



**HAL**  
open science

# Perception multisensorielle pour la localisation d'un robot mobile en environnement extérieur, application aux véhicules routiers

Jean Laneurit

► **To cite this version:**

Jean Laneurit. Perception multisensorielle pour la localisation d'un robot mobile en environnement extérieur, application aux véhicules routiers. Automatique / Robotique. Université Blaise Pascal - Clermont-Ferrand II, 2006. Français. NNT : 2006CLF21658 . tel-00692034

**HAL Id: tel-00692034**

**<https://theses.hal.science/tel-00692034>**

Submitted on 27 Apr 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.

N° d'ordre : 1658

EDSPIC : 345

Université Blaise Pascal - Clermont II

Ecole Doctorale

Sciences Pour L'Ingénieur De Clermont-Ferrand

Thèse

présentée par

Jean LANEURIT

pour obtenir le grade de

Docteur d'Université

(Spécialité : Vision pour la robotique)

---

Perception multisensorielle pour la localisation  
d'un robot mobile en environnement extérieur,  
application aux véhicules routiers

---

Soutenue publiquement **3 Juillet 2006** devant le jury :

MM.	L. Trassoudaine	Professeur Université Blaise Pascal	Président
	P. Bonnifait	Maître de Conf. Université de Compiègne	Rapporteur
	F. Peyret	Directeur de Recherche LCPC Nantes	Rapporteur
	J.G. Postaire	Professeur Université de Lille	Rapporteur
	F. Chausse	Maître de Conf. Université d'Auvergne	Examineur
	R. Chapuis	Professeur Université Blaise Pascal	Directeur de thèse



# Résumé

Le sujet traité dans cette thèse concerne la localisation précise et intégrée de robots mobiles (de véhicules routiers en l'occurrence) dans un environnement routier cartographié. Pour ceci, nous proposons une approche multisensorielle hybridant les capteurs classiquement utilisés dans ce genre d'application (un GPS naturel, un odomètre et un gyromètre) avec un système de vision. Ce dernier, en outre, est capable de déterminer très précisément la pose locale du véhicule sur la chaussée en détectant les marquages signalétiques au sol. Il peut donc par l'intermédiaire de la carte de l'environnement routier, fournir des informations à la fois redondantes et complémentaires à celles fournies par le GPS. Ainsi, cette combinaison d'informations doit garantir la précision et l'intégrité du système de localisation et ce notamment dans les zones urbaines denses où les informations du GPS ne sont pas forcément fiables. La mise en œuvre de ce système a été menée dans deux études distinctes. Dans la première, la fusion de données est effectuée par un filtre de Kalman étendu. Dans ce cas, la précision obtenue est quasi-décimétrique, cependant la gestion d'hypothèses multiples dans le cadre autoroutier ou des fortes non-linéarités introduites par le système de vision nuisent fortement à l'intégrité des résultats de localisation. C'est donc tout naturellement, que dans la seconde étude, le filtre de Kalman est remplacé par un filtre particulaire et plus particulièrement par un filtre particulaire génétique. La précision obtenue est ici similaire à la première étude et l'intégrité des résultats de la localisation assurée. En revanche, cette méthode est gourmande en temps de calcul et ne peut être utilisée en temps réel sur les ordinateurs actuels.

## Abstract

This thesis report deals with precise and uncorrupted localization of mobile robots (for example road vehicles) within a road environment associated to a precise numerical map. A multisensorial approach merging sensors basically used in such applications (low cost GPS, odometer, gyrometer) and a vision system, is proposed. Actually, this vision system is able to give precisely the local pose of the vehicle on the road while detecting road lines. Coupled with a numerical map this vision algorithm gives both redundant and complementary information comparable with the one provided by the GPS. Thus this data combination ensures accuracy and integrity of the localization system especially in dense urban areas where GPS data are not so reliable. Two distinct implementations of that system as been conducted. In the first one, data fusion is realized using an extended Kalman filter. In that case the resulting precision reaches some ten centimeters. However, the management of multi-hypothesis localization in the motorway context or the strong non-linearities involved by the vision system detracts seriously the integrity of the localization. Then in the second solution, the Kalman filter is naturally replaced with a particles filter and more specifically an genetic particles filter. The resulting precision is similar to the precedent and the integrity of the localization is insured. Anyway, this method involves a long calculation time and cannot run real time on actual PC's.



# Remerciements

Je remercie tous les membres de mon jury, Monsieur Laurent Trassoudaine pour l'avoir présidé. Monsieur Philippe Bonnifait, Monsieur Jack-Gérard Postaire et Monsieur François Peyret pour l'intérêt qu'ils ont porté à mon travail en acceptant d'être rapporteurs.

Je souhaite surtout remercier Roland Chapuis, mon directeur de thèse, pour m'avoir encadré durant toutes ces années, pour ses précieux conseils, son aide et toute la patience dont il a fait preuve lors de la correction de ce manuscrit.

Je voudrais également témoigner toute ma gratitude envers Frédéric Chausse, mon co-directeur de thèse, pour son soutien constant et ses remarques. Mais aussi, pour tous les moments partagés autour de la littérature et de la musique.

Bien sur je remercie l'ensemble des personnes que j'ai pu côtoyer au LASMEA, notamment Romuald, Serge, Christophe et Guillaume dont leur amitié et leur bonne humeur ont rendu ce travail agréable au quotidien. Je remercie également Thierry Chateau pour m'avoir fait découvrir le filtrage particulière et d'avoir supporté toutes mes questions.

Enfin mes dernières pensées vont à l'ensemble des personnes de ma famille qui m'ont toujours encouragé à poursuivre cette voie et soutenu dans les moments difficiles, à mes cousins et amis Damien (Jamel) et Xavier (Le Gros) et plus particulièrement à mes parents.



# Glossaire et notations

## Partie filtrage :

$\mathcal{N}(m, \mathbf{P})$  : densité gaussienne de moyenne  $m$  et de matrice de covariance  $\mathbf{P}$

$\epsilon = \mathcal{N}(0, \mathbf{P})$  : densité gaussienne centrée de matrice de covariance  $\mathbf{P}$

$\mathcal{U}(a, b)$  : loi de probabilité uniforme sur l'intervalle  $[a, b]$

$k$  : le temps discrétisé

### *Filtre de Kalman :*

$\underline{X}_{k|k-1}$  : état a priori

$\mathbf{P}_{k|k}$  : matrice de covariance de l'état a posteriori  $\underline{X}_{k|k}$

$\underline{X}_{k|k}$  : état a posteriori

$\mathbf{P}_{k|k-1}$  : matrice de covariance de l'état a priori  $\underline{X}_{k|k-1}$

$\mathbf{K}_k$  : gain de Kalman

$\underline{W}_k$  : bruit du système dynamique

$\mathbf{Q}_k$  : matrice de covariance du bruit dynamique

$\underline{V}_k$  : bruit de mesure

$\mathbf{R}_k$  : matrice de covariance du bruit de mesure

$\underline{U}_k$  : vecteur de commande

$\mathbf{S}_k$  : bruit du vecteur de commande

### *Filtre particulaire :*

$\underline{X}_{0:k} = (\underline{X}_0, \underline{X}_1, \dots, \underline{X}_k)^T$  : trajectoire du processus

$\underline{Y}_{1:k} = (\underline{Y}_1, \underline{Y}_2, \dots, \underline{Y}_k)^T$  : vecteurs de mesures jusqu'à l'instant  $k$

$p_{k|k} = p(\underline{X}_k | \underline{Y}_{1:k})$  : loi de probabilité a posteriori de  $\underline{X}_k$  conditionnellement à  $\underline{Y}_{1:k}$

$p_{k|k-1} = p(\underline{X}_k | \underline{Y}_{1:k-1})$  : loi de probabilité a priori de  $\underline{X}_k$  conditionnellement à  $\underline{Y}_{1:k-1}$

$\underline{X}^n$  : vecteur d'état de la particule  $n$

$w^n$  : poids de la particule  $n$

$\mathcal{S}^N = (\underline{X}^n, w^n)$  : nuage de  $N$  particules

$N_{eff}$  : critère de la taille efficace du nuage de particules

$N_{Th}$  : seuil du critère de la taille efficace

## Capteurs :



*Proprioceptifs :* $w_{gyro}$  : vitesse angulaire fournie par le gyromètre $v_{gyro}$  : incertitude sur  $w_{gyro}$  $\Delta_D$  : déplacement mesuré par l'odométrie $v_{\Delta_D}$  : incertitude sur  $\Delta_D$  $\delta_v$  : angle de braquage du volant $\delta_r$  : angle de braquage des roues*Extéroceptifs :* $\underline{Y}_{gps} = (x_{gps}, y_{gps})^T$  : informations fournies par le GPS $\mathbf{R}_{gps}$  : covariance des incertitudes des informations GPS sans modélisation AR $\mathbf{R}_{gps}^*$  : covariance des incertitudes des informations GPS avec modélisation AR $lat_g$  : latence GPS $\underline{Y}_{vision} = (L, X_0, \psi, \alpha, C_h, C_l)$  : informations fournies par le capteur "vision" $\mathbf{R}_{vision}$  : covariance des incertitudes des informations "vision" $lat_v$  : latence "Vision" $\underline{Y}_{dgps} = (x_{dgps}, y_{dgps})^T$  : informations fournies par le DGPS $\mathbf{R}_{dgps}$  : matrice de covariance des incertitudes des informations DGPS*Carte/Monde :* $x_r$  et  $y_r$  : origine du repère local à la route dans le référentiel du monde $\theta_r$  : orientation du repère local à la route dans le référentiel du monde $L_i$  : largeur de la voie de circulation  $i$  de la route $(x, y, \theta)^T$  : attitude du véhicule dans le référentiel du monde $(X_r, Y_r, \Psi_r)^T$  : attitude du véhicule dans le référentiel de la route**Abréviations :**

GPS : Global Positioning System

DGPS : Differential Global Positioning System

EURE : Equivalent User Range Error

GDOP : Geometric Dilution Of Precision

HDOP : Horizontal Dilution of Precision

VDOP : Vertical Dilution of Precision

INS : Inertial Navigation System

KF : Kalman Filter

EKF : Extended Kalman Filter

UKF : Unscented Kalman Filter

FP : Filtre particulaire

GPF : Genetic Particle Filter

SIS : Sequential Importance Resampling

GSIR : Genetic Sampling Importance Resampling

SIR : Sampling Importance Resampling

RPF : Regularized Particle Filter

RPBF : Rao-Blackwellisation Particle Filter

MISE : Mean Integrates Square Error

RMSE : Root Mean Square Error

AR : Autoregressive



# Table des matières

<b>Table des matières</b>	<b>9</b>
<b>Introduction</b>	<b>13</b>
<b>I Etat de l'art</b>	<b>17</b>
<b>1 Les méthodes de positionnement</b>	<b>19</b>
1.1 Le positionnement relatif . . . . .	19
1.1.1 Positionnement par odométrie . . . . .	19
1.1.2 Positionnement inertiel . . . . .	23
1.1.3 Autres méthodes . . . . .	28
1.1.4 Bilan . . . . .	29
1.2 Le positionnement absolu . . . . .	29
1.2.1 Description . . . . .	29
1.2.2 Positionnement par satellites . . . . .	31
1.2.3 Positionnement par téléphone cellulaire . . . . .	37
1.2.4 Positionnement par vision dans un environnement routier . . . . .	41
1.2.5 Bilan . . . . .	51
1.3 Les systèmes de positionnement hybride . . . . .	52
<b>2 Le Filtrage</b>	<b>55</b>
2.1 Le filtrage optimal . . . . .	55
2.2 Les méthodes analytiques . . . . .	57
2.2.1 Le filtre de Kalman . . . . .	57
2.2.2 Le filtre de Kalman étendu EKF . . . . .	59
2.2.3 Le filtre de Kalman sans biais UKF . . . . .	61
2.3 Les filtres de Monté Carlo . . . . .	65
2.3.1 Echantillonnage pondéré IS . . . . .	66
2.3.2 Filtre de Monté Carlo avec procédure SIS . . . . .	67
2.3.3 Le filtre de Monté Carlo avec la procédure SIS-R . . . . .	71
2.3.4 Filtre particulaire régularisé RPF . . . . .	76
2.3.5 Le filtre particulaire génétique . . . . .	80
2.4 Conclusion . . . . .	85
<b>3 Modélisation des erreurs</b>	<b>89</b>
3.1 Fondement mathématiques . . . . .	90
3.1.1 Fonction d'autocorrélation et densité spectrale . . . . .	90

3.1.2	Le bruit blanc . . . . .	92
3.2	Processus générés à partir d'un bruit blanc . . . . .	92
3.2.1	La constante aléatoire . . . . .	93
3.2.2	La marche aléatoire . . . . .	94
3.2.3	Les processus de Gauss-Markov . . . . .	95
3.2.4	Précision de la modélisation des processus . . . . .	98
3.3	Modélisation autorégressive . . . . .	99
3.3.1	Processus autorégressif AR . . . . .	100
3.3.2	Détermination des paramètres des processus AR . . . . .	101
3.4	Filtrage et informations stochastiques . . . . .	106
3.4.1	Modification du système d'état . . . . .	106
3.4.2	Exemple . . . . .	107
3.5	Conclusion . . . . .	109

## II Applications 111

<b>4</b>	<b>Choix et caractérisation des capteurs</b>	<b>113</b>
4.1	Capteurs proprioceptifs . . . . .	113
4.1.1	Choix des capteurs . . . . .	113
4.1.2	L'odomètre . . . . .	116
4.1.3	Le gyromètre . . . . .	118
4.2	Capteurs extéroceptifs . . . . .	121
4.2.1	Choix des capteurs . . . . .	121
4.2.2	Le GPS . . . . .	122
4.2.3	Le capteur "Vision" . . . . .	129
4.3	Prise en compte du temps . . . . .	140
4.4	Conclusion . . . . .	142
<b>5</b>	<b>Localisation avec filtre de Kalman</b>	<b>145</b>
5.1	Vecteur d'état . . . . .	145
5.1.1	Description . . . . .	145
5.1.2	Vecteur d'état initial et matrice de covariance associée . . . . .	146
5.2	Etape de prédiction . . . . .	147
5.2.1	Prédiction du vecteur d'état . . . . .	147
5.2.2	Prédiction de la covariance . . . . .	149
5.3	Mise à jour par information gyromètre . . . . .	150
5.4	Mise à jour par informations GPS . . . . .	151
5.4.1	Cohérence des informations GPS . . . . .	151
5.4.2	Mise à jour . . . . .	152
5.5	Mise à jour par informations "vision" . . . . .	153
5.5.1	Fonction d'observation . . . . .	153
5.5.2	Incertitudes des informations "vision" . . . . .	155
5.5.3	Gestion de la multimodalité . . . . .	156

5.5.4	Mise à jour . . . . .	158
5.6	Résultats . . . . .	158
5.6.1	Localisation selon la disponibilité des capteurs . . . . .	160
5.6.2	Localisation sans la modélisation AR des incertitudes GPS . . . . .	166
5.6.3	Multimodalité et changement de voie . . . . .	167
5.7	Conclusion . . . . .	170
<b>6</b>	<b>Localisation : filtre particulaire</b>	<b>171</b>
6.1	Le jeu de particules . . . . .	172
6.1.1	Description . . . . .	172
6.1.2	Le jeu de particules initial . . . . .	172
6.2	Etape de prédiction . . . . .	173
6.3	Mise à jour par GPS . . . . .	174
6.4	Mise à jour par "vision" . . . . .	175
6.4.1	Mise à jour "vision" unimodale . . . . .	175
6.4.2	Multimodalité des observations "vision" . . . . .	177
6.4.3	Prise en compte de la carte . . . . .	178
6.5	Choix du filtre particulaire . . . . .	179
6.5.1	Filtre particulaire avec interaction . . . . .	180
6.5.2	Filtre particulaire génétique . . . . .	182
6.6	Résultats . . . . .	187
6.6.1	Localisation selon la disponibilité des capteurs . . . . .	187
6.6.2	Localisation sans la modélisation AR des incertitudes GPS . . . . .	194
6.6.3	Multimodalité et changement de voie . . . . .	194
6.6.4	Légitimité du filtre particulaire génétique vis-à-vis du filtre particulaire avec interaction . . . . .	196
6.7	Conclusion . . . . .	199
<b>7</b>	<b>Conclusion et perspectives</b>	<b>201</b>
<b>III</b>	<b>Annexes</b>	<b>205</b>
<b>A</b>	<b>Apprentissage de l'algorithme de vision</b>	<b>207</b>
<b>B</b>	<b>Systèmes géodésiques et projections cartographiques associées</b>	<b>211</b>
<b>C</b>	<b>Le véhicule expérimental VELAC</b>	<b>221</b>
	<b>Bibliographie</b>	<b>229</b>
	<b>Table des figures</b>	<b>238</b>
	<b>Liste des tableaux</b>	<b>241</b>
	<b>Listes des Algorithmes</b>	<b>242</b>



# Introduction

La conception de véhicules automatisés ou des robots mobiles est un domaine de recherche en pleine expansion. Ces véhicules sont souvent conçus pour être utilisés dans l'industrie comme moyen de transport, d'inspection, ou d'opération et peuvent être particulièrement adaptés à des interventions en environnement hostile. En voici quelques exemples :

- la robotique agricole avec l'automatisation des engins de récolte,
- la robotique de service pour le transport, le nettoyage ou la surveillance des locaux en milieu agro-alimentaire, industriel, hospitalier ...,
- les travaux publics avec la conception d'engins de chantier automatisés,
- la robotique en milieu hostile qui intervient en particulier dans le domaine nucléaire pour la surveillance et la maintenance des centrales nucléaires (reconnaissance et cartographie des niveaux de radiation, l'aide au démantèlement ou assainissement des sites radioactifs).

Pour que de tels véhicules puissent fonctionner, il est nécessaire d'élaborer des systèmes de perceptions, de localisation, de planification trajectoire ou encore de commande. L'architecture fonctionnelle (cf figure 0.1) de ces véhicules peut être décomposée en deux tâches : la tâche de navigation et la tâche de localisation. La tâche de navigation se décompose elle-même en trois modules :

- un module de planification de trajectoire qui calcule un itinéraire local sous la forme de points de passage selon une destination définie par un opérateur humain,
- un module de guidage qui calcule les consignes de vitesse et de cap et qui permet au véhicule de suivre une trajectoire en interpolant les points calculés par le module de planification,
- un module de commande qui a pour fonction d'asservir le comportement cinématique du véhicule sur les valeurs de consignes déterminées par le module de guidage,

Quant à la tâche de localisation, elle occupe une position transverse car elle est nécessaire à tous les modules de la tâche de navigation. Entre autre elle leur fournit différentes informations sur le comportement du véhicule dans son environnement comme sa position ou sa cinématique (cap, roulis, tangage ainsi que vitesse de déplacement). Enfin pour réaliser les tâches de navigation ainsi que la tâche de localisation il est bien sûr nécessaire de connaître l'environnement du véhicule qui est le plus souvent donnée par une carte planaire de celui-ci.



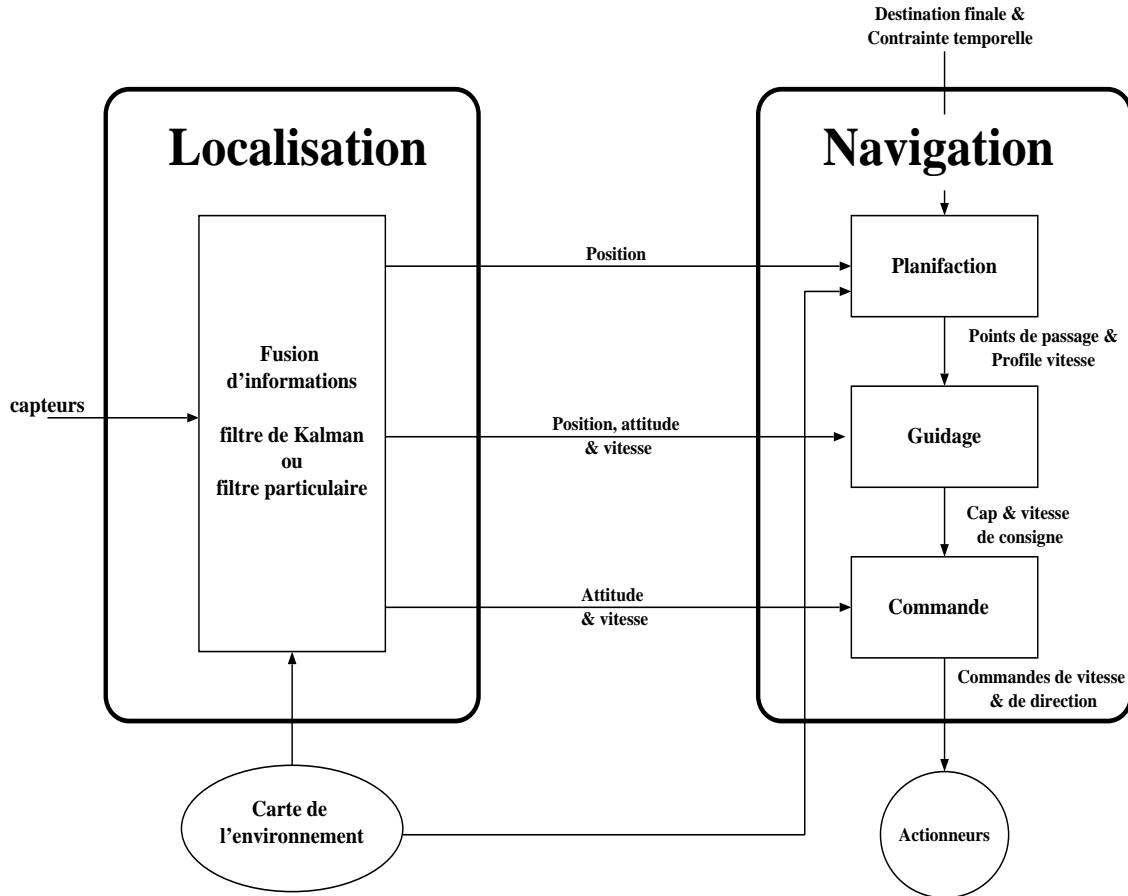


FIG. 0.1: Architecture fonctionnelle d'un véhicule automatisé

Un robot mobile est généralement conçu pour effectuer un certain nombre de tâches que l'on peut classer en fonction du type d'environnement dans lequel il évolue (milieu structuré ou semi-structuré, environnement naturel ...), du degré de connaissance de cet environnement, de son caractère évolutif (obstacles fixes ou mobiles) ou encore du domaine d'application pour lequel le véhicule a été conçu. En revanche ses performances sont étroitement liées à la précision, à la fiabilité et au temps de réponse des capteurs et des méthodes mises en oeuvre pour localiser le véhicule.

Ce travail de thèse se positionne autour du problème de localisation et notamment sur l'élaboration d'un système de localisation susceptible de fournir des informations précises et fiables en vue du contrôle d'un véhicule dans un environnement routier. La conception de ce genre de système a déjà été abordée et partiellement résolue en robotique agricole ou encore dans la conception d'engins de chantier automatisés. Dans ces cas, les approches de localisation se basent en général sur la combinaison d'informations provenant de systèmes de positionnement par satellites tels que les GPS différentiels ou cinématiques et de systèmes inertiels de haute précision. Cependant les contraintes du monde automobile :

- faible coût d'intégration,

- facilité de mise en œuvre,
- fiabilité et robustesse

des systèmes embarqués dans les véhicules empêchent l'utilisation de telles approches. En effet les GPS différentiels ou cinématiques ne répondent à aucune de ces contraintes :

- ils sont chers,
- leur mise en œuvre est relativement compliquée (nécessité d'avoir une station de base à proximité du véhicule),
- ils sont peu fiables (satellites souvent masqués en milieu urbain dense).

