\frac{\partial x_{ce}}{\partial \Delta\beta} \right) \right) \quad (C.29)$$

$$\begin{aligned} \sigma_{x_{ce}y_{ce}} = & \left(\left(1 + \frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \right) \times \cos(\beta_{min}) \times \sin(\beta_{min}) \times \sigma_{\rho_{min}}^2 \right) + \dots \\ & \sum_{i=1}^n \left(\left(\frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \right)^2 \times \cos(\beta_{min}) \times \sin(\beta_{min}) \times \sigma_{\rho_i}^2 \right) - \dots \\ & \left((\rho_{min} + r_{ce})^2 \times \cos(\beta_{min}) \times \sin(\beta_{min}) \times \sigma_{\beta_{min}}^2 \right) + \dots \\ & \sum_{i=1}^{n-1} \left(\left(\frac{\sum_{i=1}^n \rho_i}{2n} \right)^2 \times \cos(\beta_{min}) \times \sin(\beta_{min}) \times \sigma_{\Delta\beta}^2 \right) \end{aligned} \quad (C.30)$$

Covariance entre les paramètres x_{ce} et r_{ce} :

$$\sigma_{x_{ce}r_{ce}} = \sum_{i=1}^n \left(\left(\frac{\partial r_{ce}}{\partial \rho_i} \right) \times \sigma_{\rho}^2 \times \left(\frac{\partial x_{ce}}{\partial \rho_i} \right) \right) + \sum_{i=1}^{n-1} \left(\left(\frac{\partial r_{ce}}{\partial \Delta\beta} \right) \times \sigma_{\Delta\beta}^2 \times \left(\frac{\partial x_{ce}}{\partial \Delta\beta} \right) \right) \quad (C.31)$$

$$\begin{aligned} \sigma_{x_{ce}r_{ce}} = & \left(\frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \times \left(1 + \frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \right) \times \cos(\beta_{min}) \times \sigma_{\rho_{min}}^2 \right) + \dots \\ & \sum_{i=1}^n \left(\left(\frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \right)^2 \times \cos(\beta_{min}) \times \sigma_{\rho_i}^2 \right) + \dots \\ & \sum_{i=1}^{n-1} \left(\left(\frac{\sum_{i=1}^n \rho_i}{2n} \right)^2 \times \cos(\beta_{min}) \times \sigma_{\Delta\beta}^2 \right) \end{aligned} \quad (C.32)$$

Covariance entre les paramètres y_{ce} et r_{ce} :

$$\sigma_{y_{ce}r_{ce}} = \sum_{i=1}^n \left(\left(\frac{\partial r_{ce}}{\partial \rho_i} \right) \cdot \sigma_{\rho}^2 \cdot \left(\frac{\partial y_{ce}}{\partial \rho_i} \right) \right) + \sum_{i=1}^{n-1} \left(\left(\frac{\partial r_{ce}}{\partial \Delta\beta} \right) \cdot \sigma_{\Delta\beta}^2 \cdot \left(\frac{\partial y_{ce}}{\partial \Delta\beta} \right) \right) \quad (C.33)$$

$$\begin{aligned} \sigma_{y_{ce}r_{ce}} = & \left(\frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \cdot \left(1 + \frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \right) \cdot \sin(\beta_{min}) \cdot \sigma_{\rho_{min}}^2 \right) + \dots \\ & \sum_{i=1}^n \left(\left(\frac{\sum_{i=1}^{n-1} \Delta\beta}{2n} \right)^2 \cdot \sin(\beta_{min}) \cdot \sigma_{\rho_i}^2 \right) + \dots \\ & \sum_{i=1}^{n-1} \left(\left(\frac{\sum_{i=1}^n \rho_i}{2n} \right)^2 \cdot \sin(\beta_{min}) \cdot \sigma_{\Delta\beta}^2 \right) \end{aligned} \quad (C.34)$$

C.3.2 Méthode de Gauss-Newton : Formulation du critère à minimiser

Le calcul des paramètres du cercle s'appuie sur les équations paramétriques [69] :

$$\begin{aligned} x_i &= x_{ce} + (r_{ce} \cdot \cos(\varphi_i)) \\ y_i &= y_{ce} + (r_{ce} \cdot \sin(\varphi_i)) \end{aligned} \quad (C.35)$$

Les coordonnées cartésiennes $\mathbf{X}_i = [x_i, y_i]$ correspondent aux coordonnées des impacts mesurés et φ_i , les angles de chaque point, \mathbf{X}_i , référencé dans le cercle. Avec les données du problème, l'équation précédente s'écrit sous la forme :

$$\mathbf{X}_i = \mathbf{h}(\mathbf{X}_{ce}(k), \varphi(i)) + \mathbf{w}_i \quad (C.36)$$

où $\mathbf{w}(i)$ correspond à l'erreur sur la mesure, considérée gaussienne et à moyenne nulle, et $\mathbf{X}_{ce}(k)$, à l'estimation des paramètres du cercle à l'instant k . La fonction non-linéaire \mathbf{h} correspond aux équations (C.35). Les paramètres du cercle avec les angles φ_i , associés à chaque point de mesure, sont estimés. Le vecteur d'état est donc :

$$\mathbf{X}_p = [\varphi_1 \ \dots \ \varphi_i \ \dots \ \varphi_n \ x_{ce} \ y_{ce} \ r_{ce}]^T \quad (C.37)$$

Les paramètres recherchés correspondent aux trois derniers états du vecteur \mathbf{X}_p . Les paramètres φ_i sont estimés pour assurer une meilleure convergence des résultats. Toutes les mesures \mathbf{X}_i , avec une matrice de covariance associée \mathbf{R}_i , sont mises dans un vecteur \mathbf{X}_{mesure} et la matrice de covariance associée au bruit, \mathbf{R} , est diagonale et définie par :

$$\mathbf{R} = \begin{bmatrix} \ddots & & 0 \\ & \mathbf{R}_i & \\ 0 & & \ddots \end{bmatrix} \quad (\text{C.38})$$

En utilisant la formule de Newton, les paramètres du cercle sont calculés avec :

$$\mathbf{X}_p(k+1) = \mathbf{X}_p(k) + (\mathbf{J}(k)^T \cdot \mathbf{R}^{-1} \cdot \mathbf{J}(k))^{-1} \cdot \mathbf{J}(k)^T \cdot \mathbf{R}^{-1} \cdot [\mathbf{X}_{mesure} - \mathbf{h}(\mathbf{X}_p(k))] \quad (\text{C.39})$$

$\mathbf{J}(k)$ et $\mathbf{R}(k)$ correspondent respectivement à la matrice jacobienne de la fonction non-linéaire \mathbf{h} et à la matrice associée au bruit \mathbf{w}_i . La jacobienne s'écrit :

$$\mathbf{J}(k) = \begin{bmatrix} -r_{ce} \cdot \sin(\varphi_1) & \dots & 0 & 0 & 0 & \dots & 0 & 1 & \cos(\varphi_1) \\ r_{ce} \cdot \cos(\varphi_1) & \dots & 0 & 0 & 0 & \dots & 0 & 1 & \sin(\varphi_1) \\ & & \vdots & & & & & & \\ 0 & \dots & 0 & -r_{ce} \cdot \sin(\varphi_i) & 0 & \dots & 0 & 1 & \cos(\varphi_i) \\ 0 & \dots & 0 & r_{ce} \cdot \cos(\varphi_i) & 0 & \dots & 0 & 1 & \sin(\varphi_i) \\ & & \vdots & & & & & & \\ 0 & \dots & 0 & 0 & \dots & 0 & -r_{ce} \cdot \sin(\varphi_n) & 1 & \cos(\varphi_n) \\ 0 & \dots & 0 & 0 & \dots & 0 & r_{ce} \cdot \cos(\varphi_n) & 1 & \sin(\varphi_n) \end{bmatrix} \quad (\text{C.40})$$

L'erreur quadratique [12] à minimiser, $G(\mathbf{X}_p)$, est obtenue par la formule :

$$G(\mathbf{X}_p) = [\mathbf{X}_{mesure} - \mathbf{h}(\mathbf{X}_p(k))]^T \cdot [\mathbf{X}_{mesure} - \mathbf{h}(\mathbf{X}_p(k))] \quad (\text{C.41})$$

Le vecteur $\mathbf{X}_p(0)$ dans l'équation (C.41) est initialisé par le vecteur \mathbf{X}_{ce} obtenu avec la méthode de Guivant décrite précédemment. La matrice de covariance associée au vecteur d'état \mathbf{X}_{ce} est calculée avec :

$$\mathbf{C}_{X_{ce}} = [\mathbf{J}(k)^T \cdot \mathbf{R}_{mesure}^{-1} \cdot \mathbf{J}(k)]^{-1} \quad (\text{C.42})$$

C.3.3 Algorithme de Levenberg-Marquardt

Dans un premier temps, la méthode de Gauss-Newton [69] a été utilisée pour minimiser le critère défini par l'équation (C.41). Des simulations ont été réalisées avec des bruits sur les données laser identiques aux bruits expérimentaux. Il a été observé que cette méthode d'estimation donnait des estimations fausses (cf. la figure C.6). Pour répondre à ce problème, un algorithme de Levenberg-Marquardt a été implémenté (Algorithm C.1).

Le paramètre de descente du gradient de cet algorithme, r_d , a été déterminé à partir de simulations. Les résultats ont été obtenus en générant 5 points de mesure semblables aux données expérimentales. Les points de mesure sont repartis sur une section d'une longueur de 10 centimètres du périmètre du cercle. L'algorithme réalise 20 itérations. Chaque simulation est réalisée

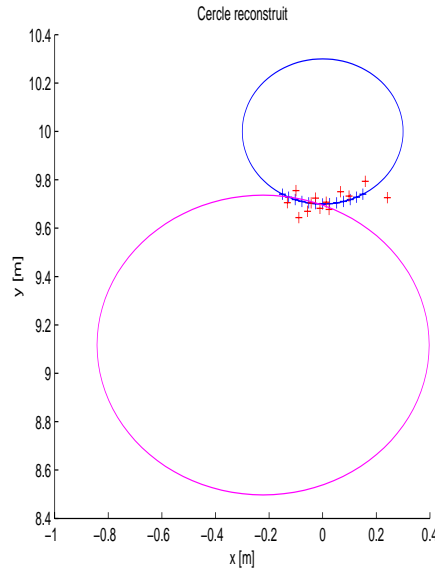


Figure C.6: Exemple de fausse détection d'un cercle : le cercle réel est représenté en bleu, les mesures bruitées de ce cercle en rouge et le cercle estimé en violet.

50 fois. Les résultats présentés sur la figure C.7 correspondent aux erreurs sur les estimées de la valeur du rayon et des coordonnées du centre du cercle en fonction de la valeur du paramètre de descente du gradient r_d . Il est observé que la valeur optimale du paramètre se situe à la valeur 100. Le coefficient r_d est fixé à 100 dans l'algorithme C.1.

C.3.4 Comparaison des trois méthodes

La figure C.8 présente une comparaison des méthodes présentées précédemment : méthode de Guivant [77] (courbes vertes), méthode de Gauss-Newton [69] (courbes violettes) et algorithme de Levenberg-Marquardt (courbes bleues). Ces courbes correspondent à l'erreur sur les paramètres estimés d'un cercle en fonction de la valeur du rayon r_{ce} .

Premièrement, l'erreur commise sur les estimées avec la méthode de Guivant [77] croît linéairement en fonction de l'augmentation de la valeur du rayon. Lorsque la valeur du rayon est très faible, les estimations sont correctes jusqu'à une valeur de 20 centimètres. Les deux méthodes itératives donnent de meilleurs résultats sur l'ensemble des valeurs du rayon du cercle.

La comparaison des deux approches itératives prouve que l'estimation avec l'algorithme de Levenberg-Marquardt est bien plus précise. Sur la figure C.8, l'erreur avec la méthode de Levenberg-Marquardt (courbe bleue) reste peu importante en comparaison avec l'algorithme de Gauss-Newton (courbes roses) dont les erreurs sur les estimées des paramètres du cercle sont beaucoup plus conséquentes.

L'emploi d'un algorithme itératif de Levenberg-Marquardt, initialisé avec les valeurs obtenues avec la méthode de Guivant, correspond à la solution optimale pour l'extraction et l'estimation des paramètres d'un cercle.

Algorithme C.1 : Algorithme de Levenberg-MarquardtIt pour l'estimation des paramètres d'un cercle

$r_d \leftarrow 100$
 $\epsilon \leftarrow 0.01$
 $\text{seuil} \leftarrow 0.035$
 $\lambda(0) \leftarrow 0.01$
 $\mathbf{X}_p(0) \leftarrow$ Estimation avec la méthode de Guivant
 $k \leftarrow 0$
 $\text{erreur} \leftarrow 10$

tant que $((\text{erreur} < \text{seuil}) \vee (k < k_{\max}))$ **faire**

$k \leftarrow k+1$

$r_\lambda = 0$

si $(\lambda(k-1) > \epsilon)$ **alors**

Calcul de $G\left(\frac{\lambda(k-1)}{r_d}\right)$

si $(G\left(\frac{\lambda(k-1)}{r}\right) < G(\mathbf{X}_p(k)))$ **alors**

$r_\lambda = 1$

fin si

fin si

si $(r_\lambda = 1)$ **alors**

$\lambda(k) \leftarrow \frac{\lambda(k-1)}{r_d}$

Calcul de $G(\mathbf{X}_p(k+1))$

sinon

Calcul de $G(\lambda(k-1))$

si $G(\lambda(k-1)) < G(\mathbf{X}_p(k))$ **alors**

$\lambda(k) \leftarrow \lambda(k-1)$

calcul de $G(\mathbf{X}_p(k+1))$

$G(\lambda(k)) \leftarrow G(\mathbf{X}_p(k+1))$

sinon

$G(\lambda(k)) = G(\lambda(k-1))$

$\text{coef}_w \leftarrow 0$

tant que $(G(\lambda(k)) > G(\mathbf{X}_p(k)))$ **faire**

$\text{coef}_w \leftarrow \text{coef}_w + 1$

$\lambda(k) \leftarrow \lambda(k-1) \cdot r_d^{\text{coef}_w}$

Calcul de $G(\lambda(k))$

fin tant que

Calcul de $G(\mathbf{X}_p(k+1))$

fin si

fin si

$\text{erreur} \leftarrow \frac{\mathbf{X}_p(k+1) - \mathbf{X}_p(k)}{\mathbf{X}_p(k)}$

fin tant que

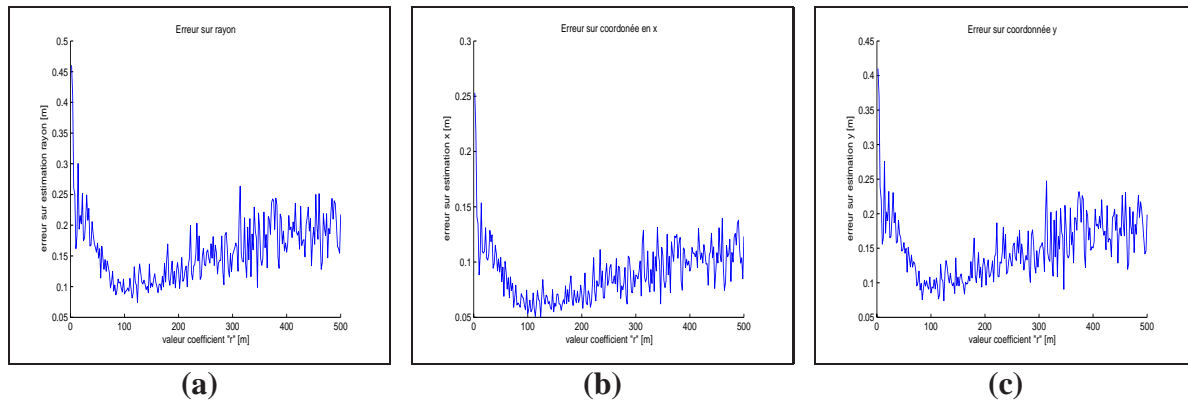


Figure C.7: Résultats de simulation pour déterminer la valeur optimum du paramètre de descente du gradient : (a) erreur commise sur le rayon, (b) erreur commise sur la coordonnée x du centre et (c) erreur commise sur la coordonnée y du centre du cercle.

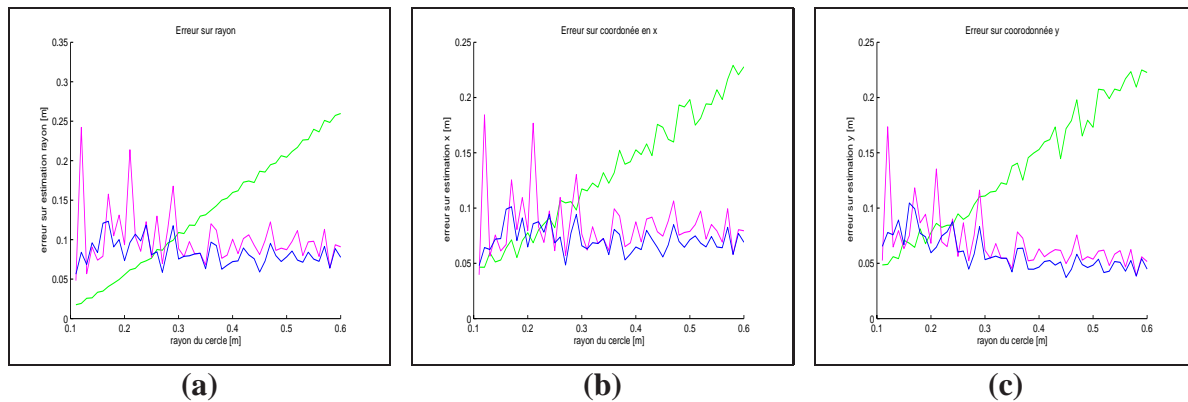


Figure C.8: Comparaison des trois algorithmes de détermination des paramètres d'un cercle : méthode de Guivant [77] (courbes vertes), méthode de Gauss-Newton [69] (courbes violettes) et algorithme de Levenberg-Marquardt (courbes bleues). La figure (a) représente l'erreur commise sur le rayon, (b) l'erreur commise sur la coordonnée x du centre et (c) l'erreur commise sur la coordonnée y du centre du cercle.

Annexe D

Description des méthodes d'analyse de la consistance des estimations et formules du filtre PEKF-SLAM

D.1 Outils pour la description de la consistance des estimées

Dans cette section, trois démonstrations, basées sur les équations du filtre EKF-SLAM, sont proposées. Elles permettent de comprendre les problèmes inhérents à cette application mais également d'en déterminer la source.

Premièrement, les limites de convergence des estimations d'une droite sont étudiées. Ces calculs permettent de statuer sur l'applicabilité des théorèmes non-linéaires (décrits dans le chapitre 3) sur une droite.

Deuxièmement, deux démonstrations, basées sur les équations du filtre EKF-SLAM, mettent en lumière l'origine des inconsistances des estimations. Ces démonstrations sont effectuées sur deux cas simples : le véhicule est soit immobile, soit en mouvement, et observe un unique amer.

D.1.1 Limite de convergence pour un amer de type droite

Pour permettre l'étude de cette limite, nous allons travailler sur un exemple simple. Le système sera composé du véhicule et d'un unique amer.

Description du système et de l'équation d'observation :

Soit un robot stationnaire dont la position est définie par le vecteur $\mathbf{X}_V = [x_v y_v \theta_v]^T$ associée à la matrice de covariance \mathbf{P}_V . Cette matrice est initialisée avec la valeur \mathbf{P}_0 :

$$\mathbf{P}_0 = \begin{bmatrix} P_{x_v y_v} & P_{x_v y_v \theta_v}^T \\ P_{x_v y_v \theta_v} & p_{\theta_v} \end{bmatrix} \quad (\text{D.1})$$

Ce robot observe un unique amer de type droite $\mathbf{X}_L = [x_l y_l \theta_l]^T$. La mesure de cette droite s'écrit $\mathbf{X}_{Lm} = [x_m y_m \theta_m]^T$ avec comme matrice de covariance associée \mathbf{R}_{Lm} . La concaténation

de ces vecteurs dans un unique vecteur s'écrit :

$$\mathbf{X}_{systeme} = \begin{bmatrix} \mathbf{X} \\ \mathbf{X}_{Lm} \end{bmatrix} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_L \\ \mathbf{X}_{Lm} \end{bmatrix} \quad (D.2)$$

La matrice de covariance associée à ce vecteur d'état s'écrit :

$$\mathbf{P}_{systeme} = \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{R}_{Lm} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_V & \mathbf{P}_{VM} & 0 \\ \mathbf{P}_{VM} & \mathbf{P}_M & 0 \\ 0 & 0 & \mathbf{R}_{Lm} \end{bmatrix} \quad (D.3)$$

La fonction d'observation d'une droite, \mathbf{h} , s'écrit :

$$\mathbf{Z} = \mathbf{h}(\mathbf{X}_{systeme}) = \ominus \mathbf{X}_L \oplus \mathbf{X}_V \oplus \mathbf{X}_{Lm} = \mathbf{0} \quad (D.4)$$

La matrice jacobienne, \mathbf{H} , associée à ce calcul s'écrit donc :

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_X & \mathbf{H}_{X_{Lm}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{X_V} & \mathbf{H}_{X_L} & \mathbf{H}_{X_{Lm}} \end{bmatrix} \quad (D.5)$$

A partir des équations (D.3) et (D.5), le calcul de la matrice de covariance, \mathbf{P}_Z , associée à l'observation \mathbf{Z} , se formule :

$$\mathbf{P}_Z = \mathbf{H} \cdot \mathbf{P}_{systeme} \cdot \mathbf{H}^T = \mathbf{H}_X \cdot \mathbf{P} \cdot (\mathbf{H}_X)^T + \mathbf{H}_{X_{Lm}} \cdot \mathbf{R}_{Lm} \cdot (\mathbf{H}_{X_{Lm}})^T \quad (D.6)$$

Pour simplifier la suite de la démonstration, la matrice de covariance associée à la mesure s'écrit :

$$\mathbf{R}_L = \mathbf{H}_{X_{Lm}} \cdot \mathbf{R}_{Lm} \cdot (\mathbf{H}_{X_{Lm}})^T \quad (D.7)$$

La fonction d'observation, \mathbf{h} , est développée en utilisant les fonctions de composition (cf. l'annexe B) :

$$\mathbf{Z} = \begin{bmatrix} (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \cos(\theta_l) + \dots \\ \dots (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \cdot \sin(\theta_l) \\ - (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \cdot \sin(\theta_l) + \dots \\ \dots (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \cdot \cos(\theta_l) \\ \theta_v - \theta_l + \theta_m \end{bmatrix} \quad (D.8)$$

A partir des équations du système (D.8), la matrice jacobienne de la fonction d'observation, \mathbf{h} , par rapport au vecteur d'état du véhicule, \mathbf{X}_V , s'écrit :

$$\mathbf{H}_{X_V} = \begin{bmatrix} \cos(\theta_l) & \sin(\theta_l) & \nabla_x H_{\theta_v} \\ -\sin(\theta_l) & \cos(\theta_l) & \nabla_y H_{\theta_v} \\ 0 & 0 & 1 \end{bmatrix} \quad (D.9)$$

Les dérivées partielles des coordonnées en x et y de la fonction \mathbf{h} par rapport à l'orientation du véhicule, θ_v , s'écrivent :

$$\begin{aligned} \nabla_x H_{\theta_v} &= -\cos(\theta_l) \cdot (-x_m \cdot \sin(\theta_v) - y_m \cdot \cos(\theta_v)) + \sin(\theta_l) \cdot (x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \\ \nabla_y H_{\theta_v} &= -\sin(\theta_l) \cdot (-x_m \cdot \sin(\theta_v) - y_m \cdot \cos(\theta_v)) + \cos(\theta_l) \cdot (x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) \end{aligned} \quad (D.10)$$

La matrice jacobienne de la fonction d'observation, \mathbf{h} , par rapport au vecteur d'état de l'amer, \mathbf{X}_L , s'écrit :

$$\mathbf{H}_{X_L} = \begin{bmatrix} -\cos(\theta_l) & -\sin(\theta_l) & \nabla_x H_{\theta_l} \\ \sin(\theta_l) & -\cos(\theta_l) & \nabla_y H_{\theta_l} \\ 0 & 0 & -1 \end{bmatrix} \quad (\text{D.11})$$

où les dérivées partielles des coordonnées en x et y de la fonction \mathbf{h} par rapport à l'orientation du véhicule, θ_l , s'écrivent :

$$\begin{aligned} \nabla_x H_{\theta_l} &= -\sin(\theta_l) \cdot (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) + \dots \\ &\quad \dots \cos(\theta_l) \cdot (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \\ \nabla_y H_{\theta_l} &= -\cos(\theta_l) \cdot (x_v - x_l + x_m \cdot \cos(\theta_v) - y_m \cdot \sin(\theta_v)) - \dots \\ &\quad \dots \sin(\theta_l) \cdot (y_v - y_l + x_m \cdot \sin(\theta_v) + y_m \cdot \cos(\theta_v)) \end{aligned} \quad (\text{D.12})$$

Description des matrices jacobienes liées à la fonction d'observation

La matrice jacobienne de la fonction d'observation, \mathbf{h} , par rapport au vecteur d'état de la mesure de la droite, \mathbf{X}_{Lm} , s'écrit :

$$\mathbf{H}_{X_{Lm}} = \begin{bmatrix} \cos(\theta_v - \theta_l) & -\sin(\theta_v - \theta_l) & 0 \\ \sin(\theta_v - \theta_l) & \cos(\theta_v - \theta_l) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{D.13})$$

Valeur de la matrice de covariance de la carte après une infinité d'observations :

Maintenant que les différentes matrices du système sont décrites, nous pouvons calculer le gain d'information apporté par l'observation d'un amer de type droite :

$$\Omega_{new} = (\mathbf{H}_X)^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_X \quad (\text{D.14})$$

Avec les équations (D.9) et (D.11), la matrice jacobienne, \mathbf{H}_X , s'écrit :

$$\mathbf{H}_X = \begin{bmatrix} -\mathbf{H}_{x_l y_l} & \mathbf{H}_{\theta_v} & \mathbf{H}_{x_l y_l} & \mathbf{H}_{\theta_l} \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad (\text{D.15})$$

Si le véhicule réalise n observations de l'amer \mathbf{X}_L , le gain d'information total sur le système s'écrit :

$$\begin{aligned} \Omega_n &= \begin{bmatrix} \Omega_0 & 0 \\ 0 & 0 \end{bmatrix} + n \cdot \Omega_{new} \\ &= \begin{bmatrix} \Omega_0 + n \cdot (\mathbf{H}_{X_L})^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_V} & n \cdot (\mathbf{H}_{X_V})^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_L} \\ n \cdot (\mathbf{H}_{X_L})^T \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_V} & n \cdot (\mathbf{H}_{X_L}) \cdot \mathbf{R}_L^{-1} \cdot \mathbf{H}_{X_L} \end{bmatrix} \end{aligned} \quad (\text{D.16})$$

En employant le lemme d'inversion de matrice, la matrice de covariance \mathbf{P}_n s'écrit :

$$\mathbf{P}_n = \begin{bmatrix} \mathbf{P}_0 & -\mathbf{P}_0 \cdot (\mathbf{H}_{X_V} \cdot (\mathbf{H}_{X_L}^{-1}))^T \\ -(\mathbf{H}_{X_L})^{-1} \cdot \mathbf{H}_{X_V} \cdot \mathbf{P}_0 & \mathbf{P}_{L_n} \end{bmatrix} \quad (\text{D.17})$$

avec :

$$\mathbf{P}_{L_n} = (\mathbf{H}_{X_L})^{-1} \cdot \mathbf{H}_{X_V} \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_V} \cdot (\mathbf{H}_{X_L}^{-1}))^T + \frac{(\mathbf{H}_{X_L})^{-1} \cdot \mathbf{R}_L \cdot (\mathbf{H}_{X_L})^{-T}}{n} \quad (\text{D.18})$$

Soit la matrice $\mathbf{H}_{X_L}^*$ telle que :

$$\mathbf{H}_{X_L}^* = -(\mathbf{H}_{X_L})^{-1} \cdot \mathbf{H}_{X_V} \quad (\text{D.19})$$

Après une infinité d'observations de l'amer \mathbf{X}_L , la limite de la matrice de covariance associée à la carte s'écrit :

$$\mathbf{P}_{L_\infty} = [\mathbf{H}_{X_L}^* \cdot \mathbf{P}_0 \cdot (\mathbf{H}_{X_L}^*)^T] \quad (\text{D.20})$$

Pour analyser la convergence de la matrice de covariance \mathbf{P}_M , il faut réaliser l'étude de la matrice (D.19) où l'inversion de la matrice \mathbf{H}_{X_L} est nécessaire.

Soit une matrice \mathbf{A} telle que :

$$\mathbf{A} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad (\text{D.21})$$

Soit l'inverse de la matrice, \mathbf{A}^{-1} , s'écrivant :

$$\mathbf{A}^{-1} = \begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} ei - fh & ch - bi & bf - ce \\ fg - di & ai - cg & cd - af \\ dh - eg & bg - ah & ae - bd \end{bmatrix} \quad (\text{D.22})$$

Le déterminant de la matrice \mathbf{A} s'écrit :

$$\det \mathbf{A} = a \cdot (ei - fh) - b \cdot (di - fg) + c \cdot (dh - eg) \quad (\text{D.23})$$

Dans notre cas d'étude, les valeurs des éléments g et h sont nuls et i est égale à -1 (équation (D.11)). Cette remarque permet de réécrire la matrice \mathbf{A}^{-1} :

$$\mathbf{A}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} -e & b & bf - ce \\ d & -a & cd - af \\ 0 & 0 & ae - bd \end{bmatrix} \quad (\text{D.24})$$

Le déterminant de la matrice \mathbf{A} se simplifie également :

$$\det \mathbf{A} = -ae + bd \quad (\text{D.25})$$

En prenant en compte les expressions de (D.24) et (D.25), l'inversion de la matrice \mathbf{A} est obtenue avec l'équation :

$$\mathbf{A}^{-1} = \begin{bmatrix} \frac{-e}{-ae+bd} & \frac{b}{-ae+bd} & \frac{bf-ce}{-ae+bd} \\ \frac{d}{-ae+bd} & \frac{-a}{-ae+bd} & \frac{cd-af}{-ae+bd} \\ 0 & 0 & -1 \end{bmatrix} \quad (\text{D.26})$$

La sous matrice composée des éléments x_1, x_2, x_4 et x_5 de A^{-1} correspond à l'inversion du bloc matrice composé des éléments a, b, d et e de la matrice \mathbf{A} :

$$\begin{bmatrix} x_1 & x_2 \\ x_4 & x_5 \end{bmatrix} = \begin{bmatrix} \frac{-e}{-ae+bd} & \frac{b}{-ae+bd} \\ \frac{d}{-ae+bd} & \frac{-a}{-ae+bd} \end{bmatrix} = \begin{bmatrix} a & b \\ d & e \end{bmatrix}^{-1} \quad (\text{D.27})$$

Cette constatation permet l'écriture de l'inverse de la matrice \mathbf{H}_{X_L} sous la forme :

$$\mathbf{H}_{X_L}^{-1} = \begin{bmatrix} \mathbf{H}_{x_1y_l}^{-1} & \mathbf{H}_{\theta_l}^* \\ 0 & -1 \end{bmatrix} \quad (\text{D.28})$$

La matrice $\mathbf{H}_{\theta_l}^*$ résulte de l'inversion de la matrice \mathbf{H}_{X_L} correspondant aux deux premières droites de la dernière colonne. Avec les expressions (D.19) et (D.28), le terme $\mathbf{H}_{X_L}^*$ s'écrit :

$$\begin{aligned} \mathbf{H}_{X_L}^* &= -1 \cdot \begin{bmatrix} \mathbf{H}_{x_1y_l}^{-1} & \mathbf{H}_{\theta_l}^* \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} -\mathbf{H}_{x_1y_l} & \mathbf{H}_{\theta_v} \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I} & -\mathbf{H}_{x_1y_l}^{-1} \cdot \mathbf{H}_{\theta_v} - \mathbf{H}_{\theta_l}^* \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (\text{D.29})$$

L'observation de la forme de cette matrice (D.29) permet de vérifier que le **théorème 2.1** est également applicable aux droites à l'instar des points (cf. le chapitre 3). En effet, lorsque le véhicule réalise une multitude d'observations d'une même droite, la matrice de covariance de la carte \mathbf{P}_M tend vers la valeur \mathbf{P}_0 si l'incertitude sur l'orientation du véhicule est nulle (équations (D.20) et (D.29)). Dans le cas contraire, la matrice \mathbf{P}_M ne sera jamais égale à la matrice \mathbf{P}_0 . De surcroît, à la différence des amers de type point, la valeur de la matrice $\mathbf{P}_{L\infty}$ dépend des matrices jacobiniennes calculées à partir des orientations du véhicule et de la droite.

D.1.2 Robot stationnaire observant un amer

Dans un filtrage de Kalman non-linéaire, le calcul de l'innovation s'écrit :

$$v = \mathbf{Z} - \mathbf{h}(\hat{\mathbf{X}}) \quad (\text{D.30})$$

v permet la mise à jour du vecteur d'état, $\hat{\mathbf{X}}$, et la matrice de covariance associée, \mathbf{P} . Reprenons l'exemple où le véhicule est stationnaire et observe un amer, \mathbf{X}_A , la quantité v s'écrit en utilisant une linéarisation de Taylor :

$$\begin{aligned} v &\approx \mathbf{h}(\mathbf{X}) - \mathbf{h}(\hat{\mathbf{X}}) \\ &\approx \mathbf{H}_X(\mathbf{X} - \hat{\mathbf{X}}) \\ &\approx \mathbf{H}_{\theta_v} \cdot (\theta_v - \hat{\theta}_v) - \mathbf{H}_{X_A} \cdot (\mathbf{X}_{x_vy_v} - \hat{\mathbf{X}}_{x_vy_v}) + \mathbf{H}_{X_A} \cdot (\mathbf{X}_A - \hat{\mathbf{X}}_A) \end{aligned} \quad (\text{D.31})$$

En séparant les valeurs exactes des valeurs estimées, l'équation (D.31) s'écrit :

$$\begin{aligned} \mathbf{H}_{\theta_v} \cdot \theta_v + \mathbf{H}_{X_A} \cdot \mathbf{X}_{x_vy_v} - \mathbf{H}_{X_A} \cdot \mathbf{X}_A &\approx -v + \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v + \mathbf{H}_{X_A} \cdot \hat{\mathbf{X}}_{x_vy_v} - \mathbf{H}_{X_A} \cdot \hat{\mathbf{X}}_A \\ \mathbf{H}_{X_A}^{-1} \cdot \mathbf{H}_{\theta_v} \cdot \theta_v + \mathbf{X}_{x_vy_v} - \mathbf{X}_A &\approx -\mathbf{H}_{X_A}^{-1} \cdot v + \mathbf{H}_{X_A}^{-1} \cdot \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v + \hat{\mathbf{X}}_{x_vy_v} - \hat{\mathbf{X}}_A \end{aligned} \quad (\text{D.32})$$

Le véhicule réalise maintenant deux observations successives de l'amer, \mathbf{Z}_1 et \mathbf{Z}_2 . Ces deux mesures permettent de calculer deux innovations, v_1 et v_2 telles que :

$$\begin{aligned}\nabla_1 \mathbf{H}_{X_A}^{-1} \cdot \nabla_1 \mathbf{H}_{\theta_v} \cdot \theta_v + \mathbf{X}_{x_v y_v} - \mathbf{X}_A &\approx -\nabla_1 \mathbf{H}_{X_A}^{-1} \cdot v_1 + \nabla_1 \mathbf{H}_{X_A}^{-1} \cdot \nabla_1 \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v + \hat{\mathbf{X}}_{x_v y_v} - \hat{\mathbf{X}}_A \\ \nabla_2 \mathbf{H}_{X_A}^{-1} \cdot \nabla_2 \mathbf{H}_{\theta_v} \cdot \theta_v + \mathbf{X}_{x_v y_v} - \mathbf{X}_A &\approx -\nabla_2 \mathbf{H}_{X_A}^{-1} \cdot v_2 + \nabla_2 \mathbf{H}_{X_A}^{-1} \cdot \nabla_2 \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v^1 + \hat{\mathbf{X}}_{x_v y_v}^1 - \hat{\mathbf{X}}_A^1\end{aligned}\quad (\text{D.33})$$

∇_1 et ∇_2 sont donc les matrices jacobiennes calculées aux positions estimées du robot et de l'amer respectivement pour les observations \mathbf{Z}_1 et \mathbf{Z}_2 . $\hat{\theta}_v^1$, $\hat{\mathbf{X}}_{x_v y_v}^1$ et $\hat{\mathbf{X}}_A^1$ représentent les estimations obtenues avec l'observation \mathbf{Z}_1 . En soustrayant les deux équations du système (D.33), un nouveau système d'équations est obtenu :

$$\begin{aligned}(\nabla_1 \mathbf{H}_{X_A}^{-1} \cdot \nabla_1 \mathbf{H}_{\theta_v} - \nabla_2 \mathbf{H}_{X_A}^{-1} \cdot \nabla_2 \mathbf{H}_{\theta_v}) \cdot \theta_v &\approx \nabla_2 \mathbf{H}_{X_A}^{-1} \cdot v_2 - \nabla_1 \mathbf{H}_{X_A}^{-1} \cdot v_1 + \hat{\mathbf{X}}_{x_v y_v} - \hat{\mathbf{X}}_{x_v y_v}^1 - \hat{\mathbf{X}}_A \\ &\quad + \hat{\mathbf{X}}_A^1 + \nabla_1 \mathbf{H}_{X_A}^{-1} \cdot \nabla_1 \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v - \nabla_2 \mathbf{H}_{X_A}^{-1} \cdot \nabla_2 \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v^1\end{aligned}\quad (\text{D.34})$$

Comme les valeurs estimées $\hat{\mathbf{X}}_A$ et $\hat{\mathbf{X}}_A^1$ ne sont pas égales, les matrices jacobiennes, $\nabla_1 \mathbf{H}_{X_A}^{-1}$ et $\nabla_2 \mathbf{H}_{X_A}^{-1}$, calculées à partir de ces estimations, ne sont pas égales. Le membre de droite de l'équation (D.34) n'est donc pas nul. Le calcul des deux innovations v_1 et v_2 apporte donc une information sur l'angle θ_v . Cependant, l'observation d'un seul amer ne doit pas améliorer la connaissance de l'orientation du véhicule (**théorème 2.1**). Par conséquent, cette information apportée au système est incorrecte. Elle rendra les estimations inconsistantes puisque l'incertitude définie par la matrice de covariance, \mathbf{P} , sera surévaluée.

D.1.3 Robot mobile observant un amer

Le robot observe un amer, X_A , au point a puis se déplace vers le point b et réobserve le même amer. Avec les mêmes équations (D.33), les innovations, v_a et v_b , calculées aux deux positions du véhicule, respectivement a et b s'écrivent :

$$\begin{aligned}\nabla_a \mathbf{H}_{X_A}^{-1} \cdot \nabla_a \mathbf{H}_{\theta_v} \cdot \theta_v^a + \mathbf{X}_{x_v y_v}^a - \mathbf{X}_A^a &\approx -\nabla_a \mathbf{H}_{X_A}^{-1} \cdot v_a + \nabla_a \mathbf{H}_{X_A}^{-1} \cdot \nabla_a \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v^a \\ &\quad + \hat{\mathbf{X}}_{x_v y_v}^a - \hat{\mathbf{X}}_A^a \\ \nabla_b \mathbf{H}_{X_A}^{-1} \cdot \nabla_b \mathbf{H}_{\theta_v} \cdot \theta_v^b + \mathbf{X}_{x_v y_v}^b - \mathbf{X}_A^b &\approx -\nabla_b \mathbf{H}_{X_A}^{-1} \cdot v_b + \nabla_b \mathbf{H}_{X_A}^{-1} \cdot \nabla_b \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v^b \\ &\quad + \hat{\mathbf{X}}_{x_v y_v}^b - \hat{\mathbf{X}}_A^b\end{aligned}\quad (\text{D.35})$$

A partir de la modélisation du déplacement du véhicule, il est possible d'écrire la linéarisation suivante :

$$\begin{aligned}\theta_v^b &\approx \theta_v^a + f_{\theta_v} \\ \mathbf{X}_{x_v y_v}^b &\approx \mathbf{X}_{x_v y_v}^a + \begin{bmatrix} \Delta t \cdot v \cdot \cos \theta_v^a \\ \Delta t \cdot v \cdot \sin \theta_v^a \end{bmatrix} + \begin{bmatrix} \Delta t \cdot v \cdot \cos \hat{\theta}_v^a \\ \Delta t \cdot v \cdot \sin \hat{\theta}_v^a \end{bmatrix} \cdot (\theta_v^a - \hat{\theta}_v^a)\end{aligned}\quad (\text{D.36})$$

avec

$$f_{\theta_v} = \frac{\Delta t.v. \tan(\varphi)}{L} \quad (\text{D.37})$$

En remplaçant les variables de l'équation (D.36) dans le système (D.35) et en soustrayant les deux équations de ce système, l'équation suivante est obtenue :

$$\begin{aligned} \theta_v^a \cdot & \left(\nabla_a \mathbf{H}_{X_A}^{-1} \cdot \nabla_a \mathbf{H}_{\theta_v} - \nabla_b \mathbf{H}_{X_A}^{-1} \cdot \nabla_b \mathbf{H}_{\theta_v} - \begin{bmatrix} \Delta t.v. \cos \theta_v^a \\ \Delta t.v. \sin \theta_v^a \end{bmatrix} \right) \approx \nabla_b \mathbf{H}_{X_A}^{-1} \cdot v_b \\ & - \nabla_a \mathbf{H}_{X_A}^{-1} \cdot v_a + \nabla_a \mathbf{H}_{X_A}^{-1} \cdot \nabla_a \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v^a + \hat{\mathbf{X}}_{x_v y_v}^a - \hat{\mathbf{X}}_A^a - \nabla_b \mathbf{H}_{X_A}^{-1} \cdot \nabla_b \mathbf{H}_{\theta_v} \cdot \hat{\theta}_v^b \\ & - \hat{\mathbf{X}}_{x_v y_v}^b + \hat{\mathbf{X}}_A^a + \nabla_b \mathbf{H}_{X_A}^{-1} \cdot \nabla_b \mathbf{H}_{\theta_v} \cdot f_{\theta_v} + \begin{bmatrix} \Delta t.v. \cos \theta_v^a \\ \Delta t.v. \sin \theta_v^a \end{bmatrix} - \begin{bmatrix} \Delta t.v. \cos \theta_v^a \\ \Delta t.v. \sin \theta_v^a \end{bmatrix} \cdot \hat{\theta}_v^a \end{aligned} \quad (\text{D.38})$$

La condition pour que les estimations soient consistantes se formule :

$$\nabla_a \mathbf{H}_{X_A}^{-1} \cdot \nabla_a \mathbf{H}_{\theta_v} = \nabla_b \mathbf{H}_{X_A}^{-1} \cdot \nabla_b \mathbf{H}_{\theta_v} - \begin{bmatrix} \Delta t.v. \cos \theta_v^a \\ \Delta t.v. \sin \theta_v^a \end{bmatrix} \quad (\text{D.39})$$

L'information sur l'orientation du véhicule, θ_v^a , est apportée par l'observation de l'amer \mathbf{X}_A . Comme les jacobiennes de cette expression sont calculées à partir des valeurs des estimations, différentes à chaque itération du filtre, la relation (D.39) n'est pas vérifiée. Le **théorème 2.1** de convergence des estimations n'est donc pas respecté lorsque le véhicule se déplace et n'observe qu'un seul amer.

D.2 Formulation du filtre PEKF-SLAM

Cette annexe complète la description mathématique de l'approche PEKF-SLAM proposée dans le chapitre 3 de ce document. Sont présentées les différentes équations de prédiction des états, du véhicule et des amers, employées dans l'estimation du filtre de Kalman non linéaire. Cette description permet de souligner l'avantage principal de l'emploi d'un tel filtre : les sommes de polynômes. Cependant, une telle représentation impose des calculs fastidieux présentés dans la suite de cette annexe.

D.2.1 Modélisation du déplacement du véhicule

Reformulation des équations non-linéaires associées au déplacement du véhicule :

Dans la modélisation de type tricycle ou d'Ackerman (cf. l'annexe B), l'orientation du véhicule évolue entre deux instants k et $k + 1$ selon l'équation :

$$\theta_v(k + 1|k) = \theta_v(k|k) + \frac{\Delta t.v. \tan(\varphi)}{L} \quad (\text{D.40})$$

L'évolution du cosinus de l'angle θ_v s'écrit donc :

$$\cos(\theta_v)(k + 1|k) = \cos(\theta_v(k + 1|k)) = \cos(\theta_v(k|k) + \frac{\Delta t.v. \tan(\varphi)}{L}) \quad (\text{D.41})$$

En appliquant les formules trigonométriques associées au cosinus, l'équation (D.41) s'écrit :

$$\cos(\theta_v)(k+1|k) = \cos(\theta_v(k|k)) \cdot \cos\left(\frac{\Delta t \cdot v(k) \cdot \tan(\varphi(k))}{L}\right) - \sin(\theta_v(k|k)) \cdot \sin\left(\frac{\Delta t \cdot v(k) \cdot \tan(\varphi(k))}{L}\right) \quad (\text{D.42})$$

L'équation d'évolution de la variable $\sin(\theta_v)$ est décomposée en employant les mêmes formules trigonométriques :

$$\sin(\theta_v)(k+1|k) = \sin(\theta_v(k|k)) \cdot \cos\left(\frac{\Delta t \cdot v(k) \cdot \tan(\varphi(k))}{L}\right) + \cos(\theta_v(k|k)) \cdot \sin\left(\frac{\Delta t \cdot v(k) \cdot \tan(\varphi(k))}{L}\right) \quad (\text{D.43})$$

La nouvelle modélisation du déplacement du véhicule entre deux instants k et $k+1$ repose toujours sur le vecteur de commande \mathbf{u} . Dans la suite des explications, ce vecteur est modifié pour s'adapter à la nouvelle modélisation du déplacement du véhicule.