Quant aux systèmes inertiels de haute précision ils sont eux trop chers et souvent trop difficiles à étalonner pour pouvoir être embarqués dans un véhicule routier.

Ainsi les approches existantes de localisation précise élaborées dans la communauté de la robotique mobile ne sont que peu enclines à être utilisées dans le monde automobile. Cette thèse présente un système de localisation répondant à chacune de ces contraintes. Ce système se base sur la combinaison d'informations provenant d'un ensemble de capteurs proprioceptifs et extéroceptifs de faibles coûts tels qu'un GPS naturel, un odomètre et un gyromètre de technologie MEMS, un système de vision ainsi que d'une carte précise de l'environnement routier. Le système de vision, encore appelé capteur "vision" permet :

- de localiser précisément le véhicule sur la chaussée c'est-à-dire de connaître sa position latérale et son orientation sur la route,
- de localiser le véhicule dans le monde en effectuant une mise en correspondance avec la carte de l'environnement routier et donc fournir une information de localisation redondante avec le GPS.

Deux avantages importants résultent de l'utilisation de ce capteur :

- améliorer la précision du système GPS + capteurs proprioceptifs bas coûts permettant d'approcher les performances d'un système combinant GPS différentiels et système inertiel de haute précision,
- amélioration de la robustesse et de la fiabilité du système en suppléant le GPS lorsque l'intégrité de ses informations est remise en cause en particulier dans les zones urbaines denses où le GPS peut être mis en défaut par les occultations des satellites provoqués par les bâtiments, les tunnel etc...

Ce mémoire se divise en deux parties auxquelles se rajoutent les annexes.

La première partie se décompose en trois chapitres. Un premier chapitre où sont présentées les diverses méthodes de positionnement les plus souvent usitées dans le domaine de la robotique mobile à savoir : les méthodes de positionnement absolu, les méthodes de positionnement relatif et enfin les méthodes de positionnement hybrides qui combinent les deux précédentes. L'approche proposée dans cette thèse est une méthode de positionnement de type hybride, elle nécessite donc la mise en place d'un système de fusion de données pour combiner les informations provenant de différents capteurs embarqués dans le véhicule. Pour cela le filtrage bayésien a été retenu, les principes en sont présentés dans le deuxième chapitre où l'accent est surtout mis sur le filtre de Kalman et le filtre de Kalman étendu pour les méthodes analytiques et sur les filtres de Monté-Carlo et les dif-

férents filtres particuliers pour les méthodes numériques. Comme les filtres de Kalman sont depuis longtemps bien définis, un simple rappel est effectué dans ce mémoire. En revanche les filtres de Monté-Carlo et notamment les filtres particuliers avec interaction dont les travaux sont encore récents sont plus détaillés dans ce chapitre. Enfin pour espérer réaliser un filtrage optimal, il est nécessaire de caractériser au mieux les informations utilisées et en particulier leur caractère stochastique, le dernier chapitre de cette partie présente les différentes manières de caractériser puis de modéliser les informations stochastiques comme la modélisation de Gauss-Markov, la modélisation Autorégressive etc...

La seconde partie rassemble l'ensemble des développements originaux de cette thèse. Le quatrième chapitre aborde le choix des différents capteurs pour notre approche de localisation hybride ainsi que la caractérisation de leurs informations, de leurs incertitudes et de leur comportement temporel. Notamment ce chapitre insiste surtout sur la caractérisation et la modélisation des incertitudes des informations provenant du gyromètre et du GPS dont le comportement stochastique est très souvent négligé dans de telles applications ainsi que sur la description détaillée du capteur hybride "vision". Deux méthodes de filtrages bayésiens ont été retenues pour réaliser la combinaison d'informations provenant de ces divers capteurs : le filtre de Kalman étendu et le filtre particulière génétique. Ainsi le cinquième et le sixième chapitre présentent respectivement la mise en œuvre de notre approche de localisation avec chacun de ces deux filtres ainsi que de nombreux résultats attestant la validité de ces approches.

Enfin ce mémoire se termine en donnant quelques conclusions et perspectives sur ce travail.

Première partie

Etat de l'art



# Chapitre 1

## Les méthodes de positionnement

En robotique mobile, il existe trois méthodes différentes de positionnement. (1) **La localisation relative** où le déplacement relatif du véhicule est déterminé par intégrations successives d'informations décrivant la cinématique du véhicule, c'est-à-dire les composantes de vitesse et d'accélération selon une attitude d'origine connue du véhicule. (2) **La localisation absolue** où les capteurs fournissent des informations permettant de déterminer directement la position du robot dans un référentiel donné (ex : référentiel géodésique pour le positionnement par GPS, le référentiel de la carte des amers visuels pour le positionnement par vision etc...). Et enfin (3) **le positionnement hybride** où le robot est localisé en combinant les méthodes de positionnement relatif et de positionnement absolu, améliorant ainsi la localisation du véhicule en compensant les différents problèmes rencontrés dans les deux méthodes précédentes (dérives à long terme de la position du véhicule pour le positionnement relatif et les problèmes d'occultations pour le positionnement absolu). En revanche la combinaison des informations oblige la mise en œuvre de processus de fusion de données réalisés en général par des méthodes de filtrage bayésien. Une étude bibliographique de ces différentes méthodes va être présentée dans les trois prochaines sections.

### 1.1 Le positionnement relatif

Le positionnement relatif peut être lui même décomposé en trois catégories : le positionnement par odométrie, le positionnement inertiel et le positionnement relatif "hybride" dans lequel positionnement par odométrie et positionnement inertiel sont combinés.

#### 1.1.1 Positionnement par odométrie

La localisation par odométrie est très répandue en robotique mobile de par sa facilité, son faible coût de mise en œuvre ainsi qu'une cadence d'acquisition des mesures très élevée. L'odométrie permet de fournir une quantification des déplacements curvilignes du véhicule en mesurant la rotation des roues. Ces mesures sont généralement réalisées par des codeurs incrémentaux couplés à l'axe des roues, la précision de la mesure dépend ici

directement de la résolution du codeur. L'information d'orientation peut être aussi déduite de la différence de distance parcourue par la roue droite et gauche du véhicule. Le calcul de la position relative et du cap du véhicule  $(x, y, \theta)^T$  est réalisé par intégrations successives de ses déplacements élémentaires. La connaissance du diamètre des roues, l'entre axes des roues et de la cinématique du véhicule est ici requise.

### 1.1.1.1 Le modèle cinématique de type char

Considérons que la cadence d'acquisition des mesures odométriques soit assez élevée pour réaliser des petits déplacements élémentaires et que le véhicule se déplace sans glissement (cf figure 1.1). En premier lieu, calculons la vitesse longitudinale, la vitesse latérale du véhicule selon l'axe  $(x_v, y_v)$  et la vitesse de rotation du véhicule en fonction des vitesses angulaires des roues  $w_g$  et  $w_d$ .

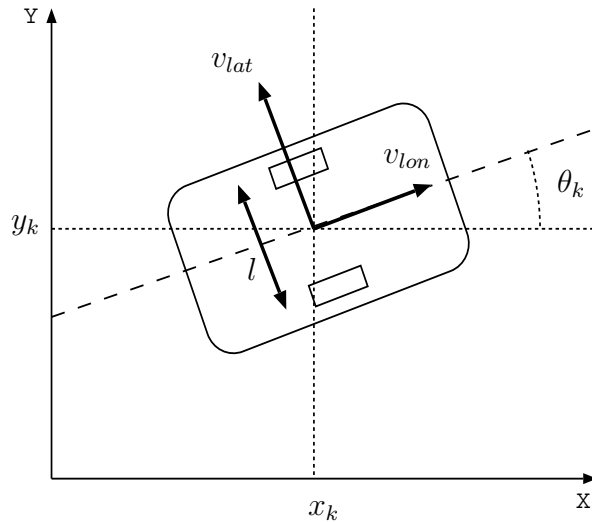


FIG. 1.1: Modèle cinématique de type char

$$\begin{cases} v_{lon} &= \frac{r(w_d + w_g)}{2} \\ v_{lat} &= 0 \\ \dot{\theta} &= \frac{r(w_d - w_g)}{l} \end{cases} \quad (1.1)$$

où  $r$  est le diamètre des roues (supposé identique pour les deux roues) et  $l$  la distance séparant les deux roues. Notons que l'hypothèse de déplacement sans glissement impose que la vitesse selon l'axe latéral du véhicule soit considérée nulle.

La vitesse longitudinale et la vitesse latérale du véhicule selon l'axe  $(x, y)$  sont données par les relations :

$$\begin{cases} v_{lat} &= -\dot{x} \sin(\theta) + \dot{y} \cos(\theta) \\ v_{lon} &= \dot{x} \cos(\theta) + \dot{y} \sin(\theta) \end{cases} \quad (1.2)$$

En tenant compte des équations (1.1) et (1.2) et en intégrant dans le temps les vitesses longitudinale, latérale et de rotation du véhicule il est possible de connaître les déplacements élémentaires du véhicule. Ainsi le déplacement du véhicule d'un instant  $k - 1$  à un instant  $k$  est décrit par :

$$\begin{cases} x_k = x_{k-1} + \frac{r(w_d + w_g)}{2} \Delta_T \cos(\theta_k) \\ y_k = y_{k-1} + \frac{r(w_d + w_g)}{2} \Delta_T \sin(\theta_k) \\ \theta_k = \theta_{k-1} + \frac{r(w_d - w_g)}{l} \Delta_T \end{cases} \quad (1.3)$$

avec  $\Delta_T$  le temps écoulé entre les instant  $k$  et  $k - 1$

### 1.1.1.2 Modèle cinématique de type tricycle

Ce modèle se ramène à considérer un véhicule avec deux roues arrières et une roue directrice avant (cf figure 1.2). L'état du véhicule est alors décrit par le vecteur de position  $(x, y, \theta)^T$  et l'angle de braquage de la roue  $\delta_r$ . Selon que les capteurs sont montés sur les roues arrières où les roues avant du véhicule, deux formulations de ce modèle cinématique existent, et sont successivement décrites par les équations (1.4) et (1.5) :

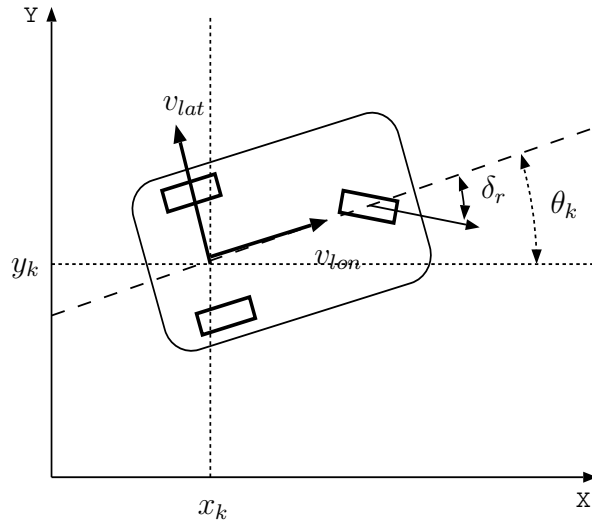


FIG. 1.2: Modèle cinématique de type tricycle

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\delta}_r \end{pmatrix} = \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \\ \tan \frac{\delta_r}{l} \\ 0 \end{pmatrix} v_{lon} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v_r \quad (1.4)$$



$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\delta}_r \end{pmatrix} = \begin{pmatrix} \cos(\delta_r) \cos(\theta) \\ \sin(\delta_r) \sin(\theta) \\ \sin \frac{\delta_r}{l} \\ 0 \end{pmatrix} v_{lon} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v_r \quad (1.5)$$

où  $\dot{\delta}_r$  est la vitesse de braquage de la roue directrice et  $v_{lon}$  la vitesse longitudinale des roues arrières ou des roues avants du véhicule selon la disposition des odomètres.

Considérons l'équation (1.4), en admettant que l'angle de braquage de la roue directrice soit connu à tout instant (l'estimation de  $\dot{\delta}_r$  devient inutile) et en intégrant dans le temps  $\dot{x}$ ,  $\dot{y}$ ,  $\dot{\theta}$ , le déplacement du véhicule entre les instant  $k$  et  $k - 1$  est :

$$\begin{cases} x_k = x_{k-1} + v_{lon} \Delta_T \cos(\theta_k) \\ y_k = y_{k-1} + v_{lon} \Delta_T \sin(\theta_k) \\ \theta_k = \theta_{k-1} + \tan \frac{\delta_r}{l} \end{cases} \quad (1.6)$$

Ce modèle est réaliste sous la contrainte qu'à tout instant  $|\theta| \leq \theta_{max} \leq \frac{\pi}{2}$ , ce qui traduit que l'angle de braquage de la roue directrice doit être borné. Cette contrainte peut être réécrite en fonction du rayon de courbure instantanée maximal  $R_{max}$  et la vitesse longitudinale ainsi

$$|\theta| \leq |v_{lon}| R_{max} \quad \text{où} \quad R_{max} = \frac{l}{\tan(\delta_{r_{max}})} \quad (1.7)$$

### 1.1.1.3 Modélisation des erreurs odométriques

Le principal défaut de cette méthode est une dérive de la position proportionnelle à la distance parcourue par le véhicule, principalement une grande erreur sur la position, ce à cause de l'accumulation d'erreurs lors des intégrations successives. D'après Boreinstein dans [Boreinstein *et al.*, 1996a] ces erreurs peuvent être classées en deux catégories :

1. **les erreurs systématiques** : ces erreurs s'ajoutent à chaque itération de la localisation du véhicule et sont de deux natures différentes : a) les erreurs sur les paramètres mécaniques du véhicule (les roues de diamètres différents, erreurs sur la longueur entreaxe des roues, non alignement des roues, etc ...) et b) les erreurs de mesures (résolution des codeurs, fréquence d'échantillonnage inexacte, etc ...). Les erreurs systématiques du premier du type sont en général compensées en utilisant des méthodes de calibrage alors que celles du second type ne peuvent être corrigées et doivent être prises en considération dans le modèle cinématique du véhicule.
2. **les erreurs non-systématiques** : ces erreurs sont directement dues aux contacts véhicule / terrain. Elles présentent un caractère aléatoire et dépendent en général des caractéristiques du sol : surface de roulement irrégulière (trous et bosses), présence d'objets sur la route ou encore glissement du véhicule (virages, terrain glissant). Du

fait de l'impossibilité de les prédire, les erreurs non-systématiques sont quasiment impossibles à compenser sans recours à des méthodes de positionnement absolu.

Selon l'environnement dans lequel évolue le véhicule, l'une ou l'autre de ces sources d'erreurs devient prépondérante devant l'autre. Par exemple dans le cadre d'applications en environnement intérieur, les erreurs non-systématiques étant quasi-inexistantes, ce sont les erreurs systématiques qui constituent la cause principale des dérives de la position. En revanche dans un environnement extérieur, ce qui est notre cas, ce sont les erreurs non-systématiques qui jouent un rôle primordial dans cette dérive.

Une analyse des phénomènes physiques à l'origine des erreurs odométriques est très compliquée à mettre en œuvre. D'après [Rives *et al.*, 2001] on peut considérer que l'écart-type sur la distance parcourue  $\Delta_{D_k}$  par un odomètre à l'instant  $k$ , est constitué d'un terme représentant l'erreur sur la mesure du codeur (de variance  $\sigma_q$ ) et d'un terme représentant l'influence des erreurs sur les paramètres de localisation. Ce terme est modélisé par un bruit blanc dont l'écart-type est proportionnel à la distance mesurée ( $\epsilon_{\Delta_D} \Delta_{D_k}$ ) :

$$\sigma_{\Delta_{D_k}} = \sqrt{\sigma_q^2 + (\epsilon_{\Delta_D} \Delta_{D_k})^2} \quad (1.8)$$

### 1.1.2 Positionnement inertiel

Cette technique de localisation se base sur l'utilisation des systèmes inertiels de navigation (INS), dont le nom est dû au fait de l'utilisation des propriétés inertielles de la matière et de la lumière. D'après les lois de la mécanique formulées par Isaac Newton, la localisation du véhicule est déduite de la double intégration des composantes accélérométriques  $a$  du véhicule :

$$\text{vitesse} \quad v = \int a \, dt \quad (1.9)$$

$$\text{position} \quad p = \int v \, dt \quad (1.10)$$

Pour réaliser le positionnement dans un système de référence il est alors nécessaire de connaître l'orientation spatiale des composantes accélérométriques. Ainsi les systèmes INS sont composés de trois accéléromètres orientés à l'aide de trois gyroscopes.

$$\text{orientation} \quad \theta = \int w \, dt \quad (1.11)$$

#### 1.1.2.1 Les différents systèmes INS

Les différents systèmes inertiels peuvent être classés en deux catégories : les systèmes "à plates-formes stabilisées" et les systèmes "à composant liés". Selon les performances et le prix de revient envisagés pour la localisation du robot, ces systèmes peuvent être composés de différents types de capteurs inertiels. Dans la suite de cette section, les

deux catégories de systèmes et les différents capteurs inertiels les plus utilisés sont décrits succinctement, de plus amples informations pourront être trouvées dans [Shaikh *et al.*, ].

**les systèmes inertiels "à plates-formes stabilisées" :** ce sont les premiers systèmes de navigation inertielle, les capteurs sont solidaires d'une plate-forme isolée des mouvements mécaniques (figure 1.3) et le positionnement est simplement réalisé par une double intégration des composantes d'accélération du véhicule. Après un phase initiale d'alignement, l'attitude de la plate-forme est asservie mécaniquement en fonction des mesures gyroscopiques et conserve donc une attitude constante définissant le repère du système de référence. Ce genre de système est très performant et offre l'avantage de protéger les capteurs de la plupart des vibrations encourues par le véhicule, prolongeant ainsi la durée de fonctionnement du système. En revanche les mécanismes d'asservissement sont complexes, coûteux et encombrants.

**les systèmes inertiels "à composant liés" :** encore appelés systèmes inertiels "strap-down", ils sont les fruits du progrès technologique et en particulier de l'informatique. Dans ce dispositif, les accéléromètres et les gyromètres sont directement solidaires du corps du véhicule. Les accélérations mesurées directement dans le repère lié au véhicule sont converties dans le repère de référence en fonction de l'attitude mesurée par les gyromètres. Ainsi les systèmes d'asservissement mécaniques sont complètement remplacés par un système informatique diminuant ainsi son coût de fabrication. Ces systèmes sont donc de structure très simple avec un faible encombrement spatial et résistants aux chocs, ils sont donc particulièrement appropriés pour le domaine automobile. Néanmoins, cette configuration présente l'inconvénient de soumettre les capteurs aux vibrations du véhicule, leurs mesures sont alors très bruitées et nécessitent d'importants pré-traitements dégradant ainsi la précision par rapport aux systèmes à plates-formes stabilisées.

**les capteurs inertiels :** les capteurs inertiels touchent un large éventail des sciences contemporaines comme la mécanique, l'électronique, l'optronique ou encore la physique atomique. Dans les dernières décennies, trois catégories de gyroscopes se sont démarquées, 1)les Ring Laser Gyros , 2)les Fiber Optic Gyros et 3)les Micro-Electro-Mechanical Systems gyros.

1. le RLG : créé dans les années 60 et apparu dans les systèmes strapdown dans les années 80 est un excellent capteur inertielle. Il a une très bonne stabilité et linéarité du facteur d'échelle, une faible sensibilité à l'accélération ainsi qu'une bonne répétabilité de ses performances en fonction des variations de la température (compensation possible). Cependant ce capteur reste très onéreux et n'est utilisé que dans les applications militaires et l'aviation.
2. le FOG : inventé dans les années 80, il devait être une simple alternative bas coût aux RLG. Actuellement, il obtient des performances équivalentes voire supérieures

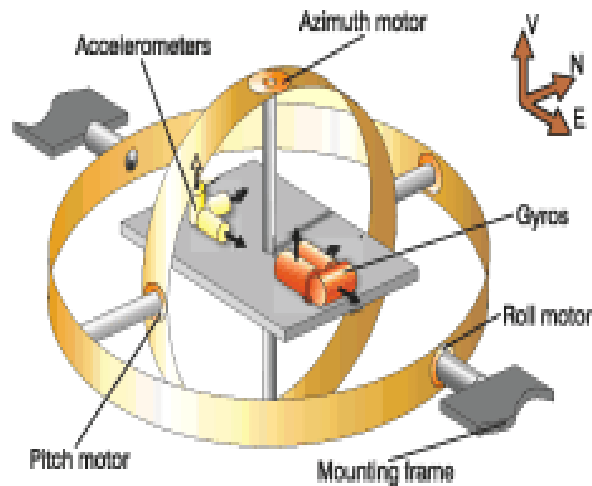


FIG. 1.3: Système inertiel à plate-forme stabilisée

aux RLG avec en particulier une très grande linéarité du facteur d'échelle et une insensibilité envers les vibrations. Il est désormais très compétitif dans les applications militaires et les applications commerciales.

3. les gyromètres MEMS : sont présentés comme les remplaçants des deux précédents capteurs et ceci dans de nombreux domaines d'application. Ils bénéficient d'une réduction de taille et de prix très important. Cependant ces avantages impliquent une baisse des performances telle que l'augmentation du bruit de mesure, une faible sensibilité (facteur d'échelle) etc... Mais cette technologie est en plein essor et on peut parier que ces capteurs deviennent de plus en plus compétitifs.

Quant aux accéléromètres les capteurs électromécaniques ont toujours dominé tous les domaines d'applications non seulement parce que leurs prix sont raisonnables mais parce qu'aucune technologie alternative leur ont succédé à part les accéléromètres à quartz résonant uniquement utilisés dans le guidage de missiles tactiques. Cependant depuis quelques années, là aussi les capteurs MEMS envahissent le marché et sont de plus en plus compétitifs. Une description plus détaillée de l'état de l'art des capteurs inertiels peut être trouvée dans [Barbour *et al.*, 1998] et [Boreinstein *et al.*, 1996a].

### 1.1.2.2 Modélisation des erreurs des capteurs inertiels

Dans [Hou, 2004] ou encore [Park, 2004] les auteurs affirment que l'on peut considérer que trois sources d'erreurs influencent la mesure d'un capteur inertiel (figure 1.4) à savoir : un biais sur la mesure, un mauvais calibrage du facteur d'échelle ou encore les non-linéarités de celui-ci et bien sur le bruit de mesure. A partir de ces hypothèses les mesures

d'un accéléromètre  $\tilde{a}$  et d'un gyromètre  $\tilde{w}$  peuvent être modélisées de la manière suivante :

$$\tilde{a} = K_a a + b_a + \epsilon_a \quad (1.12)$$

$$\tilde{w} = K_w w + b_w + \epsilon_w \quad (1.13)$$

avec :

- $a$  et  $w$  mesures fournies par les capteurs
- $K$  le facteur d'échelle,
- $b$  le biais sur la mesure,
- $\epsilon$  le bruit de mesure.