Modification du vecteur de commande :

Les deux données fournies par les capteurs proprioceptifs correspondent à la vitesse du véhicule, v et l'angle de braquage des roues avant, φ . Le vecteur de commande est $\mathbf{u} = [v \ \varphi]^T$. Comme les deux commandes sont indépendantes, la matrice de covariance associée au vecteur \mathbf{u} s'écrit :

$$\mathbf{Q}_u = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\varphi^2 \end{bmatrix} \quad (\text{D.44})$$

σ_v^2 représente la variance sur la donnée v et σ_φ^2 , la variance sur la donnée φ . Le nouveau vecteur de commande créé, \mathbf{U} , comportant quatre commandes distinctes :

$$\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} v \cdot \Delta t \\ \frac{\Delta t \cdot v \cdot \tan(\varphi)}{L} \\ \sin\left(\frac{\Delta t \cdot v \cdot \tan(\varphi)}{L}\right) \\ \cos\left(\frac{\Delta t \cdot v \cdot \tan(\varphi)}{L}\right) \end{bmatrix} \quad (\text{D.45})$$

La matrice de covariance, \mathbf{Q}_U , associée au vecteur, \mathbf{U} , se calcule avec la formule :

$$\mathbf{Q}_U = \mathbf{F}_u \cdot \mathbf{Q}_u \cdot (\mathbf{F}_u)^T \quad (\text{D.46})$$

\mathbf{F}_U est la matrice jacobienne de l'équation (D.45) par rapport au vecteur de commande \mathbf{u} :

$$\mathbf{F}_u = \begin{bmatrix} \Delta t & 0 \\ \frac{\Delta t \cdot \tan(\varphi)}{L} & \frac{\Delta t \cdot v}{L \cdot \cos(\varphi)^2} \\ \frac{\Delta t \cdot \tan(\varphi)}{L} \cdot \cos\left(\frac{\Delta t \cdot v \cdot \tan(\varphi)}{L}\right) & \frac{\Delta t \cdot v}{L \cdot \cos(\varphi)^2} \cdot \cos\left(\frac{\Delta t \cdot v \cdot \tan(\varphi)}{L}\right) \\ -\frac{\Delta t \cdot \tan(\varphi)}{L} \cdot \sin\left(\frac{\Delta t \cdot v \cdot \tan(\varphi)}{L}\right) & -\frac{\Delta t \cdot v}{L \cdot \cos(\varphi)^2} \cdot \sin\left(\frac{\Delta t \cdot v \cdot \tan(\varphi)}{L}\right) \end{bmatrix} \quad (\text{D.47})$$

D.2.2 Equations d'évolution des états du système

Dans l'approche SLAM présentée, l'environnement est considéré comme statique. Seul le véhicule et donc, seuls les états définissant sa position dans la carte estimée, évoluent lors de l'étape de prédiction. Cependant, en redéfinissant les états du système, les états décrivant les positions des amers sont couplés aux états du véhicule. Cette nouvelle écriture impose de faire évoluer tous les états.

D.2.2.1 Équations d'évolution des états du véhicule

Le nouvel état du véhicule, \mathcal{Z}_V évolue en suivant les équations précédemment définies :

$$\mathcal{Z}_V(k+1|k) = \begin{bmatrix} z_1(k|k) + z_5(k|k).u_1 \\ z_2(k|k) + z_4(k|k).u_1 \\ z_3(k|k) + u_2 \\ z_4(k|k).u_4 + z_5(k|k).u_3 \\ z_5(k|k).u_4 - z_4(k|k).u_3 \\ z_6(k|k).(u_4)^2 + z_7(k|k).(u_3)^2 + 2.z_{12}(k|k).u_3.u_4 \\ z_7(k|k).(u_4)^2 + z_6(k|k).(u_3)^2 - 2.z_{12}(k|k).u_3.u_4 \\ z_8(k|k).u_4 + z_9(k|k).u_3 + z_7(k|k).u_1.u_3 + z_{12}(k|k).u_1.u_4 \\ z_9(k|k).u_4 - z_8(k|k).u_3 + z_7(k|k).u_1.u_4 - z_{12}(k|k).u_1.u_2 \\ z_{10}(k|k).u_4 + z_{11}(k|k).u_3 + z_6(k|k).u_1.u_4 + z_{12}(k|k).u_1.u_3 \\ z_{11}(k|k).u_4 - z_{10}(k|k).u_3 - z_6(k|k).u_1.u_3 + z_{12}(k|k).u_1.u_4 \\ z_{12}(k|k).(u_4)^2 - z_6(k|k).u_3.u_4 + z_7(k|k).u_3.u_4 - z_{12}(k|k).(u_3)^2 \end{bmatrix} \quad (\text{D.48})$$

Le calcul de la matrice de covariance, associée au vecteur d'état du véhicule \mathcal{Z}_V (D.48), s'écrit :

$$\mathbf{P}_V(k+1|k) = \mathbf{J}_{\mathcal{Z}_V} \cdot \mathbf{P}_V(k|k) \cdot (\mathbf{J}_{\mathcal{Z}_V})^T + \mathbf{J}_U \cdot \mathbf{Q}_U \cdot \mathbf{J}_U^T \quad (\text{D.49})$$

La matrice jacobienne $\mathbf{J}_{\mathcal{Z}_V}$, dérivée de l'équation (D.48) par rapport à \mathcal{Z}_V , s'écrit :

$$\mathbf{J}_{\mathcal{Z}_V} = \begin{bmatrix} 1 & 0 & 0 & 0 & u_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & u_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_4 & u_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -u_3 & u_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & (u_4)^2 & (u_3)^2 & 0 & 0 & 0 & 0 & 2.u_3.u_4 \\ 0 & 0 & 0 & 0 & 0 & (u_3)^2 & (u_4)^2 & 0 & 0 & 0 & 0 & -2.u_3.u_4 \\ 0 & 0 & 0 & 0 & 0 & 0 & u_1.u_3 & u_4 & u_3 & 0 & 0 & u_1.u_4 \\ 0 & 0 & 0 & 0 & 0 & 0 & u_1.u_4 & -u_3 & u_4 & 0 & 0 & -u_1.u_3 \\ 0 & 0 & 0 & 0 & 0 & u_1.u_4 & 0 & 0 & 0 & u_4 & u_3 & u_1.u_3 \\ 0 & 0 & 0 & 0 & 0 & -u_1.u_3 & 0 & 0 & 0 & -u_3 & u_4 & u_1.u_4 \\ 0 & 0 & 0 & 0 & 0 & -u_3.u_4 & u_3.u_4 & 0 & 0 & 0 & 0 & (u_4)^2 - (u_3)^2 \end{bmatrix} \quad (\text{D.50})$$

L'application des puissances de Kronecker sur les états du système impose une réécriture du vecteur de commande, \mathbf{U} . Ce vecteur de commande agrandi, \mathcal{U} , associé à la matrice de cova-

riance $\mathbf{Q}_{\mathcal{U}}$, s'écrit sous la forme :

$$\mathcal{U} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ (u_3)^2 \\ (u_4)^2 \\ u_1 \cdot u_3 \\ u_1 \cdot u_4 \\ u_3 \cdot u_4 \end{bmatrix} \quad (\text{D.51})$$

La matrice jacobienne $\mathbf{J}_{\mathcal{U}}$, dérivée de l'équation (D.48) par rapport au vecteur de commande \mathcal{U} , équation (D.51), s'écrit :

$$\mathbf{J}_{\mathcal{U}} = \begin{bmatrix} z_5(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_4(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z_5(k|k) & z_4(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -z_4(k|k) & z_5(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & z_7(k|k) & z_6(k|k) & 0 & 0 & 0 & 2z_{12}(k|k) \\ 0 & 0 & 0 & 0 & 0 & z_6(k|k) & z_7(k|k) & 0 & 0 & 0 & -2z_{12}(k|k) \\ 0 & 0 & z_9(k|k) & z_8(k|k) & 0 & 0 & 0 & z_7(k|k) & z_{12}(k|k) & 0 & 0 \\ 0 & 0 & -z_8(k|k) & z_9(k|k) & 0 & 0 & 0 & -z_{12}(k|k) & z_7(k|k) & 0 & 0 \\ 0 & 0 & z_{11}(k|k) & z_{10}(k|k) & 0 & 0 & 0 & z_{12}(k|k) & z_6(k|k) & 0 & 0 \\ 0 & 0 & -z_{10}(k|k) & z_{11}(k|k) & 0 & 0 & 0 & -z_6(k|k) & z_{12}(k|k) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -z_{12}(k|k) & z_{12}(k|k) & 0 & 0 & 0 & z_7(k|k) - z_6(k|k) \end{bmatrix} \quad (\text{D.52})$$

D.2.2.2 Équations d'évolution des états d'un amer point

Le nouveau vecteur associé à un amer point, \mathcal{Z}_{pi} , introduit des états liant la position du véhicule avec la position d'un amer. La carte estimée est considérée comme statique mais le véhicule se déplace dans l'environnement cartographié. La nouvelle écriture du vecteur \mathcal{Z}_{pi} impose donc l'évolution d'une partie de ces états telle que :

$$\mathcal{Z}_{pi}(k+1|k) = \begin{bmatrix} z_{pi1}(k|k) \\ z_{pi2}(k|k) \\ z_{pi3}(k|k) \cdot u_4 + z_{pi4}(k|k) \cdot u_3 \\ z_{pi4}(k|k) \cdot u_4 - z_{pi3}(k|k) \cdot u_3 \\ z_{pi5}(k|k) \cdot u_4 + z_{pi6}(k|k) \cdot u_3 \\ z_{pi6}(k|k) \cdot u_4 - z_{pi5}(k|k) \cdot u_3 \end{bmatrix} \quad (\text{D.53})$$

Le calcul de la matrice de covariance est réalisé de la même manière que pour le calcul de la matrice de covariance associée au vecteur d'état du véhicule.

$$\mathbf{P}_{pi}(k+1|k) = \mathbf{J}_{\mathcal{Z}_{pi}} \cdot \mathbf{P}_{pi}(k|k) \cdot (\mathbf{J}_{\mathcal{Z}_{pi}})^T + \mathbf{J}_{\mathcal{U}_{pi}} \cdot \mathbf{Q}_{\mathcal{U}} \cdot (\mathbf{J}_{\mathcal{U}_{pi}})^T \quad (\text{D.54})$$

La matrice jacobienne $\mathbf{J}_{\mathcal{Z}_{pi}}$, dérivée de l'équation (D.53) par rapport à \mathcal{Z}_{pi} , s'écrit :

$$\mathbf{J}_{\mathcal{Z}_{pi}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & u_4 & u_3 & 0 & 0 \\ 0 & 0 & -u_3 & u_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & u_4 & u_3 \\ 0 & 0 & 0 & 0 & -u_3 & u_4 \end{bmatrix} \quad (\text{D.55})$$

La matrice jacobienne $\mathbf{J}_{\mathcal{U}_{pi}}$, dérivée de l'équation (D.53) par rapport au vecteur de commande \mathcal{U} (formule (D.51)) s'écrit :

$$\mathbf{J}_{\mathcal{U}_{pi}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z_{pi4}(k|k) & z_{pi3}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -z_{pi3}(k|k) & z_{pi4}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z_{pi6}(k|k) & z_{pi5}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -z_{pi5}(k|k) & z_{pi6}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.56})$$

D.2.2.3 Équations d'évolution des états d'un amer droite

A l'instar des points, le vecteur d'état \mathcal{Z}_{li} définissant les droites est également composé d'informations liées à la position du véhicule. Ce vecteur est donc modifié lors de l'étape de prédiction du filtre de Kalman :

$$\mathcal{Z}_{li}(k+1|k) = \begin{bmatrix} z_{li1}(k|k) \\ z_{li2}(k|k) \\ z_{li3}(k|k) \\ z_{li4}(k|k) \\ z_{li5}(k|k) \\ z_{li6}(k|k) \\ z_{li7}(k|k) \\ z_{li8}(k|k) \\ z_{li9}(k|k) \\ z_{li10}(k|k) + z_{li16}(k|k).u_1 \\ z_{li11}(k|k) + z_{li17}(k|k).u_1 \\ z_{li12}(k|k) + z_{li14}(k|k).u_1 \\ z_{li13}(k|k) + z_{li15}(k|k).u_1 \\ z_{li14}(k|k).u_4 + z_{li16}(k|k).u_3 \\ z_{li15}(k|k).u_4 + z_{li17}(k|k).u_3 \\ z_{li16}(k|k).u_4 - z_{li14}(k|k).u_3 \\ z_{li17}(k|k).u_4 - z_{li15}(k|k).u_3 \end{bmatrix} \quad (\text{D.57})$$

La matrice de covariance, \mathbf{P}_{li} , associée au vecteur \mathcal{Z}_{li} , est calculée avec la formule :

$$\mathbf{P}_{li}(k+1|k) = \mathbf{J}_{\mathcal{Z}_{li}} \cdot \mathbf{P}_{li}(k|k) \cdot (\mathbf{J}_{\mathcal{Z}_{li}})^T + \mathbf{J}_{\mathcal{U}_{li}} \cdot \mathbf{Q}_{\mathcal{U}} \cdot (\mathbf{J}_{\mathcal{U}_{li}})^T \quad (\text{D.58})$$

La matrice jacobienne $\mathbf{J}_{\mathcal{Z}_{li}}$, dérivée de l'équation (D.57) par rapport à $\mathcal{Z}_{li}(k)$, s'écrit :

$$\mathbf{J}_{\mathcal{Z}_{li}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.59})$$

La matrice jacobienne $\mathbf{J}_{\mathcal{U}_{li}}$, dérivée de l'équation (D.57) par rapport vecteur de commande \mathcal{U} , s'écrit :

$$\mathbf{J}_{\mathcal{U}_{li}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_{li16}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_{li17}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_{li14}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_{li15}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z_{li16}(k|k) & z_{li14}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z_{li17}(k|k) & z_{li15}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -z_{li14}(k|k) & z_{li16}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -z_{li15}(k|k) & z_{li17}(k|k) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.60})$$

D.2.3 Equations d'observation

La définition de nouveaux états permet une reformulation des équations d'observation en somme de polynômes. Ces équations sont désormais fonction des états du vecteur d'état du système \mathcal{Z} . Cette nouvelle décomposition réduit les effets dus à la non-linéarité du système.

D.2.3.1 Equation d'observation d'un amer point

La fonction d'observation associée à un amer de type point, \mathcal{Z}_{pi} , s'écrit :

$$\mathbf{X}_{pi}(k+1|k) = \mathbf{h}_{pi}(\mathcal{Z}(k+1|k)) = \begin{bmatrix} z_{pi4}(k+1|k) + z_{pi5}(k+1|k) - z_9(k+1|k) - z_{10}(k+1|k) \\ -z_{pi3}(k+1|k) + z_{pi6}(k+1|k) + z_8(k+1|k) - z_{11}(k+1|k) \end{bmatrix} \quad (\text{D.61})$$

Maintenant, l'équation d'observation (D.61) est composée par une somme d'états ou polynômes. La matrice de covariance associée à la prédiction de l'observation de l'amer point \mathbf{X}_{pi} s'écrit :

$$\mathbf{C}_{X_{pi}}(k+1|k) = \mathbf{H}_{X_{pi}} \cdot \mathbf{P}(k+1|k) \cdot (\mathbf{H}_{X_{pi}})^T \quad (\text{D.62})$$

La matrice jacobienne utilisée dans l'équation (D.62), $\mathbf{H}_{X_{li}}$, s'écrit :

$$\mathbf{H}_{X_{li}} = \begin{bmatrix} \mathbf{H}_{Z_v} & 0 & \cdots & 0 & \mathbf{H}_{Z_{pi}} & 0 & \cdots & 0 \end{bmatrix} \quad (\text{D.63})$$

Cette jacobienne (D.63) se décompose en deux jacobienes, \mathbf{H}_{Z_v} et $\mathbf{H}_{Z_{pi}}$. Ces matrices sont calculées respectivement par rapport au vecteur d'état du véhicule, $\mathcal{Z}_v(k+1|k)$, et par rapport au vecteur d'état de l'amer considéré $\mathcal{Z}_{pi}(k+1|k)$. La première matrice jacobienne \mathbf{H}_{Z_v} s'écrit :

$$\mathbf{H}_{Z_v} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix} \quad (\text{D.64})$$

La deuxième matrice jacobienne $\mathbf{H}_{Z_{pi}}$, dérivée de l'équation d'observation (D.62) par rapport au vecteur d'état associée au point, $\mathcal{Z}_{pi}(k+1|k)$, se formule :

$$\mathbf{H}_{Z_{pi}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \quad (\text{D.65})$$

D.2.3.2 Equation d'observation d'un amer droite

La mesure de la droite observée est décrite par le vecteur \mathbf{X}_{lm} avec une matrice de covariance associée \mathbf{C}_{lm} . L'écriture de l'observation \mathbf{X}_{li} en fonction du vecteur d'état du système \mathcal{Z} et du vecteur de la mesure \mathbf{X}_{lm} , est donnée par :

$$\mathbf{X}_{li}(k+1|k) = \mathbf{h}_{li}(\mathcal{Z}, \mathbf{X}_{lm}) = \begin{bmatrix} x_{lm} \cdot (z_{li14}(k+1|k) + z_{li17}(k+1|k)) + y_{lm} \cdot (z_{li16}(k+1|k) - z_{li15}(k+1|k)) + \cdots \\ \cdots + z_{li11}(k+1|k) - z_{li7}(k+1|k) + z_{li12}(k+1|k) - z_{li8}(k+1|k) \\ x_{lm} \cdot (z_{li15}(k+1|k) - z_{li16}(k+1|k)) + y_{lm} \cdot (z_{li17}(k+1|k) + z_{li14}(k+1|k)) + \cdots \\ \cdots + z_{li13}(k+1|k) - z_{li9}(k+1|k) - z_{li10}(k+1|k) + z_{li6}(k+1|k) \\ \theta_{lm} - z_{li3}(k+1|k) + z_3(k+1|k) \end{bmatrix} \quad (\text{D.66})$$

Le calcul de la matrice de covariance associée à la prédiction de l'observation de l'amer droite $\mathbf{X}_{li}(k+1|k)$ s'écrit :

$$\mathbf{C}_{X_{li}}(k+1|k) = \mathbf{H}_{X_{li}} \cdot \mathbf{P}(k+1|k) \cdot (\mathbf{H}_{X_{li}})^T + \mathbf{H}_{X_{lm}} \cdot \mathbf{C}_{lm} \cdot (\mathbf{H}_{X_{lm}})^T \quad (\text{D.67})$$

La matrice jacobienne $\mathbf{H}_{X_{lm}}$ correspond aux dérivées partielles des équations d'observation d'une droite par rapport au vecteur d'état de la mesure, \mathbf{X}_{lm} :

$$\mathbf{H}_{X_{lm}} = \begin{bmatrix} (z_{li14}(k+1|k) + z_{li17}(k+1|k)) & (z_{li16}(k+1|k) - z_{li15}(k+1|k)) & 0 \\ (z_{li15}(k+1|k) - z_{li16}(k+1|k)) & (z_{li14}(k+1|k) + z_{li17}(k+1|k)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{D.68})$$

La matrice jacobienne du calcul précédent, $\mathbf{H}_{X_{li}}$, s'écrit :

$$\mathbf{H}_{X_{li}} = [\mathbf{H}_{Z_V} \quad 0 \quad \cdots \quad 0 \quad \mathbf{H}_{Z_{li}} \quad 0 \quad \cdots \quad 0] \quad (\text{D.69})$$

La matrice jacobienne de l'équation d'observation (D.67) par rapport au vecteur Z_V , s'écrit :

$$\mathbf{H}_{Z_V} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.70})$$

La matrice jacobienne de l'équation d'observation (D.67) par rapport au vecteur Z_{li} , s'écrit :

$$\mathbf{H}_{Z_{li}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & 1 & 0 & x_{lm} & -y_{lm} & y_{lm} & x_{lm} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & y_{lm} & x_{lm} & -x_{lm} & y_{lm} \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.71})$$

D.2.3.3 Conclusion sur la nouvelle formulation des observations

L'intérêt principal de ce filtre réside dans l'utilisation des nouveaux états. Avec cette formulation, les équations d'observation (D.61) et (D.66) sont désormais linéaires par rapport au vecteur d'état du système \mathcal{Z} . Cette nouvelle écriture permet d'isoler les effets non-linéaires des équations de mise à jour des états et ainsi d'améliorer les estimations des incertitudes sur les positions des éléments de la carte.

D.2.4 Equations de cartographie d'un nouvel amer

Les équations présentées dans cette partie diffèrent de celles de l'approche classique EKF-SLAM (cf. la chapitre 2). Désormais, certains états, décrivant la position des amers, sont liés aux états du véhicule. Comme les vecteurs d'états sont agrandis, les équations de cartographie des amers point et droite doivent être modifiées.

D.2.4.1 Equations de cartographie d'un amer point

La fonction de cartographie d'un amer correspond à la translation des coordonnées de l'amer mesuré de la position du véhicule vers le repère de la carte estimée. L'ajout d'un nouvel amer \mathbf{X}_{pn} , dans la carte estimée \mathcal{Z} , utilise donc la position du véhicule Z_V . Les états définissant ce

nouvel amer \mathcal{Z}_{pn} sont obtenus par les équations :

$$\mathcal{Z}_{pn} = \mathbf{g}(\mathcal{Z}, \mathbf{X}_{pn}) = \begin{bmatrix} x_{pn} \cdot z_5 - y_{pn} \cdot z_4 + z_1 \\ x_{pn} \cdot z_4 + y_{pn} \cdot z_5 + z_2 \\ -y_{pn} \cdot z_6 + x_{pn} \cdot z_{12} + z_8 \\ x_{pn} \cdot z_7 - y_{pn} \cdot z_{12} + z_9 \\ x_{pn} \cdot z_6 + y_{pn} \cdot z_{12} + z_{10} \\ y_{pn} \cdot z_7 + x_{pn} \cdot z_{12} + z_{11} \end{bmatrix} \quad (\text{D.72})$$

La matrice de covariance du système, \mathbf{P}_{aug} , après l'insertion d'un nouvel amer point, est calculée à partir des matrices de covariance du système \mathbf{P} et de la mesure \mathbf{C}_{pn} :

$$\mathbf{P}_{aug} = \mathbf{G} \cdot \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{C}_{pn} \end{bmatrix} \cdot (\mathbf{G})^T \quad (\text{D.73})$$

La matrice jacobienne \mathbf{G} correspond à :

$$\mathbf{G} = \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{G}_Z & \mathbf{G}_{pn} \end{bmatrix} \quad (\text{D.74})$$

\mathbf{G}_Z est la matrice jacobienne de l'équation (3.72) par rapport au vecteur d'état \mathcal{Z} :

$$\mathbf{G}_Z = \begin{bmatrix} 1 & 0 & -y_{pn} & x_{pn} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & x_{pn} & y_{pn} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -y_{pn} & 0 & 1 & 0 & 0 & 0 & x_{pn} \\ 0 & 0 & 0 & 0 & 0 & x_{pn} & 0 & 1 & 0 & 0 & -y_{pn} \\ 0 & 0 & 0 & 0 & x_{pn} & 0 & 0 & 0 & 1 & 0 & y_{pn} \\ 0 & 0 & 0 & 0 & 0 & y_{pn} & 0 & 0 & 0 & 1 & x_{pn} \end{bmatrix} \quad (\text{D.75})$$

\mathbf{G}_{pn} est la matrice jacobienne de l'équation (3.72) par rapport au vecteur \mathbf{X}_{pn} :

$$\mathbf{G}_{pn} = \begin{bmatrix} z_4 & -z_3 \\ z_3 & z_4 \\ z_{11} & -z_5 \\ z_6 & -z_{11} \\ z_5 & z_{11} \\ z_{11} & z_6 \end{bmatrix} \quad (\text{D.76})$$

D.2.4.2 Equations de cartographie d'un amer droite

Soit un nouvel amer de type droite, avec le vecteur \mathbf{X}_{ln} et la matrice de covariance associée \mathbf{C}_{ln} . L'insertion de ce nouvel amer correspond à la fonction de cartographie :

$$\mathcal{Z}_{ln} = \mathbf{g}(\mathbf{X}_{ln}, \mathcal{Z}) = \begin{bmatrix} x_{ln} \cdot z_5 - y_{ln} \cdot z_4 + z_1 \\ x_{ln} \cdot z_4 + y_{ln} \cdot z_5 + z_2 \\ \theta_{ln} + z_3 \\ \cos(\theta_{ln}) \cdot z_4 + \sin(\theta_{ln}) \cdot z_5 \\ \cos(\theta_{ln}) \cdot z_5 - \sin(\theta_{ln}) \cdot z_4 \\ x_{ln} \cdot (\cos(\theta_{ln}) \cdot z_{12} + \sin(\theta_{ln}) \cdot z_7) - y_{ln} \cdot (\cos(\theta_{ln}) \cdot z_6 + \sin(\theta_{ln}) \cdot z_{12}) + \cos(\theta_{ln}) \cdot z_8 + \sin(\theta_{ln}) \cdot z_9 \\ x_{ln} \cdot (\cos(\theta_{ln}) \cdot z_7 - \sin(\theta_{ln}) \cdot z_{12}) + y_{ln} \cdot (\sin(\theta_{ln}) \cdot z_6 - \cos(\theta_{ln}) \cdot z_{12}) + \cos(\theta_{ln}) \cdot z_9 - \sin(\theta_{ln}) \cdot z_8 \\ x_{ln} \cdot (\cos(\theta_{ln}) \cdot z_6 + \sin(\theta_{ln}) \cdot z_{12}) + y_{ln} \cdot (\sin(\theta_{ln}) \cdot z_7 + \cos(\theta_{ln}) \cdot z_{12}) + \cos(\theta_{ln}) \cdot z_{10} + \sin(\theta_{ln}) \cdot z_{11} \\ x_{ln} \cdot (\cos(\theta_{ln}) \cdot z_{12} - \sin(\theta_{ln}) \cdot z_6) + y_{ln} \cdot (\cos(\theta_{ln}) \cdot z_7 - \sin(\theta_{ln}) \cdot z_{12}) + \cos(\theta_{ln}) \cdot z_{11} - \sin(\theta_{ln}) \cdot z_{10} \\ \cos(\theta_{ln}) \cdot z_8 + \sin(\theta_{ln}) \cdot z_9 \\ \cos(\theta_{ln}) \cdot z_9 - \sin(\theta_{ln}) \cdot z_8 \\ \cos(\theta_{ln}) \cdot z_{10} + \sin(\theta_{ln}) \cdot z_{11} \\ \cos(\theta_{ln}) \cdot z_{11} - \sin(\theta_{ln}) \cdot z_{10} \\ \cos(\theta_{ln}) \cdot z_6 + \sin(\theta_{ln}) \cdot z_{12} \\ \cos(\theta_{ln}) \cdot z_{12} - \sin(\theta_{ln}) \cdot z_6 \\ \cos(\theta_{ln}) \cdot z_{12} + \sin(\theta_{ln}) \cdot z_7 \\ \cos(\theta_{ln}) \cdot z_7 - \sin(\theta_{ln}) \cdot z_{12} \end{bmatrix} \quad (\text{D.77})$$

Dans le but de simplifier les calculs, le vecteur de mesure, \mathbf{X}_{ln} , est augmentée. Cette réécriture permet d'inclure le cosinus et le sinus de l'orientation du repère associé à la mesure de la droite, θ_{ln} . Le vecteur de mesure agrandi, \mathcal{X}_{ln} , est obtenu par les équations :

$$\mathcal{X}_{ln} = \begin{bmatrix} x_{ln1} \\ x_{ln2} \\ x_{ln3} \\ x_{ln4} \\ x_{ln5} \\ x_{ln6} \\ x_{ln7} \\ x_{ln8} \\ x_{ln9} \end{bmatrix} = \begin{bmatrix} x_{ln} \\ y_{ln} \\ \theta_{ln} \\ \sin(\theta_{ln}) \\ \cos(\theta_{ln}) \\ x_{ln} \cdot \sin(\theta_{ln}) \\ x_{ln} \cdot \cos(\theta_{ln}) \\ y_{ln} \cdot \sin(\theta_{ln}) \\ y_{ln} \cdot \cos(\theta_{ln}) \end{bmatrix} \quad (\text{D.78})$$

La matrice de covariance $\mathbf{C}_{\mathcal{X}_{ln}}$, associée au vecteur \mathcal{X}_{ln} est obtenue par les calculs suivants :

$$\mathbf{C}_{\mathcal{X}_{ln}} = \mathbf{J}_{\mathcal{X}_{ln}} \cdot \mathbf{C}_{ln} \cdot (\mathbf{J}_{\mathcal{X}_{ln}})^T \quad (\text{D.79})$$

La matrice jacobienne de l'équation (D.79) s'écrit :

$$\mathbf{J}_{\mathcal{X}_{l_n}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \cos(\theta_{l_n}) \\ 0 & 0 & -\sin(\theta_{l_n}) \\ \sin(\theta_{l_n}) & 0 & x_{l_n} \cdot \cos(\theta_{l_n}) \\ \cos(\theta_{l_n}) & 0 & -x_{l_n} \cdot \sin(\theta_{l_n}) \\ 0 & \sin(\theta_{l_n}) & y_{l_n} \cdot \cos(\theta_{l_n}) \\ 0 & \cos(\theta_{l_n}) & -y_{l_n} \cdot \sin(\theta_{l_n}) \end{bmatrix} \quad (\text{D.80})$$

La définition du vecteur \mathcal{X}_{l_n} impose une réécriture des équations de cartographie (D.77) :

$$\mathcal{Z}_{l_n} = \mathbf{g}(\mathcal{X}_{l_n}, \mathcal{Z}) = \begin{bmatrix} x_{l_n1} \cdot z_5 - x_{l_n2} \cdot z_4 + z_1 \\ x_{l_n1} \cdot z_4 + x_{l_n2} \cdot z_5 + z_2 \\ x_{l_n3} + z_3 \\ x_{l_n5} \cdot z_4 + x_{l_n4} \cdot z_5 \\ x_{l_n5} \cdot z_5 - x_{l_n4} \cdot z_4 \\ x_{l_n7} \cdot z_{12} + x_{l_n6} \cdot z_7 - x_{l_n9} \cdot z_6 - x_{l_n8} \cdot z_{12} + x_{l_n5} \cdot z_8 + x_{l_n4} \cdot z_9 \\ x_{l_n7} \cdot z_7 - x_{l_n6} \cdot z_{12} + x_{l_n8} \cdot z_6 - x_{l_n9} \cdot z_{12} + x_{l_n5} \cdot z_9 - x_{l_n4} \cdot z_8 \\ x_{l_n7} \cdot z_6 + x_{l_n6} \cdot z_{12} + x_{l_n8} \cdot z_7 + x_{l_n9} \cdot z_{12} + x_{l_n5} \cdot z_{10} + x_{l_n4} \cdot z_{11} \\ x_{l_n7} \cdot z_{12} - x_{l_n6} \cdot z_6 + x_{l_n9} \cdot z_7 - x_{l_n8} \cdot z_{12} + x_{l_n5} \cdot z_{11} - x_{l_n4} \cdot z_{10} \\ x_{l_n5} \cdot z_8 + x_{l_n4} \cdot z_9 \\ x_{l_n5} \cdot z_9 - x_{l_n4} \cdot z_8 \\ x_{l_n5} \cdot z_{10} + x_{l_n4} \cdot z_{11} \\ x_{l_n5} \cdot z_{11} - x_{l_n4} \cdot z_{10} \\ x_{l_n5} \cdot z_6 + x_{l_n4} \cdot z_{12} \\ x_{l_n5} \cdot z_{12} - x_{l_n4} \cdot z_6 \\ x_{l_n5} \cdot z_{12} + x_{l_n4} \cdot z_7 \\ x_{l_n5} \cdot z_7 - x_{l_n4} \cdot z_{12} \end{bmatrix} \quad (\text{D.81})$$

L'insertion du nouvel amer droite, \mathcal{Z}_{l_n} , dans le vecteur \mathcal{Z} impose une augmentation de la matrice de covariance \mathbf{P} telle que :

$$\mathbf{P}_{aug} = \mathbf{G}_{\mathcal{Z}_L} \cdot \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \mathbf{C}_{\mathcal{X}_{l_n}} \end{bmatrix} \cdot (\mathbf{G}_{\mathcal{Z}_L})^T \quad (\text{D.82})$$

La matrice jacobienne $\mathbf{G}_{\mathcal{Z}_L}$ se décompose en plusieurs matrices jacobiennes tel que :

$$\mathbf{G}_{\mathcal{Z}_L} = \begin{bmatrix} I & 0 \\ \mathbf{G}_{\mathcal{Z}_V} & \mathbf{G}_{\mathcal{Z}_{l_n}} \end{bmatrix} \quad (\text{D.83})$$

\mathbf{G}_{Z_V} est la matrice jacobienne de l'équation (D.80) par rapport au vecteur d'état Z_V :

$$\mathbf{G}_{Z_V} = \begin{bmatrix} 1 & 0 & 0 & -z_{ln2} & z_{ln1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & z_{ln1} & z_{ln2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_{ln5} & z_{ln4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_{ln4} & z_{ln5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -z_{ln9} & z_{ln6} & z_{ln5} & z_{ln4} & 0 & 0 & z_{ln7} - z_{ln8} \\ 0 & 0 & 0 & 0 & 0 & z_{ln8} & z_{ln7} & -z_{ln4} & z_{ln5} & 0 & 0 & -z_{ln6} - z_{ln9} \\ 0 & 0 & 0 & 0 & 0 & z_{ln7} & z_{ln8} & 0 & 0 & z_{ln5} & z_{ln4} & z_{ln6} + z_{ln9} \\ 0 & 0 & 0 & 0 & 0 & -z_{ln6} & z_{ln9} & 0 & 0 & -z_{ln4} & z_{ln5} & z_{ln7} - z_{ln8} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & z_{ln5} & z_{ln4} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -z_{ln4} & z_{ln5} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & z_{ln5} & z_{ln4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -z_{ln4} & z_{ln5} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & z_{ln5} & 0 & 0 & 0 & 0 & 0 & z_{ln4} \\ 0 & 0 & 0 & 0 & 0 & -z_{ln4} & 0 & 0 & 0 & 0 & 0 & z_{ln5} \\ 0 & 0 & 0 & 0 & 0 & 0 & z_{ln4} & 0 & 0 & 0 & 0 & z_{ln5} \\ 0 & 0 & 0 & 0 & 0 & 0 & z_{ln5} & 0 & 0 & 0 & 0 & -z_{ln4} \end{bmatrix} \quad (\text{D.84})$$

$\mathbf{G}_{Z_{ln}}$ est la matrice jacobienne de l'équation (D.80) par rapport au vecteur Z_{ln} :

$$\mathbf{G}_{Z_{ln}} = \begin{bmatrix} z_5 & -z_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_4 & z_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_5 & z_4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_4 & z_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_9 & z_8 & z_7 & z_{12} & -z_{12} & -z_6 & 0 \\ 0 & 0 & 0 & -z_8 & z_9 & -z_{12} & z_7 & z_6 & -z_{12} & 0 \\ 0 & 0 & 0 & z_{11} & z_{10} & z_{12} & z_6 & z_7 & z_{12} & 0 \\ 0 & 0 & 0 & -z_{10} & z_{11} & -z_6 & z_{12} & -z_{12} & z_7 & 0 \\ 0 & 0 & 0 & z_9 & z_8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_8 & z_9 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_{11} & z_{10} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_{10} & z_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_{12} & z_6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_6 & z_{12} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_7 & z_{12} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_{12} & z_7 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.85})$$

D.2.4.3 Remarque sur la nouvelle formulation de la cartographie des amers

L'étape de cartographie permet l'insertion d'un nouvel amer dans la carte estimée en agrandissant le vecteur Z et la matrice de covariance \mathbf{P} . Lors de cette phase, les corrélations entre les différents états du système sont initialisées. Ces corrélations jouent un rôle essentiel lors de la

mise à jour de la carte. Elles permettent la correction de l'ensemble de la carte à partir d'une seule mesure.

L'étude des équations de cartographie d'un point (3.72) et d'une droite (D.81) permet de souligner un fait important. Les états du véhicule, \mathcal{Z}_V , ne sont pas tous corrélés aux états de l'amer cartographié. Par exemple, les états z_1 à z_5 ne sont corrélés qu'avec les états z_{pi1} à z_{pi2} pour un point et z_{li1} à z_{li5} pour une droite. Des remarques similaires peuvent être formulées pour les autres états de ces vecteurs. Comme le prouvent les résultats des simulations (cf. le chapitre 2), ces différentes corrélations permettent de réduire certains problèmes posés par la non-linéarité des équations d'un système SLAM.

D.3 PEKF-SLAM : étape d'association de données

Pour permettre une meilleure association des données en se basant sur une estimation des l'incertitude sur la position des amers plus cohérentes, les équations d'observation des amers de type cercle ou ligne sont reformulées. Dans la suite de cette annexe, les formules permettant cette reformulation sont présentées.

Reformulation de l'équation d'observation d'un amer point :

La fonction d'observation (D.61) est réécrite pour être fonction des états z_1 à z_5 du véhicule et z_{pi1} et z_{pi2} . La fonction finale d'observation d'un amer de type point s'écrit :

$$\mathbf{X}_{pi}(k+1|k) = \begin{bmatrix} (z_{pi1} \cdot z_5) + (z_{pi2} \cdot z_4) - (z_1 \cdot z_5) - (z_2 \cdot z_4) \\ (z_{pi2} \cdot z_5) - (z_{pi1} \cdot z_4) + (z_1 \cdot z_4) - (z_2 \cdot z_5) \end{bmatrix} \quad (\text{D.86})$$

Grâce à cette nouvelle formulation, la matrice de covariance associée à $\mathbf{X}_{pi}(k+1|k)$ sera calculée à partir des incertitudes dont les valeurs sont proches du cas idéal.

$$\mathbf{C}_{X_{pi}}(k+1|k) = \mathbf{H}_{X_{pi}} \cdot \mathbf{P}(k+1|k) \cdot (\mathbf{H}_{X_{pi}})^T \quad (\text{D.87})$$

La matrice jacobienne utilisée dans l'équation (D.62), $\nabla H_{X_{li}}$, s'écrit :

$$\mathbf{H}_{X_{li}} = [\mathbf{H}_{Z_v} \quad 0 \quad \cdots \quad 0 \quad \mathbf{H}_{Z_{pi}} \quad 0 \quad \cdots \quad 0] \quad (\text{D.88})$$

Les nouvelles matrices jacobienes, \mathbf{H}_{Z_v} et $\mathbf{H}_{Z_{pi}}$ s'écrivent :

$$\mathbf{H}_{Z_v} = \begin{bmatrix} -z_5 & -z_4 & 0 & -z_2 + z_{pi2} & -z_1 + z_{pi1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ z_4 & -z_5 & 0 & z_1 - z_{pi1} & -z_2 + z_{pi2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.89})$$

et

$$\mathbf{H}_{Z_{pi}} = \begin{bmatrix} z_5 & z_4 & 0 & 0 & 0 & 0 \\ -z_4 & z_5 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{D.90})$$

Le même raisonnement s'applique aux amers de type droite.

Reformulation de l'équation d'observation d'un amer ligne :

$$\mathbf{h}_{li}(\mathcal{Z}, \mathbf{X}_{lm}) = \begin{bmatrix} x_{lm} \cdot z_5 \cdot z_{li5} + x_{lm} \cdot z_4 \cdot z_{li4} - y_{lm} \cdot z_4 \cdot z_{li5} + y_{lm} \cdot z_5 \cdot z_{li4} + z_1 \cdot z_{li5} + \dots \\ \dots + z_2 \cdot z_{li4} - z_{li1} \cdot z_{li5} - z_{li2} \cdot z_{li4} \\ x_{lm} \cdot z_4 \cdot z_{li5} - x_{lm} \cdot z_5 \cdot z_{li4} + y_{lm} \cdot z_5 \cdot z_{li5} + y_{lm} \cdot z_4 \cdot z_{li4} - z_1 \cdot z_{li4} + \dots \\ \dots + z_2 \cdot z_{li5} + z_{li1} \cdot z_{li4} - z_{li2} \cdot z_{li5} \\ \theta_{lm} - z_{li3} + z_3 \end{bmatrix} \quad (\text{D.91})$$

Le calcul de la matrice de covariance associée au vecteur (D.91) s'écrit maintenant en fonction des incertitudes sur les positions des éléments de la carte tel que :

$$\mathbf{C}_{X_{li}}(k+1|k) = \mathbf{H}_{X_{li}} \cdot \mathbf{P}(k+1|k) \cdot (\mathbf{H}_{X_{li}})^T + \mathbf{H}_{X_{lm}} \cdot \mathbf{C}_{lm} \cdot (\mathbf{H}_{X_{lm}})^T \quad (\text{D.92})$$

La matrice jacobienne $\mathbf{H}_{X_{lm}}$ associée à la mesure \mathbf{X}_{lm} s'écrit :

$$\mathbf{H}_{X_{lm}} = \begin{bmatrix} (z_5 \cdot z_{li5} + z_4 \cdot z_{li4}) & (z_5 \cdot z_{li4} - z_4 \cdot z_{li5}) & 0 \\ (z_4 \cdot z_{li5} - z_5 \cdot z_{li4}) & (z_5 \cdot z_{li5} + z_4 \cdot z_{li4}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{D.93})$$

La matrice jacobienne de la fonction \mathbf{h} par rapport au vecteur d'état du système \mathcal{Z} s'écrit :

$$\mathbf{H}_{X_{li}} = [\mathbf{H}_{Z_V} \quad 0 \quad \dots \quad 0 \quad \mathbf{H}_{Z_{li}} \quad 0 \quad \dots \quad 0] \quad (\text{D.94})$$

Les deux matrices jacobiennes composant la matrice (D.94) s'écrivent :

$$\mathbf{H}_{Z_V} = \begin{bmatrix} z_{li5} & z_{li4} & 0 & x_{lm} \cdot z_{li4} - y_{lm} \cdot z_{li5} & x_{lm} \cdot z_{li5} + y_{lm} \cdot z_{li4} & 0 & \dots & 0 \\ -z_{li4} & z_{li5} & 0 & x_{lm} \cdot z_{li5} + y_{lm} \cdot z_{li4} & y_{lm} \cdot z_{li5} - x_{lm} \cdot z_{li4} & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad (\text{D.95})$$

et

$$\mathbf{H}_{Z_{li}} = \begin{bmatrix} -z_5 & -z_4 & 0 & x_{lm} \cdot z_4 + y_{lm} \cdot z_5 + z_2 - z_{li2} & x_{lm} \cdot z_5 - y_{lm} \cdot z_4 + z_1 - z_{li1} & 0 & \dots & 0 \\ z_4 & -z_5 & 0 & -x_{lm} \cdot z_5 + y_{lm} \cdot z_4 - z_1 + z_{li1} & x_{lm} \cdot z_4 + y_{lm} \cdot z_5 + z_2 - z_{li2} & 0 & \dots & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad (\text{D.96})$$

Cette approche PEKF-SLAM avec une association de données "améliorée" sera comparée à l'approche classique PEKF ainsi qu'à l'approche EKK SLAM à travers différentes simulations et expérimentations. L'objectif est de quantifier la précision et la consistance des estimations obtenues avec l'approche PEKF-SLAM dans des contextes de simulations plus compliqués. Ces contextes sont similaires à ceux employés dans le chapitre précédent. Ils permettent l'étude du comportement des estimées des différentes applications lors de la fermeture de la trajectoire du véhicule avec des cartes composées uniquement d'amers point ou des cartes hétérogènes composées d'amer de type ligne et de type cercle (considéré comme des amers de type point).

Annexe E

Simulateurs et contextes expérimentaux

Cette annexe présente successivement :

- les deux simulateurs employés dans mes travaux ;
- le véhicule expérimental avec ses capteurs embarqués (télémètre laser, capteurs odométriques et capteur GPS différentiel) ;
- les sites et la plateforme expérimentale où ont été recueillies les mesures.

E.1 Présentation des simulateurs

Deux simulateurs de données ont été implémentés et employés.

E.1.1 Simulateur de base

Ce simulateur a été inspiré des codes proposés par Tim Bailey [11]. Dans cette implémentation, les mesures correspondent à des positions cartésiennes de points, associées à des incertitudes sur leurs coordonnées (cf. la figure E.1). Aucune segmentation n'est réalisée pour obtenir ces positions d'amers.

Pour ces données simulées, le véhicule évolue à une vitesse constante de 3 ms^{-1} . Les commandes simulées de celui-ci correspondent à sa vitesse et à l'angle de braquage de ses roues avant. Pour simuler ces commandes, une modélisation de type tricycle (cf. l'annexe B) a été employée.

Les données proprioceptives et les mesures télémétriques sont bruitées. Les valeurs de ces bruits, pour les différentes simulations présentées dans ce manuscrit, sont résumées dans la tableau E.1.