**Facteur d'échelle et non-linéarités :** le facteur d'échelle n'évolue pas ou peu dans le temps, en conséquence une procédure de calibration permet d'estimer ce paramètre de manière définitive. Comme pour tout capteur, des non-linéarités sont présentes, leur prise en compte peut être réalisée en modélisant le facteur d'échelle par un polynôme d'ordre supérieur ou égal à 2. Dans ce cas les paramètres d'ordre 0 et d'ordre 1 représentent le biais et le facteur d'échelle et les paramètres supérieurs ou égaux à 2 décrivent les non-linéarités du facteur d'échelle :

$$\tilde{a} = K_{a_0} + K_{a_1} a + K_{a_2} a^2 + \dots \quad (1.14)$$

$$\tilde{w} = K_{w_0} + K_{w_1} w + K_{w_2} w^2 + \dots \quad (1.15)$$

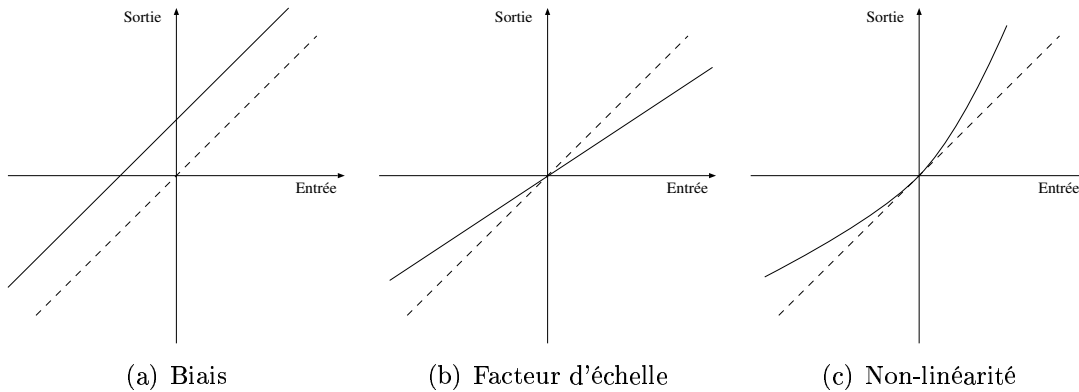


FIG. 1.4: Erreurs classiques de mesures inertielles

**Le biais :** il pourrait être facilement déterminé lorsque le système inertiel ne subit aucun mouvement. Cependant il est différent à chaque mise sous tension, les variations possibles du biais encore appelé "stabilité de biais" sont souvent données dans les spécifications techniques de la centrale inertielle. En outre, ce paramètre évolue en fonction de la température. Ce phénomène de dérive du biais est donc alors très important pendant les premières minutes après la mise sous tensions où les composants électroniques s'échauffent.

Plusieurs manières de modéliser cette dérive du biais existent dans la littérature :

1. la marche aléatoire :

$$b_k = b_{k-1} + \epsilon_b \quad (1.16)$$

cette modélisation considère que le biais est constant et permet d'avoir un ordre de grandeur de ce paramètre pendant une courte période d'observation notamment juste après la phase de préchauffage de la centrale inertielle.

2. le processus markovien :

$$b_k = -\beta_b b_{k-1} + \epsilon_b \quad (1.17)$$

cette modélisation utilisée dans [Park, 2004] [Abuhadrous, 2005] permet de prendre en compte les variations du biais dans le temps où  $\frac{1}{\beta_b}$  est le temps de corrélation de la variable aléatoire  $b$ .

3. modèle en fonction de la température :

$$b_k = b_0 + b_1 T_k + b_2 T_k^2 \quad (1.18)$$

cette modélisation proposée dans [Chung *et al.*, 2001] considère que le biais peut être estimé à tout instant par un polynôme du second ordre fonction de la température  $T$ . Les paramètres  $b_i$  ne sont pas constants mais varient en fonction de la vitesse de rotation  $w$ , il est alors nécessaire de calibrer ces paramètres en fonction de  $w$ .

**Le bruit de mesure :** inévitable comme dans tous capteurs, il a pour origine le bruit des composants électroniques, le bruit de quantification etc..., il est en général considéré comme un bruit blanc gaussien.

### 1.1.2.3 Influence des erreurs sur le positionnement

Reprenons les équations (1.10) et (1.11) et prenons en compte les modèles d'erreurs précédemment exposés, la vitesse et la position du véhicule sont alors :

$$\begin{aligned} \tilde{v} &= \int K_a a + b_a + \epsilon_a dt \\ &= v + b_a t + \int \epsilon_a dt \end{aligned} \quad (1.19)$$

$$\begin{aligned} \tilde{p} &= \iint K_a a + b_a + \epsilon_a dt \\ &= p + \frac{b_a t^2}{2} + \iint \epsilon_a dt \end{aligned} \quad (1.20)$$

l'orientation est :

$$\begin{aligned} \tilde{\theta} &= \int K_w w + b_w + \epsilon_w dt \\ &= \theta + b_w t + \int \epsilon_w dt \end{aligned} \quad (1.21)$$

A partir de ces équations, il est aisé de voir que le biais est le terme le plus pondérant dans la mesure de la vitesse, de la position et de l'orientation du mobile. En effet il cause une dérive qui augmente linéairement en fonction du temps pour la vitesse et l'orientation et quadratiquement pour la position.

Pour limiter ces phénomènes on doit soit :

- estimer le biais à chaque instant en prenant en compte un des modèles de biais vu dans la section précédente et bien sûr des informations provenant d'un autre capteur tel qu'un GPS,
- recalibrer le biais en ligne à chaque fois que le véhicule reste immobile ce qui est assez fréquent pour les véhicules routiers.

Enfin, les bruits de mesure introduisent un phénomène de marche aléatoire dans les différentes mesures qui est impossible à compenser.

### 1.1.3 Autres méthodes

#### 1.1.3.1 Méthode de positionnement hybride

Une localisation précise par une approche purement inertielle semble incompatible avec les applications robotiques en raison du prix des capteurs inertiels. En effet les capteurs accessibles en terme de coûts présentent l'inconvénient majeur de dériver rapidement avec le temps, (processus d'intégration présenté dans la section précédente). Pour remédier à ce problème de nombreux systèmes [Vaganay, 1993][Thrapp *et al.*, 2001] proposent de combiner simplement l'odométrie pour mesurer la vitesse linéaire et un gyromètre pour mesurer la vitesse de rotation du véhicule. La localisation est dans ce cas moins précise et nécessite d'être recalée par des informations extéroceptives mais elle présente l'avantage d'être facile à mettre en œuvre et peu onéreuse.

Une autre méthode proposée dans [Boreinstein *et al.*, 1996b] appelée gyrodométrie repose sur le constat expérimental qu'en présence de sources d'erreurs non-systématiques, les mesures de cap fournies par l'odométrie et la gyrométrie diffèrent considérablement alors qu'en absence de ces sources d'erreurs elles restent très similaires. Les auteurs proposent de déduire le cap du véhicule en utilisant seulement les informations odométriques sauf lorsqu'elles sont trop différentes de des données gyrométriques. Celles-ci sont alors utilisées pour estimer le cap. Ce système permet de minimiser les dérives dans l'estimation de la position du véhicule liées à l'utilisation de gyromètre bas de gamme.

#### 1.1.3.2 Positionnement par effet Doppler

Le radar doppler fournit une estimation instantanée de la vitesse linéaire du mobile par rapport à un objet de la scène (en général le sol) en se basant sur l'effet Doppler Fizeau. Le principe est de diriger un faisceau électromagnétique de fréquence  $F$  vers le sol, le faisceau reçu après la réflexion sur le sol présente un décalage de fréquence  $\Delta F$

proportionnel à la vitesse  $V$  relative au déplacement du véhicule par rapport au sol. Enfin l'intégration de la mesure de la vitesse donnera une estimation du déplacement du véhicule. Ce système surtout utilisé dans le domaine agricole pour remplacer l'odométrie [Boreinstein *et al.*, 1996a] [Stentz *et al.*, 2002], présente l'avantage d'être insensible au glissement des roues par rapport au sol et aux erreurs induites par la pression des pneus dans le modèle cinématique du véhicule. En revanche, un tel système est lourd à mettre en œuvre tant au point de vue de l'électronique que du traitement de l'information. De plus la précision sur la mesure se dégrade fortement avec la diminution de la vitesse du véhicule et est très dépendante des incertitudes d'angle d'incidence du radar (surtout dans le cas de fort tangage du véhicule).

### 1.1.4 Bilan

Le positionnement basé sur les mesures odométriques est un système peu coûteux, rapide et précis à court terme. D'ailleurs dans le cas d'applications routières, il est devenu aisé de mettre en œuvre puisque les véhicules récents fournissent ces mesures par l'intermédiaire des systèmes ABS. Cependant plusieurs sources d'erreurs peuvent dégrader le positionnement, en particulier les erreurs systématiques et les erreurs non-systématiques, mais il ne faut pas perdre de vue que cette méthode de positionnement est exclusivement 2D et ne peut pas prendre en compte la géométrie du terrain (angle de site, de gisement et d'azimut) qui peut induire très fortement le système en erreur.

Le positionnement par système inertiel a pour avantages de fournir de nombreuses informations telles que la position, la vitesse, l'orientation, l'accélération et la rotation du véhicule, d'être indépendant des paramètres géométriques du véhicules, peu sensible aux conditions de roulement et donc quasi-insensible aux erreurs systématiques et non-systématiques qui affectent l'odométrie. Cependant pour des raisons économiques les capteurs inertiels utilisés dans le positionnement de véhicule routier sont en général de moindres coûts, donc de moindres qualités, ainsi le positionnement est alors soumis à de fortes dérives dues à la double intégration du biais et au phénomène de marche aléatoire lié à l'intégration des bruits de mesures.

Enfin pour diminuer l'influences des dérives dans ces deux méthodes, il est nécessaire d'effectuer un recalage régulier par un système de positionnement absolu.

## 1.2 Le positionnement absolu

### 1.2.1 Description

Les méthodes de positionnement absolu reposent généralement sur la perception d'éléments remarquables disposés dans l'environnement du véhicule ou du robot. Celui-ci est alors localisé dans un repère global lié à cet environnement. Pour répondre à la problématique de la localisation absolue deux types de stratégies peuvent être rencontrées :

- soit les éléments remarquables transmettent des informations au système de perception et la localisation est simplement réalisée par une trilatération ou une triangulation de ces informations,



- soit les éléments remarquables sont passifs (balises artificielles ou amers naturels disposés dans l’environnement) auquel cas le système de perception doit d’abord les détecter avant d’effectuer la localisation du véhicule.

Bien sûr, quel que soit le cas de figure, le processus de positionnement nécessite toujours une connaissance de l’environnement généralement donnée sous forme de carte.

**Éléments remarquables actifs :** Dans ce cas, les informations liées aux éléments remarquables sont en général transmises au système de perception par des moyens de communication radio fréquence et la localisation est réalisée par une trilatération de ces informations. Cette méthode est largement utilisée pour le positionnement de bateaux [Monroe *et al.*, 1998], d’avions [Kayton *et al.*, 1997] et est le principe de positionnement par GPS (cf section 1.2.2) ou encore par téléphonie mobile (cf section 1.2.3).

**Éléments remarquables passifs :** Dans cette situation, le positionnement du véhicule repose sur la perception et la cartographie de l’environnement proche du véhicule à l’aide de capteurs extéroceptifs tels que le radar [Dissanayake *et al.*, 2001], les capteurs télémétriques que ce soit par laser [Guivant *et al.*, 2000] [Brenneke *et al.*, 2003] ou par ultrason [Kleeman *et al.*, 1995] [Tardos *et al.*, 2002] ou encore par des systèmes de vision. Le processus de localisation se décompose alors en trois étapes : 1) construction d’une carte locale de l’environnement en fonction des informations provenant des capteurs extéroceptifs, 2) mise en correspondance de cette carte avec une carte globale de l’environnement et 3) calcul (en général triangulation des informations extéroceptives une fois la mise en correspondance effectuée) de la position et de l’orientation du véhicule.

Fondamentalement deux cas de figures peuvent être distingués, soit la carte globale de l’environnement est disponible et le positionnement s’effectue comme présenté ci-avant. Soit elle n’est pas disponible auquel cas le système de perception doit construire sa propre carte globale au fur et à mesure de son exploration de l’environnement, on parle alors de SLAM<sup>1</sup> [Dissanayake *et al.*, 2001] [Guivant *et al.*, 2000].

L’avantage des méthodes de positionnement absolu par rapport aux méthodes de positionnement relatif est l’élimination du phénomène de dérive rencontré avec ces dernières. En contre partie la localisation dépend directement de la visibilité des éléments remarquables par le système de perception et nécessitent bien souvent un aménagement adéquate de l’environnement et donc un investissement lourd en équipement. De plus tout changement de configuration de cet environnement implique une remise en cause de son aménagement et de sa carte. Enfin les traitements des informations issues des différents capteurs sont en général plus lourds que pour les capteurs proprioceptifs.

**Méthodes de positionnement absolu envisageables pour notre approche :** Tout d’abord, nous rappelons que notre approche doit permettre la localisation d’un véhicule dans le réseau routier en considérant que sa carte existe (carte décrivant les limites des

---

<sup>1</sup>Simultaneous Localization And Mapping

voies de circulation) et qu'aucun aménagement de son infrastructure n'est envisageable (pas de balises artificielles). Alors seulement deux méthodes de positionnement absolu présentées ci-dessus peuvent être utilisables : les méthodes de positionnement à partir d'éléments remarquables actifs comme le positionnement par GPS ou par téléphonie mobile et la méthode de positionnement avec éléments remarquables passifs "naturels" (dans notre cas marquages signalétiques au sol, limites de la chaussée) par système de vision. Ces différentes méthodes sont présentées dans la suite de cette section.

## 1.2.2 Positionnement par satellites

### 1.2.2.1 Description des systèmes existants

Trois systèmes [Thurston, 2002] de positionnement par satellites existent de par le monde, deux sont déjà opérationnels ce sont le système américain GPS NAVSTAR et le système russe GLONASS. Le troisième, c'est-à-dire le système européen GALILEO, est en cours de construction.

**GPS NAVSTAR :** Le système GPS (Global Positioning System) NAVSTAR a été mis en oeuvre par le ministère de la défense Américaine dès les années 1970. De ce fait il est entièrement sous contrôle américain et leur permet une disponibilité sélective du signal en cryptant certaines informations. Ce système est composé de 24 satellites NAVSTAR sur 6 orbites inclinés de  $55^\circ$  à une altitude de 20184 km et de 5 stations américaines au sol dont la principale se trouve aux Etats-Unis. Le système GPS est mondialement connu et gratuit et permet une localisation avec une précision de 10m à  $2\sigma$ , ce qui le rend le plus utilisé au monde. Néanmoins il faut rappeler que certaines zones géographiques ne sont pas couvertes et que le système peut voir ses performances dégradées volontairement par les américains en cas de guerre.

**GLONASS :** GLONASS (GLObal Navigation Satellite System) [Glonass, 2002] est l'équivalent Russe du GPS. Il a été développé à partir de 1982 par les militaires Russes et est opérationnel depuis 1997. GLONASS est composé de 24 satellites sur 3 orbites (permet une réorganisation plus rapide des satellites en cas de panne) inclinés de  $64.8^\circ$  à 19130 km d'altitude et de 5 stations au sol. La particularité de GLONASS est que chaque satellite possède sa propre fréquence émission ce qui augmente la résistance au brouillage. Malheureusement, à l'heure actuelle seuls 7 satellites sont actifs à cause des problèmes économiques que connaît la Russie.

**GALILEO :** GALILEO [Benedicto *et al.*, 2001] [Galileo, 2002] sera le futur système de positionnement par satellites à l'initiative de l'Union Européenne et de l'Agence Spatiale Européenne. La nécessité d'un système de positionnement Européen répond au besoin d'indépendance face au GPS américain. En effet, en cas de conflit ou mésentente avec les USA, l'accès des européens au signal GPS pourrait être dégradé. Ainsi l'Europe mise sur un système novateur et performant qui supprimera les restrictions (précision et fiabilité)

du GPS . L'un des atouts de GALILEO sera d'être entièrement sous contrôle civil. Le calendrier de réalisation de GALILEO est le suivant :

- 2001-2006 : Développement
- 2006-2008 : validation en orbite
- 2008-2010 : Déploiement
- A partir de 2010 : Exploitation commerciale.

Le système GALILEO sera composé de 27 satellites actifs (plus 3 de secours) placés sur 3 orbites à 23616 Km d'altitude et 2 centres de contrôles en Européen coordonnant un minimum de 20 stations au sol. De plus, GALILEO diffusera 10 signaux sur 3 bandes de fréquences, 6 pour les services gratuits, 2 pour le service commercial et 2 pour le service public réglementé. Ces différents services sont :

- Service ouvert (ou OS pour Open Service)
- Service commercial ( ou CS pour Commercial Service)
- Service de la sûreté de la vie (ou SOL pour Safety of Life Service)
- Service de recherche et secours (ou SAR pour Search And Rescue service)
- Service public réglementé (ou PRS pour Public Regulated Service)

En conclusion, GALILEO est à la fois le concurrent et le complémentaire du GPS, fournissant une plus grande fiabilité et une moindre vulnérabilité.

### 1.2.2.2 Principe du positionnement

Le positionnement par satellites peut être réalisé selon deux méthodes différentes. La première est une simple trilatération en fonction des informations émis par les satellites visibles, cette méthode se nomme le positionnement absolu par satellites. L'autre méthode utilise une station de base (dont la position est parfaitement connue) qui transmet au récepteur les corrections des erreurs de trilatérations, permettant ainsi un positionnement très précis du mobile, cette méthode s'appelle le positionnement relatif par satellites.

**Positionnement absolu** Le positionnement par satellite obéit au principe de triangulation. Grâce à l'horloge atomique que possède chaque satellite, les ondes radio émises sur deux porteuses L1 et L2 émettent, à des instants connus, des signaux codés selon des fréquences pseudo-aléatoires. Une réplique de la séquence du code du satellite est générée par le récepteur en même temps que le satellite. Le décalage que doit subir la réplique afin de coïncider avec le code reçu correspond au temps de propagation, pris par le signal, pour parcourir la distance satellite-récepteur. Cette différence de temps multipliée par la vitesse de la lumière dans le vide donne une mesure de distance appelée pseudo-distance.

Une seule distance n'est pas suffisante pour déterminer un point, il faut donc théoriquement, comme dans le principe d'une trilatération, réaliser la même opération avec deux autres satellites, portant à trois le nombre de pseudo-distances. Ces trois "pseudo-distances" ne sont exploitables que si les coordonnées de chaque satellite sont connues. Or, le signal émis par chaque satellite contient un code propre à celui-ci qui peut être identifié par le récepteur, celui-ci connaissant les éphémérides des différents satellites peut obtenir les coordonnées des trois satellites au moment de l'émission du signal. La position

du récepteur est l'intersection des 3 sphères dont les centres sont chacun des satellites et de rayon la "pseudo-distance" séparant le satellite du récepteur (cf figure 1.5).

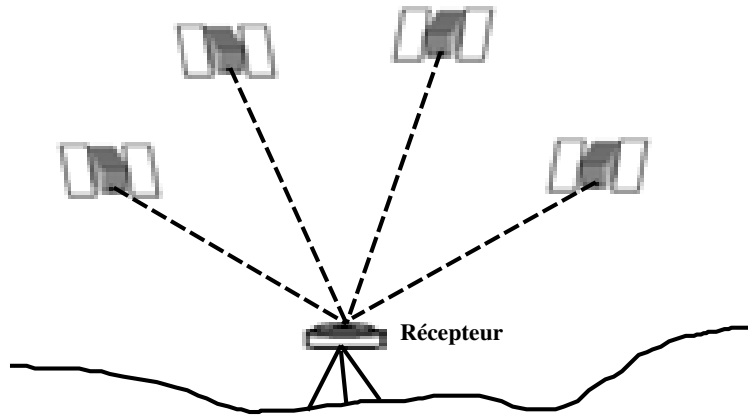


FIG. 1.5: Positionnement absolu par satellites

Cependant, dans la pratique les erreurs de synchronisation d'horloge entre les satellites et le récepteur ne permettent pas d'obtenir un point dans un repère en trois dimensions mais seulement une zone assez vaste, contenant le récepteur. Les erreurs de synchronisation peuvent être traduites comme une quatrième inconnue dans le processus de triangulation, ainsi une mesure d'une quatrième pseudo-distance permet de résoudre ce problème et de fournir les coordonnées tridimensionnelles ainsi que l'erreur d'horloge du récepteur.

**Mesure de la vitesse** Le système GPS permet également de mesurer la vitesse à laquelle se déplace l'utilisateur. Cette évaluation se fait par mesure Doppler sur le signal provenant d'un satellite GPS. En effet, le signal perçu par le récepteur GPS n'a pas exactement la même fréquence que lorsqu'il est généré par le satellite. Suivant le principe d'effet Doppler, le rapport des fréquences est fonction des vitesses relatives du satellite et de l'utilisateur. En outre, cette mesure permet de détecter et de corriger les sauts de phase potentiellement présents dans les mesures de phase (cf section suivante).

### 1.2.2.3 Positionnement relatif

**Le GPS différentiel :** Le GPS différentiel (DGPS) fonctionne exactement comme le système décrit précédemment, mais offre des performances nettement supérieures. Le changement s'effectue au niveau du récepteur, qui est équipé d'un boîtier supplémentaire permettant de recevoir des informations, ou plutôt des corrections en provenance de stations DGPS. Le principe du DGPS est relativement simple ; les mesures effectuées en un lieu de position connue sont comparées aux valeurs théoriques, et leur différence fournit alors une correction qui est retransmise vers l'utilisateur, dont le récepteur intégrera les corrections dans le calcul de sa position, améliorant considérablement sa précision. L'application de cette méthode peut se faire à différents niveaux, soit sur la position calculée, soit directement sur la mesure des pseudo-distances. Un récepteur GPS est placé en

un lieu de position connue : la station de référence. Ce récepteur reçoit les signaux émis par les satellites et mesure les pseudo-distances. Il les compare ensuite avec les distances théoriques, c'est-à-dire les distances entre le récepteur et chacun des satellites servant à la mesure de positionnement. La position exacte des satellites est connue grâce à des éphémérides que le récepteur charge en mémoire. Les corrections sont ensuite élaborées et retransmises vers l'utilisateur, via un message connu sous le nom de RTCM SC 104. Le récepteur mobile peut donc appliquer les corrections à ses propres mesures de pseudo-distances.

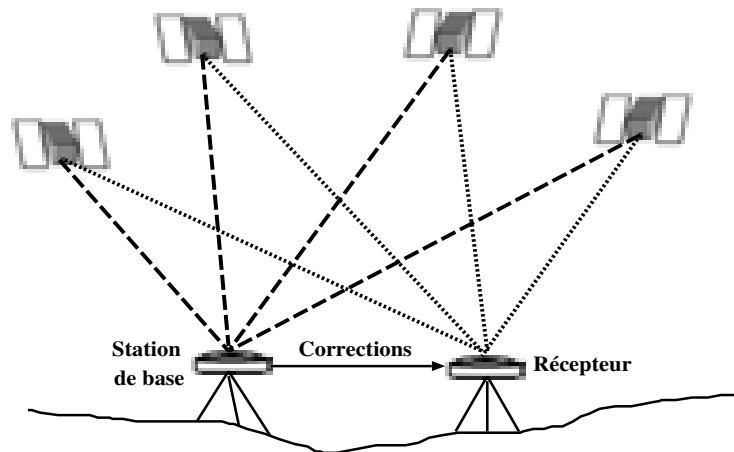


FIG. 1.6: Positionnement relatif par satellites

La technologie DGPS augmente considérablement la précision la faisant passer de 30 m à moins d'un mètre. L'inconvénient d'un tel système est sa portée, qui n'excède pas plusieurs centaines de kilomètres autour des stations d'émission. Cependant la couverture peut être nettement améliorée grâce à la retransmission des corrections par l'intermédiaire de satellites géostationnaires (cf section 1.2.2.5).

**La mesure de phase :** Les récepteurs les plus perfectionnés permettent d'exploiter directement la phase sur les ondes porteuses L1 et L2 en déterminant le déphasage entre le satellite et le récepteur. Le principe de ce système est la comparaison de la phase du signal reçu par le récepteur ( donc émis par le satellite ) avec la phase du signal générée par l'oscillateur à quartz interne au récepteur et qui est une réplique du signal satellite. A partir de cette mesure de phase, on peut obtenir une distance satellite-récepteur en multipliant cette phase par la longueur des ondes porteuses (19,0cm pour L1 et 24,4cm pour L2). La distance obtenue est la distance satellite-récepteur. Dans ce cas, le traitement des données peut se faire de deux façons : soit en mode postopératoire à l'aide de logiciels spécialisés ; soit en temps réel, auquel cas on parle alors de RTK<sup>2</sup>. Le principal inconvénient de ce système de positionnement est une phase d'initialisation et de réinitialisation très longue (plusieurs dizaines de secondes), ce phénomène est d'autant plus pénalisant

<sup>2</sup>Real Time Kinematic

dans les situations urbaines où la mesure de phase est souvent perdue par le récepteur à cause des occultations dues aux bâtiments, aux ponts etc... Malgré cet inconvénient et un prix élevé, la précision du GPS RTK (de l'ordre de 1 à 5 cm) peut justifier son utilisation dans de nombreuses applications de positionnement en environnement extérieur.

#### 1.2.2.4 Sources d'erreurs

Il existe de nombreuses sources d'erreurs pouvant dégrader le positionnement : la synchronisation de l'horloge des satellites, la géométrie de la constellation des satellites, la traversée des différentes couches de l'atmosphère, les effets de multi-trajet du signal GPS et les erreurs de mesures du récepteur.

**Horloge des satellites :** La synchronisation entre les horloges des satellites et celles des récepteurs est une source d'erreur importante sur la précision, pour remédier à cet obstacle technique, le positionnement s'effectue au moins avec 4 satellites. Cependant la nécessité d'avoir 4 satellites pour effectuer une mesure pose un problème, celui de la couverture GPS. En effet, la configuration actuelle des satellites ne couvre que 95% de la planète et donc dans 5% des cas la mesure de positionnement n'est possible qu'avec trois satellites, ce qui nuit grandement à sa précision.

**Géométrie des satellites :** La disposition de la constellation induit une forte dégradation géométrique de la précision deux fois par jours dans certaines zones limitées pour des durées de 5 à 30 minutes. Par ailleurs l'élévation des satellites et leurs positions relatives par rapport au récepteur va affecter la qualité des positions calculées. Cette qualité peut être caractérisée par le calcul des facteurs de dégradation géométrique appelés DOP<sup>3</sup> (cf annexe B) donnant ainsi à l'utilisateur une idée de la confiance à apporter à une mesure.

**La réfraction dans l'ionosphère :** Les ondes radios issues des satellites du système GPS orbitant à 20 000 km d'altitude doivent traverser en premier l'ionosphère, la propagation du signal GPS est alors perturbée comme toute onde électromagnétique traversant un milieu conducteur (l'ionosphère est composée d'ions et d'électrons). Cette perturbation se manifeste par un retard et conduit à une surestimation du temps de propagation du signal et donc la distance satellite-récepteur.

**La réfraction dans la troposphère :** Le temps de propagation de l'onde GPS est ici affecté par la teneur en vapeur d'eau de cette couche basse de l'atmosphère. Il serait donc nécessaire de connaître cette quantité avec précision tout le long du trajet suivi par l'onde pour établir les imprécisions sur les mesures de pseudo-distances qui en découlent.

---

<sup>3</sup>Dilution Of Precision

**Les trajets multiples :** Le phénomène des trajets multiples est dû à la réflexion des ondes radios émises par les satellites, par les infrastructures en milieu urbain. Ces réflexions vont souvent augmenter la distance du trajet satellite-récepteur et donc aussi la durée de propagation du signal. Le récepteur va alors calculer une distance satellite-récepteur fautive et donc commettre une erreur sur la position.

**Le bruit des mesures :** Le bruit de la mesure est estimé à moins de 1% de la longueur du signal sur lequel se fait la mesure, dans le cas d'une mesure de phase comme dans celui d'un calcul de pseudo-distance.

Dans la réalité, ces erreurs sont indépendantes les unes des autres et s'additionnent. Néanmoins, comme la trilateration est réalisée par une méthode des moindres carrés, la précision du système GPS est d'environ de 15m et celle du DGPS inférieure à un mètre.

Source d'erreurs	erreurs GPS (m)	erreurs DGPS (m)
Horloge atomique	10,5	0
Position du satellite	10	0
Traversée ionosphère	3,9	0,4
Traversée troposphère	9,8 à 19,6	0,2
Précision du récepteur	2,9	0,3
Réflexion parasites	2,4	0,6

TAB. 1.1: Erreurs des systèmes GPS et DGPS

Un tableau de facteurs d'erreurs dans le cas du GPS RTK n'est pas réalisable car les erreurs rencontrées dans les méthodes précédentes sont éliminées dans ce type de mesure pour obtenir une précision de quelques millimètres.

#### 1.2.2.5 Amélioration du système

**Système d'augmentation par satellites SBAS :** De façon à améliorer la sécurité des applications de navigation aérienne et maritime, différents pays placent des satellites géostationnaires dont le rôle est de transmettre aux utilisateurs sur la fréquence L1 des corrections précises (orbites GPS, horloges, ionosphère et troposphère) calculées par un réseau important de stations de contrôle au sol et des données d'intégrité informant des erreurs du système. Un gain de précision en positionnement absolu, surtout sur la composante verticale doit permettre d'atteindre une précision de 1 à 2m ainsi qu'une précision décimétrique en positionnement différentiel (cf tableau 1.2). Les systèmes d'augmentation par satellites [Fuller *et al.*, 1998] [Abousalem *et al.*, 2000] les plus connus sont WAAS<sup>4</sup> [Hughes *et al.*, 2005] pour l'Amérique du Nord, MSAS<sup>5</sup> pour le Japon et EGNOS

<sup>4</sup>Wide Area Augmentation System

<sup>5</sup>Multi-fonctional Satellite Augmentation System

Caractéristiques	GPS + WAAS	GALILEO + EGNOS	GNSS
Nbre de Satellites	28 + 3	30 + 3	58 + 6
HDOP - VDOP	1 - 3	1 - 3	0,7 - 2
Précision pos. absolu (m)	1 à 2	1 à 2	0,6 à 1,3

TAB. 1.2: Précisions obtenues par le système GNSS

<sup>6</sup> [Benedicto *et al.*, 1999] pour la Communauté Européenne.

**Global Navigation Satellite System GNSS :** Le GNSS est la réalisation de l'interopérabilité des différents systèmes de radionavigation par satellites existant dans le monde : actuellement contribuent au GNSS les systèmes GPS, GLONASS et les systèmes géostationnaires (WAAS, EGNOS...), le système GALILEO sera intégré aux alentours de 2010. Le GNSS reposera alors essentiellement sur les deux systèmes GPS (USA) et GALILEO (EU). L'indépendance et l'interopérabilité des deux systèmes apportera une continuité de service en cas de défaillance de l'un des deux systèmes ainsi que l'émission de messages d'intégrité propres à chacun des systèmes.

### 1.2.3 Positionnement par téléphone cellulaire

#### 1.2.3.1 Introduction

Ces dernières années, des systèmes de communication mobile ont été développés pour fournir la possibilité de localiser l'utilisateur d'un téléphone mobile par rapport à une station de base, l'idée principale étant de pouvoir localiser son utilisateur en cas d'urgence. Grâce à la densité du réseau de télécommunication dans la plupart des villes et des régions et de la possibilité de déterminer la position d'un téléphone, l'idée d'intégrer ce service dans les systèmes de navigation en complément du GPS est apparu. En effet les téléphones portables de deuxième génération peuvent déterminer la position d'un utilisateur n'importe où et n'importe quand avec une précision de 125m grâce au réseau GSM <sup>7</sup>. Et les téléphones portables de troisième génération utilisant le réseau UMTS <sup>8</sup> permettent une localisation environ dix fois plus précise.

#### 1.2.3.2 Principe du positionnement

Les opérateurs peuvent employer différentes techniques pour fournir les informations de localisation [Akerblom, 2000] [Wölfle *et al.*, 2002] :

- l'angle d'arrivée (AoA <sup>9</sup>)
- le temps d'arrivée (ToA <sup>10</sup>)

---

<sup>6</sup>European Geostationary Navigation Overlay Service

<sup>7</sup>Global System for Mobile communication

<sup>8</sup>Universal System for Mobile Communication

<sup>9</sup>Angle of arrival

<sup>10</sup>Time of arrival



- la différence de temps d'arrivée (TDoA <sup>11</sup>)
- la différence de temps d'arrivée dite augmentée (E-OTD <sup>12</sup>)

**AoA** : Avec au moins deux stations de base, la position est obtenue par l'intersection de deux lignes passant par chacune des stations de base et dont leurs orientations peuvent être déterminées (figure 1.7). De ce fait les angles de directions  $\Theta_i$  des signaux reçus par chacune des stations de base doivent être mesurés en utilisant des antennes appropriées. L'inconvénient majeur de cette méthode est dans le cas où aucune ligne n'existe entre le téléphone mobile et la station de base car la dernière mesure valide de l'angle est utilisée pour effectuer la localisation. De plus dans le cas de multi-trajet les signaux réfléchis peuvent interférer dans la mesure de l'angle.

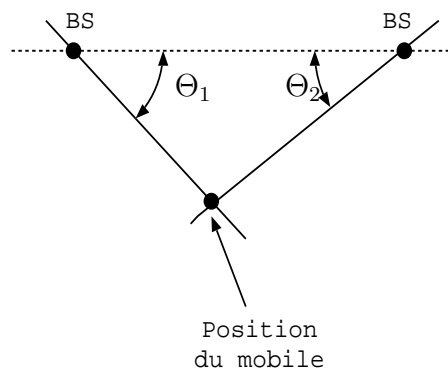


FIG. 1.7: Illustration de la méthode AoA

**ToA** : Dans ce cas, le principe de triangulation est utilisé pour calculer la position du téléphone mobile (figure 1.8). Ici les distances  $\delta_i$  obtenues par la mesure du temps de vol du signal émis par le mobile et au moins trois stations de base sont mesurées, cette mesure requière une synchronisation du récepteur et des stations de base. La position du mobile est déduite de l'intersection de trois cercles centrés autour des stations de base et de rayon  $\delta_i$ . Lorsque plus de 3 stations de base sont utilisées, des informations redondantes apparaissent lors du calcul de la position, ainsi il est possible de calculer la latence du mobile ainsi que son erreur d'horloge par la méthode des moindres carrés. Cette méthode est en général onéreuse car la synchronisation des stations de base et du mobile nécessite des modifications majeures de la station de base tant au point de vue de matériel que logiciel, en contrepartie aucune modification du téléphone mobile n'est nécessaire.

**TDoA** : Cette technique utilise des hyperboles pour déterminer la position (figure 1.9). Au lieu de mesurer le temps d'arrivée comme pour la méthode précédente, la méthode TDOA mesure au même instant la différence de temps du signal d'arrivée entre différentes

<sup>11</sup>Time of Difference of Arrival

<sup>12</sup>Enhanced observed time difference

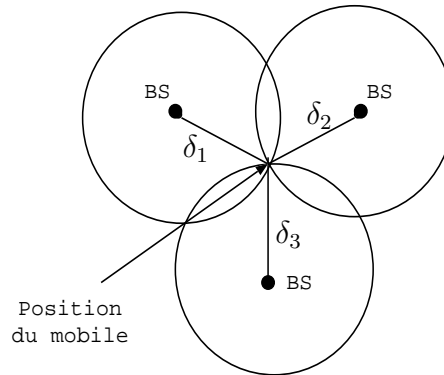


FIG. 1.8: Illustration de la méthode TOA

stations de base. Cette différence de temps permet de calculer une distance constante entre deux stations de base, définissant une hyperbole qui indique les positions possibles du téléphone mobile. Si au moins trois stations de base sont utilisées, la position du mobile est donc déterminée par l'intersection de toutes les hyperboles. L'avantage de cette méthode est la suppression des erreurs d'horloge dans l'équation d'observation et l'atténuation des effets des conditions atmosphériques sur la détermination de la position. En revanche, en plus des modifications des stations de base nécessaires à la méthode TOA, il est obligatoire que les stations de base puissent communiquer ensemble pour réaliser la synchronisation.

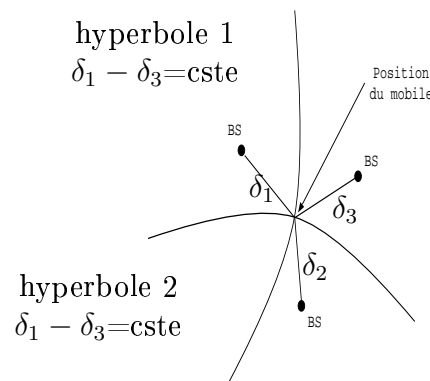


FIG. 1.9: Illustration de la méthode TDOA

**E-OTD** : Dans cette dernière méthode la position du mobile est encore calculée par l'intersection de plusieurs hyperboles. Mais par rapport à la méthode TDOA, le signal envoyé par une station de base est reçu à la fois par le téléphone mobile et par une station de référence dont la position est parfaitement connue (cette station doit être ajoutée au réseau). La différence de temps, et donc par la même occasion la distance entre la station

de base et la station de référence est alors déterminée en corrélant les deux signaux reçus. Comme la distance déduite contient toujours les erreurs d'horloge liées au mobile, trois mesures sont nécessaires pour estimer la position du mobile.

- Le précision sur la localisation obtenue par l'ensemble de ces techniques dépend :
- du nombres de mesures, on obtient donc un meilleur fonctionnement dans les zones urbaines car les stations de base sont plus nombreuses,
  - de la géométrie du réseau, en effet en général les stations de base n'ont pas été placées pour avoir une bonne géométrie mais pour fournir la meilleure couverture possible du réseau,
  - mais aussi de la surcharge du réseau de communication qui peut jouer un rôle prépondérant sur la précision.

Le tableau 1.3 présente la précision obtenue en fonction de chacune de ces méthodes et de la génération des téléphones mobiles.

	GSM	UMTS
TDOA E-OTD	90...160m	5...20m
AOA	100...200m	100...200m
ToA	N/A	50...100m

TAB. 1.3: Précisions obtenues avec les générations GSM/UMTS

### 1.2.3.3 Les systèmes existant

De nombreux projets ont vu le jour depuis la dernière décennie, il est bien sûr impossible d'en fournir une liste exhaustive. Les principaux systèmes existant dans le commerce sont présentés dans cette section.

**Telesentinel :** Le projet Telesentinel développé à la base par la société KSI est un système de localisation selon la technique AOA. L'angle d'arrivée est ici mesuré par des capteurs spécifiques composés de trois antennes omnidirectionnelles situées avec les stations de base. Les mesures d'au moins deux capteurs sont envoyées à une station de contrôle qui détermine la position du mobile. Les capteurs peuvent aussi lire le contenu des trames de contrôle du réseau téléphonique, trouver le numéro de téléphone du mobile et l'envoyer à la station de contrôle. Ainsi la position estimée et le numéro téléphonique sont alors stockés dans une base de données accessible aux utilisateurs du système de localisation. Les essais ont été menés dans la zone d'Annandale en Virginie du Nord, les capteurs couvrant une zone de  $2km \times 2km$ , la précision obtenue par ce système est supérieure à 34m dans 50% des cas, supérieure à 45m dans 67% des cas, supérieure à 89m dans 90% des cas et supérieure à 282m dans 99.9% des cas. A l'heure actuelle, la

société TruePosition continue de développer ce système en utilisant une technique hybride de localisation TDOA+AOA.

**CURSOR** : Le système de localisation CURSOR [Larder, 2001] développé par la compagnie britannique Cambridge Positioning System (CPS) est basé sur la méthode E-OTD. La synchronisation requise entre les stations de base est obtenue en observant leurs signaux par un récepteur spécifique appelé Location Measurement Unit (LMU) situé à une position fixe et le plus souvent avec une station de base (approximativement un LMU est nécessaire pour 4 stations de base). Le calcul de la position est réalisé par le Serving Mobile Location Center (SMLC) et les messages de synchronisation sont envoyés par le service SMS du réseaux GSM. Après de nombreux essais expérimentaux, CPS a fourni les constats suivants :

- Cambridge, zone péri-urbain de  $5km^2$  : précision supérieure à 50m dans 67% des cas,
- Houston, zone péri-urbaine : précision supérieure à 50m dans 67% des cas,
- Hong Kong, zone urbaine dense : précision supérieure à 125m dans 67% des cas.

D'après ces résultats, les effets de multi-trajet dans les centres villes dégradent les performances du système.

**SnapTrack** : Le projet SnapTrack [Moeglein *et al.*, 98] peut être considéré comme une solution A-GPS (assited GPS solution). Dans cette méthode, les signaux GPS sont reçus par des stations réceptrices fixes et de positions connues, les messages de navigation sont alors décodés puis transmis au serveur effectuant la localisation du mobile. Quant au module GPS intégré au mobile, il décode les mesures des pseudo-distances séparant celui-ci et les différents satellites visibles dans la constellation. Ces mesures sont elles-mêmes envoyées au serveur et le calcul de la position s'effectue en prenant en compte les corrections différentielles déduites des messages de navigations fournis par les stations réceptrices obtenant ainsi une localisation proche d'un système DGPS. D'après les résultats expérimentaux ce système permet une couverture totale dans les zones urbaines denses ainsi qu'une bonne couverture dans les environnements intérieurs. La précision obtenue en extérieur est supérieure à 4m dans 67% des cas et en intérieur à 83m dans 67% des cas.

**Véhicule routier** A l'heure actuelle, aucun système n'a été conçu pour le domaine routier, ce en partie par le manque de maturité des systèmes de localisation par téléphonie cellulaire, néanmoins des exemples de projet de recherche peuvent être trouvés dans [Retscher *et al.*, 2001] [Wang *et al.*, 2004].

#### 1.2.4 Positionnement par vision dans un environnement routier

Depuis les vingt dernières années, de nombreux travaux sur le thème de la localisation de véhicule routier par vision ont vu le jour [Desouza *et al.*, 2002] [Bertozzi *et al.*, 2002].

En général le processus de localisation peut être décomposé en cinq étapes 1) l'acquisition de l'image, 2) un prétraitement, 3) la segmentation de l'image, 4) l'interprétation géométrique de la route à partir de l'image segmentée ainsi que l'estimation des paramètres de localisation du véhicule sur la route et enfin 5) une étape de suivi prenant en compte ou non les informations cinématiques du véhicule qui permet de déterminer les zones d'intérêt pour la segmentation à l'itération suivante. Un organigramme de la localisation de véhicule par vision dans un environnement routier est donné par la figure 1.10.

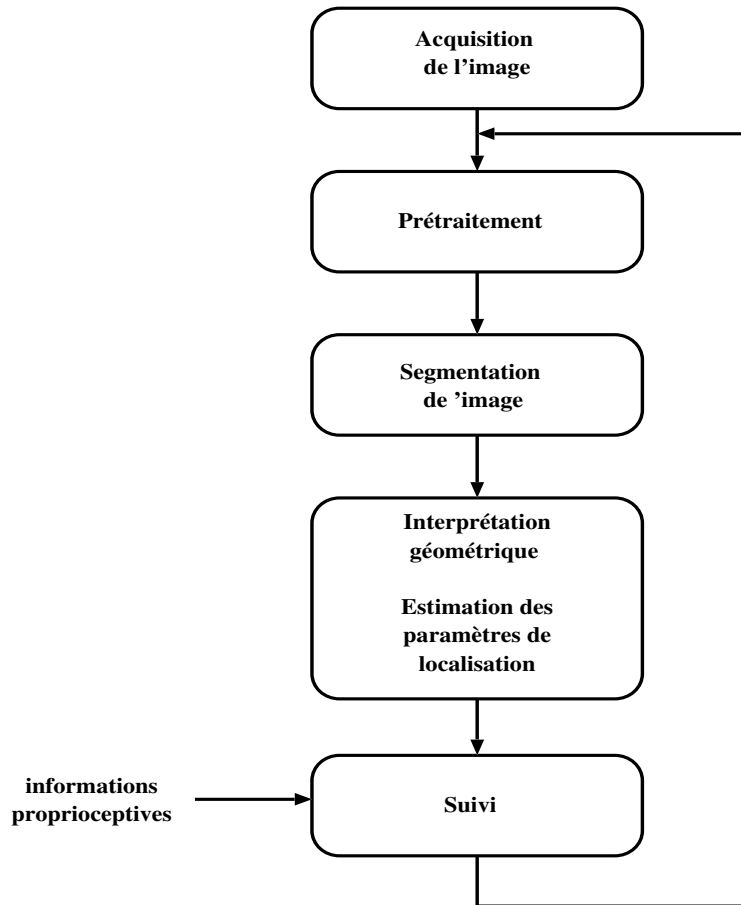


FIG. 1.10: Organigramme d'un système de localisation basée sur la vision

L'approche de segmentation dépend directement de l'environnement dans lequel évolue le robot ou le véhicule. On peut distinguer deux types d'environnements routiers : 1) les environnements routiers structurés pour lesquels des éléments remarquables dans la scène tels que les marquages au sol suffisent pour identifier la route dans l'image et 2) les environnements routiers semi-structurés où il n'y a pas ou peu d'éléments remarquables permettant d'identifier les limites de la chaussée. Ainsi pour les environnements structurés, les techniques de segmentation reposent principalement sur la détection des marquages au sol en utilisant une segmentation de type contour et pour les environnements

non-structurés les techniques de segmentation ont pour but d'identifier l'ensemble de la route dans l'image et reposent sur des principes de segmentation en région. Ces deux approches de segmentation sont présentées dans les sections 1.2.4.1 et 1.2.4.2.

Une fois la segmentation de l'image effectuée, l'estimation de la position du véhicule ou du robot sur la route est réalisée par une interprétation géométrique de la scène (cf section 1.2.4.4). De nombreuses méthodes ont été proposées dans la littérature, et pour chacune d'elles il est nécessaire de choisir un modèle tridimensionnel et/ou un modèle bi-dimensionnel selon que l'on choisit de résoudre le problème de localisation dans le monde 3D et/ou dans l'image. Ces différents modèles sont présentés dans la section 1.2.4.3.

#### 1.2.4.1 Segmentation par contour

Les méthodes de segmentation par contour s'appliquent généralement dans les contextes de routes structurées qui sont délimitées par des marquages au sol et où les transitions route / marquages sont très contrastées. Dans le cas d'images de niveaux de gris, ces transitions sont détectées en appliquant de simples opérateurs différentiels à celles-ci et les contours sont alors trouvés en recherchant des maximums locaux correspondant aux variations brutales de luminosité dans l'image. Dans le cas d'images couleurs, d'autres paramètres tels que la teinte ou la saturation peuvent être ajoutés à la luminosité pour faciliter la détection des marquages aux sols. De plus, pour améliorer la détection des marquages au sol deux techniques sont souvent utilisés :

1. les transitions détectées sont appariées selon les lignes de l'image et comparées à un gabarit de marquages au sol le plus souvent un modèle "plateau" (c'est-à-dire un modèle de luminosité décrivant la transition chaussée/marquage/chaussée), celles-ci sont alors considérées comme les contours des marquages aux sols en recherchant le maximum de corrélation entre le modèle détecté et le gabarit.
2. la deuxième solution est d'augmenter le rapport signal sur bruit, la recherche des contours ne s'effectue plus alors dans toute l'image mais dans des zones d'intérêts (zones rectangulaires ou parallélépipédiques) ou encore simplement sur quelques lignes de l'image. Dans ce cas, ces zones sont souvent déduites de l'étape de suivi et accélèrent considérablement l'étape de segmentation.