E.1.2 Simulateur de cartes hétérogènes

Ce simulateur fournit des mesures similaires à celles obtenues dans un contexte réel (cf. la figure E.2). L'environnement est composé de formes géométriques simples : des rectangles et des cercles. De telles formes s'apparentent à des arbres et des bâtiments présents dans les environnements expérimentaux. L'objectif est de tester l'ensemble de notre système (extraction d'amers plus filtrage) avec une représentation hétérogène de l'environnement. Les valeurs des

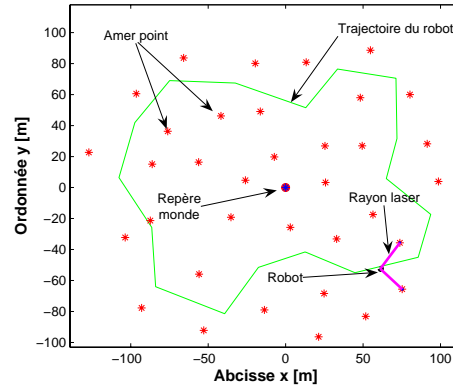


Figure E.1: Premier simulateur employé dans ces travaux : le véhicule est modélisé par un point et la carte est uniquement composée d'amers de type point

Valeurs des bruits				
Télémetre laser		données de contrôle		
	σ_ρ [m]	σ_β [rad]	σ_v [ms^{-1}]	σ_φ [rad]
Simulation	0.1	$\pi/180$	0.2	$3.\pi/180$

TABLE E.1: Valeur des bruits sur les données proprioceptives et extéroceptives : simulateur de base

bruits sur les mesures simulées, proprioceptives et extéroceptives, sont résumées dans le tableau E.2.

E.2 Contexte expérimental

Dans ces travaux de thèse, trois jeux de données différents ont été traités. Le véhicule expérimental utilisé, ainsi que les capteurs embarqués, sont présentés dans le premier paragraphe. Ensuite, à partir de photos, sont présentés les trois environnements où ont été recueillies les données.

E.2.1 Véhicule expérimental : CYCAB

Le laboratoire est équipé d'une flotte de 5 véhicules expérimentaux, nommés CYCAB (cf. la figure E.3). Ces véhicules sont des prototypes de petits véhicules électriques. Ils sont équipés de quatre roues motrices, de capteurs embarqués proprioceptifs et extéroceptifs.

Les **capteurs proprioceptifs** correspondent à des encodeurs placés sur les essieux des quatre roues et sur l'essieu de direction des roues avant. Ils permettent d'obtenir respectivement les vitesses des quatre roues et la valeur de l'angle de braquage des roues avant. Ces

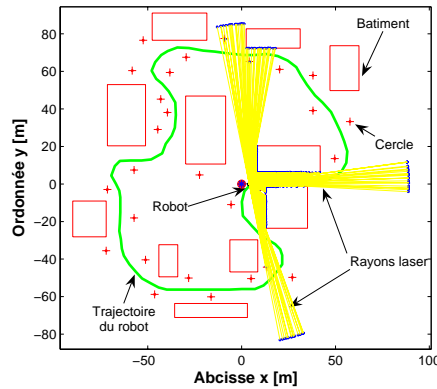


Figure E.2: Deuxième simulateur de cartes hétérogènes : le véhicule est représenté par un point et les mesures sont similaires à celles obtenues avec un capteur réel

Valeurs des bruits				
Télémètre laser		données de contrôle		
	σ_ρ [m]	σ_β [rad]	σ_v [ms^{-1}]	σ_φ [rad]
Simulation	0.1	$\pi/180$	0.02	$3.\pi/180$

TABLE E.2: Valeur des bruits sur les données proprioceptives et extéroceptives : simulateur de cartes hétérogènes



Figure E.3: Véhicule expérimental : CYCAB

encodeurs optiques incrémentaux mesurent le nombre de tour d'un essieu dans un intervalle de temps.

Le **capteur extéroceptif** employé dans mes travaux est un télémètre laser SICK LMS 221 [1]. Chaque mesure télémétrique fournit 361 impacts laser. Ces points sont représentés en coordonnées polaires. Ils sont repartis sur une plage allant de 0 à 180 degrés avec une résolution de 0.5 degrés. La portée maximale de ce capteur est de 80 mètres.

Les valeurs des bruits sur ces mesures, déterminées à partir d'une calibration, sont présentées dans le tableau E.3.

	Valeurs des bruits			
	Télémètre laser		données de contrôle	
	σ_ρ [m]	σ_β [rad]	σ_v [ms ⁻¹]	σ_φ [rad]
Données expérimentales	0.2	$\pi/36$	0.15	$\pi/360$

TABLE E.3: Valeur des bruits sur les données proprioceptives et extéroceptives des capteurs embarqués sur le véhicule expérimental

E.2.2 Présentation des lieux d'expérimentation

Dans ce manuscrit, trois types de données expérimentales sont utilisées pour décrire les performances des différents systèmes :

- environnement intérieur correspondant au hall de l'aéroport d'Aulnat (à proximité de Clermont Ferrand) ;
- environnement extérieur semi-structuré composé de bâtiments et d'arbres ;
- environnement extérieur correspondant à la plate forme expérimentale de PAVIN situé à proximité du LASMEA.

Hall de l'aéroport d'Aulnat :

Elle correspond à la première expérimentation réalisée lors de mes travaux. Le traitement de ces jeux de données a permis d'obtenir les premiers résultats de SLAM. La vérité de terrain (position du véhicule) n'était pas disponible. Pour qualifier les résultats obtenus, des photographies ainsi que le plan du hall de l'aéroport, furent utilisés.

La figure E.4 présente le contexte de l'expérimentation. Le véhicule expérimental a réalisé plusieurs parcours à l'intérieur du hall de l'aéroport. La trajectoire présentée dans ces travaux de thèse est représentée sur la figure E.5. Le point de départ, marqué en vert sur la figure, correspond aux deux photos (a) et (b) sur la figure E.4. Le véhicule a parcouru une distance d'environ 100 mètres.

Parc situé dans l'enceinte du Cemagref :

Ce deuxième jeu de données a été réalisé dans l'enceinte d'un parc, sur le campus universitaire. Ce lieu est principalement composé d'arbres et de bâtiments (cf. la figure E.6).

Le traitement de ces jeux de données représentent les limites de notre application au niveau de la représentation de l'environnement. En effet, l'environnement est difficilement modélisable en deux dimensions. Les résultats présentés dans le deuxième chapitre de cette thèse le prouvent.

Plate-forme expérimentale PAVIN :

La plate forme expérimentale de PAVIN a été construite en 2008. Ce site est une représentation d'un centre urbain avec des pans de murs, des feux de signalisations, des passages cloutés et un rond point (cf. la figure E.7 (a)). Cet environnement expérimental est particulièrement



Figure E.4: Manipulations réalisées à l'aéroport d'Aulnat : (a) position initiale du véhicule, (b) vue d'ensemble du hall, (c) accueil du hall et exemple d'obstacles rencontrés, (d) fuite du véhicule expérimental

propice à une modélisation en 2D de l'environnement sous une forme hétérogène. De surcroît, un plan d'architecte d'une partie des installations est disponible (cf. la figure E.7 (b)). La traduction de ce plan nous a permis de réaliser les tests de la méthode SLAM-Contraint présentée dans le quatrième chapitre.

Le plan de ce site expérimental est considéré comme imparfait. En d'autres termes, la construction des différents murs, représentant la mini ville, ne correspondent pas exactement au plan de l'architecte. Les valeurs de ces incertitudes, sur les tracés des bâtiments (amers de type droite), sont présentées dans le tableau E.4.

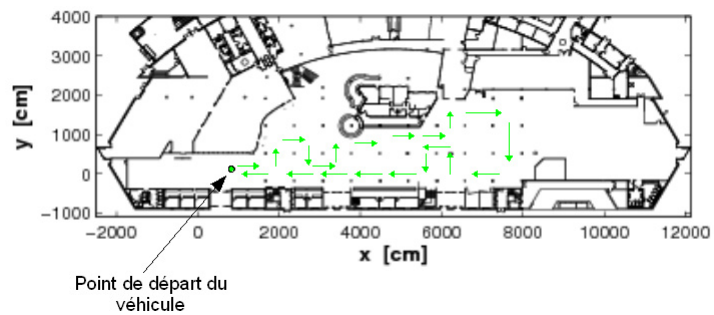


Figure E.5: *Plan de l'aéroport de Aulnat*



Figure E.6: *Vue aérienne de l'enceinte du CEMAGREF*

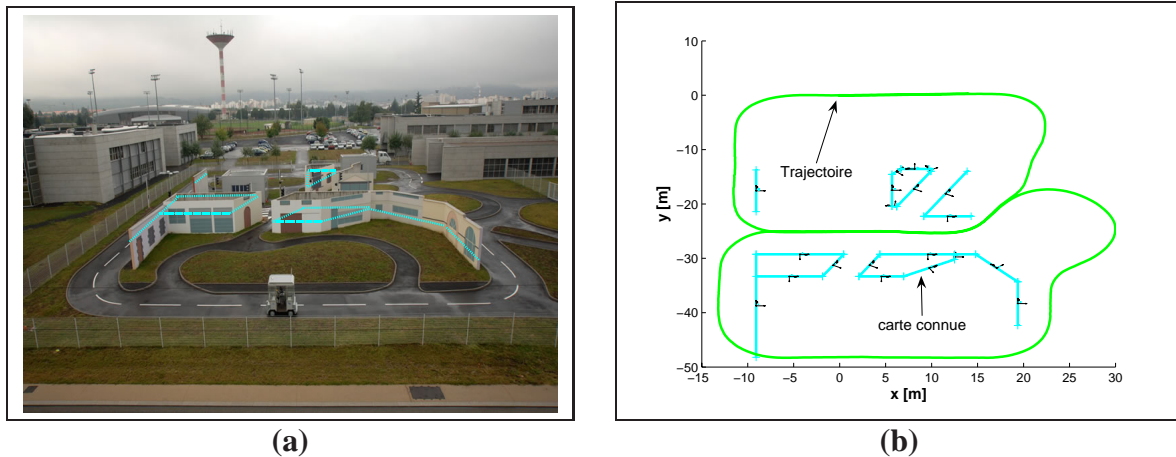


Figure E.7: Manipulations sur la plate-forme expérimentale de PAVIN : (a) photographie du site avec la carte connue de cet environnement (trait bleu), (b) trajectoire du véhicule (capteur DGPS) avec la carte connue de l'environnement (traits bleus)

Valeurs des écart-type sur les états des droites		
σ_x [m]	σ_y [m]	σ_{theta} [m]
0.2	0.2	$0.05 * \pi / 180$

TABLE E.4: Valeurs des incertitudes sur les tracés du plan d'architecte de la plate-forme expérimentale PAVIN

Annexe F

Méthodes SLAM fondées sur la théorie de Kalman

F.1 Introduction

Ce chapitre présente les principales variantes de la méthode SLAM fondée sur un filtrage non-linéaire de Kalman. Seules cinq approches, citées ou employées dans mes travaux, sont détaillées :

- approche *Worldcentric* [53],
- approche *Robotcentric* [30],
- approche *Iterated Extended Kalman Filter* [12],
- approche utilisant un filtre de kalman sans parfum ¹ (UKF) [88],
- approche multi-cartes [146].

Ces méthodes diffèrent soit par leur représentation de l'environnement, soit par leur méthode d'approximation des équations non-linéaires. Pour chacune de ces méthodes, leurs fonctions non-linéaires de prédiction, d'observation et de cartographie des amers de la carte sont présentées. Elles sont accompagnées d'explication sur les caractéristiques mathématiques liées à chacune d'elles.

F.2 Approche EKF-SLAM *Worldcentric*

L'approche EKF *Worldcentric* correspond à la première solution mise en oeuvre pour résoudre le problème de localisation et cartographie simultanées [53][142]. Elle a été introduite en même temps que l'acronyme SLAM.

F.2.1 Présentation

Dans cette approche, toutes les positions des éléments de la carte et du véhicule sont référencées par rapport à un repère global W_g (cf. la figure F.1). La position de ce repère correspond

1. Unscented Kalman Filter, ce filtre est également appelé Sigma Point Filter

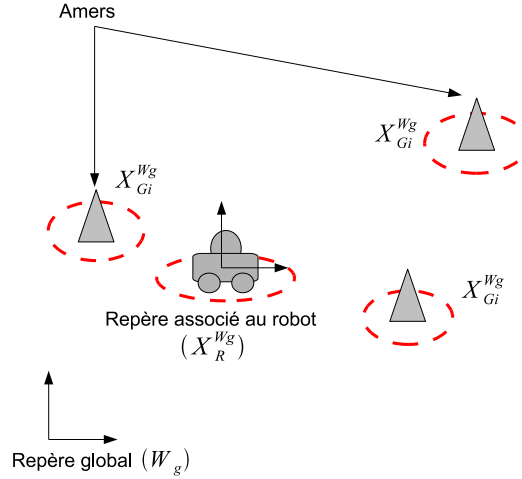


Figure F.1: Représentation de la carte et du robot : approche WorldCentric

généralement à la position initiale du véhicule. Le vecteur d'état du système s'écrit :

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_M \end{bmatrix} = \begin{bmatrix} \mathbf{X}_R^{W_g} \\ \mathbf{X}_{G_1}^{W_g} \\ \vdots \\ \mathbf{X}_{G_m}^{W_g} \end{bmatrix} \quad \text{et la matrice de covariance : } \mathbf{P} \quad (\text{F.1})$$

Les vecteurs $\mathbf{X}_R^{W_g}$ et \mathbf{X}_V correspondent à la position du véhicule dans la carte estimée. Les vecteurs $\mathbf{X}_{G_i}^{W_g}$ avec $i = \{1 \dots m\}$ correspondent aux positions des amers référencées dans le repère W_g .

F.2.2 Fonctions non linéaires régissant le système

F.2.2.1 Fonction d'évolution des états

L'environnement est toujours considéré comme statique. Comme tous les éléments sont représentés par rapport à un repère fixe, seuls les états associés au véhicule vont évoluer. La fonction d'évolution d'état, \mathbf{f} , s'écrit :

$$\mathbf{X}(k+1|k) = \mathbf{f}(\mathbf{X}(k|k), \mathbf{U}(k)) = \begin{bmatrix} \mathbf{f}_v(\mathbf{X}_V(k|k), \mathbf{U}(k)) + \mathbf{w}_v(k) \\ \mathbf{X}_M(k|k) \end{bmatrix} \quad (\text{F.2})$$

Le déplacement du véhicule est modélisé par la fonction \mathbf{f}_v (cf. l'annexe B). Le vecteur \mathbf{U} représente la commande sur le véhicule et \mathbf{w}_v , le bruit sur celle-ci.

F.2.2.2 Fonction d'observation d'un amer de type point ou cercle

Cette fonction non-linéaire correspond à la projection des amers exprimés dans le repère globale W_g vers un repère centré sur la position du véhicule. Elle est définie avec des opérations

dite de composition (cf. l'annexe B) :

$$\begin{aligned}
\mathbf{Z}(k+1|k) &= \mathbf{h}_i(\mathbf{X}_R^{W_g}(k+1|k), \mathbf{X}_{G_i}^{W_g}(k+1|k)) \\
&= \ominus \mathbf{X}_R^{W_g}(k+1|k) \oplus \mathbf{X}_{G_i}^{W_g}(k+1|k) \\
&= \mathbf{X}_{G_i}^R(k+1|k)
\end{aligned} \tag{F.3}$$

Le calcul de la matrice de covariance associée est donné par :

$$\mathbf{P}_Z(k+1|k) = \mathbf{H} \cdot \mathbf{P}(k+1|k) \cdot \mathbf{H}^T \tag{F.4}$$

\mathbf{H} correspond à la matrice jacobienne des fonctions de composition définies par les signes mathématiques \oplus et \ominus . Cette matrice s'écrit :

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{X_V} & 0 & \cdots & 0 & \mathbf{H}_{X_{G_i}} & 0 & \cdots & 0 \end{bmatrix} \tag{F.5}$$

$$\tag{F.6}$$

La matrice jacobienne \mathbf{H}_{X_V} associée à l'état du véhicule s'écrit :

$$\mathbf{H}_{X_V} = \begin{bmatrix} \mathbf{J}_{1\oplus}(\mathbf{X}_{W_g}^R, X_{G_i}^{W_g}) \cdot \mathbf{J}_{\ominus}(\mathbf{X}_R^{W_g}) \end{bmatrix} \tag{F.7}$$

La matrice jacobienne $\mathbf{H}_{X_{G_i}}$ associée à l'état de l'amer se formule :

$$\mathbf{H}_{X_{G_i}} = \begin{bmatrix} \mathbf{J}_{2\oplus}(\mathbf{X}_{W_g}^R, \mathbf{X}_{G_i}^{W_g}) \end{bmatrix} \tag{F.8}$$

F.2.2.3 Fonction d'observation d'un amer de type ligne

Dans une représentation cartésienne d'une droite, la formulation de la fonction d'observation prend en compte l'information angulaire apportée par celui-ci. Lorsque un amer de type ligne $\mathbf{X}_{G_i}^{W_g}$ est associé avec une mesure de ce même amer $\mathbf{X}_{L_j}^R$, la fonction d'observation s'écrit :

$$\mathbf{Z}(k+1|k) = \mathbf{h}_{ij}(\mathbf{X}_{W_g}^R, \mathbf{X}_{G_i}^{W_g}, X_{L_j}^R) = \ominus \mathbf{X}_{G_i}^{W_g} \oplus \mathbf{X}_R^{W_g} \oplus \mathbf{X}_{L_j}^R = 0 \tag{F.9}$$

La matrice jacobienne de cette fonction \mathbf{h} s'écrit donc :

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_X & \mathbf{H}_{X_{L_j}^R} \end{bmatrix}$$

La matrice jacobienne \mathbf{H}_X est calculée par rapport au vecteur \mathbf{X} et s'écrit :

$$\begin{aligned}
\mathbf{H}_X &= \begin{bmatrix} \mathbf{H}_{X_V} & 0 & \cdots & 0 & \mathbf{H}_{X_{G_i}} & 0 & \cdots & 0 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{J}_{1\oplus}(\mathbf{X}_R^{G_i}, \mathbf{X}_{L_j}^R) \cdot \mathbf{J}_{1\oplus}(\mathbf{X}_{W_g}^{G_i}, \mathbf{X}_R^{W_g}) \cdot \mathbf{J}_{\ominus}(\mathbf{X}_{G_i}^{W_g}) & 0 & \cdots & 0 & \mathbf{J}_{2\oplus}(\mathbf{X}_R^{G_i}, \mathbf{X}_{L_j}^R) & 0 & \cdots & 0 \end{bmatrix}
\end{aligned} \tag{F.10}$$

F.2.2.4 Fonction de cartographie

La fonction de cartographie correspond à l'ajout d'une position d'amer mesuré depuis la position du véhicule, $\mathbf{X}_{A_i}^R$, dans le vecteur d'état du système. Comme tous les amers sont exprimés dans le repère W_g , la fonction de cartographie correspond donc à une projection de la position de l'amer A_i du repère centré sur le robot R vers le repère W_g .

$$\begin{aligned} \mathbf{X}_{A_i}^{W_g}(k+1|k+1) &= \mathbf{g}_i(\mathbf{X}_R^{W_g}(k+1|k+1), \mathbf{X}_{A_i}^R(k)) \\ &= \mathbf{X}_R^{W_g}(k+1|k) \oplus \mathbf{X}_{G_i}^{W_g}(k+1|k) \end{aligned} \quad (\text{F.11})$$

La matrice jacobienne \mathbf{G}_X de la fonction \mathbf{g} s'écrit :

$$\begin{aligned} \mathbf{G}_X &= \begin{bmatrix} \mathbf{G}_{X_V} & 0 & \cdots & 0 & \mathbf{G}_{X_{A_i}} & 0 & \cdots & 0 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{J}_{1\oplus}(\mathbf{X}_R^{W_g}, \mathbf{X}_{G_i}^{W_g}) & 0 & \cdots & 0 & \mathbf{J}_{2\oplus}(\mathbf{X}_R^{W_g}, \mathbf{X}_{G_i}^{W_g}) & 0 & \cdots & 0 \end{bmatrix} \end{aligned} \quad (\text{F.12})$$

F.3 Approche EKF-SLAM *RobotCentric*

L'approche *Worldcentric* est une représentation de l'environnement dite absolue. Pour minimiser les erreurs découlant des changements de repère entre la carte globale et la position du véhicule, Castellanos et al. proposent une représentation relative à la position courante du robot, nommée *Robotcentric* [30].

F.3.1 Présentation

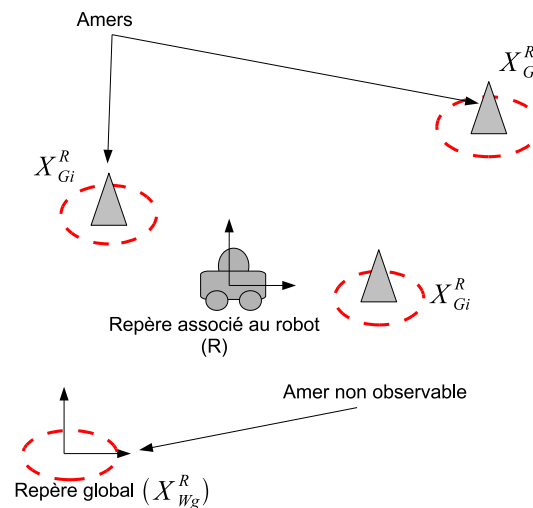


Figure F.2: Représentation de la carte et du robot : approche *RobotCentric*.

Dans une approche *RobotCentric*, tous les amers de la carte sont référencés par rapport à la position du véhicule à l'instant k (cf. la figure F.2). Le vecteur d'état s'écrit :

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_{W_g}^R \\ \mathbf{X}_{G_1}^R \\ \vdots \\ \mathbf{X}_{G_m}^R \end{bmatrix} \quad (\text{F.13})$$

Le vecteur $\mathbf{X}_{W_g}^R$ correspond à la position du repère global de la carte, par rapport à la position du véhicule. Il est considéré comme un amer non observable dans la carte. Les vecteurs $\mathbf{X}_{G_i}^R$ avec $i = \{1 \dots m\}$ correspondent aux positions des amers référencés dans le repère centré sur la position du robot R .

F.3.2 Fonctions régissant le système

F.3.2.1 Fonction d'évolution des états

Dans cette approche, le déplacement relatif du véhicule entre deux instants, k et $k + 1$, est estimé à partir des données odométriques $U(k)$ et noté $\mathbf{X}_{R(k+1)}^{R(k)}$:

$$\mathbf{X}_{R(k+1)}^{R(k)} = \mathbf{f}_v(\mathbf{X}_0, \mathbf{U}(k)) \quad (\text{F.14})$$

La fonction \mathbf{f}_v correspond à la modélisation du tricycle et \mathbf{X}_0 le vecteur du déplacement initialisé avec une valeur nulle. La matrice de covariance associée à $\mathbf{X}_{R(k+1)}^{R(k)}$ sera notée \mathbf{P}_Q .