Voici dans l'ordre chronologique, certains exemples d'approches de segmentation par contour dédiées à la localisation de véhicule en milieux structurés :

**LANEOK** : dans cette approche [Kenue, 1989], les primitives images sont détectées par une simple étape de filtrage (opérateur de Sobel) et les bords de voies sont déterminés par une transformée de Hough en considérant que les bords de la route peuvent être décrits dans l'image par deux lignes droites. Cependant cette hypothèse s'avère peu adéquate dans les situations de virages à forte courbure, pour résoudre ce problème l'auteur dans [Kenue *et al.*, 1992] décide alors de décrire les bords de voies de circulation par un modèle explicite composé de cinq segments. Leurs positions étant ajustées à l'ensemble

des primitives détectées dans l'image par une méthode des moindres carrés médians. Malgré cette amélioration, cette méthode est sensible aux occultations des primitives images notamment les occultations induites par des obstacles se trouvant la chaussée ou encore les changements d'illumination dans la scène.

**YARF**<sup>13</sup> : dans cette approche [Kluge *et al.*, 1995a] [Dellaert *et al.*, 1998] la segmentation de l'image est réalisée dans l'espace HSI. En effet les auteurs proposent une méthode originale qui permet de reconnaître les marquages au sol en fonction de leurs teintes (aux Etats-Unis deux teintes existent pour les marquages au sol : le blanc ou le jaune). Ainsi les pixels appartenant à ces deux teintes sont détectés dans l'image et les bandes blanches ou jaunes sont extraites dans des zones d'intérêts en cherchant le maximum de corrélation entre le gradient d'un modèle "plateau" (passage revêtement chaussée/marquage au sol/chaussée) et le gradient horizontale des pixels appartenant a priori au marquage au sol. L'étape de suivi prend en compte l'hypothèse que la teinte des marquages au sol est invariante aux conditions météorologiques d'une image à l'autre et permet de définir des zones d'intérêts pour limiter la recherche des pixels blancs ou jaunes.

**VELAC** : Aufrère dans [Aufrère *et al.*, 2000b] [Aufrère, 2001] propose une approche statistique de localisation qui utilise un modèle explicite dans l'image pour représenter les marquages au sol et un modèle implicite pour décrire la position du véhicule et la géométrie de la chaussée. Ici le modèle explicite est défini par  $4n$  paramètres dont  $2n$  sont les positions des points caractéristiques du polygone représentant la voie de circulation dans l'image et  $2n$  les incertitudes sur la position de chaque points caractéristiques et le modèle implicite est défini par des arcs de cercle. Pour améliorer le processus de localisation, l'auteur propose d'effectuer une phase d'apprentissage qui permet d'apprendre toutes les dépendances statistiques entre ces deux modèles en fonction de l'ensemble des scènes autoroutières rencontrables dans la réalité. Dans ce cas, il est alors possible de réunir l'étape de segmentation et d'interprétation géométrique en une seule étape et l'état du système est alors constitué des paramètres de position du véhicule, de géométrie de la route ainsi que les paramètres du modèles explicites. L'originalité de cette méthode tient au fait que pour chaque point du modèle explicite détecté dans l'image, il est possible d'obtenir directement une estimation de la position du véhicule et qu'il n'est pas alors obligatoire de détecter tous les points pour obtenir une estimation correcte de la localisation du véhicule, ce qui rend cette méthode robuste aux occultations et aux changements d'illuminations.

Pour améliorer l'étape de segmentation, dans certaines approches l'image originale peut subir au préalable des transformations (étape de prétraitement). Deux situations peuvent être rencontrées selon le choix du prétraitement : soit celui-ci n'a aucune influence sur l'espace et l'étape de segmentation s'effectue tout simplement dans l'image prétraitée, soit celui-ci effectue un changement d'espace auquel cas il est nécessaire d'établir des outils permettant de mettre en évidence la chaussée dans ce nouvel espace.

---

<sup>13</sup>Yet Another Road Follower

**ARGO** : ici les auteurs [Bertozzi *et al.*, 1998] [Broggi *et al.*, 1999] proposent de corriger autant que possible les effets de perspectives en travaillant dans une vue aérienne de la scène routière. Cette vue est obtenue par une transformation IPM<sup>14</sup> de l'image originale (voir figure 1.11). En considérant que la largeur de la chaussée est constante et connue, les bords de la chaussée peuvent être considérés comme parallèles dans cette nouvelle image et déterminés de la manière suivante : 1) l'ensemble des transitions brutales de luminosité sont détectées par un simple gradient horizontal, 2) ces transitions sont appariées deux à deux sur chaque ligne de l'image en tenant compte des contraintes sur la largeur de la route et 3) les bords des voies de circulation sont déterminés en effectuant une corrélation verticale de ces couples.

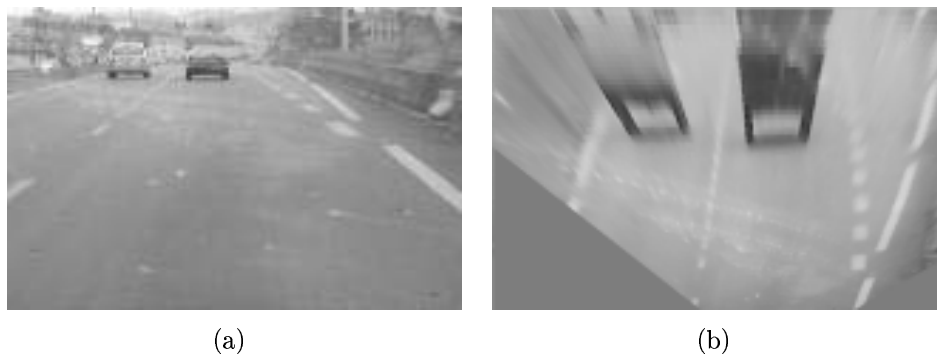


FIG. 1.11: Inverse perspective mapping a) figure originale b) projection IPM de la figure originale

**LANA**<sup>15</sup> : Kreusher dans [Kreusher *et al.*, 1999] propose une nouvelle méthode de détection de bords de voie qui effectue une transformation image/fréquence dont les caractéristiques fréquentielles sont obtenues par une transformation en cosinus discrète (DCT). Les hautes fréquences dont l'amplitude et la phase sont directement reliées à la position et l'orientation des contours dans l'image initiale et sont alors aisément détectables dans ce nouvel espace. De cette manière les auteurs espèrent ainsi s'affranchir des problèmes liés aux changements d'illumination dans la scène.

#### 1.2.4.2 Segmentation par région

Les méthodes de segmentation par région s'appliquent principalement dans les contextes de routes semi-structurées et non-structurées dont les bords de la route sont souvent peu marqués et peu contrastés et ne peuvent être détectés par des approches de segmentation par contour. L'idée générale de cette méthode est de décomposer l'image en plusieurs zones homogènes dont au moins une doit correspondre à la chaussée, les pixels appartenant à chacune de ces régions sont alors classés en fonction de critères tels que leur couleur (dans

<sup>14</sup>Inverse Perspective Mapping

<sup>15</sup>Lanefinding in ANother domAin



l'espace RGB<sup>16</sup> ou HSI<sup>17</sup>) ou leur texture souvent définie par des critères statistiques tels que l'homogénéité, l'entropie, l'énergie ou encore le contraste. Enfin comme les conditions d'illuminations de la route sont souvent variables dans de telles scènes, le niveau de gris est souvent exclu de ces méthodes car il ne peut rarement constitué un critère suffisamment discriminant.

**Classification par couleur :** dans ce cas, la classification des pixels de l'image est réalisée dans l'espace des couleurs (RGB, HSI ou autre espace de couleur) en effectuant un seuillage des informations multidimensionnelles délivrées par l'image. Cette technique a surtout été utilisée dans la fin des années 1980 :

**FMC Corporation :** Kuan dans [Kuan *et al.*, 1988] [Kuan *et al.*, 1987] propose de classer les pixels en effectuant un simple seuillage dans l'espace RGB néanmoins cette technique est sensible aux changements d'éclairement de la scène vue par la caméra. Pour atténuer ce phénomène, les auteurs proposent d'apprendre dynamiquement les couleurs caractéristiques des zones de l'image censées appartenir à la route.

**VITS :** Turk dans [Turk *et al.*, 1987] [Turk *et al.*, 1988] propose de travailler uniquement dans l'espace des couleurs RB. L'auteur justifie ce choix en prônant qu'en général la composition spectrale dominante pour les pixels appartenant à la route tend vers le bleu alors que le reste de l'image tend globalement vers le rouge et montre ainsi qu'il est possible de séparer les pixels "route" et "non-route" par un simple plan. De plus, pour prendre en compte les diverses spécificités d'une scène routière à savoir la présence ou non d'ombres, de flaques d'eau, de neige, de terre ou de goudron etc.. l'auteur considère plusieurs classes pour décrire la route (cf figure 1.12). Lorsque la classification est effectuée, le système détermine les pixels appartenant au bord de la route (sous l'hypothèse que ces pixels ont 8 pixels voisins appartenant à la classe "non-route" et 4 pixels voisins appartenant à la classe "route"), la position et l'orientation du véhicule sur la chaussée sont alors déterminées en prenant compte un modèle de pente et de vallée<sup>18</sup> pour représenter la géométrie de la route.

**Classification couleur bayésienne :** l'hypothèse émise lors d'une telle classification est que les classes "route" et "non-route" auxquelles appartient potentiellement un pixel peuvent être représentées par différentes distributions gaussiennes prédéfinies dans l'espace des couleurs. Dans ce cas, chaque distribution est représentée par la moyenne et la covariance des composantes de couleurs d'une classe et la probabilité qu'un pixel appartienne à une de ces classes est calculée à partir du théorème de Bayes. Enfin la probabilité que le pixel appartienne à la route  $P(\text{pixel}|\text{route})$  est définie par le maximum de probabilité que ce pixel appartienne à une des classes routes. Ainsi si les différentes classes

---

<sup>16</sup>Red Green Blue

<sup>17</sup>Hue Saturation Intensity

<sup>18</sup>Hill and Dale en anglais

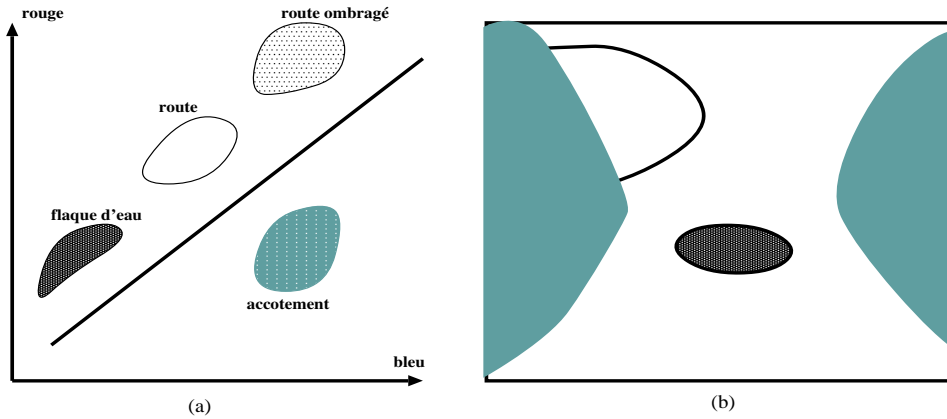


FIG. 1.12: Classification des pixels dans l'espace de couleurs RB, figure (a) les classes dans l'espace de couleur RB, figure (b) l'image originale vue par la caméra

"route" sont  $w_r = \{w_{r_1}, w_{r_1} \dots, w_{r_n}\}$  où  $w_r$  sont définis par les deux premiers moments des distributions gaussiennes  $\underline{m}_r$  et  $\mathbf{c}_r$ , la probabilité qu'un pixel appartienne à la route est :

$$P(\text{pixel}|\text{route}) = \arg \max_{\underline{c}} \left[ \frac{1}{\sqrt{(2\pi)^3 |\mathbf{c}_r|}} e^{-\frac{1}{2}(\underline{c}-\underline{m}_r)^T \mathbf{c}_r^{-1} (\underline{c}-\underline{m}_r)} \right] \quad (1.22)$$

avec  $\underline{c}$  le vecteur composé par les composantes de couleur du pixel. De nombreux travaux basés sur cette segmentation ont vu le jour au début des années 1990 on pensera surtout aux systèmes :

**SCARF** : qui a été développé à l'université de Canergie Mellon [Crisman *et al.*, 1991] [Crisman *et al.*, 1993]. Ce système utilise 4 classes de couleurs pour représenter la classe "route" et 4 classes pour représenter pour la classe "non-route". Ce choix permet ainsi de prendre en compte les changements d'illumination de la scène, les changements de couleur du revêtement de la route, etc... Une fois les pixels de l'image classés, une image binaire (route/non-route) de la scène est obtenue, cette image est alors comparée avec des images binaires prédéfinies fonction des différents scénarios de route et d'intersection pouvant être rencontrés dans la réalité (cf figure 1.13.) Ainsi l'image binaire prédéfinie obtenant le meilleur score de corrélation avec l'image binaire fournie par le processus de classification est considérée comme représentant au mieux la scène routière et le véhicule peut être localisé en fonction de ces caractéristiques géométriques.

**VELAC** : dans le même contexte Aufrère [Aufrère *et al.*, 2000a] propose une méthode de localisation où la classification est effectuée à partir d'un modèle de luminance. Ainsi en décomposant l'image en secteurs de taille 6x6 pixels et en calculant leur moyenne et leur variance des niveaux de gris, chaque secteur peut être classé en utilisant l'équation

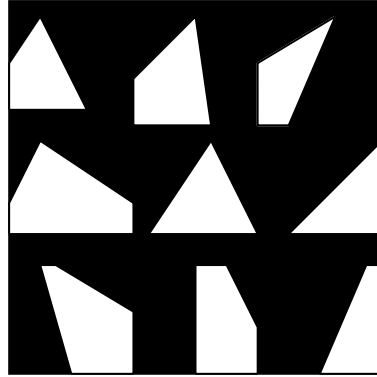


FIG. 1.13: Images binaires prédéfinies

(1.22). Ici l'auteur propose de prendre en compte un seul modèle de luminance pour représenter la classe "route", ce modèle étant caractérisé en fonction des informations de luminance d'une zone pyramidale positionnée en bas et au centre de l'image. Une fois la classification effectuée une étape de relaxation permet de ne garder que les secteurs appartenant réellement à la route. Enfin en ne gardant que les secteurs représentant les bords de la route et en considérant que les bords de la route peuvent être décrits par des droites, la position et l'orientation du véhicule est calculée.

**Classification par transformation morphologique :** l'idée principale de cette approche est de segmenter l'image en régions uniformes en exploitant les propriétés d'une transformation morphologique plus connue sous le nom de "ligne de partage des eaux" [Beucher *et al.*, 1994a] [Beucher *et al.*, 1994b]. Puis de reconstruire la continuité des marquages aux sols (si ils existent) ou des bords de la chaussée en ajoutant une étape de filtrage temporel de la séquence d'image.

**Classification selon des critères statistiques :** ces méthodes de classification utilisent des attributs statistiques [Unser, 1984] comme primitives élémentaires de texture qui sont définis par le comportement spatial dans l'image des pixels et de leur voisinage. Les modèles statistiques les plus utilisés exploitent les matrices de co-occurrence, les coefficients d'autocorrélation ou encore les histogrammes de sommes et de différences.

**ROMANE :** Kaske dans [Kaske, 1997] et [Kaske *et al.*, 1997] propose une méthode de localisation où les bords de la route sont détectés en utilisant les critères d'énergie, d'entropie, de contraste et d'homogénéité. Pour alléger les calculs, l'image est décomposée en secteurs de taille 8x8 pixels, les paramètres statistiques sont alors calculés en fonction de ces secteurs et des matrices de co-occurrence correspondantes aux critères statistiques. Ainsi un pixel est considéré comme appartenant à un bord de route en recherchant le minimum des paramètres d'énergie et d'homogénéité ou le maximum du contraste et d'entropie. La localisation du véhicule est déterminée par une méthode de moindres carrés en considérant que les bords de la chaussée sont représentés par deux hyperboles séparées de

la largeur de la route et corrélées entre elles.

**DALA** : Cervantes dans [Cervantes, 2005] présente une méthode de segmentation pour environnement semi-structuré. Dans un premier temps, une simple étape de segmentation en région est réalisée, pour chacune de ces régions, douze paramètres sont alors déterminés à savoir : les 3 composantes moyennes de couleur, des critères de textures comme l'énergie, la corrélation, le contraste, l'entropie, l'homogénéité, la nuance et la prééminence ainsi que la position de celles-ci dans l'image. Une classification de ces régions en fonction de ces paramètres est alors possible et réalisée soit par la technique de Support Vector Machine soit par la méthode des k plus proches voisins. Selon les résultats de la classification les régions voisines ayant la même nature (c'est-à-dire appartenant à la même classe) sont fusionnées afin de corriger les erreurs de sur-segmentation et les bords de voies sont alors retrouvés en détectant les contours communs des régions dites "route" et leurs régions connexes.

#### 1.2.4.3 Modélisation de la chaussée

En général deux modèles de chaussée interviennent dans un algorithme de localisation par vision : un modèle 2D utilisé pour représenter les bords de la chaussée dans l'image et qui permet de définir les zones d'intérêt dans l'image et un modèle tridimensionnel qui définit la géométrie de la route et permet à la fois de déterminer la position et l'orientation du véhicule dans la scène et de prendre en compte l'aspect cinématique du véhicule et le temps (approche 4D). D'après [Kluge *et al.*, 1989], deux sortes de modèles existent : les modèles implicites (ou paramétriques) et les modèles explicites. Souvent les modèles implicites représentent les bords de la chaussée par des droites, des arcs de cercle, des paraboles ou des clothoïdes (souvent sous l'hypothèse de route plane) alors que les modèles explicites décrivent les limites de la chaussée par un ensemble de points ou un ensemble de segments. Les modèles implicites semblent plus appropriés pour décrire des routes structurées et les modèles explicites semblent plus adéquat dans des contextes de routes non-structurées dont les limites de chaussée ne peuvent être exprimées analytiquement. Dans la suite de cette section nous allons présenter les différents modèles implicites et explicites que l'on peut rencontrer dans la littérature.

**Les modèles implicites ou paramétriques** : dans la réalité, les routes sont constituées d'un ensemble de segments possédant chacun des caractéristiques géométriques fixes tels des droites pour les segments rectilignes, des arcs de cercle pour les virages ou encore des clothoïdes pour les segments de transition. Naturellement, les modèles implicites reposent eux aussi sur les mêmes caractéristiques. Cependant pour avoir un modèle décrivant toute ces situations routières, les portions rectilignes ne sont en général pas représentées par des droites mais par des arcs de cercle de rayon infini. Alors les principaux modèles implicites dans l'espace 3D sont soit des arcs de cercle [Chapuis *et al.*, 1995] [Kreusher *et al.*, 1999] (auquel cas on considère que la courbure de la route est toujours constante même dans les zones de transition) soit des clothoïdes [Maurer *et al.*, 1998] [Dickmanns *et al.*, 1992] qui permettent de tenir compte de la variation de la courbure en

fonction de l'axe longitudinal de route.

Dans l'image, les modèles implicites définissent le contour de la chaussée. Ces modèles ne correspondent pas forcément à la projection perspective d'un modèle implicite de l'espace 3D mais sont choisis de telle sorte que leurs paramètres puissent être déterminés par une simple méthode de régression. Bien sûr le modèle le plus simple est la ligne droite mais aussi le plus restrictif car il ne permet que de décrire des portions de route rectiligne. Afin de tenir compte d'éventuel courbure de la route, il faut introduire des paramètres supplémentaires. Ces modèles sont alors des arcs de parabole [Kenue *et al.*, 1992] caractérisés par des polynômes du second degré ou le plus souvent des hyperboles [Kluge *et al.*, 1995b] [Chapuis *et al.*, 1995] (projection d'un arc de cercle dans le plan image) dont un de leurs axes est défini par la ligne d'horizon apparaissant dans l'image.

**Les modèles explicites** : ces modèles permettent une description point par point de la route en fonction des coordonnées des limites de la chaussée. En général un modèle explicite est déterminé en faisant correspondre ces points à une forme géométrique prédéfinie (ex : polygones) et fournit une représentation fidèle de la chaussée. Dans l'espace tridimensionnel, contrairement aux modèles implicites, ils sont définis par un grand nombre de paramètres pour garder tous les degrés de liberté de la forme géométrique. Dans l'image [Kenue *et al.*, 1992] [Wang *et al.*, 1998], certains degrés de liberté sont éliminés des modèles explicites puisque la dimension spatiale du problème diminue. Dans ce cas, les ambiguïtés sur les positions réelles des éléments remarquables (ambiguïtés souvent rencontrées dans l'espace tridimensionnel) disparaissent, les paramètres de ces modèles sont ainsi mieux déterminés et la localisation du véhicule est alors plus précise.

#### 1.2.4.4 Interprétation géométrique et localisation du véhicule

La localisation du véhicule est réalisée lors de l'interprétation de la géométrie de la scène routière. En général, cette étape fournit des informations sur l'attitude du véhicule sur la chaussée telles que sa position latérale et son orientation ainsi que parfois des informations sur la géométrie de la route telles que sa largeur, sa courbure, etc... Pour réaliser cette étape, il existe deux manières de procéder : soit l'interprétation géométrique est effectuée dans l'image auquel cas un certain nombre de paramètres caractérisant la forme géométrique de la route dans l'image (droites, paraboles, polynôme etc...) sont déterminés et utilisés pour estimer l'attitude du véhicule dans le monde 3D, soit l'interprétation géométrique est directement effectuée dans l'espace 3D auquel cas l'estimation de l'attitude du véhicule est immédiate.

Selon le type de modèle géométrique, l'interprétation géométrique peut être réalisée dans l'image par diverses méthodes. Pour les modèles implicites, les méthodes les plus courantes sont la transformée de Hough [Kenue, 1989] [Crisman *et al.*, 1993], dans le cas où les bords de voies sont représentés par des droites ; ou la méthode des moindres carrés et ses diverses extensions comme les moindres carrés médians [Kluge *et al.*, 1995a],

pondérés [Kenue *et al.*, 1992], récursifs [Guichard *et al.*, 1999] lorsque les bords de voies sont représentés par des paraboles, des hyperboles ou encore des polynômes. Pour les modèles explicites, la méthode utilisée est en général la méthode des moindres carrés [Kenue *et al.*, 1992]. Enfin, quelques fois les méthodes de Monté Carlo ou de recuit simulé sont elles aussi utilisées pour résoudre le problème de l'interprétation géométrique (et ce quelque soit le type de modèle géométrique utilisé) comme le montre [Lakshmanan *et al.*, 1995] ou encore [Wang *et al.*, 1998].

Quant à l'estimation de l'attitude du véhicule sur la chaussée, et ce qu'elle soit directement réalisée dans l'espace 3D ou qu'elle tienne compte de l'interprétation géométrique 2D, est essentiellement basée sur l'estimation bayésienne de modèles implicites 3D. Le plus souvent, cette estimation est effectuée par un filtre de Kalman [Dickmanns *et al.*, 1992] [Aufrère, 2001] mais aussi quelque fois par un filtre à particules [Macek *et al.*, 2004] [Apostoloff *et al.*, 2003].

### 1.2.5 Bilan

Le positionnement par satellites est la méthode par excellence de localisation de robots mobiles en environnement extérieur. Selon le coût envisagé pour le récepteur, la précision de la localisation peut varier d'environ 15m (GPS classique d'un coût de l'ordre de 100€) à quelques centimètres (GPS RTK à 20000€). Néanmoins si le système est soumis à des restrictions budgétaires le prix du récepteur devient un inconvénient majeur. De plus, les occultations et les effets de multi-trajet provoqués par les immeubles, les tunnels, les arbres etc... dégradent fortement la précision du système de positionnement et peut même engendrer un arrêt total de la localisation.

Le positionnement par téléphone cellulaire est sûrement l'avenir du positionnement en environnement urbain et représentera l'avantage de suppléer le GPS là où ses performances sont dégradées. Cependant à l'heure actuelle, le manque de maturité de ce système de positionnement ainsi qu'une couverture approximative du territoire par le réseau UMTS (30% du territoire regroupé autour des grandes agglomérations françaises) ne permettent pas une localisation fiable et précise du mobile à tout endroit et à tout instant sur le sol français.

Le positionnement par vision dans un environnement routier est l'approche de localisation qui se rapproche le plus du système de perception humain. Grâce à un nombre incroyable d'informations fournies par l'image il est possible de déterminer la position et l'orientation du véhicule dans la plupart des scènes routières. Potentiellement, le seul point faible de cette méthode est l'extraction des primitives de la masse d'informations. En effet toutes primitives non détectées (occultations par des obstacles, changement d'illuminations, conditions climatiques) peuvent mettre en défaut l'interprétation géométrique et fausser la localisation du véhicule. Néanmoins de nombreux travaux ont prouvé que cette méthode de localisation était fiable dans de nombreuses situations. Par contre, il ne faut pas perdre de vue que cette méthode de positionnement fournit une position et une orientation du véhicule dans le référentiel de la route qui ne peuvent être combinées avec d'autres informations de position absolue sans l'utilisation d'une carte de l'environnement

routier.

### 1.3 Les systèmes de positionnement hybride

Comme le montre le tableau 1.4, les différentes méthodes de positionnement présentent divers inconvénients et avantages. Le principal avantage du positionnement relatif est une cadence d'acquisition des informations élevée qui permet de calculer précisément le déplacement du robot à court terme. Par contre, à long terme, cette méthode diverge en raison de l'accumulation des erreurs lors des intégrations successives des déplacements élémentaires. En revanche le positionnement absolu n'est pas affecté par ce problème mais le résultat de la localisation est sujet à la visibilité des sources d'information, la visibilité des satellites pour le positionnement par GPS, la visibilité des stations émettrices pour le positionnement par téléphone cellulaire ou encore la visibilité des amers visuels pour le positionnement par vision. De par ce constat, une seule méthode de positionnement ne peut permettre une localisation fiable d'un robot. On peut attribuer ce problème au fait que les informations sensorielles utilisées dans ces diverses méthodes de positionnement donnent une mesure imparfaite, c'est-à-dire qu'elles sont des mesures partielles et/ou entachées d'erreurs des phénomènes physiques que l'on cherche à appréhender (dans notre cas : la position, l'orientation, l'accélération etc... du véhicule).

Pour répondre à ce problème, la plupart des systèmes de positionnement sont réalisés par des systèmes hybrides combinant plusieurs sources d'informations. Dans ce cas on peut distinguer deux cas de figure, soit ces informations sont de même nature et sont :

1. redondantes : auquel cas la combinaison des informations tente d'accroître la précision et le degré de confiance du système de positionnement en fonction de plusieurs observations distinctes,
2. complémentaires : dans ce cas la fusion des informations permet en général de fournir une information plus complète sur le comportement du véhicule dans son environnement qui ne peut être obtenue en utilisant ces capteurs séparément.

Soit ces informations ne sont pas de mêmes natures (informations proprioceptives et extéroceptives), auquel cas le système de positionnement absolu est en général chargé de corriger régulièrement l'estimation du système de positionnement relatif afin de pallier les problèmes de dérives rencontrés avec celui-ci. Cette méthode de correction dynamique est désignée sous le terme de recalage dynamique.

Pour mettre en œuvre de tels systèmes de positionnement, de nombreuses techniques de fusion de données sont disponibles. Une très bonne étude bibliographique est présentée dans [Gruyer *et al.*, 2004]. Ces techniques peuvent être séparées en quatre grandes classes : 1) les approches analytiques telles que le filtre de Kalman et ses extensions (cf section 2.2.1), 2) les approches numériques telles que le filtre particulaire et ses extensions (cf section 2.3), 3) les approches ensemblistes que ce soit les approches à erreur bornée [Halbwachs, 1997] ou les approches par analyse d'intervalles [Kieffer, 1999] [Seigneur *et al.*, 2005] et enfin 4) les approches basées sur la théorie des croyances

Positionnement relatif		
Méthode	Avantages	Inconvénients
Positionnement par odométrie	<ul style="list-style-type: none"> <li>• peu cher</li> <li>• rapide</li> </ul>	<ul style="list-style-type: none"> <li>• positionnement 2D</li> <li>• diverge à long terme</li> <li>• sensible aux conditions de roulement</li> </ul>
Positionnement inertiel	<ul style="list-style-type: none"> <li>• calcul : position, vitesses, accélérations</li> <li>• insensible aux conditions de roulement</li> <li>• rapide</li> </ul>	<ul style="list-style-type: none"> <li>• diverge à long terme</li> <li>• précision/prix</li> </ul>
Positionnement absolu		
Méthode	Avantages	Inconvénients
Positionnement par satellites	<ul style="list-style-type: none"> <li>• ne diverge pas à long terme</li> <li>• couverture mondiale</li> <li>• mise en œuvre facile</li> </ul>	<ul style="list-style-type: none"> <li>• sensible aux occultations des satellites</li> <li>• précision/prix</li> </ul>
Positionnement par téléphone cellulaire	<ul style="list-style-type: none"> <li>• ne diverge pas à long terme</li> <li>• couverture accrue en zone urbaine</li> <li>• localisation dans les immeubles (ex : parking)</li> </ul>	<ul style="list-style-type: none"> <li>• sensible aux occultations des stations de base</li> <li>• aménagement du réseau téléphonique</li> </ul>
Positionnement par vision	<ul style="list-style-type: none"> <li>• ne diverge pas à long terme</li> <li>• adaptation</li> </ul>	<ul style="list-style-type: none"> <li>• sensible aux occultations des amers visuels</li> <li>• aménagement de l'environnement</li> </ul>

TAB. 1.4: Avantages et inconvénients des méthodes de positionnement relatif et absolu

[Royère, 2002] [Najjar *et al.*, 2002] ou la théorie des possibilités [Oussalah, 1998]. Bien sûr, dans cette thèse, il nous a été impossible d'étudier l'ensemble de ces méthodes et d'en justifier le choix. Néanmoins, comme les approches analytiques et notamment le filtre de Kalman étendu sont les plus usitées dans les approches hybrides de localisation en robotique mobile, nous avons donc opté dans un premier temps pour celui-ci. D'ailleurs la théorie ainsi que les résultats expérimentaux de l'approche de localisation utilisant cette technique de fusion sont présentés dans le chapitre 5. Cependant comme le montreront les résultats expérimentaux, le choix du filtre de Kalman étendu est discutable dans des



environnements routiers comportant plusieurs voies de circulation car il est nécessaire de gérer des situations multimodales (situations généralement incompatibles avec les approches analytiques). Pour résoudre ce problème, notre choix c'est alors portée vers une méthode numérique et plus particulièrement vers le filtre particulaire génétique comme le montre l'approche de localisation présentée dans le chapitre 6. Afin de comprendre les mécanismes de ces deux techniques de fusion, une étude approfondie des approches analytiques et numériques est proposée dans le prochain chapitre.

# Chapitre 2

## Le Filtrage

Dans ce chapitre, le problème du filtrage dans le cas général est abordé en exposant les différentes approches classiques utilisées pour le résoudre. Les deux grandes classes de méthodes sont présentées : les méthodes analytiques, comme le filtre de Kalman et les méthodes numériques comme le filtrage à particules, montrant ainsi les limites et faiblesses de chacune de ces méthodes.

### 2.1 Le filtrage optimal

Le problème du filtrage consiste à estimer l'état de variables d'un système dynamique (variables d'état) sujet à des perturbations et observées partiellement. La modélisation de l'évolution dans le temps des états  $\underline{X}_{t \geq 0}$  du système considéré s'écrit alors sous la forme d'une partie d'évolution déterministe et d'une partie stochastique,

$$\underline{X}_{t \geq 0} = \underline{X}_0 + \int_0^t f(\underline{X}_t, \underline{W}_t, t) dt \quad (2.1)$$

où  $\underline{X}_0$  est l'état initial des variables d'état,  $f$  est une fonction déterministe décrivant la dynamique et  $\underline{W}_t$  est un processus de Wiener standard modélisant les perturbations aléatoires de la dynamique du système. Le processus  $\underline{X}_t$  est alors un processus Markovien.

Dans le but de déterminer l'état d'un système discret, on est amené à construire une équation d'observation qui relie à des instants  $k$  l'observation  $\underline{Y}_k$  à l'état courant  $\underline{X}_k$  sachant que les mesures sont entachées d'erreurs dues à l'imperfection des capteurs.

$$\underline{Y}_k = h_k(\underline{X}_k) + \underline{V}_k \quad (2.2)$$

où  $h_k$  est une fonction d'observation connue et  $\underline{V}_k$  une suite de variables aléatoires de statistique connue modélisant l'imperfection des observations. On cherche à déterminer l'état du système  $\underline{X}_k$  à partir des observations  $(\underline{Y}_1, \dots, \underline{Y}_n)$ . Suivant la valeur de  $n$  on parle :

- d'un problème de lissage si  $n > k$ ,

- d'un problème de filtrage si  $n = k$ ,
- d'un problème de prédiction si  $n < k$ .

Les hypothèses classiques du filtrage sont les suivantes :

- les observations ( $\underline{Y}_k, k \geq 1$ ) sont mutuellement indépendantes, conditionnellement à  $\underline{X}_{k \geq 0}$ .
- le processus d'état  $\underline{X}_k$  est Markovien et le bruit de mesure  $\underline{V}_k$  est un ensemble de variables aléatoires indépendantes du processus  $\underline{X}_k$ .
- $\underline{X}_0$  l'état initial, présente une distribution de probabilité supposée connue et indépendante de  $\underline{V}_k$  et de  $\underline{W}_k$ .

Afin d'obtenir un filtrage optimal, on cherche classiquement à minimiser la variance de l'erreur de filtrage. Ainsi, si  $\hat{\underline{X}}_k$  désigne un estimateur de l'état  $\underline{X}_k$ , celui-ci sera optimal s'il minimise l'erreur moyenne quadratique :

$$E[(\hat{\underline{X}}_k - \underline{X}_k)(\hat{\underline{X}}_k - \underline{X}_k)^T] \quad (2.3)$$

Partant de l'hypothèse que les seules informations disponibles sur le système sont les mesures  $(Y_1, \dots, Y_k)$ , alors l'estimateur de variance minimale non biaisé est donné par :

$$\hat{\underline{X}}_k = E[\underline{X}_k | \underline{Y}_1, \dots, \underline{Y}_k] \quad (2.4)$$

Dans le cas où la loi conditionnelle  $p(\underline{X}_k | \underline{Y}_1, \dots, \underline{Y}_k)$  est multimodale, il est souhaitable de prendre l'estimateur du maximum a posteriori

$$\hat{\underline{X}}_k = \text{Arg Max}_{\underline{X}_k} [p(\underline{X}_k | \underline{Y}_1, \dots, \underline{Y}_k)] \quad (2.5)$$

Le filtrage optimal consiste à calculer la densité conditionnelle  $p_{k|k}$  de l'état  $\underline{X}_k$ , sachant les observations  $\underline{Y}_{1:k} = (\underline{Y}_1, \dots, \underline{Y}_k)$  jusqu'à l'instant courant  $k$

$$p_{k|k} = p(\underline{X}_k | \underline{Y}_{1:k}) \quad (2.6)$$

Dans la pratique, la taille du vecteur d'observation  $\underline{Y}_{1:k}$  augmentant à chaque instant  $k$ , il est souhaitable que l'on parvienne à contenir l'information ainsi accumulée dans un vecteur dont la dimension ne croisse pas de même. Il est alors intéressant d'établir une expression récurrente de la densité conditionnelle  $p_{k|k}$ . Ainsi  $p_{k|k}$  ne doit être exprimée qu'en fonction de la dernière observation  $\underline{Y}_k$  et de la loi conditionnelle  $p_{k-1|k-1}$  de l'instant précédent. Le passage de  $p_{k-1|k-1}$  à  $p_{k|k}$  fait intervenir la loi conditionnelle de  $p_{k|k-1}$  de l'état  $p_{k|k}$  sachant les observations  $\underline{Y}_{1:k-1} = (\underline{Y}_1, \dots, \underline{Y}_{k-1})$

$$p_{k|k-1} = p(\underline{X}_k | \underline{Y}_{1:k-1}) \quad (2.7)$$

La suite des lois de probabilité  $p_{k|k-1}$ , est appelée filtre de prédiction et se décompose en deux étapes essentielles :

- Prédiction : l'étape de prédiction du filtre utilise la connaissance a priori de l'état du système et de son noyau de transition  $p(\underline{X}_k, \underline{X}_{k-1})$  (défini par le système dynamique du système), en réalisant la transition de  $p_{k-1|k-1}$  à  $p_{k|k-1}$  de la manière suivante,

$$p_{k|k-1} = \int_{\mathcal{R}_d} p(\underline{X}_k, \underline{X}_{k-1}) p_{k-1|k-1} d\underline{X}_{k-1} \quad (2.8)$$

ou  $\mathcal{R}_d$  désigne l'espace d'état.

- Correction : l'étape de correction utilise l'observation  $\underline{Y}_k$  à travers la densité conditionnelle  $p(\underline{Y}_k | \underline{X}_k)$  (encore appelée densité de vraisemblance), et permet ainsi de corriger la densité prédite  $p_{k|k-1}$ , par application de la formule de Bayes.

$$p_{k|k} = \frac{p(\underline{Y}_k | \underline{X}_k) p(\underline{X}_k | \underline{Y}_{1:k-1})}{p(\underline{Y}_k | \underline{Y}_{1:k-1})} \quad (2.9)$$

soit

$$p_{k|k} = \frac{p(\underline{Y}_k | \underline{X}_k) p_{k|k-1}}{\int_{\mathcal{R}_d} p(\underline{Y}_k | \underline{X}_k) p_{k|k-1} d\underline{X}_k} \quad (2.10)$$

L'application de ces étapes donne la solution théorique du filtrage non linéaire optimal. Son calcul effectif exige le calcul d'intégrales multidimensionnelles. Malheureusement, dès que l'équation d'état et/ou l'équation d'observation sont non-linéaires, ces intégrales ne se calculent pas analytiquement. On a alors recours à des approximations soit par des méthodes analytiques (par exemple le filtre de Kalman étendu qui considère le modèle d'état linéarisé), soit par des techniques numériques (le filtrage particulaire).

## 2.2 Les méthodes analytiques

### 2.2.1 Le filtre de Kalman

Développé par Kalman [Kalman., 1960] pour le cas discret et repris par Kalman et Bucy [Kalman *et al.*, 1961] pour le cas continu, on trouvera une présentation détaillée du filtre de Kalman dans [Gelb, 1974]. On considère que le modèle d'état est linéaire, avec un bruit additif Gaussien et l'observation est une fonction linéaire de l'état entachée d'un bruit additif Gaussien, et représenté par les équations (2.11) et (2.12)

$$\underline{X}_k = \mathbf{F}_k \underline{X}_{k-1} + \mathbf{G}_k \underline{U}_k + \mathbf{\Phi} \underline{W}_k \quad (2.11)$$

où

- Les vecteurs  $\underline{X}_k, \underline{X}_{k-1} \in \mathcal{R}^{d_1}$  sont les vecteurs d'état à l'instant  $k$  et  $k-1$ ,
- $\underline{U}_k \in \mathcal{R}^{d_2}$  est le vecteur de commande du système,
- $\underline{W}_k \in \mathcal{R}^{d_3}$  est un vecteur de bruit additif gaussien sur l'état,
- Les matrices  $\mathbf{F}_k, \mathbf{G}_k$  et  $\mathbf{\Phi}$  sont des matrices déterministes de tailles  $d_1 \times d_1, d_1 \times d_2, d_1 \times d_3$  représentant la dynamique du système, l'interaction du vecteur de commande et du bruit gaussien additif dans le système.

$$\underline{Y}_k = \mathbf{H}_k \underline{X}_k + \Psi \underline{V}_k \quad (2.12)$$

où

- $\underline{Y}_k \in \mathcal{R}^{d_4}$  est la mesure à l'instant  $k$ ,
- $\mathbf{H}_k$  la matrice d'observation du système de taille  $d_4 \times d_1$ ,
- $\underline{V}_k \in \mathcal{R}^{d_5}$  est le bruit de mesure,
- $\Psi$  la matrice d'interaction du bruit de mesure dans l'équation d'observation de taille  $d_4 \times d_5$ .

De plus, les vecteurs de variables aléatoires  $\underline{W}_k$  et  $\underline{V}_k$  sont supposés mutuellement indépendants, indépendants de l'état initial  $\underline{X}_0$  du système, blancs et suivant une distribution de probabilité normale :

$$\begin{aligned} p(\underline{W}_k) &\sim \mathcal{N}(0, \mathbf{Q}_k) \\ p(\underline{V}_k) &\sim \mathcal{N}(0, \mathbf{R}_k) \end{aligned} \quad (2.13)$$

où les matrices  $\mathbf{Q}_k$  et  $\mathbf{R}_k$  sont les matrices de covariance associées respectivement au processus et aux mesures. Enfin l'état initial  $\underline{X}_0$  du filtre doit suivre la loi normale  $\mathcal{N}(\bar{\underline{X}}_0, \mathbf{P}_0)$  avec :

$$\begin{aligned} \bar{\underline{X}}_0 &= E[\underline{X}_0] \\ \mathbf{P}_0 &= E[(\underline{X}_0 - \bar{\underline{X}}_0)(\underline{X}_0 - \bar{\underline{X}}_0)^T] \end{aligned} \quad (2.14)$$

De par le fait de la linéarité des équations d'état et de mesure, du caractère gaussien de l'état et des observations, la loi de probabilité a posteriori gouvernant l'état du système est la loi normale  $\mathcal{N}(\hat{\underline{X}}_{k|k}, \mathbf{P}_{k|k})$  qui peut être entièrement décrite par ses deux premiers moments. En tenant compte des équations de prédiction (2.8) et de correction (2.10), le filtre de Kalman peut se résumer par les trois étapes suivantes.

$$\begin{cases} p(\underline{X}_{k-1} | \underline{Y}_{1:k-1}) = \mathcal{N}(\underline{X}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) \\ p(\underline{X}_k | \underline{Y}_{1:k-1}) = \mathcal{N}(\underline{X}_{k|k-1}, \mathbf{P}_{k|k-1}) \\ p(\underline{X}_k | \underline{Y}_{1:k}) = \mathcal{N}(\underline{X}_{k|k}, \mathbf{P}_{k|k}) \end{cases} \quad (2.15)$$

où les deux premiers moments peuvent être à chaque instant donnés par les équations suivantes :

$$\underline{X}_{k|k-1} = \mathbf{F} \underline{X}_{k-1|k-1} + \mathbf{G} \underline{U}_k \quad (2.16)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F} \mathbf{P}_{k-1|k-1} \mathbf{F}^T + \Phi \mathbf{Q}_k \Phi^T \quad (2.17)$$

et

$$\underline{X}_{k|k} = \underline{X}_{k|k-1} + \mathbf{K}_k (\underline{Y}_k - \mathbf{H} \underline{X}_{k|k-1}) \quad (2.18)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_{k|k-1} \quad (2.19)$$

avec

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}^T (\mathbf{H} \mathbf{P}_{k|k-1} \mathbf{H}^T + \Psi \mathbf{R}_k \Psi)^{-1} \quad (2.20)$$

En général, on appelle l'innovation l'écart entre la mesure et sa prédiction. Elle est donnée par  $\tilde{\underline{Y}}_{k|k} = \underline{Y}_k - \underline{Y}_{k|k-1}$  en sachant que  $\underline{Y}_{k|k-1} = \mathbf{H}_k^T \mathbf{H}_k \underline{X}_{k|k-1}$ . Cet écart pondéré

par le gain de Kalman  $\mathbf{K}_k$  représente la quantité d'information supplémentaire à apporter à la correction lors d'une nouvelle mesure. Le gain de Kalman, toujours compris entre 0 et 1, joue un rôle très important dans l'étape de correction. Il permet de connaître la confiance que l'on peut apporter à la mesure par rapport à la prédiction. Il devient d'autant plus important que la covariance  $\mathbf{R}_k$  du bruit de mesure tend vers une matrice nulle, ce qui privilégie la mesure devant la prédiction. A l'inverse, si la covariance  $\mathbf{P}_{k|k-1}$  prédite tend vers 0, et donc lorsque la matrice  $\mathbf{Q}_k$  tend elle-même vers 0, le gain de Kalman tend aussi vers 0. Alors le filtre privilégie la prédiction devant la mesure.

Il existe au moins trois approches différentes conduisant aux équations du filtre de Kalman- Bucy. Une première approche bayésienne consiste à calculer explicitement les deux étapes de prédiction et de correction. Une deuxième approche par projection qui consiste à montrer que les innovations  $\underline{Y}_k - \mathbf{H}_k \hat{\underline{X}}_{k|k-1}$  constituent une suite de variables aléatoires indépendantes et de projeter le processus d'état sur la base orthonormale associée [Anderson *et al.*, 1979]. Une troisième est de noter que le problème du filtrage linéaire gaussien est équivalent au problème des moindres carrés récursifs.

Si l'hypothèse de bruits gaussiens n'est plus vérifiée (en considérant le système dynamique toujours linéaire), alors on montre que le filtre de Kalman-Bucy fournit une estimation à variance minimale de l'état du système si l'on dispose de leurs deux premiers moments. Dans ce cas  $\underline{X}_{k|k}$  ne désigne plus l'espérance conditionnelle, mais son estimée par le filtre de Kalman-Bucy que l'on dit sous optimal.

### 2.2.2 Le filtre de Kalman étendu EKF

Comme décrit dans le chapitre précédemment, le filtre de Kalman étendu s'inscrit dans les problèmes de filtrage optimal dont le processus et/ou les fonctions de mesures sont linéaires. Malheureusement dans de nombreuses situations ces contraintes ne sont pas vérifiées. Dans [Maybeck, 1979] l'auteur démontre que le filtre de Kalman étendu permet d'estimer l'état d'un système dont le processus et/ou les fonctions de mesures sont non-linéaires en linéarisant les équations du filtre de Kalman autour d'un point de fonctionnement à chaque instant. Pour réaliser ce changement nous devons dans un premier temps modifier les équations (2.11) et (2.12) de la manière suivante :

$$\begin{cases} \underline{X}_k = f_k(\underline{X}_{k-1}, \underline{U}_k, \underline{W}_k) \\ \underline{Y}_k = h_k(\underline{X}_k, \underline{V}_k) \end{cases} \quad (2.21)$$

où  $h_k$  et  $f_k$  sont des fonctions non linéaires. On considère cette fois que le vecteur de commande  $\underline{U}_k$  est entaché d'un bruit blanc supposé gaussien défini par la matrice de covariance  $\mathbf{S}_k$ . Ce bruit est également indépendant des bruits de mesure et d'état ainsi que de l'état initial du système.