Le déplacement, $\mathbf{X}_{R(k+1)}^{R(k)}$, n'est pas ajouté directement à l'état du véhicule comme dans l'approche *Worldcentric*. Il est inclus dans le vecteur d'état $\mathbf{X}^{R(k)}$:

$$\mathbf{X}^{R(k)} = \begin{bmatrix} \mathbf{X}_{W_g}^{R(k)} \\ \mathbf{X}_{G_1}^{R(k)} \\ \vdots \\ \mathbf{X}_{G_m}^{R(k)} \\ \mathbf{X}_{R(k+1)}^{R(k)} \end{bmatrix} \quad (\text{F.15})$$

En supposant que $\mathbf{X}_{R(k+1)}^{R(k)}$ est indépendant par rapport aux autres éléments du vecteur (F.15), la matrice de covariance associée à ce vecteur d'état s'écrit :

$$\mathbf{P}^{R(k)}(k+1|k) = \begin{bmatrix} \mathbf{P}^{R_k}(k|k) & 0 \\ 0 & \mathbf{P}_Q \end{bmatrix} \quad (\text{F.16})$$

F.3.2.2 Fonction d'observation associée à un amer de type point ou cercle

En utilisant les notations introduites précédemment, le vecteur d'un amer mesuré s'écrit $\mathbf{X}_{A(i)}^{R(k+1)}$. La fonction d'observation \mathbf{h} s'écrit donc :

$$\mathbf{Z}(k+1) = \mathbf{h}_i(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_i}^{R(k)}) + \mathbf{w}_i \quad (\text{F.17})$$

avec :

$$\mathbf{h}_i(\mathbf{X}_{R(k+1)}^{R(k)}, \mathbf{X}_{G_i}^{R(k)}) = \mathbf{X}_{G_i}^{R(k+1)} = \ominus \mathbf{X}_{R(k+1)}^{R(k)} \oplus \mathbf{X}_{G_i}^{R(k)} \quad (\text{F.18})$$

Le calcul de la matrice de covariance associée $P_Z(k+1|k)$ s'écrit :

$$\mathbf{P}_Z(k+1|k) = \mathbf{H}_X \cdot P(k+1|k) \cdot \mathbf{H}_X^T \quad (\text{F.19})$$

\mathbf{H}_X représente la matrice jacobienne correspondant à la dérivée de la fonction h par rapport au vecteur d'état $\mathbf{X}^{R(k)}$:

$$\begin{aligned} \mathbf{H}_X &= \begin{bmatrix} 0 & \cdots & 0 & \mathbf{H}_{X_{G_i}^{R(k)}} & 0 & \cdots & 0 & \mathbf{H}_{X_{R(k+1)}^{R(k)}} \end{bmatrix} \\ &= \begin{bmatrix} 0 & \cdots & 0 & \mathbf{J}_{2\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_i}^{R(k)}) & 0 & \cdots & 0 & \mathbf{J}_{1\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_i}^{R(k)}) \cdot \mathbf{J}_{\ominus}(\mathbf{X}_{R(k+1)}^{R(k)}) \end{bmatrix} \end{aligned} \quad (\text{F.20})$$

F.3.2.3 Fonction d'observation associée à un amer de type ligne

Dans une représentation de 3 états d'une ligne, la formulation de la fonction d'observation diffère dû à la présence d'une information angulaire apportée par l'amer. Lorsque un amer de type ligne $\mathbf{X}_{G_i}^{R(k)}$ est associé avec une mesure de ce même amer $\mathbf{X}_{L_j}^{R(k+1)}$, la fonction d'observation s'écrit :

$$\mathbf{Z}(k+1|k) = \mathbf{h}_{ij}(\mathbf{X}_{G_i}^{R(k)}, \mathbf{X}_{R(k+1)}^{R(k)}, \mathbf{X}_{L_j}^{R(k+1)}) = \ominus \mathbf{X}_{G_i}^{R(k)} \oplus \mathbf{X}_{R(k+1)}^{R(k)} \oplus \mathbf{X}_{L_j}^{R(k+1)} = 0 \quad (\text{F.21})$$

La matrice jacobienne de cette fonction \mathbf{h} s'écrit donc :

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_X & \mathbf{H}_{X_{L_j}^{R(k+1)}} \end{bmatrix}$$

La matrice jacobienne \mathbf{H}_X est calculée par rapport au vecteur \mathbf{X} et s'écrit :

$$\mathbf{H}_X = \begin{bmatrix} 0 & \cdots & 0 & \mathbf{H}_{X_{G_i}^{R(k)}} & 0 & \cdots & 0 & \mathbf{H}_{X_{R(k+1)}^{R(k)}} \end{bmatrix} \quad (\text{F.22})$$

Les matrices jacobienes s'écrivent :

$$\begin{aligned} \mathbf{H}_{X_{G_i}^{R(k)}} &= \begin{bmatrix} \mathbf{J}_{1\oplus}(\mathbf{X}_{R(k+1)}^{G_i}, \mathbf{X}_{L_j}^{R(k+1)}) \cdot \mathbf{J}_{1\oplus}(\mathbf{X}_{R(k)}^{G_i}, \mathbf{X}_{R(k+1)}^{R(k)}) \cdot \mathbf{J}_{\ominus}(\mathbf{X}_{G_i}^{R(k)}) \end{bmatrix} \\ \mathbf{H}_{X_{R(k+1)}^{R(k)}} &= \begin{bmatrix} \mathbf{J}_{2\oplus}(\mathbf{X}_{R(k+1)}^{G_i}, \mathbf{X}_{L_j}^{R(k+1)}) \end{bmatrix} \end{aligned} \quad (\text{F.23})$$

F.3.2.4 Fonction de transformation

Après avoir réalisé l'étape de mise jour du filtre EKF, il faut réaliser une dernière étape, nommée *composition*, pour exprimer les positions des amers de la carte par rapport à la nouvelle position du véhicule estimée, $R(k+1)$. Cette fonction \mathbf{l} correspond donc à l'ajout du déplacement du véhicule $\mathbf{X}_{R(k+1)}^{R(k)}$ à tous les éléments de la carte pour obtenir le vecteur $\mathbf{X}^{R(k+1)}$:

$$\mathbf{X}^{R(k+1)} = \mathbf{l}(\mathbf{X}_{R(k+1)}^{R(k)}, \mathbf{X}^{R(k)}) = \begin{bmatrix} \ominus \mathbf{X}_{R(k+1)}^{R(k)} \oplus \mathbf{X}_{W_g}^{R(k)} \\ \ominus \mathbf{X}_{R(k+1)}^{R(k)} \oplus \mathbf{X}_{G_1}^{R(k)} \\ \vdots \\ \ominus \mathbf{X}_{R(k+1)}^{R(k)} \oplus \mathbf{X}_{G_n}^{R(k)} \end{bmatrix} \quad (\text{F.24})$$

Le calcul de la matrice de covariance associée est :

$$\mathbf{P}^{R(k+1)} = \begin{bmatrix} \mathbf{L}_{X^{R(k)}} \\ \mathbf{L}_{X_{R(k+1)}^{R(k)}} \end{bmatrix} \mathbf{P}^{R(k)} \cdot \begin{bmatrix} \mathbf{L}_{X^{R(k)}} \\ \mathbf{L}_{X_{R(k+1)}^{R(k)}} \end{bmatrix}^T \quad (\text{F.25})$$

Les jacobiennes $\mathbf{L}_{X_{R(k+1)}^{R(k)}}$ et $\mathbf{L}_{X^{R(k)}}$ sont fonctions d'opérations de composition (cf. l'annexe F) :

$$\mathbf{L}_{X_{R(k+1)}^{R(k)}} = \begin{bmatrix} \mathbf{J}_{1\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{W_g}^{R(k)}) & \mathbf{J}_{\ominus}(\mathbf{X}_{R(k+1)}^{R(k)}) \\ \mathbf{J}_{1\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_1}^{R(k)}) & \mathbf{J}_{\ominus}(\mathbf{X}_{R(k+1)}^{R(k)}) \\ \vdots & \\ \mathbf{J}_{1\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_n}^{R(k)}) & \mathbf{J}_{\ominus}(\mathbf{X}_{R(k+1)}^{R(k)}) \end{bmatrix} \quad (\text{F.26})$$

$$\mathbf{L}_{X^{R(k)}} = \begin{bmatrix} \mathbf{J}_{2\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{W_g}^{R(k)}) & 0 & \cdots & 0 \\ 0 & \mathbf{J}_{2\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_1}^{R(k)}) & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{J}_{2\oplus}(\mathbf{X}_{R(k)}^{R(k+1)}, \mathbf{X}_{G_n}^{R(k)}) \end{bmatrix} \quad (\text{F.27})$$

F.3.2.5 Fonction de cartographie

Après l'étape de composition, tous les amers sont référencés par rapport à la position courante du robot. C'est de cette même position du robot que les mesures des amers ont été effectuées à l'instant $k + 1$. L'amer \mathbf{X}_{A_i} avec comme matrice de covariance associée \mathbf{C}_{A_i} est référencé dans le repère $R(k+1)$. Cet amer est donc directement ajouté au vecteur d'état du système sans aucun changement de repère :

$$\mathbf{X}^{R(k+1)} = \begin{bmatrix} \mathbf{X}^{R(k+1)} \\ \mathbf{X}_{A_i} \end{bmatrix} \quad (\text{F.28})$$

$$\mathbf{P}^{R(k+1)} = \begin{bmatrix} \mathbf{P}^{R(k+1)} & 0 \\ 0 & \mathbf{C}_{A_i} \end{bmatrix} \quad (\text{F.29})$$

F.3.3 Remarque sur cette approche

Cette approche *RobotCentric* donne de meilleurs résultats que l'approche *WorldCentric* d'après [30]. En effet, tous les éléments de la carte sont référencés par rapport à la position du véhicule à l'instant k . La projection de la position des amers à la position du véhicule à l'instant $k + 1$ est réalisée sur une distance très courte. Les erreurs, commises sur la linéarisation des fonctions non-linéaires, correspondent essentiellement à celles du déplacement du véhicule entre deux instants. Elles sont donc faibles et la linéarisation plus exacte. Dans l'approche *WorldCentric*, les amers sont projetés du repère global vers la position courante du véhicule. Cette distance est largement plus importante, ce qui entraîne des approximations de la linéarisation des fonctions non-linéaires plus importantes, et donc les erreurs sur les estimations rendent les calculs de plus en plus inexacts.

F.4 Filtrage de Kalman Itératif

Cette approche est introduit dans de nombreux ouvrages dont le livre [12]. Ce filtre repose toujours sur la théorie de Kalman. Cependant, dans le but de réduire les erreurs introduites par les modèles non-linéaires d'observation, la mise à jour des états est modifiée. Dans cette approche, le calcul de l'état mis à jour ne correspond plus à une moyenne conditionnelle approximée mais à une estimation à vraisemblance maximale a priori. Cette estimation est obtenue par une itération supplémentaire équivalente à une relinéarisation des équations de mesure (cf. l'algorithme F.1). Seule l'étape du filtre de Kalman correspondant à la mise à jour des estimations est modifiée. Les deux autres étapes, la prédiction des états et la cartographie des nouveaux amers, restent inchangées.

Algorithme F.1 : Étape de mise à jour des estimations itérative du filtre IEKF :
Algorithme

```

N ← nombre d'itération
 $\hat{\mathbf{X}}_0(k+1|k+1) \leftarrow \hat{\mathbf{X}}_0(k+1|k+1)$ 
i ← 1
pour i = 0 à N - 1 faire
    Calcul de la matrice jacobienne  $\mathbf{H}_i$  : équation (F.30)
    Calcul de la matrice de covariance  $\mathbf{P}_i$  : équation (F.31)
    Calcul de l'estimation  $\hat{\mathbf{X}}_{i+1}$  : équation (F.33)
fin pour
pour i = N faire
    Calcul de la matrice de covariance  $\mathbf{P}_i$ 
fin pour

```

Dans la suite de ce paragraphe, nous allons décrire les différents calculs permettant de réaliser les itérations du filtre IEKF lors de la mise à jour des estimations. En premier, le calcul de la matrice Jacobienne d'observation repose sur la dérivée de la fonction d'observation d'un amer dans le repère du robot. Ce calcul s'écrit :

$$\mathbf{H}_i = \frac{\partial \mathbf{h}(\hat{\mathbf{X}}_i)}{\partial \hat{\mathbf{X}}_i} \quad (\text{F.30})$$

A l'aide de la matrice (F.30), la matrice de covariance \mathbf{P}_i est calculée pour la $i^{\text{ème}}$ itération telle que :

$$\mathbf{P}_i = \mathbf{P}_0 - \mathbf{P}_0 \cdot \mathbf{H}_i^T \cdot (\mathbf{S}_i)^{-1} \cdot \mathbf{H}_i \cdot \mathbf{P}_0 \quad (\text{F.31})$$

La matrice de covariance associée à l'innovation s'écrit :

$$\mathbf{S}_i = \mathbf{H}_i \cdot \mathbf{P}_0 \cdot \mathbf{H}_i^T + \mathbf{R} \quad (\text{F.32})$$

Une fois les matrices de covariance \mathbf{P}_i et jacobienne \mathbf{H}_i obtenues, la valeur de l'estimation de l'état du système est mise à jour :

$$\hat{\mathbf{X}}_{i+1} = \hat{\mathbf{X}}_i + \mathbf{P}_i \cdot \mathbf{H}_i^T \cdot \mathbf{R}^{-1} \cdot v_i - \mathbf{P}_i \cdot \mathbf{P}_0^{-1} \cdot [\hat{\mathbf{X}}_i - \hat{\mathbf{X}}_0] \quad (\text{F.33})$$

La variable v_i représente l'innovation calculée entre l'observation $\mathbf{h}(\hat{\mathbf{X}}_i)$ et la mesure de l'amer z . Cette innovation s'écrit :

$$v_i = z - \mathbf{h}(\hat{\mathbf{X}}_i) \quad (\text{F.34})$$

F.5 Filtrage sans parfum dans une application SLAM

Proposé dans [88], ce filtre de Kalman sans parfum est une extension du filtre de Kalman pour les systèmes non-linéaires. Il repose sur le principe qu'il est plus simple d'approximer une distribution gaussienne avec un nombre de paramètres fixés (les sigma points) que d'approximer arbitrairement une fonction non linéaire via une linéarisation de Taylor dans le cas du filtre de Kalman Étendu. La loi de probabilité est ainsi représentée par un ensemble de points aux quels sont appliqués les modèles non-linéaires du SLAM. Ce filtre a été appliqué au SLAM dans les travaux [111] [4].

Cette méthode ressemble au principe des filtres de Monte Carlo mais avec la différence fondamentale suivante : les échantillons (aussi appelés sigma points) ne sont plus tirés de manière aléatoire mais de manière déterministe grâce à la transformation sans parfum (*Unscented Transformation*).

F.5.1 Principe de la transformation *unscented*

Cette transformation repose sur la création d'un ensemble de $p+1$ particules ou *sigma points* $\mathcal{S} = \{\mathcal{S}^i, W^i\}$. Cet ensemble doit satisfaire une condition écrite sous la forme :

$$\mathcal{L}(\mathcal{S}, p(\mathbf{X})) = 0 \quad (\text{F.35})$$

$p(\mathbf{X})$ représente la densité de probabilité de la variable \mathbf{X} et la fonction \mathcal{L} détermine l'information qui doit être capturée sur la variable \mathbf{X} . Dans un cas gaussien, la condition (F.35) sur la valeur moyenne $\hat{\mathbf{X}}$ s'écrit :

$$\mathcal{L}_1 = \sum_{i=0}^p (W^i \cdot \mathcal{X}^i) - \mathbf{X}(k|k) \quad (\text{F.36})$$

et celle sur la matrice de covariance $P(k|k)$:

$$\mathcal{L}_2 = \sum_{i=0}^p (W^i \cdot (\mathcal{X}^i - \mathbf{X}(k|k)) \cdot (\mathcal{X}^i - \mathbf{X}(k|k))^T) - \mathbf{P}(k|k) \quad (\text{F.37})$$

L'ensemble de points qui satisfait les conditions \mathcal{L}_1 et \mathcal{L}_2 est défini par :

$$\begin{aligned} \mathcal{X}^0(k|k) &= \mathbf{X}(k|k) & \text{et } W^0 &= \frac{\kappa}{n + \kappa} \\ \mathcal{X}^i(k|k) &= \mathbf{X}(k|k) + (\sqrt{(n + \kappa)\mathbf{P}(k|k)})_i & \text{et } W^i &= \frac{1}{2 \cdot (n + \kappa)} \quad \text{avec } i = 1, \dots, n \\ \mathcal{X}^{i+n}(k|k) &= \mathbf{X}(k|k) - (\sqrt{(n + \kappa)\mathbf{P}(k|k)})_i & \text{et } W^{i+n} &= \frac{1}{2 \cdot (n + \kappa)} \quad \text{avec } i = 1, \dots, n \end{aligned} \quad (\text{F.38})$$

La valeur $(\sqrt{(n + \kappa)\mathbf{P}(k|k)})_i$ correspond à la $i^{\text{ième}}$ colonne de la décomposition de Cholesky de la matrice $\mathbf{P}(k|k)$ et W^i est le poids associé au $i^{\text{ième}}$ point. Le terme κ est utilisé pour échantillonner les termes d'ordre supérieur à 2 et n est la dimension de l'espace d'état augmenté. C'est à dire le nombre d'état et le bruit.

F.5.2 Application de l'UKF dans un contexte SLAM

La transformation sans parfum est appliquée aux différentes étapes de l'EKF qui nécessite une linéarisation : prédiction et observation.

F.5.2.1 Etape de prédiction

Pour tenir compte du bruit sur les données odométriques dans le calcul des sigma points, le vecteur d'état \mathbf{X} et sa matrice de covariance associée \mathbf{P} doivent être agrandis :

$$\mathbf{X}^*(k|k) = \begin{bmatrix} \mathbf{X}(k|k) \\ \mathbf{U}(k) \end{bmatrix} \text{ et la matrice de covariance : } \mathbf{P}^*(k|k) = \begin{bmatrix} \mathbf{P}(k|k) & 0 \\ 0 & \mathbf{Q}_U(k) \end{bmatrix} \quad (\text{F.39})$$

Ensuite, l'évolution de ce nuage de points utilise la fonction non linéaire \mathbf{f} :

$$\mathcal{X}^i(k+1|k) = \mathbf{f}(\mathcal{X}^i(k|k), u_k, \mathcal{V}^i(k)) \quad (\text{F.40})$$

$\mathcal{V}^i(k)$ correspond au bruit gaussien associé au point $\mathcal{X}^i(k|k)$. Le vecteur d'état et la matrice de covariance sont obtenus par les calculs :

$$\mathbf{X}(k+1|k) = \sum_{i=0}^p (W^i \cdot \mathcal{X}^i(k+1|k)) \quad (\text{F.41})$$

$$\mathbf{P}(k+1|k) = \sum_{i=0}^p (W^i \cdot (\mathcal{X}^i(k+1|k) - \mathbf{X}(k+1|k)) \cdot (\mathcal{X}^i(k+1|k) - \mathbf{X}(k+1|k))^T) \quad (\text{F.42})$$

F.5.2.2 Etape de mise à jour : calcul de l'innovation

Pour tenir compte du bruit sur les mesures extéroceptives dans le calcul des sigma points, le vecteur d'état \mathbf{X} et sa matrice de covariance associée \mathbf{P} doivent être agrandis de nouveau :

$$\mathbf{X}^*(k+1|k) = \begin{bmatrix} \mathbf{X}(k+1|k) \\ \mathbf{Z}(k) \end{bmatrix} \text{ et la matrice de covariance : } \mathbf{P}^*(k+1|k) = \begin{bmatrix} \mathbf{P}(k+1|k) & \mathbf{0} \\ \mathbf{0} & \mathbf{R}(k) \end{bmatrix} \quad (\text{F.43})$$

Ensuite, l'observation de ce nuage de points repose sur la fonction non linéaire \mathbf{h} :

$$\mathcal{Z}^i(k+1|k) = \mathbf{h}(\mathcal{X}^i(k+1|k)) + \mathcal{W}^i(k) \quad (\text{F.44})$$

$\mathcal{W}^i(k)$ correspond au bruit gaussien associée au point $\mathcal{X}^i(k+1|k)$. Le vecteur d'état et la matrice de covariance sont obtenues par les calculs :

$$\mathbf{Z}(k+1|k) = \sum_{i=0}^p (W^i \cdot \mathcal{Z}^i(k+1|k)) \quad (\text{F.45})$$

$$\mathbf{P}_Z(k+1|k) = \sum_{i=0}^p (W^i \cdot (\mathcal{Z}^i(k+1|k) - \mathbf{Z}(k+1|k)) \cdot (\mathcal{Z}^i(k+1|k) - \mathbf{Z}(k+1|k))^T) \quad (\text{F.46})$$

F.5.2.3 Etape de mise à jour : calcul des estimations

La matrice de covariance entre les observations \mathbf{Z} et la prédiction de l'état $\mathbf{X}(k+1|k)$ est obtenue avec :

$$\mathbf{P}_{XZ}(k+1|k) = \sum_{i=0}^p (W^i \cdot (\mathcal{X}^i(k+1|k) - \mathbf{X}(k+1|k)) \cdot (\mathcal{Z}^i(k+1|k) - \mathbf{Z}(k+1|k))^T) \quad (\text{F.47})$$

Cette matrice permet de calculer le gain du filtre de Kalman \mathbf{W} :

$$\mathbf{W} = \mathbf{P}_{XZ}(k+1|k) \cdot (\mathbf{P}_Z(k+1|k))^{-1} \quad (\text{F.48})$$

Le calcul des estimations du vecteur d'état et de la matrice de covariance associée s'obtiennent facilement à partir des équations classiques du filtre de Kalman :

$$\mathbf{X}(k+1|k+1) = \mathbf{X}(k+1|k) + \mathbf{W} \cdot \tilde{\mathbf{Z}}(k+1|k) \quad (\text{F.49})$$

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k) - \mathbf{W} \cdot \mathbf{P}_Z(k+1|k) \cdot \mathbf{W}^T \quad (\text{F.50})$$

F.6 Approche multi-cartes

F.6.1 Présentation de l'approche

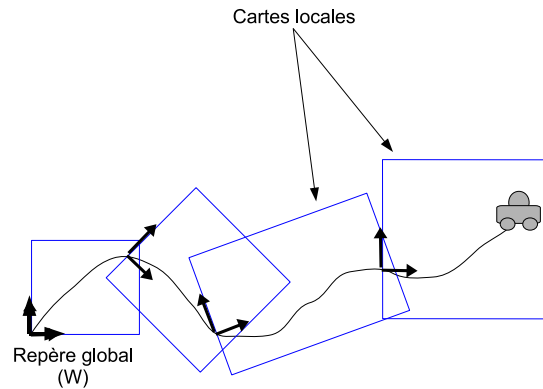


Figure F.3: Représentation de la carte et du robot dans une approche multi-cartes.

L'approche multi-cartes [146] permet de découper la carte de l'environnement en plusieurs cartes locales (cf. la figure F.3). Le système débute par la construction d'une première carte locale qui est stoppée après plusieurs itérations². Cette carte est alors sauvegardée et une nouvelle

2. Le nombre optimal d'itérations par cartes locales est analysé dans les articles [124] et [125]

carte est initialisée et construite. Au final, la carte de l'environnement est représentée par une série de cartes locales. L'obtention de la carte globale de l'environnement passe par la fusion de toutes les cartes locales. Cette opération appelée *Joining Local Map* [146] est décrite dans la suite de cette section.

F.6.2 Systèmes d'état

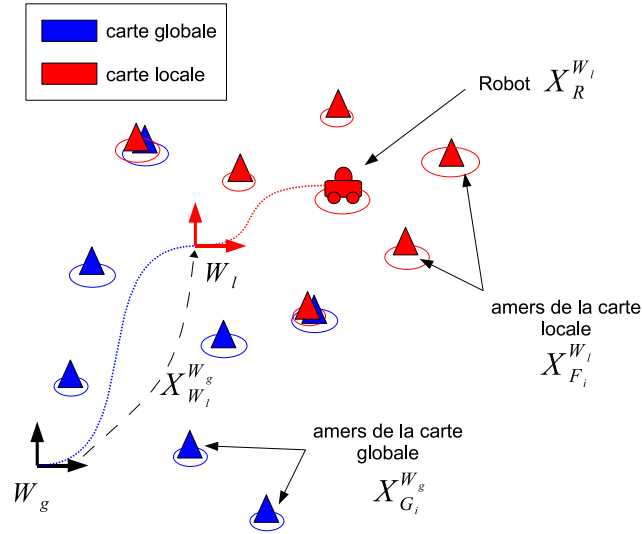


Figure F.4: Mise à jour d'une carte globale avec une carte locale.

Pour présenter le concept de fusion d'une carte locale dans une carte globale, les cartes vont adopter une représentation *Worldcentric* (cf. la figure F.4). Le vecteur d'une carte locale s'écrit :

$$\mathbf{X}_{local} = \begin{bmatrix} \mathbf{X}_R^{W_l} \\ \mathbf{X}_{F_1}^{W_l} \\ \vdots \\ \mathbf{X}_{F_n}^{W_l} \end{bmatrix} \text{ et la matrice de covariance associée : } \mathbf{C}_{local} \quad (\text{F.51})$$

Les vecteurs $\mathbf{X}_{F_i}^{W_l}$ avec $i = \{1 \dots n\}$ correspondent aux positions des amers référencés dans le repère W_l de la carte locale. Le vecteur $\mathbf{X}_R^{W_l}$ représente la position du robot dans cette carte. Il est initialisé avec une valeur nulle. La carte globale est représentée par le vecteur :

$$\mathbf{X}_{global} = \begin{bmatrix} \mathbf{X}_{W_l}^{W_g} \\ \mathbf{X}_{G_1}^{W_g} \\ \vdots \\ \mathbf{X}_{G_m}^{W_g} \end{bmatrix} \text{ et la matrice de covariance associée : } \mathbf{C}_{global} \quad (\text{F.52})$$

Les vecteurs $\mathbf{X}_{G_i}^{W_g}$ avec $i = \{1 \dots m\}$ correspondent aux positions des amers référencés dans le repère W_g de la carte globale. Le vecteur $\mathbf{X}_{W_l}^{W_g}$ représente la position courante du robot dans la carte globale. Il correspond donc à la position du repère de la carte locale dans la carte globale.

F.6.3 Fonctions régissant la fusion des cartes locales

La carte finale de l'environnement est obtenu en fusionnant successivement toutes les cartes locales soit de manière itérative, soit dans un processus annexe. Dans notre approche, la carte globale de l'environnement est mise à jour à chaque nouvelle carte locale construite. Cette mise à jour est réalisée en quatre étapes : concaténation des variables locales et globales, mise en correspondance des deux cartes, mise à jour de la carte globale et cartographie des nouveaux amers.

Vecteur d'état du système :

Tout d'abord, le vecteur de la carte locale \mathbf{X}_{local} est concaténé avec le vecteur de la carte globale \mathbf{X}_{global} :

$$\mathbf{X}_{system} = \begin{bmatrix} \mathbf{X}_{global} \\ \mathbf{X}_{local} \end{bmatrix} = \begin{bmatrix} \mathbf{X}_{W_g}^{W_g} \\ \mathbf{X}_{G_1}^{W_g} \\ \vdots \\ \mathbf{X}_{G_m}^{W_g} \\ \mathbf{X}_{R}^{W_i} \\ \mathbf{X}_{F_1}^{W_i} \\ \vdots \\ \mathbf{X}_{F_n}^{W_i} \end{bmatrix} \quad (\text{F.53})$$

Comme il est supposé que les deux cartes ne sont pas corrélées, la matrice de covariance associée au vecteur \mathbf{X}_{system} s'écrit :

$$\mathbf{P}_{system} = \begin{bmatrix} \mathbf{C}_{global} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{local} \end{bmatrix} \quad (\text{F.54})$$

Etape de mise à jour :

Pour mettre en correspondance les amers de chacune des deux cartes, tous les amers de la carte globale sont translatés dans le repère de la carte locale via la fonction d'observation \mathbf{h} . Cette fonction est définie avec des opérations de composition (cf. l'annexe B) :

$$\mathbf{Z}(k+1|k) = \mathbf{h}_{ij}(\mathbf{X}_{system}) = \ominus \mathbf{X}_{G_j}^{W_g} \oplus \mathbf{X}_{W_l}^{W_g} \oplus \mathbf{X}_{F_i}^{W_l} = 0 \quad (\text{F.55})$$

Dans ce calcul, il est supposé que l'amer $\mathbf{X}_{F_i}^{W_l}$ de la carte locale a été mis en correspondance avec l'amer $\mathbf{X}_{G_j}^{W_g}$ de la carte globale durant une étape d'association de données (cf. le chapitre 2). La matrice de covariance associée à la prédiction \mathbf{Z} s'écrit :

$$\mathbf{P}_Z(k+1|k) = \mathbf{H}_{ij} \cdot \mathbf{P}_{system} \cdot \mathbf{H}_{ij}^T \quad (\text{F.56})$$

La matrice jacobienne associée à la fonction d'observation \mathbf{h} s'écrit :

$$\mathbf{H}_{ij} = \begin{bmatrix} \mathbf{H}_{W_g} & \mathbf{0} & \mathbf{H}_{G_j} & \mathbf{0} & \mathbf{H}_{F_i} & \mathbf{0} \end{bmatrix} \quad (\text{F.57})$$

\mathbf{H}_{W_g} correspond à la matrice Jacobienne de la fonction \mathbf{h}_{ij} calculée par rapport à \mathbf{H}_{W_g} :

$$\mathbf{H}_{W_g} = \left[\mathbf{J}_{1\oplus}(\mathbf{X}_{W_l}^{G_j}, \mathbf{X}_{F_i}^{W_l}) \cdot \mathbf{J}_{2\oplus}(\mathbf{X}_{W_g}^{G_j}, \mathbf{X}_{W_l}^{W_g}) \right] \quad (\text{F.58})$$

\mathbf{H}_{G_j} correspond à la matrice jacobienne de \mathbf{h}_{ij} calculée par rapport à $\mathbf{X}_{W_l}^{W_g}$:

$$\mathbf{H}_{G_j} = \left[\mathbf{J}_{1\oplus}(\mathbf{X}_{W_l}^{G_j}, \mathbf{X}_{F_i}^{W_l}) \cdot \mathbf{J}_{1\oplus}(\mathbf{X}_{W_g}^{G_j}, \mathbf{X}_{W_l}^{W_g}) \cdot \mathbf{J}_{\ominus}(\mathbf{X}_{G_j}^{W_g}) \right] \quad (\text{F.59})$$

La matrice \mathbf{H}_{F_i} correspond à la jacobienne de la fonction \mathbf{h}_{ij} calculée par rapport à $\mathbf{X}_{F_i}^{W_l}$:

$$\mathbf{H}_{W_g} = \left[\mathbf{J}_{2\oplus}(\mathbf{X}_{W_l}^{G_j}, \mathbf{X}_{F_i}^{W_l}) \right] \quad (\text{F.60})$$

Le calcul des matrices jacobiennes reposent sur les dérivées des opérations de composition (cf. l'annexe B). Les différentes matrices et calculs précédents sont ensuite employés dans les équations d'un filtre EKF pour mettre à jour le système (cf. l'annexe A).

Cartographie des amers :

Une fois la mise à jour terminée, les amers de la carte locale qui ont été associés sont éliminés du vecteur \mathbf{X}_{system} et la matrice de covariance associée \mathbf{P}_{system} . Les amers non associés sont cartographiés via la fonction \mathbf{g} :

$$\begin{aligned} \mathbf{X}_{F_i}^{W_g} &= \mathbf{g}_i(\mathbf{X}_{W_L}^{W_g}, \mathbf{X}_{F_i}^{W_L}(k)) \\ &= \mathbf{X}_{W_L}^{W_g} \oplus \mathbf{X}_{F_i}^{W_L} \end{aligned} \quad (\text{F.61})$$

La matrice jacobienne \mathbf{G}_X de la fonction g s'écrit :

$$\begin{aligned} \mathbf{G}_X &= \left[\mathbf{G}_{X_{W_g}} \quad 0 \quad \cdots \quad 0 \quad \mathbf{G}_{X_{F_i}} \quad 0 \quad \cdots \quad 0 \right] \\ &= \left[J_{1\oplus}(\mathbf{X}_{W_L}^{W_g}, \mathbf{X}_{F_i}^{W_L}) \quad 0 \quad \cdots \quad 0 \quad J_{2\oplus}(\mathbf{X}_{W_L}^{W_g}, \mathbf{X}_{F_i}^{W_L}) \quad 0 \quad \cdots \quad 0 \right] \end{aligned} \quad (\text{F.62})$$

F.6.4 Remarques sur cette approche

Une telle approche a pour avantage de réduire le nombre d'amer à traiter à chaque itération du filtre de Kalman et donc de réduire d'une manière conséquente le temps de calcul. Elle est très usitée et donc très étudiée [132][146][160]. De nombreux aspects de cette approche ont été améliorés. Par exemple, la taille optimale des cartes locales a été déterminée dans l'article [125]. De plus, les mêmes auteurs proposent une stratégie efficace de fusion des cartes locales sur plusieurs « niveaux » hiérarchiques nommée *Divide and Conquer* [124], permettant de diminuer les temps de calcul. Le temps de mise à jour de la carte globale est égal à $O(n)$. Plus récemment, une nouvelle technique [128] permet de partager une information entre deux cartes locales. Le suivi d'un objet mobile d'une carte locale vers la suivante devient donc possible.

L'ensemble de ces travaux soulignent l'efficacité d'une telle approche au niveau de la représentation de l'environnement, des temps de calcul des mises à jour de la carte mais également de la consistance des estimations obtenues avec le filtre de Kalman [29] [131]. Pour toutes ces raisons, cette méthode est employée dans les travaux présentés dans le chapitre 4.

Bibliographie

- [1] SICK LMS 220/221 Technical Information.
- [2] M. Altermatt, A. Martinelli, N. Tomatis, and R. Siegwart. Slam with Corner Features Based on a Relative Map. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan*, pages 1053–1058, October 2004.
- [3] J. Andrade-Cetto and A. Sanfeliu. The Effects of Partial Observability in SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 397–402, April 2004.
- [4] J. Andrade-Cetto, T. Vidal-Calleja, and A. Sanfeliu. Unscented Transformation of Vehicle States in SLAM. In *Proceedings, IEEE International Conference on Robotic and Automation, Barcelona, Spain*, pages 324–339, April 2005.
- [5] K. O. Arras and R. Siegwart. Feature Extraction and scene interpretation for map-based navigation and map building. In *Proceedings of SPIE, Mobile Robotics*, pages 42–53, February 1997.
- [6] K.O. Arras. An Introduction To Error Propagation : Derivation, Meaning and Examples of Equation. Technical report, Autonomous Systems Lab, Institute of Robotic Systems, Swiss Federal Institute of Technology Lausanne (EPFL), September 1998.
- [7] O. Aycard, F. Charpillet, D. Fohr, and J.F. Mari. Real Time Self Localization in Unknown Indoor Environments using a Panorama Laser Range Finder. In *Proceedings, IEEE International Conference on Intelligent Robots and Systems*, pages 697–702, October 1997.
- [8] T. Bailey. *Mobile robot Localisation and Mapping in extensive Outdoor Environments*. PhD thesis, University of Sidney, 2002.
- [9] T. Bailey and H. Durrant-Whyte. Simultaneous Localisation and Mapping (SLAM) : Part II State of the Art. *IEEE Robotics and Automation Magazine*, 13(3) :108–117, September 2006.
- [10] T. Bailey, E.M. Nebot, J.K. Rosenblatt, and H.F. Durrant-Whyte. Data Association for Mobile Robot Navigation : A graph Theoretic Approach. In *Proceedings, IEEE International Conference on Robotics and Automation, Washington, U.S.*, 2000.
- [11] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot. Consistency of the EKF-SLAM Algorithm. In *Proceedings, IEEE/RSJ International Conference on Robots and Systems, Beijing, China*, pages 3562–3568, 2006.
- [12] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with Applications To Tracking and Navigation*. John Wiley and SONS, INC.

- [13] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11 :451–460, 1975.
- [14] Bay, Harbert, Tuytelaars, Tinne, V. Gool, and Luc. SURF : Speeded Up Robust Features. In *Proceedings of the 9th European Conference on Computer Vision*, Graz, Austria, may 2006.
- [15] K.R. Beevers and W.H. Huang. Loop Closing in Topological Maps. In *Proceedings, IEEE International Conference on Robotic and Automation, Barcelona, Spain*, pages 4378–4383, April 2005.
- [16] K.R. Beevers and W.H. Huang. Fixed-lag Sampling Strategies for Particle Filtering SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 2433–2438, April 2007.
- [17] P.J. Besl and N.D. McKay. A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2) :239–256, February 1992.
- [18] J. Bom, B. Thuillot, F. Marmoiton, and P. Martinet. Nonlinear Control for Urban Vehicles Platooning, relying upon a Unique Kinematic GPS. In *Proceedings, IEEE International Conference on Robotic and Automation, Barcelona, Spain*, pages 4138–4143, April 2005.
- [19] J. Borenstein and L. Feng. Measurement and Correction of Systematic Odometry errors in Mobile Robots. *IEEE Transactions on Robotics and automation*, 12(6) :869–880, December 1996.
- [20] G.A. Borges and M.J Aldon. Line Extraction in 2D Range Images for Mobile Robotics. *Intelligent Robotics Systems*, 40(3) :267–297, 2004.
- [21] M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller. An Atlas framework for scalable mapping. In *Proceedings, IEEE International Conference on Robotics and Automation, Taipei, Taiwan*, pages 1899–1906, September 2003.
- [22] M. Bosse and J. Roberts. Histogram Matching and Global Initialization for Laser-only SLAM in Large Unstructured Environments. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 4820–4826, April 2007.
- [23] D. Caltabiano, G. Muscato, and F. Russo. Localization and Self Calibration of a Robot for Volcano Exploration. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 586–591, April 2004.
- [24] F. Carravetta, A. Germani, and M. Raimondi. Polynomial Filtering of Discrete-Time Stochastic Linear Systems with Multiplicative State Noise. *IEEE Transactions on Automatic Control*, 42(8) :1106–1126, May 1997.
- [25] Francesco Carravetta, Alfredo Germani, and Massimo Raimondi. Polynomial Filtering for Linear Discrete Time Non-Gaussian Systems. *SIAM Journal on Control and Optimization*, 34(5) :1666–1690, 1996.
- [26] G. Casella and C.P. Robert. Rao-blackwellisation of sampling schemes. *Biometrika*, 83(1) :81–94, 1996.

- [27] J. Castellanos, J. Montiel, J. Neira, and J. Tardos. The SPmap : A probabilistic framework for Simultaneous Localization and Map Building. *IEEE Transactions on Robotics and Automation*, 15(5) :948–953, 1999.
- [28] J. Castellanos, J. Neira, O. Strauss, and J. Tardos. Detecting high level features for mobile robot localization. In *Proceedings, IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems, Washington, U.S.*, pages 611–618, December 1996.
- [29] J.A. Castellanos, R. Martinez-Cantin, J.D. Tardos, and J. Neira. Robocentric map joining : Improving the consistency of EKF-SLAM. *Robotics and Autonomous Systems*, 55(1) :21–29, February 2007.
- [30] J.A. Castellanos, J. Neira, and J.D. Tardos. Limits to the consistency of EKF-based SLAM. In *Proceedings, IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon, Portugal*, July 2004.
- [31] J.A. Castellanos, J.D. Tardos, and G. Schmidt. Building a global map of the environment of a mobile robot : The importance of correlations. In *Proceedings, IEEE International Conference on Robotics and Automation, Taipei, Taiwan*, pages 1053–1059, April 1997.
- [32] H.J. Chang, C.S.G. Lee, Y.H. Lu, and Y.C. Hu. A Computational Efficient SLAM Algorithm Based on Logarithmic-Map Partitioning. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan*, pages 1041–1046, October 2004.
- [33] F. Chanier, P. Checchin, C. Blanc, and L. Trassoudaine. Impact of Perturbation Estimator on EKF-SLAM Results. In *Proceedings, IFAC Symposium on Intelligent Autonomous Vehicles, Toulouse, France*, July 2007.
- [34] F. Chanier, P. Checchin, C. Blanc, and L. Trassoudaine. Comparison of EKF and PEKF in a SLAM context. In *Proceedings, The 11th International IEEE Conference on Intelligent Transportation Systems, Beijing, China*, pages 1078–1083, October 2008.
- [35] F. Chanier, P. Checchin, C. Blanc, and L. Trassoudaine. Map fusion based on a multi-map SLAM framework. In *Proceedings, IEEE International Conference on Multisensor Fusion for Intelligent Systems, Seoul, Korea*, pages 533–538, August 2008.
- [36] F. Chanier, P. Checchin, C. Blanc, and L. Trassoudaine. Polynomial Extended Kalman Filter in a SLAM framework. In *Proceedings, workshop of the IEEE International Conference on Robots and Systems, Nice, France*, September 2008.
- [37] F. Chanier, P. Checchin, C. Blanc, and L. Trassoudaine. SLAM process using Polynomial Extended Kalman Filter : Experimental Assessment. In *Proceedings, The 10th International Conference on Control, Automation, Robotics and Vision, Hanoi, Vietnam*, pages 365–370, December 2008.
- [38] F. Chanier, P. Checchin, C. Blanc, and L. Trassoudaine. *Multisensor Fusion and Integration for Intelligent Systems. An Edition of the Selected Papers from the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems 2008*, volume 35 of *Lecture Notes in Electrical Engineering*, chapter Map Fusion based on a multi-map SLAM Framework, pages 439–154. Springer, springer berlin heidelberg edition, 2009.

- [39] R. Chatila and J.P. Laumond. Position referencing and consistent world modeling. In *Proceedings, IEEE International Conference on Robotics and Automation*, pages 138–145, April 1985.
- [40] M. Choi, R. Sakhivel, and W.K. Chung. Neural Network-Aided Extended Kalman Filter for SLAM Problem. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 1686–1690, April 2007.
- [41] K.S. Chong and L. Kleeman. Large scale sonar array mapping using multiple connected local maps. *Field and Service Robotics*, 12 :507–514, 1998.
- [42] H. Choset and K. Nagatani. Topological Simultaneous Localization and Mapping (SLAM) : Toward Exact Localization Without Explicit Localization. *IEEE Transactions on Robotics and Automation*, 17(2) :125–137, April 2001.
- [43] D.M. Cole and P.M. Newman. Using Laser Range Data for 3D SLAM in Outdoor Environments. In *Proceedings, IEEE International Conference on Robotics and Automation, Orlando, Floride*, pages 1556–1563, May 2006.
- [44] C.P. Connette, O. Meister, M. Hagele, and G.F. Trommer. Decomposition of Line Segments into Corner and Statistical Grown Line Features in an EKF-SLAM Framework. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA*, pages 3884–3891, October 2007.
- [45] F. Conticelli, A. Bicchi, and A. Balestrino. Observability and nonlinear observers for mobile robot localization. In *Proceedings, IFAC International Symposium on Robot Control, SyRoCo*, May 2000.
- [46] I.J. Cox. Blanche-an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and automation*, 7(2) :193–204, April 1991.
- [47] M. Csorba. *Simultaneous Localization and Map Building*. PhD thesis, University of Oxford, 2003.
- [48] A. Davison. *Mobile Robot Navigation Using Active Vision*. PhD thesis, Oxford University, 1999.
- [49] F. Dellaert and M. Kaess. Square Root SAM : Simultaneous Location and Mapping via Square Root Information Smoothing,. *International Journal of Robotics Research*, 25 :1181–1203, 2006.
- [50] K.C.J. Dietmayer, J. Sparbert, and D. Streller. Model based Object Classification and Object Tracking in Traffic scenes from Range Images. In *Proceedings, IEEE International Conference on Intelligent Vehicles, Tokyo, Japan*, 2001.
- [51] A. Diosi and L. Kleeman. Uncertainty of Line Segments Extracted from Static SICK PLS Laser Scans. In *Proceedings, Australian Conference on Robotics and Automation*, 2003.
- [52] A. Diosi and L. Kleeman. Laser Scan Matching in Polar Coordinates with Application to SLAM. In *Proceedings, IEEE International Conference on Robots and Systems*, pages 1439–1444, August 2005.