Le filtre de Kalman étendu repose sur l'approximation gaussienne de la loi a posteriori

gouvernant l'état. Ainsi les équations (2.15) deviennent :

$$\begin{cases} p(\underline{X}_{k-1}|\underline{Y}_{1:k-1}) \simeq \mathcal{N}(\underline{X}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) \\ p(\underline{X}_k|\underline{Y}_{1:k-1}) \simeq \mathcal{N}(\underline{X}_{k|k-1}, \mathbf{P}_{k|k-1}) \\ p(\underline{X}_k|\underline{Y}_{1:k}) \simeq \mathcal{N}(\underline{X}_{k|k}, \mathbf{P}_{k|k}) \end{cases} \quad (2.22)$$

En faisant l'approximation de Taylor au premier au ordre des deux premiers moments l'équation d'état peut être linéarisée autour de l'estimée  $\underline{X}_{k-1|k-1}$  pour devenir :

$$\underline{X}_{k|k} = f_k(\underline{X}_{k-1|k-1}, \underline{U}_k, 0) + \mathbf{F}_k [\underline{X}_{k-1} - \underline{X}_{k-1|k-1}] + \mathbf{\Phi}_k \underline{W}_k \quad (2.23)$$

où

$$\mathbf{F}_k = \left. \frac{\partial f_k(X, \underline{U}_k, 0)}{\partial X} \right|_{X=\underline{X}_{k-1|k-1}} \quad (2.24)$$

$$\mathbf{G}_k = \left. \frac{\partial f_k(\underline{X}_{k-1|k-1}, U, 0)}{\partial U} \right|_{U=\underline{U}_k} \quad (2.25)$$

$$\mathbf{\Phi}_k = \left. \frac{\partial f_k(\underline{X}_{k-1|k-1}, \underline{U}_k, W)}{\partial W} \right|_{W=\underline{W}_k} \quad (2.26)$$

De la même manière, l'équation d'observation peut être linéarisée autour  $\underline{X}_{k|k-1}$  pour donner :

$$\underline{Y}_k = h_k(\underline{X}_{k|k-1}, 0) + \mathbf{H}_k [\underline{X}_k - \underline{X}_{k|k-1}] + \mathbf{\Psi}_k \underline{V}_k \quad (2.27)$$

où

$$\mathbf{H}_k = \left. \frac{\partial h_k(X, 0)}{\partial X} \right|_{X=\underline{X}_{k|k-1}} \quad (2.28)$$

$$\mathbf{\Psi}_k = \left. \frac{\partial h_k(\underline{X}_{k|k-1}, V)}{\partial V} \right|_{V=\underline{V}_k} \quad (2.29)$$

En appliquant le filtre de Kalman à ces équations linéarisées on obtient alors les équations du filtre de Kalman étendu :

$$\underline{X}_{k|k-1} = f_k(\underline{X}_{k-1|k-1}, \underline{U}_k, 0) \quad (2.30)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{G}_k^T \mathbf{S}_k \mathbf{G}_k + \mathbf{\Phi}_k \mathbf{Q}_k \mathbf{\Phi}_k^T \quad (2.31)$$

et

$$\underline{X}_{k|k} = \underline{X}_{k|k-1} + \mathbf{K}_k (\underline{Y}_k - h_k(\underline{X}_{k|k-1}, 0)) \quad (2.32)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \quad (2.33)$$

où

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{\Psi}_k \mathbf{R}_k \mathbf{\Psi}_k)^{-1} \quad (2.34)$$

Il y a deux sources d'inexactitude qui peuvent influencer l'estimation du filtre de Kalman étendu. La première est la linéarisation des modèles d'état et d'observation à chaque instant  $k$ . Et la seconde est de faire l'hypothèse que l'estimation a priori et a posteriori

de l'état restent toujours gaussiennes. En effet, malgré le théorème de la limite centrale les non-linéarités dans les modèles conduisent souvent les distributions de probabilité a priori et a posteriori vers des distributions de probabilité non gaussiennes. Ainsi les deux premiers moments estimés par le filtre ne sont plus que des approximations de la réalité.

Intuitivement, plus la linéarisation est fine (développement de Taylor à des ordres supérieurs), plus le filtre doit donner de meilleurs résultats. D'ailleurs l'exactitude de la linéarisation dépend des quantités :

$$\frac{E[(X_{k|k-1} - \underline{X}_{k|k-1})(X_{k|k-1} - \underline{X}_{k|k-1})^T]}{E[(X_{k-1|k-1} - \underline{X}_{k-1|k-1})(X_{k-1|k-1} - \underline{X}_{k-1|k-1})^T]} \quad (2.35)$$

ainsi que du degré de non-linéarité des équations d'état et d'observation  $f$  et  $h$ . Plus ces quantités sont petites meilleure est l'approximation. Une approximation de ces quantités peut être fournie par les traces de  $\mathbf{P}_{k-1|k-1}$  et de  $\mathbf{P}_{k|k-1}$ . Dans la pratique, si le système comporte des non-linéarités trop importantes, le système linéarisé est trop éloigné du système réel, par conséquent le filtre de Kalman accordant trop de confiance au système linéarisé a tendance à diverger. Pour remédier à ce problème trois méthodes ont été proposées :

- soit on considère que le modèle de prédiction n'est pas très adapté et on augmente artificiellement la covariance du bruit associée à l'état pour compenser les erreurs de prédiction,
- soit on accorde moins de confiance au passé en augmentant la covariance  $\mathbf{P}_{k|k-1}$
- soit on utilise un processus itératif pour améliorer la linéarisation du modèle. Dans ce cas la mesure courante est alors utilisée pour mettre à jour l'estimation précédente de l'état autour de laquelle un nouveau filtre de Kalman étendu est appliqué. On parle alors de filtre de Kalman étendu itéré [Maybeck, 1979].

Enfin, l'EKF ne pouvant fournir qu'une approximation gaussienne de la loi conditionnelle n'est pas du tout capable de gérer les cas de multimodalité. L'heuristique classiquement utilisée dans ces cas est d'approcher la loi conditionnelle par un mélange de gaussiennes dont chaque composante est estimée par un EKF. Mais l'efficacité de cette technique n'est pas garantie [Sorenson *et al.*, 1971] [Alspach *et al.*, 1972].

### 2.2.3 Le filtre de Kalman sans biais UKF

Proposé dans [Julier *et al.*, 1996] et [Julier *et al.*, 2001], le filtre de Kalman sans biais (Unscented Kalman Filter) est une autre extension du filtre de Kalman pour les systèmes non linéaires. Il repose sur le principe qu'il est plus aisé d'approcher une loi gaussienne par un nuage de points que de linéariser les fonctions d'état et de mesure du filtre de Kalman traditionnel. Ainsi la loi de probabilité a posteriori est représentée par un ensemble d'échantillons captant totalement la loi  $\mathcal{N}(\underline{X}_{k|k-1}, \mathbf{P}_{k|k-1})$ . C'est alors à ces échantillons que sont appliqués les modèles non-linéaires d'état et de mesure.



Cette méthode ressemble au principe des filtres de Monté Carlo mais avec cependant 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 unscented (Unscented Transformation).

### 2.2.3.1 Principe de la transformation "unscented"

Décrite dans [Julier, 1999] et [Merwe *et al.*, 2001] la transformation "unscented" est formulée de la manière suivante. Soit  $\underline{A}$  une variable aléatoire de dimension  $n$  ayant pour moyenne  $\bar{A}$  et covariance  $\mathbf{P}_A$ . Soit une variable  $\underline{B} = g(\underline{A})$  fonction non linéaire de  $\underline{A}$ . On construit les  $2n + 1$  vecteurs  $\underline{\chi}_i$  de poids  $w_i$  de la manière suivante :

$$\begin{cases} \underline{\chi}_0 = \bar{A} & w_0 = \frac{\kappa}{n + \kappa} \\ \underline{\chi}_i = \bar{A} + (\sqrt{(n + \kappa)\mathbf{P}_A})^i & w_i = \frac{1}{2}(n + \kappa) \quad i = 1, \dots, n \\ \underline{\chi}_i = \bar{A} - (\sqrt{(n + \kappa)\mathbf{P}_A})^i & w_i = \frac{1}{2}(n + \kappa) \quad i = n + 1, \dots, 2n \end{cases} \quad (2.36)$$

où  $\kappa$  est un facteur d'échelle qui contrôle la répartition des échantillons autour de la moyenne et  $(\sqrt{(n + \kappa)\mathbf{P}_A})^i$  est la  $i^{eme}$  colonne de la matrice de la décomposition de Cholesky de la matrice  $(n + \kappa)\mathbf{P}_A$ .

$$\kappa = n(\alpha^2 - 1) \quad (2.37)$$

où le paramètre  $\alpha$  détermine la diffusion des sigma points sachant que  $10^{-4} \leq \alpha \leq 1$ . Ensuite les vecteurs  $\underline{\chi}_i$  sont propagés selon la fonction non linéaire  $g$  :

$$B_i = g(\underline{\chi}_i), \quad i = 0, \dots, 2n \quad (2.38)$$

La moyenne et la matrice de covariance de la variable aléatoire  $\underline{B}$  sont données par :

$$\bar{B} \approx \sum_{i=0}^{2n} w_i B_i \quad (2.39)$$

$$\mathbf{P}_B \approx \sum_{i=0}^{2n} w_i (B_i - \bar{B})(B_i - \bar{B})^T \quad (2.40)$$

### 2.2.3.2 Etapes du filtre de Kalman unscented

Considérons un système d'état défini par l'équation (2.21). Pour que les bruits de mesures puissent être pris en compte dans le calcul des sigma points, le système d'état doit être modifié, le vecteur d'état est alors augmenté en intégrant les bruits de prédiction et d'observation, c'est-à-dire :

$$\underline{X}_{k-1|k-1}^a = \begin{pmatrix} \underline{X}_{k-1|k-1}^a \\ \underline{W}_k \\ \underline{V}_k \end{pmatrix} \quad (2.41)$$

et les équations de prédiction et d'observation sont réécrites de la manière suivante :

$$\begin{cases} \underline{X}_{k|k-1}^a &= f^a(\underline{X}_{k-1|k-1}^a, \underline{U}_k) \\ \underline{Y}_{k|k-1} &= h^a(\underline{X}_{k|k-1}^a) \end{cases} \quad (2.42)$$

Enfin, la mise en œuvre du filtre de Kalman unscented suit alors les étapes suivantes :

**Prédiction de l'état :** La première étape consiste à générer un nuage de sigma points répartis dans le volume paramétré par la loi  $\mathcal{N}(\underline{X}_{k-1|k-1}^a, \mathbf{P}_{k-1|k-1}^a)$  puis de faire évoluer ce nuage selon le modèle d'état. Enfin l'état a priori  $\underline{X}_{k|k-1}^a$  ainsi que la matrice de covariance  $\mathbf{P}_{k|k-1}^a$  sont calculés :

1. Evolution du nuage de sigma points :

$$(\chi_k)_i = f_k^a((\chi_{k-1})_i, \underline{U}_k) \quad i = 0 \dots 2n \quad (2.43)$$

2. L'état a priori est estimé :

$$\underline{X}_{k|k-1}^a = \sum_{i=0}^{2n} W_i (\chi_k)_i \quad (2.44)$$

où les poids  $W_i$  associés au sigma point  $(\chi_k)_i$  :

$$\begin{aligned} W_0 &= \frac{\kappa}{n + \kappa} \\ W_i &= \frac{1}{2(n + \kappa)} \quad i = 1 \dots 2n \end{aligned} \quad (2.45)$$

3. La matrice de covariance représentant les erreurs de l'état a priori est définie par :

$$\mathbf{P}_{k|k-1}^a = \sum_{i=0}^{2n} W_i ((\chi_k)_i - \underline{X}_{k|k-1}^a)((\chi_k)_i - \underline{X}_{k|k-1}^a)^T \quad (2.46)$$

où  $\mathbf{Q}_k$  est la matrice représentant les erreurs du processus d'état et les poids  $W_i$  décrits par :

$$\begin{aligned} W_0 &= \frac{\kappa}{n + \kappa} + (1 - \alpha^2 + \beta) \\ W_i &= \frac{\kappa}{n + \kappa} \quad i = 1 \dots 2n \end{aligned} \quad (2.47)$$

Le paramètre  $\beta$  est intégré dans le calcul des poids pour introduire une connaissance a priori sur la distribution de probabilité du vecteur d'état. Ainsi si  $p(X)$  est une gaussienne la valeur optimale du paramètre  $\beta$  est 2.

**Prédiction de l'observation :** Comme l'étape précédente, il faut faire évoluer le nuage de sigma points selon le modèle d'observation, puis calculer l'observation a priori  $\underline{Y}_{k|k-1}$  et sa matrice de covariance  $\mathbf{P}_{\underline{Y}_{k|k-1}\underline{Y}_{k|k-1}}$  :

1. Evolution du nuage de sigma points :

$$(Y_k)_i = h_k^a((\chi_{k-1})_i) \quad i = 0 \dots 2n \quad (2.48)$$

2. L'observation a priori est estimée :

$$\underline{Y}_{k|k-1} = \sum_{i=0}^{2n} W_i (Y_k)_i \quad (2.49)$$

3. La matrice de covariance :

$$\mathbf{P}_{\underline{Y}_{k|k-1}\underline{Y}_{k|k-1}} = \sum_{i=0}^{2n} W_i ((Y_k)_i - \underline{Y}_{k|k-1}) ((Y_k)_i - \underline{Y}_{k|k-1})^T \quad (2.50)$$

où  $\mathbf{R}_k$  est la matrice représentant les erreurs du processus d'observation.

**Mise à jour :** dans cette dernière étape, la matrice de covariance croisée  $\mathbf{K}$  entre l'état a priori et l'observation a priori doit être calculée. Cette matrice correspond en fait au gain de Kalman. L'état est finalement mis à jour par les équations du filtre de Kalman traditionnelles.

1. Le gain de Kalman est donc

$$\mathbf{K} = \mathbf{P}_{\underline{X}_{k|k-1}^a \underline{Y}_{k|k-1}} \mathbf{P}_{\underline{Y}_{k|k-1} \underline{Y}_{k|k-1}}^{-1} \quad (2.51)$$

avec

$$\mathbf{P}_{\underline{X}_{k|k-1}^a \underline{Y}_{k|k-1}} = \sum_{i=0}^{2n} W_i ((\chi_k)_i - \underline{X}_{k|k-1}^a) ((Y_k)_i - \underline{Y}_{k|k-1})^T \quad (2.52)$$

2. La mise à jour est alors réalisée, l'état et la matrice de covariance a posteriori sont :

$$\underline{X}_{k|k}^a = \underline{X}_{k|k-1}^a + \mathbf{K} (\underline{Y}_k - \underline{Y}_{k|k-1}) \quad (2.53)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K} \mathbf{P}_{\underline{X}_{k|k-1}^a \underline{Y}_{k|k-1}} \mathbf{K}^T \quad (2.54)$$

### 2.2.3.3 Conclusion sur l'UKF

Le filtre de Kalman unscented est plus robuste aux non-linéarités que le filtre de Kalman étendu. Dans [Julier *et al.*, 2004] l'auteur montre qu'il est équivalent à l'EKF avec une linéarisation à l'ordre 2. Il ne demande aucune linéarisation du modèle et donc aucun calcul explicite des matrices jacobiniennes n'est nécessaire. Enfin son coût de calcul est quasiment identique à celui de l'EKF. En revanche, pour un bon fonctionnement de ce filtre, il est absolument nécessaire que les distributions de probabilité a priori et a posteriori soient de nature gaussienne, ce qui n'est pas forcément le cas dans les méthodes de localisation (étape de prédiction biaisée par les capteurs ou le modèle cinématique).

## 2.3 Les filtres de Monté Carlo

Le principe de base des filtres de Monté Carlo est présenté dans cette partie. Comme toutes méthodes de Monté Carlo, ces filtres reposent sur un grand nombre de simulations. Dans le cas du filtrage, les paramètres simulés sont appelés des particules et représentent la trajectoire d'un système dynamique, d'où l'appellation de filtre particulaire ou encore de filtre à particules. Cette méthode considère le problème de filtrage d'un point de vue probabiliste. Ainsi le filtre optimal est l'estimation de la loi de probabilité de l'état  $\underline{X}_k$  conditionnellement à l'ensemble des mesures  $\underline{Y}_1, \dots, \underline{Y}_k$  c'est-à-dire  $p(\underline{X}_k | \underline{Y}_1 \dots \underline{Y}_k)$ . Pour cela, un ensemble de trajectoires d'état est généré selon le système dynamique, à partir duquel une mesure empirique est construite en fonction de l'ensemble des observations. Grâce à la loi des grands nombres cette mesure empirique converge vers la mesure associée  $\underline{X}_k | \underline{Y}_1 \dots \underline{Y}_k$ . [Crisan *et al.*, 2002] [Arulampalam *et al.*, 2002]

La première grande famille de filtres de Monté Carlo est basée sur la procédure SIS (Sequential Importance Sampling). Ainsi dans les années 70 le filtre de Monté Carlo pondéré vit le jour [Handschin, 1970]. L'idée de base est de faire évoluer  $N$  particules indépendantes les unes des autres et de les pondérer à chaque instant  $k$  en fonction de leurs vraisemblances avec les observations. La convergence d'une telle méthode demandant la simulation d'un grand nombre de trajectoires, les capacités de calcul des ordinateurs de l'époque empêchèrent leur développement et ces méthodes ne connurent un véritable intérêt qu'à la fin des années 80. Cependant malgré l'évolution de l'informatique, cette méthode peut avoir tendance à long terme à rassembler l'ensemble des particules en un seul point de l'espace d'état et donc diverger.

Pour endiguer ce problème, une deuxième famille de filtres de Monté Carlo basée sur la procédure SIS-R (Sequential Importance Samplin-Resampling) a été introduite dans les années 90. Ce nouveau filtre présente une différence notable dans l'évolution des trajectoires. Ainsi elles n'évoluent plus de manière indépendante mais subissent une sélection en fonction de leur vraisemblance avec les observations : certaines sont alors dupliquées et d'autres éliminées. Suivants les différents auteurs comme Del Moral et al dans [Moral, 1996], Gordon dans [Gordon *et al.*, 1993], Kitawaga dans [Kitawaga, 1996] ou encore Isard et Blake dans [Isard *et al.*, 1998] on parle respectivement de filtre particulaire avec interaction ou interacting particle filter, de filtre bootstrap ou "bootstrap filter", de filtre de Monté Carlo ou encore de Condensation. Cependant malgré cette amélioration il peut arriver que ces filtres divergent encore à long terme.

Fin des années 90, plusieurs auteurs ont essayé de résoudre le problème de divergence des filtres particulaires avec interaction. Par exemple, Oudjane [Oudjane, 2000] propose un nouveau filtre à particules qui utilise non plus une représentation discrète de la densité de probabilité mais plutôt une représentation continue grâce à l'utilisation de noyau de régularisation. Il espère ainsi échapper au phénomène de dégénérescence des particules. On parle alors de filtre particulaire régularisé. Higuchi [Higuchi, 1997] propose de résoudre ce problème en intégrant la théorie de l'évolution génétique dans l'étape de prédiction du

filtre particulière avec interaction. Les particules sont alors considérées comme des individus ou encore des chromosomes qui évoluent selon la dynamique du système d'état et qui peuvent subir des mutations génétiques. L'auteur espère ainsi obtenir une couverture optimale de l'espace d'état par le jeu de particules et endiguer le phénomène de dégénérescence. Ce nouveau filtre est appelé le filtre particulière génétique.

Tout d'abord dans ce chapitre le principe de l'échantillonnage pondéré, nécessaire pour la compréhension de tout filtre particulière, est décrit. Puis quelques uns des principaux filtres particuliers énoncés ci-dessus sont présentés dans l'ordre chronologique.

### 2.3.1 Echantillonnage pondéré IS

Le tirage suivant une fonction d'importance est une méthode de Monté Carlo très utilisée dans les chaînes de Markov [Marshall, 1956] [Rubin, 1988]. Considérons le système suivant :

- le vecteur d'état  $\underline{X}_k$  à estimer est un processus Markovien de distribution initiale  $\Pi_0$  et de densité de probabilité de transition  $p(\underline{X}_k|\underline{X}_{k-1})$ ,
- les observations  $\underline{Y}_k$  sont indépendantes conditionnellement à la variable  $\underline{X}_k$  et ont pour distribution de probabilité  $p(\underline{Y}_k|\underline{X}_k)$ .

Pour toute fonction  $f$  intégrable par rapport à la loi de transition  $p(\underline{X}_{0:k}|\underline{X}_{0:k-1})$ , son espérance est alors :

$$I(f) = \int f(\underline{X}_{0:k})p(\underline{X}_{0:k}|\underline{Y}_{1:k})d_{\underline{X}_{0:k}} \quad (2.55)$$

Dans les approches classiques de Monté Carlo pour estimer  $I(f)$ , il suffit de générer les trajectoires de  $N$  variables aléatoires  $X_{0:k}^i$ ,  $i = 1, \dots, N$  suivant la loi de transition  $p(\underline{X}_{0:k}|\underline{Y}_{1:k})$ , puis de calculer l'espérance de  $p(\underline{X}_{0:k}|\underline{Y}_{0:k})$  à partir des trajectoires :

$$p^N(\underline{X}_{0:k}|\underline{Y}_{1:k}) = \frac{1}{N} \sum_{i=1}^N \delta_{\underline{X}_{0:k}^i} \quad (2.56)$$

et d'estimer  $I(f)$  :

$$I_N(f) = \frac{1}{N} \sum_{i=1}^N f(\underline{X}_{0:k}^i) \quad (2.57)$$

Alors, selon la loi des grands nombres on peut considérer que  $I_N(f)$  approche au mieux  $I(f)$ , si  $N$  tend vers l'infini.

$$\lim_{N \rightarrow \infty} I_N(f) = I(f) \quad (2.58)$$

Du point de vue théorique cette méthode d'estimation est tout à fait satisfaisante, cependant dans la pratique il est impossible de générer de manière efficace les trajectoires selon  $p(\underline{X}_{0:k}|\underline{Y}_{1:k})$ , car  $p(\underline{X}_{0:k}|\underline{Y}_{1:k})$  n'est connue qu'à un coefficient près de normalisation.