- [53] M.W.M.G. Dissanayake, P. Newman and S. Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the Simultaneous Localization and Map building (SLAM) problem. *IEEE Transactions on Robotics and automation*, 17(3) :229–241, June 2001.
- [54] M.W.M.G. Dissanayake, S.B. Williams, H.F. Durrant-Whyte, and T. Bailey. Map Management for Efficient Simultaneous Localization and Mapping (SLAM). *Autonomous Robots*, 12 :267–286, 2002.
- [55] A. Doucet. On Sequential Simulation-Based Methods for Bayesian Filtering. Technical Report CUED/F-INFENG/TR. 310, Cambridge University Department of Engineering, 1998.
- [56] A. Doucet, N. F., K. Murphy, and S. Russell. Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. pages 176–183, April 2000.
- [57] A. Doucet, N. Freitas, and N. Gordon. *An introduction to sequential Monte Carlo methods*, pages 3–14. Springer Verlag, April 2001.
- [58] H. Durrant-Whyte and J.E. Guivant. Simultaneous Localisation and Mapping (SLAM) : Part I The Essential Algorithms. *IEEE Transactions on Robotics and Automation*, 13(2) :99–110, June 2006.
- [59] A. Elfes. Using Occupancy Grids for Mobile Robot Perception and Navigation. *IEEE Computer*, 22(6) :46–57, June 1989.
- [60] C. Estrada, J. Neira, and J.D. Tardos. Hierarchical SLAM : real-time accurate mapping of large environments. *IEEE Transactions on Robotics and Automation*, 21(4) :588–596, August 2005.
- [61] R. Eustice, M. Walter, and J. Leonard. Sparse Extended Information Filters : Insights into Sparsification. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada*, pages 641–648, August 2005.
- [62] R.M. Eustice, H. Singh, and J.J. Leonard. Exactly Sparse Delayed-State Filters for View-Based SLAM. *IEEE Transactions on Robotics and automation*, 22(6) :1100–1114, December 2006.
- [63] J. Folkesson and H. Christensen. Graphical SLAM - A Self-Correcting Map. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 383–390, April 2004.
- [64] U. Forssell, P. Hall, S. Ahlqvist, and F. Gustafsson. Map-Aided Positioning System. In *Proceedings, International Federation of Automotive Engineering Societies*, April 2002.
- [65] O. Frank, J. Nieto, J. Guivant, and S. Scheduling. Multiple target tracking using sequential monte carlo methods and statistical data association. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, U.S.*, pages 2718–2723, October 2003.
- [66] U. Frese. Treemap : An $O(\log n)$ Algorithm for Simultaneous Localization and Mapping. *Spatial Cognition 4*, C. Freska, Ed Springer Varlag, pages 455–476, 2005.
- [67] U. Frese and G. Hirzinger. Simultaneous Localization and Mapping - A Discussion. In *Proceedings of the IJCAI Workshop on Reasoning with Uncertainty in Robotics, Seattle, U.S.*, May 2001.

- [68] U. Frese, P. Larsson, and T. Duckett. A Multilevel Relaxation Algorithm for Simultaneous Localisation and Mapping. *Robotics and Autonomous Systems*, 21(2) :196–207, April 2005.
- [69] W. Gander, G.H. Golub, and R. Strebler. Least-Squares Fitting of Circles and Ellipses. *BIT*, 43 :558–578, december 1994.
- [70] A. Germani, C. Manes, and P. Palumbo. Polynomial Extended Kalman Filtering for Discrete-Time Nonlinear Stochastic Systems. In *Proceedings, IEEE International Conference on Decision and Control, Maui, Hawaii, U.S.*, pages 886–891, December 2003.
- [71] A. Germani, C. Manes, and P. Palumbo. Polynomial Extended Kalman Filter. *IEEE Transactions on Automatic Control*, 50(12) :2059–2064, December 2005.
- [72] F. Gerossier, P. Checchin, C. Blanc, R. Chapuis, and L. Trassoudaine. Trajectory-oriented EKF-SLAM using the Fourier-Mellin Transform applied to Microwave Radar Images. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, St-Louis, USA*, April 2009.
- [73] J.J. Gonzalez-Barbosa and S. Lacroix. Fast dense panoramic stereovision. In *Proceedings, IEEE International Conference on Robotic and Automation, Barcelona, Spain*, pages 1222–1227, April 2005.
- [74] W.E.L. Grimson and D.P. Huttenlocher. On the Sensitivity of the Hough Transform for Object Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3) :255–274, 1990.
- [75] G. Grisetti, G.D. Tipaldi, C. Stachniss, W. Burgard, and D. Nardi. Fast and accurate SLAM with Rao Blackwellized particle filters. *Robotics and Autonomous Systems*, 55(1) :30–38, 2007.
- [76] J. Guivant and F. Masson. Using Absolute Non-Gaussian Non-White Observations in Gaussian SLAM. In *Proceedings, IEEE International Conference on Robotic and Automation, Barcelona, Spain*, pages 338–343, April 2005.
- [77] J.E. Guivant. *Efficient Simultaneous Localization and Mapping in Large Environments*. PhD thesis, University of Sidney, 2002.
- [78] J.E. Guivant, E.M. Nebot, and S. Baiker. Autonomous navigation and map building using Laser Range Sensors in outdoor applications. *Robotics Systems*, 17(10) :565–593, October 2000.
- [79] J.S. Gutmann and C. Schlegel. AMOS : Comparison of Scan Matching Approaches for Self-Localization in Indoor Environments. In *Proceedings of the First Euromicro Workshop on Advanced Mobile Robots, Kaiserslautern, Germany*, pages 61–67, October 1996.
- [80] D. Hahnel, W. Burgard, D. Fox, and S. Thrun. An Efficient FastSLAM Algorithm for Generating Maps of Large-Scale Cyclic Environments from Raw Laser Range Measurements. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada, U.S.*, pages 206–211, October 2003.
- [81] R. Hermann and A.J. Krener. Nonlinear Controllability and Observability. *IEEE Transactions on Automatic Control*, 22(5) :728–740, October 1977.

- [82] G.P. Huang, A.I. Mourikis, and S.I. Roumeliotis. Analysis and Improvement of the Consistency of Extended Kalman Filter based SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, Pasadena, U.S.*, pages 473–479, May 2008.
- [83] S. Huang and G. Dissanayake. Convergence analysis for extended Kalman filter based SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, Orlando, Floride*, pages 412–417, May 2006.
- [84] S. Huang and G. Dissanayake. Convergence and Consistency Analysis for Extended Kalman Filter based SLAM. *IEEE Transactions on Robotics and Automation*, 21(5) :1036–1049, October 2007.
- [85] P. Jensfelt and H.I. Christensen. Pose Tracking Using Laser Scanning and Minimalistic Environmental Models. *IEEE Transactions on Robotics and Automation*, 17(2) :138–147, April 2001.
- [86] S.J. Julier and J.K. Uhlmann. A New Extension of the Kalman Filter to Nonlinear Systems. In *Proceedings of AeroSense, 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Orlando, U.S.*, October 1997.
- [87] S.J. Julier and J.K. Uhlmann. Using covariance intersection for SLAM. *Robotics and Autonomous Systems*, 2007, 55 :3–20, 2007.
- [88] S.J. Julier, J.K. Uhlmann, and H.F. Durrant-Whyte. A new Approach for filtering Nonlinear Systems. In *Proceedings, the American Control Conference, Seattle, USA*, pages 1628–1632, October 1995.
- [89] S.J. Julier and K. Uhlmann. A counter example to the theory of simultaneous localization and map building. In *Proceedings, IEEE International Conference on Robotics and Automation, Seoul, Korea*, pages 4238–4243, October 2001.
- [90] S.J. Julier and K. Uhlmann. Simultaneous localisation and map building using split covariance intersection. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, U.S.*, pages 406–411, November 2001.
- [91] M. Kaess, A. Ranganathan, and F. Dellaert. iSAM : Fast Incremental Smoothing and Mapping with Efficient Data Association. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 1670–1677, April 2007.
- [92] K. Kowalski and W.H. Steeb. *Nonlinear Dynamical Systems and Carleman Linearization*. World Scientific (Singapore, Teaneck, N.J).
- [93] B. Kuipers. The Spatial Semantic Hierarchy. *Artificial Intelligence*, 117(1-2) :191–233, December 2000.
- [94] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, and F. Savelli. Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 212–217, April 2004.
- [95] S. Kwon, K.W. Yang, and S. Park. An Effective Kalman Filter Localization Method for Mobile Robots. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China*, pages 1524–1529, October 2005.

- [96] S.J Kwon and W.K. Chung. A discrete-time design and analysis of perturbation observer for motion control applications. *IEEE Transactions on Control Systems Technology*, 11(3) :399–407, May 2003.
- [97] S.J Kwon, K.W. Yang, S. Park, and Y. Ryuh. Robust Mobile Robot Localization with Combined Kalman Filter-Perturbation Estimator. In *Proceedings, IEEE International Conference on Robots and Systems*, pages 190–195, August 2005.
- [98] K.W. Lee, S. Wijesoma, and J.I. Guzman. A constrained SLAM approach to robust and accurate localisation of autonomous ground vehicles. *Robotics and Autonomous Systems*, 55(7) :527–540, February 2007.
- [99] K.W. Lee, W.S. Wijesoma, and J.I. Guzman. On the Observability and Observability Analysis of SLAM. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China*, pages 3569–3574, October 2005.
- [100] J.J. Leonard and H.F. Durrant Whyte. Simultaneous map building and localization for an autonomous mobile robot. In *Proceedings, IEEE/RSJ International Conference on Robots and Systems, Osaka, Japan*, pages 1442–1447, November 1991.
- [101] K. Lingemann, H. Surmann, A. Nuchter, and J. Hertzberg. Indoor and Outdoor Localization for Fast Mobile Robots. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan*, pages 2185–2190, October 2004.
- [102] G.S. Lionis and K.J. Kyriakopoulos. A Laser Scanner based Mobile Robot SLAM Algorithm with Improved Convergence Properties. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland*, pages 582–587, October 2002.
- [103] Y. Liu and S. Thrun. Results for outdoor-SLAM using Sparse Extended Information Filters. In *Proceedings, IEEE International Conference on Robotics and Automation, Taipei, Taiwan*, pages 1227–1233, September 2003.
- [104] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(20) :91–110, November 2004.
- [105] F. Lu and E. Milius. Globally Consistent Range Scan Alignment for Environment Mapping. *Autonomous Robots*, 4(4) :333–349, October 1997.
- [106] F. Lu and E. Milius. Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans. *Intelligent and Robotic Systems*, 18(3) :249–275, 1997.
- [107] C. Manes, A. Martinelli, F. Martinelli, and P. Palumbo. Mobile Robot Localization based on a Polynomial Approach. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 3539–3544, April 2007.
- [108] A. Martinelli, V. Nguyen, N. Tomatis, and R. Siegwart. A relative map approach to SLAM based on shift and rotation invariants. *Robotics and Autonomous Systems*, 55(1) :50–61, February 2007.
- [109] A. Martinelli, N. Tomatis, and R. Siegwart. Some Results on SLAM and the Closing the Loop Problem. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada*, pages 2917–3432, August 2005.

- [110] A. Martinelli, N. Tomatis, A. Tapus, and R. Siegwart. Simultaneous Localization and Odometry Calibration for Mobile Robot. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada*, pages 1499–1504, October 2003.
- [111] R. Martinez-Cantin and J.A. Castellanos. Unscented SLAM for Large-Scale Outdoor Environments. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada*, pages 3427–3432, August 2005.
- [112] R. Martinez-Cantin and J.A. Castellanos. Bounding Uncertainty in EKF-SLAM : The Robocentric Local Approach. In *Proceedings, IEEE International Conference on Robotics and Automation, Orlando, Floride*, pages 430–435, May 2006.
- [113] R. Martinez-Cantin, N. de Freitas, and J.A. Castellanos. Analysis of Particle Methods for Simultaneous Robot Localization and Mapping and a New Algorithm : Marginal-SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 2415–2420, April 2007.
- [114] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0 : An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico*, 2003.
- [115] J. Neira. *Sensors used for autonomous navigation*, pages 135–156. 1999.
- [116] J. Neira and J.D. Tardos. Data Association in Stochastic Mapping Using the Joint Compatibility Test. *IEEE Transactions on Robotics and automation*, 17(6) :890–897, June 2001.
- [117] J. Neira, J.D. Tardos, J. Horn, and G. Schmidt. Fusing Range and Intensity Images for Mobile Robot Localization. *IEEE Transactions on Robotics and automation*, 15(1) :76–84, February 1999.
- [118] P. Newman, D. Cole, and K. Ho. Outdoor SLAM using Visual Appearance and Laser Ranging. In *Proceedings, IEEE International Conference on Robotics and Automation, Orlando, Floride*, pages 1180–1187, May 2006.
- [119] P.M. Newman and H.F. Durrant-Whyte. A new solution to the simultaneous Localization and Map building (SLAM) Problem - the GPF. In *Proceedings of SPIE*, pages 1180–1187, 2001.
- [120] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart. A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada*, August 2005.
- [121] K. Ni, D. Steedly, and F. Dellaert. Tectonic SAM : Exact, Out-of-Core, Submap-Based SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 1678–1685, April 2007.
- [122] J. Nieto, T. Bailey, and E. Nebot. Recursive Scan-Matching SLAM. *Journal of Robotics and Autonomous Systems*, 55(1) :39–49, January 2007.

- [123] J. Nygard. *On robot Feedback from Range sensors : Reliable Control by Active Reduction of Uncertainty and Ambiguities*. PhD thesis, Lindkoping University, 1998.
- [124] L.M. Paz, P. Jensfelt, J.D. Tardos, and J. Neira. EKF SLAM updates in $O(n)$ with Divide and Conquer SLAM. In *Proceedings, IEEE International Conference on Robotics and Automation, Roma, Italy*, pages 1657–1663, April 2007.
- [125] L.M. Paz and J. Neira. Optimal local map size for EKF-based SLAM. In *Proceedings, IEEE International Conference on Intelligent Robots and Systems, Beijing, China*, pages 5019–5025, October 2005.
- [126] L.D.L. Perera, W.S. Wijesoma, and M.D. Adams. On Multidimensional Assignment Data Association for Simultaneous Robot Localization and Mapping. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 860–865, April 2004.
- [127] S.T. Pfister, S.I. Roumeliotis, and J.W. Burdick. Weighted Line Fitting Algorithms for Mobile Robot Map Building and Efficient Data Representation. In *Proceedings, IEEE International Conference on Robotics and Automation, Taipei, Taiwan*, pages 1304–1311, April 2003.
- [128] P. Pinies and J.D. Tardos. Scalable SLAM building Conditionally Independent Local Maps. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, U.S.*, pages 3466–3471, October 2007.
- [129] C. Premebida and U. Nunes. Segmentation and Geometric Primitives Extraction from 2D Laser Range Data for mobile robot applications. In *Proceedings, Robotica, Coimbra, Switzerland*, pages 17–25, April 2005.
- [130] D. Rodriguez-Losada and F. Matia. Integrating Segments and Edges in Feature-based SLAM. In *Proceedings, IEEE International Conference on Advanced Robotics, Coimbra, Portugal*, pages 1717–1722, July 2003.
- [131] D. Rodriguez-Losada, F. Matia, and R. Galan. Building geometric feature based maps for indoor service robots. *Robotics and Autonomous Systems*, 2006, 54 :546–558, 2006.
- [132] D. Rodriguez-Losada, F. Matia, and A. Jimenez. Local Maps Fusion for Real Time Multirobot Indoor Simultaneous Localization and Mapping. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 1308–1313, April 2004.
- [133] D. Rodriguez-Losada, F. Matia, L. Pedraza, A. Jimenez, and R. Galan. Consistency of SLAM-EKF algorithms for indoor environments. *Intelligent and Robotics Systems*, 50(4) :375–397, December 2007.
- [134] S.I. Roumeliotis and G.A. Bekey. SEGMENTS : A Layered, Dual-Kalman filter Algorithm for Indoor Feature Extraction. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Takamatsu, Japan*, pages 454–461, October 2000.
- [135] E. Royer, J. Bom, M. Dhome, B. Thuillot, and M. Lhuillet. Outdoor autonomous navigation using monocular vision. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada*, pages 3395–3400, August 2005.

- [136] S. Santos, J.E. Faria, F. Soares, R. Araújo, and U. Nunes. Segmentation and Geometric Primitives Extraction from 2D Laser Range Data for mobile robot applications. In *3rd National Festival of Robotics Scientific Meeting (ROBOTICA), Lisboa, Portugal*, pages 59–65, May 2003.
- [137] F. Savelli and B. Kuipers. Loop-Closing and Planarity in Topological Map-Building. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan*, pages 1511–1517, October 2004.
- [138] S. Scheding, E. Nebot, and H. Durrant-Whyte. High-integrity navigation : A frequency-domain approach. *IEEE Transactions on Control Systems Technology*, 8(4) :676–694, 2000.
- [139] A.C. Schultz and W. Adams. Continuous Localization using evidence Grids. In *Proceedings, IEEE International Conference on Robotics and Automation, Seoul, Korea*, pages 2833–2839, May 1998.
- [140] R. Sim. Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling. In *Proceedings, IEEE International Conference on Robotic and Automation, Barcelona, Spain*, pages 2422–2427, April 2005.
- [141] R.C. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Proceedings, 2nd workshop on Uncertainty in Artificial Intelligence (AAAI)*, pages 1–21, August 1986.
- [142] R.C. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *Proceedings, Fourth International Symposium of Robotics Research*, pages 467–474, 1987.
- [143] C. Stachniss, G. Grisetti, W. Burgard, and N. Roy. Analyzing Gaussian Proposal Distributions for Mapping with Rao-Blackwellized Particle Filters. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, U.S.*, pages 3485–3490, October 2007.
- [144] C. Stachniss, D Hahnel, and W. Burgard. Exploration with Active Loop-Closing for FastSLAM. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan*, pages 1505–1510, October 2004.
- [145] J. D. Tardos. Representing Partial and Uncertain Sensorial Information Using the Theory of Symmetries. In *Proceedings, IEEE International Conference on Robots and Automation, Nice, France*, pages 1799–1804, 1998.
- [146] J.D. Tardos, J. Neira, P.M. Newman, and J.J. Leonard. Robust Mapping and Localization in Indoor Environments using Sonar Data. *International Journal of Robotics Research*, 21(4) :311–330, June 2002.
- [147] C. Tessier, C. Debain, R. Chapuis, and F. Chausse. Simultaneous landmarks detection and data association in noisy environment for map aided localization. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA*, pages 1396–1401, October 2007.
- [148] S. Thrun. Learning Occupancy Grids with Forward Models. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, Hawaii, U.S.*, pages 1676–1681, November 2001.

- [149] S. Thrun. *Robotic Mapping : A Survey*. Morgan Kaufmann, 2002.
- [150] S. Thrun, W. Burgard, and D. Fox. A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots. 31(1-3) :29–53, 1998.
- [151] S. Thrun, D. Koller, Z. Ghaaramani, and H. Durrant-Whyte. Simultaneous Mapping and Localization With Sparse Extended Information Filters : Theory and Initial Results. Technical report, 2002.
- [152] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot. FastSLAM : An Efficient Solution to the Simultaneous Localization And Mapping Problem with Unknown Data Association. *Journal of Machine Learning Research*, 2004.
- [153] B. Thuillot, C. Cariou, P. Martinet, and M. Berducat. Automatic guidance of a frame tractor relying on a single CP-DGPS. *Autonomous Robots*, 13(1) :53–71, 2002.
- [154] N. Tomatis, I. Nourbakhsh, and R. Siegwart. Hybrid simultaneous localization and map building : a natural integration of topological and metric. *Robotics and Autonomous Systems*, 44(1) :3–14, July 2003.
- [155] Bill Triggs, Philip McLauchlan, Richard Hartley, and Andrew Fitzgibbon. Bundle Adjustment – A Modern Synthesis. In W. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms : Theory and Practice*, LNCS, pages 298–375. Springer Verlag, 2000.
- [156] J. Uhlmann, S. Julier, and M. Csorba. Non-divergent simultaneous map building and localization using covariance intersection. In *Proceedings, SPIE International Conference on Navigation and control Technologies for Unmanned systems*, pages 2–11, October 1997.
- [157] C.C. Wang and C. Thorpe. Simultaneous Localization and Mapping with Detection and Tracking of Moving Objects. In *Proceedings, IEEE International Conference on Robotics and Automation, Wasington, U.S.*, pages 406–411, May 2002.
- [158] Z. Wang, S. Huang, and G. Dissanayake. Decoupling Localization and Mapping in SLAM Using Compact Relative Maps. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Canada*, pages 1041–1046, August 2005.
- [159] G. Weiss and E.V. Puttkamer. A map based on laser scans without geometric interpretation. pages 403–407, 1995.
- [160] S.B. Williams, G. Dissanayake, and H. Durrant-Whyte. An efficient approach to the simultaneous localisation and mapping problem. In *Proceedings, IEEE International Conference on Robotics and Automation, Washington, U.S.*, pages 406–411, May 2002.
- [161] O. Wulf, A. Nuchter, J. Hertzberg, and B. Wagner. Ground Truth Evaluation of Large Urban 6D SLAM. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, U.S.*, pages 650–657, October 2007.
- [162] E. Zalama, G. Candela, J. Gomez, and S. Thrun. Concurrent Mapping And Localization For Mobile Robots With Segmented Local Maps. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland*, pages 546–551, October 2002.

-
- [163] L. Zhang and B.K. Ghosh. Line Segment Based Map Building and Localization Using 2D Laser Rangefinder. In *Proceedings, IEEE International Conference on Robotics and Automation, San Francisco, U.S.*, pages 2538–2543, April 2000.
 - [164] S. Zhang, L. Xie, and M.D. Adams. An Efficient Data Association Approach to Simultaneous Localization and Map Building. In *Proceedings, IEEE International Conference on Robotics and Automation, New Orleans, U.S.*, pages 854–859, October 2004.
 - [165] S. Zhang, L. Xie, and M.D. Adams. Feature extraction for outdoor mobile robot navigation based on a modified gauss newton optimization approach. *Robotics and Autonomous Systems*, 54(4) :277–287, April 2006.

Résumé

Pour être autonome, un robot mobile doit être capable de décrire l'environnement dans lequel il évolue. Pour ce faire, le robot doit pouvoir se localiser et construire une carte de l'environnement si celle-ci n'existe pas. Grâce à ces informations, il pourra par exemple éviter les obstacles et donc naviguer en toute sécurité. Cette capacité correspond à une étape obligatoire sur la route de son autonomie totale.

Le sujet traité dans ce manuscrit répond à cette problématique. Il concerne la localisation d'un véhicule et la cartographie d'un environnement en simultanée, **SLAM** pour *Simultaneous Localization And Mapping*. Après un état de l'art des approches proposées durant la dernière décennie, deux points importants ont été constatés lors de l'emploi d'un filtre EKF-SLAM : les estimations obtenues avec ce filtre ne sont pas consistantes et cette approche, telle qu'elle a été introduite, ne répond pas aux critères d'observabilité d'un système. Pour chacun de ces problèmes, une solution est proposée et critiquée à l'aide de résultats de simulation et d'expérimentation.

Mots clefs : cartographie, consistance, filtre de Kalman, localisation, observabilité, robot mobile

Abstract

In order to be autonomous, a mobile robot must be able to describe the environment where it evolves. To this end, the robot must localize itself and make a map of the environment if no one is available. With this information, it will be able, for example, to avoid obstacles and therefore to navigate safely. This ability is a compulsory step on the way of its complete autonomy.

The subject handled in this manuscript deals with this problem. It refers to simultaneous vehicle localization and environment mapping (**SLAM**). After a state of the art of approaches proposed during the last decade, two important points have been noticed when an EKF-SLAM filter is used: obtained estimates are not consistent and this approach, like it has been introduced, does not answer the system observability criteria. For each studied problem, a solution is introduced and criticized by means of simulation and experiment results.

Key words: mapping, consistency, Kalman filter, localization, observability, mobile robot