Une solution alternative est d'utiliser la méthode d'échantillonnage pondéré. L'idée de cette méthode est la suivante. Prenons une fonction d'importance  $q()$  dont la loi de probabilité est  $q(\underline{X}_{0:k}|\underline{Y}_{1:k})$  qui soit facilement échantillonnable et ayant pour seule contrainte :

$$p(\underline{X}_{0:k}|\underline{Y}_{1:k}) > 0 \Rightarrow q(\underline{X}_{0:k}|\underline{Y}_{1:k}) > 0 \quad (2.59)$$

Alors, les trajectoires sont générées suivant la fonction d'importance et l'espérance de  $f$  s'écrit :

$$\begin{aligned} I(f) &= \int f(\underline{X}_{0:k}) \frac{p(\underline{X}_{0:k}|\underline{Y}_{1:k})}{q(\underline{X}_{0:k}|\underline{Y}_{1:k})} q(\underline{X}_{0:k}|\underline{Y}_{1:k}) d\underline{X}_{0:k} \\ I(f) &= \int f(\underline{X}_{0:k}) \frac{p(\underline{Y}_{1:k}|\underline{X}_{0:k})p(\underline{X}_{0:k})}{p(\underline{Y}_{1:k})q(\underline{X}_{0:k}|\underline{Y}_{1:k})} q(\underline{X}_{0:k}|\underline{Y}_{1:k}) d\underline{X}_{0:k} \\ &= \frac{E_q[f(\underline{X}_{0:k})\tilde{w}(\underline{X}_{0:k})]}{E_q\tilde{w}(\underline{X}_{0:k})} \end{aligned} \quad (2.60)$$

avec les poids d'importance :

$$\tilde{w}(\underline{X}_{0:k}) = \frac{p(\underline{Y}_{1:k-1}|\underline{X}_{0:k})p(\underline{X}_{0:k})}{q(\underline{X}_{0:k}|\underline{Y}_{1:k})} \quad (2.61)$$

Pour estimer  $I(f)$ , il faut générer les  $N$  trajectoires suivant la loi d'importance  $q(\underline{X}_{0:k}|\underline{Y}_{0:k})$ , les pondérer selon  $\tilde{w}$  et enfin normaliser les poids :

$$\hat{I}_N(f) = \frac{\frac{1}{N} \sum_{i=1}^N f(\underline{X}_{0:k}^i) \tilde{w}^i}{\frac{1}{N} \sum_{i=1}^N \tilde{w}^i} = \sum_{i=1}^N f(s_{0:k}^i) w^i \quad (2.62)$$

où  $w^i = \tilde{w}^i / \sum_{i=1}^N \tilde{w}^i$  sont les poids normalisés.

Cette méthode plus connue sous le nom de "Bayesian importance sampling" est théoriquement viable, cependant elle contraint à recalculer l'ensemble des trajectoires à chaque instant  $k$ .

### 2.3.2 Filtre de Monté Carlo avec procédure SIS

La procédure SIS (Sequential Importance Sampling) est simplement la généralisation séquentielle de la procédure IS et permet de disposer d'une formulation récursive des trajectoires ainsi que des poids. Ainsi en itérant ce raisonnement la fonction d'importance est alors :

$$q(\underline{X}_{0:k}|\underline{Y}_{1:k}) = q(\underline{X}_{0:k-1}|\underline{Y}_{1:k-1})q(\underline{X}_k|\underline{X}_{0:k-1}, \underline{Y}_{1:k}) \quad (2.63)$$

et donc séquentiellement :

$$q(\underline{X}_{0:k}|\underline{Y}_{1:k}) = q(\underline{X}_0|\underline{Y}_0) \prod_{k=1}^N q(\underline{X}_k|\underline{X}_{k-1}, \underline{Y}_{1:k}) \quad (2.64)$$

Cette fonction d'importance permet alors d'évaluer de manière récurrente les poids d'importance (processus de Markov d'ordre 1) :

$$\tilde{w}_k^n = \tilde{w}_{k-1}^n \frac{p(\underline{Y}_k | \underline{X}_k^n) p(\underline{X}_k^n | \underline{X}_{k-1}^n)}{q(\underline{X}_k^n | \underline{X}_{k-1}^n, \underline{Y}_{0:k})} \quad (2.65)$$

Le filtre de Monté Carlo avec procédure SIS peut se résumer par l'algorithme 2.1.

---

#### Algorithme 2.1 : Filtre de Monté Carlo à procédure SIS

---

- **Pour**  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n)$  selon la loi initiale  $\mathcal{L}_0$ .
  - pondérer les particules  $\tilde{w}_0^n = \frac{p(\underline{Y}_0 | \underline{X}_0^n) p(\underline{X}_0^n)}{q(\underline{X}_0^n | \underline{Y}_0)}$
  - normaliser les  $w_0^n = \tilde{w}_0^n / \sum_{n=1}^N \tilde{w}_0^n$
- **Pour**  $k \geq 1$ 
  - **Evolution** : générer  $\underline{X}_k^n$  selon la fonction d'importance  $q(\underline{X}_k^n | \underline{X}_{k-1}^n, \underline{Y}_{0:k})$  et poser que  $\underline{X}_{0:k}^n = (\underline{X}_{0:k-1}^n, \underline{X}_k^n)$ .
  - **Correction** :

- pour toutes les particules calculer les poids non normalisés :

$$\tilde{w}_k^n = \tilde{w}_{k-1}^n \frac{p(\underline{Y}_k | \underline{X}_k^n) p(\underline{X}_k^n | \underline{X}_{k-1}^n)}{q(\underline{X}_k^n | \underline{X}_{k-1}^n, \underline{Y}_{0:k})}$$

- puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$ .
    - **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n}$$


---

#### 2.3.2.1 Le filtre de Monté Carlo pondéré

Le filtre de Monté Carlo pondéré est un cas particulier du filtre de Monté Carlo avec procédure SIS où la fonction d'importance est remplacée par la fonction d'évolution a priori  $p(\underline{X}_k^n, \underline{X}_{k-1}^n)$  du système d'état. La prise en compte de la dynamique du système justifie le choix de cette fonction d'importance et permet de simplifier l'algorithme. Cependant, dans ce cas, les particules ne sont plus pondérées en fonction des mesures  $\underline{Y}_{1:k}$ . Le filtre de Monté Carlo pondéré peut être résumé par l'algorithme 2.2.

---

**Algorithme 2.2 : Filtre de Monté Carlo pondéré**


---

- **Pour**  $k = 0$  générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = \frac{1}{N})$  selon la loi initiale  $\mathcal{L}_0$ .

- **Pour**  $k \geq 1$

- **Evolution** : générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k | \underline{X}_{k-1})$ .

- **Correction** :

- pour toutes les particules calculer les poids non normalisés :

$$\tilde{w}_k^n = p(\underline{Y}_k | \underline{X}_k^n) w_{k-1}^n$$

- puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$ .

- **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n}$$


---

Cette technique s'avère très simple à mettre en œuvre et est indépendante de la dimension de l'état et des non-linéarités du système. Il suffit d'être capable de générer les particules en fonction des lois conditionnelles et de les faire évoluer en fonction du temps. Cependant en pratique cet algorithme n'est pas viable. En effet il s'avère que l'ensemble des poids a tendance à se regrouper sur une seule particule provoquant ainsi la dégénérescence des pondérations et donc la divergence du filtre. De plus, toutes les particules possédant un poids très faible et qui ne sont donc plus significatives dans la densité de probabilité, sont quand même prises en compte dans l'algorithme et contribue à gaspiller du temps de calcul.

Pour remédier à ce problème, plusieurs variantes du filtre de Monté Carlo ont été introduites. La première variante est le filtre de Monté à mémoire limitée où seules les  $K$  dernières observations sont prises en compte pour effectuer la correction, de cette manière les poids des particules ne peuvent plus tendre vers zéro indéfiniment. Une autre solution est le filtre de Monté Carlo à oubli exponentiel. Ce filtre à pour particularité d'oublier le passé progressivement et donc plus les observations sont anciennes moins elles influencent les poids des particules diminuant ainsi leur dégénérescence. Ces deux filtres sont rapidement décrits dans les deux prochains paragraphes.

**Le filtre de Monté Carlo à mémoire limitée :** L'idée principale est d'estimer la loi conditionnelle a posteriori en fonction des  $K$  plus récentes observations, ainsi on ne s'intéresse plus à la loi  $p(\underline{X}_k | \underline{Y}_{1:k})$  mais à la loi  $p(\underline{X}_k | \underline{Y}_{k-K:k})$  ainsi on notera la loi conditionnelle



a posteriori :

$$\mathcal{L}_{k|k,K} = \begin{cases} p(\underline{X}_k | \underline{Y}_{1:k}) & \text{si } k \leq K \\ p(\underline{X}_k | \underline{Y}_{k-K:k}) & \text{sinon} \end{cases} \quad (2.66)$$

Le filtre de Monté Carlo à mémoire limitée est décrit par l'algorithme 2.3.

---

Algorithme 2.3 : Filtre de Monté Carlo à mémoire limitée

---

- **Pour**  $k = 0$  générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = \frac{1}{N})$  selon la loi initiale  $\mathcal{L}_0$ .
- **Pour**  $k \geq 1$ 
  - **Evolution** : générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k | \underline{X}_{k-1})$ .
  - **Correction** :
    - pour toutes les particules calculer les poids non normalisés :

$$\tilde{w}_k^n \sim p(\underline{Y}_k | \underline{X}_k^n) w_{k-1}^n \quad \text{si } k \leq K$$

$$\tilde{w}_k^n \sim \frac{p(\underline{Y}_k | \underline{X}_k^n)}{p(\underline{Y}_k | \underline{X}_{k-K}^n)} w_{k-1}^n \quad \text{sinon}$$

- puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$ .
  - **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{k:k-K}) = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n}$$


---

Le filtre de Monté Carlo à mémoire limitée semble avoir une convergence similaire au filtre optimal sous de fortes contraintes sur la loi d'évolution et la fonction d'observation [Oudjane, 2000]. En dehors de ces hypothèses ce filtre ne fournit pas de performances suffisantes pour être utilisable.

**Le filtre de Monté Carlo à oubli exponentiel** : Le filtre de Monté Carlo à oubli exponentiel reprend l'idée du filtre à mémoire limitée, cette fois l'influence des observations n'est pas tronquée dans le temps mais oubliée progressivement. Cet algorithme reste similaire au filtre de Monté Carlo pondéré, seule l'étape de correction est modifiée avec l'introduction d'un facteur d'oubli  $\alpha \in [0, 1]$ . Les poids des particules pour chaque nouvelle observation sont donnés à un facteur près par :

$$w_k^n \sim h_k(\underline{Y}_k | \underline{X}_k^n) w_{k-1}^{n,\alpha} \quad (2.67)$$

Ainsi la loi conditionnelle a posteriori  $\mathcal{L}_{k|k}^{N,\alpha}$  approximant la loi  $p(\underline{X}_k|\underline{Y}_k, \alpha)$  est :

$$\mathcal{L}_{k|k}^{N,\alpha} = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n} \quad (2.68)$$

Intuitivement, plus le facteur d'oubli  $\alpha$  est proche de 1 plus on prend en compte le passé dans l'estimation de la loi conditionnelle a priori. En revanche, plus  $\alpha$  est proche de zéro, moins le passé est pris en compte dans l'estimation. Comme pour le filtre à mémoire limitée ce filtre a une convergence similaire au filtre optimal que sous de fortes contraintes sur les propriétés de l'étape de prédiction [Oudjane, 2000]. Et donc son comportement n'est pas satisfaisant dans la pratique.

### 2.3.2.2 Conclusion

En conclusion, les filtres de Monté Carlo sont bien souvent peu viables dans la pratique. Le principal défaut de ces méthodes est l'étape de prédiction qui génère les particules de manière indépendante sans prendre en compte les observations. Ainsi les particules ont peu de chance d'explorer correctement les zones intéressantes de l'espace d'état entraînant la divergence à long terme des filtres de Monté Carlo. Pour remédier à ce problème, les filtres particulaires avec interaction ont été introduits.

### 2.3.3 Le filtre de Monté Carlo avec la procédure SIS-R

Contrairement aux filtres de Monté Carlo avec procédure SIS, dans les filtres de Monté Carlo avec procédure SIS-R l'évolution des particules ne s'effectue plus de manière indépendante mais selon un critère de sélection. Ainsi, lors de l'étape de prédiction, les particules les plus vraisemblables au regard des observations ont une forte probabilité d'évoluer et a contrario les particules les moins vraisemblables ont une forte probabilité d'être éliminées du jeu de particules. Pour que le nombre de particules reste constant dans le temps, les particules les plus vraisemblables sont dupliquées avant d'évoluer. De cette manière, les particules se positionnent naturellement dans les parties intéressantes de l'espace d'état. Enfin, dans la pratique, le critère de sélection s'effectue avant l'étape de prédiction à l'aide d'une étape de rééchantillonnage du nuage de particules. La forme générale d'un tel algorithme est décrite par l'algorithme 2.4.

#### 2.3.3.1 Filtre particulaire avec interaction SIR

Le filtre particulaire avec interaction introduit initialement par [Moral *et al.*, 1992] est un cas particulier de l'algorithme avec procédure SIS-R. Tout comme le filtre de Monté Carlo pondéré la fonction d'importance est remplacée par la fonction d'évolution a priori  $p(\underline{X}_k^n, \underline{X}_{k-1}^n)$ . Une description de ce filtre est donné par l'algorithme 2.5.

Ce filtre est actuellement un des plus performants en matière d'estimation associée au filtrage optimal. Comme pour le filtre particulaire SIS-R, il oriente les particules en fonctions des observations dans les zones intéressantes de l'espace d'état et permet d'endiguer

---

**Algorithme 2.4 : Filtre de Monté Carlo avec procédure SIS-R**


---

- **Pour**  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n)$  selon la loi initiale  $\mathcal{L}_0$ .
  - pondérer les particules  $\tilde{w}_0^n = \frac{p(\underline{Y}_0|\underline{X}_0^n)p(\underline{X}_0^n)}{q(\underline{X}_0^n|\underline{Y}_0)}$
  - normaliser les  $w_0^n = \tilde{w}_0^n / \sum_{n=1}^N \tilde{w}_0^n$
- **Pour**  $k \geq 1$ 
  - **Evolution** : générer  $\underline{X}_k^n$  selon la fonction d'importance  $q(\underline{X}_k^n|\underline{X}_{k-1}^n, \underline{Y}_{0:k})$  et poser que  $\underline{X}_{0:k}^n = (\underline{X}_{0:k-1}^n, \underline{X}_k^n)$ .
  - **Correction** :
    - pour toutes les particules, calculer les poids non normalisés :

$$\tilde{w}_k^n = \tilde{w}_{k-1}^n \frac{p(\underline{Y}_k|\underline{X}_k^n)p(\underline{X}_k^n|\underline{X}_{k-1}^n)}{q(\underline{X}_k^n|\underline{X}_{0:k-1}^n, \underline{Y}_{0:k})}$$

- puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$ .
  - **Rééchantillonnage** :
    - effectuer un tirage aléatoire avec remise parmi  $\mathcal{S}_k^N$  et proportionnellement aux poids  $w_k^n$
    - mettre les poids  $w_k^n$  à la valeur  $\frac{1}{N}$
  - **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k|\underline{Y}_{1:k}) = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n}$$


---

le phénomène de dégénérescence des particules lié aux filtres de Monté Carlo. Cependant deux situations peuvent mettre en défaut cette méthode (figure 2.1(a) et 2.1(b)) :

1. la situation où l'observation est très précise et la prédiction n'apporte que très peu d'informations. Seules les particules proches de la mesure ont une forte probabilité au regard de la fonction de vraisemblance et donc la majorité des particules sont éliminées par l'étape de rééchantillonnage ne gardant que très peu de particules représentatives de l'état. Ce phénomène appelé l'appauvrissement des états qui peut être atténué par un échantillonnage périodique.
2. la situation où l'intersection de la loi conditionnelle a priori et de la loi de vraisemblance est relativement faible. La plupart des particules ont une faible probabilité au regard de la fonction de vraisemblance. Après l'étape de rééchantillonnage, la loi a priori n'est que très peu modifiée et n'approche pas véritablement la loi réelle. Ce

---

**Algorithme 2.5 : Filtre particulaire avec interaction SIR**


---

- **Pour**  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = \frac{1}{N})$  selon la loi initiale  $\mathcal{L}_0$ .
- **Pour**  $k \geq 1$ 
  - **Evolution** : générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k^n | \underline{X}_{k-1}^n)$ .
  - **Correction** :
    - pour toutes les particules calculer les poids non normalisés :

$$\tilde{w}_k^n = p(\underline{Y}_k | \underline{X}_k^n) \tilde{w}_{k-1}^n$$

- puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$ .
  - **Rééchantillonnage** :
    - si le critère de rééchantillonnage est vérifié alors :  
(taille efficace ou entropie du jeu de particules)
      - effectuer un tirage aléatoire avec remise parmi  $\mathcal{S}_k^N$  et proportionnellement aux poids  $w_k^n$
      - mettre les poids  $w_k^n$  à la valeur  $\frac{1}{N}$
  - **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n}$$


---

phénomène est bien sûr la dégénérescence des particules.

### 2.3.3.2 Les techniques de rééchantillonnage

Comme on l'a vu dans la section précédente, pour endiguer le phénomène de dégénérescence il est nécessaire d'effectuer un rééchantillonnage du jeu de particule, en générant un autre nuage de particules de même taille où sont favorisées les particules de plus grand poids. Celles-ci sont dupliquées et celles avec un poids faible sont éliminées. Nous allons dans la suite de cette section présenter les principales techniques de rééchantillonnage utilisées dans la pratique, toutes ces méthodes assurent que le nombre de descendants d'une particule est toujours proportionnel à son poids, c'est-à-dire que  $E[N_k^n] = N w_k^n$ . En revanche la variance des poids des particules diffère d'une méthode à l'autre.

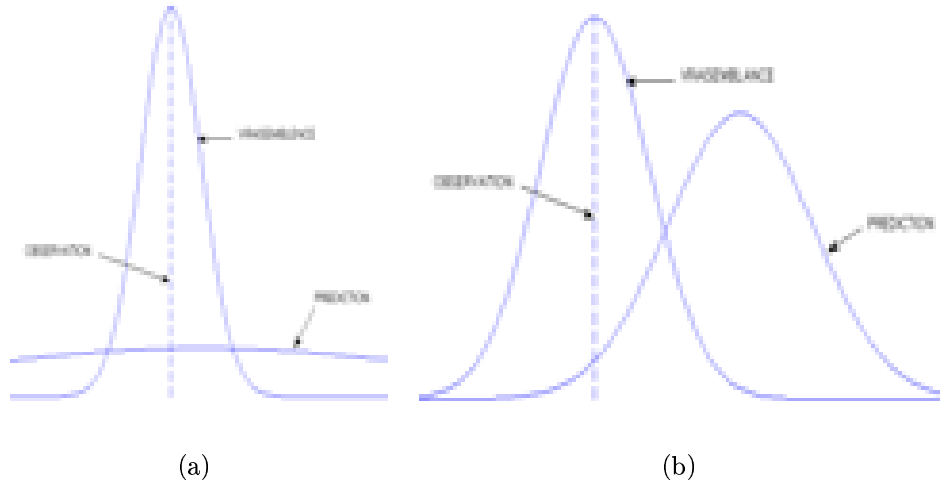


FIG. 2.1: Situations mettant en défaut le filtre particulaire avec interaction, figure (a) : observation très précise, figure (b) : observation incohérente avec la prédiction

### Le rééchantillonnage multinomial

Décrite dans [Liu *et al.*, 1995], cette technique est la plus couramment utilisée, chaque particule est sélectionnée en fonction de l'importance de son poids  $w_k^i$  suivant cette démarche :

- un ensemble de variables aléatoires ordonnées  $u^n \in [0, 1], i = 1, \dots, N$  est généré,
- si la variable aléatoire  $u^j$  est comprise dans l'intervalle  $[\sum_{n=1}^j w_k^n, \sum_{n=1}^{j-1} w_k^n]$ , la particule de poids  $w^j = [\sum_{n=1}^j w_k^n - \sum_{n=1}^{j-1} w_k^n]$  est sélectionnée, si  $m$  variables aléatoires  $u^j$  sont comprises dans cet intervalle alors la particule de poids  $w^j$  est dupliquée  $m$  fois,
- enfin les poids de chacune des particules sont réinitialisés à  $\frac{1}{N}$ .

Si on note  $N_k^{n'}$  le nombre de particules dupliquées pour chaque particule  $n$ , tel que si  $N_k^{n'} = 0$  la particule est éliminée, alors  $E[N_k^{n'}] = Nw_k^n$  et  $var(N_k^{n'}) = Nw_k^n(1 - w_k^n)$ .

### Le rééchantillonnage résiduel

Cette technique de rééchantillonnage [Liu *et al.*, 1998] est une amélioration du rééchantillonnage multinomial qui vise à diminuer la variance  $var(N_k^{n'})$  des particules dupliquées :

- tout d'abord  $\lfloor Nw_k^n \rfloor$  descendants de la particule  $n$  sont générés,  $\lfloor \cdot \rfloor$  dénote la partie entière d'un nombre,
- puis afin de conserver un nombre de particules constant, il faut choisir  $\bar{N}$  particules où

$$\bar{N} = \sum_{n=1}^N (Nw_k^n - \lfloor Nw_k^n \rfloor)$$

- enfin un rééchantillonnage multinomial est utilisé pour générer les  $\bar{N}$  échantillons pondérés par le poids résiduel

$$w_k^{n'} = \frac{w_k^n - \lfloor Nw_k^n \rfloor}{\bar{N}}$$

### Le rééchantillonnage à variance minimale

Cette méthode proposée dans [Kitawaga, 1996] et [Carpenter *et al.*, 1999] est une variante de la technique de rééchantillonnage multinomial. A l'encontre des méthodes précédentes, ce ne sont plus  $N$  variables aléatoires ordonnées qui sont générées mais plutôt un ensemble de variables semi-déterministes  $u'^n = (n-m)/N$  avec  $m \in [0, 1]$  et  $n = 1, \dots, N$ . Ainsi cette fois  $E[N_k^{n'}] \neq Nw_k^n$  mais  $var(N_k^{n'})$  est très petite.

### Critères de rééchantillonnage

Si l'étape de rééchantillonnage permet de "rafraîchir" le nuage de particules, elle peut entraîner un appauvrissement des états dans la distribution de probabilité. Pour diminuer ce phénomène il est conseillé de réaliser le rééchantillonnage périodiquement. Ainsi les particules de poids faible peuvent explorer l'espace d'état pendant plusieurs itérations avant d'être potentiellement éliminées. Dans la littérature deux principaux critères ont été proposés pour déterminer la nécessité du rééchantillonnage. Le critère de la taille efficace du nuage de particules  $N_{eff}$  proposé par Kong et Liu [Kong *et al.*, 1994]. Ce critère repose sur la variance de la distribution de probabilité a posteriori et peut être approché par :

$$N_{eff} \approx \frac{1}{\sum_{n=1}^N (w_k^n)^2} \quad (2.69)$$

Ainsi le rééchantillonnage est effectué lorsque l'inégalité  $N_{eff} < N_{T_h}$  est vérifiée avec  $N_{T_h} \in [0, N]$ . L'utilisateur fixe le seuil selon les nécessités de l'application.

Le second critère de rééchantillonnage a été introduit dans [Pham, 2001]. Ce critère est cette fois basé sur l'entropie des pondérations des particules constituant l'approximation de la distribution de probabilité a posteriori :

$$Ent = \log(N) + \sum_{i=1}^N w_k^n \log w_k^n \quad (2.70)$$

On peut remarquer certaines propriétés liées à ce critère, d'abord celui-ci est toujours compris entre  $0 < Ent \leq \log(N)$ . Il est nul si tous les poids des particules sont égaux à  $1/n$  et vaut  $\log[N]$  si l'un des poids est égal à 1 (appauvrissement sévère du jeu de particules). Ainsi l'étape de rééchantillonnage est effectuée si l'inégalité  $Ent > E_{T_h}$  est vérifiée.  $E_{T_h}$  est un seuil fixé par l'utilisateur en fonction de l'application notamment de l'amplitude de la variation du signal de mesure en fonction de l'état.

Qu'il soit systématique ou adaptatif, le rééchantillonnage rompt l'indépendance des particules ainsi les théorèmes classiques de convergence tels que la loi des grands nombres où encore le théorème de la limite centrale sont inapplicables. Des résultats théoriques sur la convergence des méthodes de filtrage particulaire avec interaction peuvent être trouvés dans [Moral, 1995][Crisan *et al.*, 1999]. Du point de vue pratique la dépendance des particules introduite par l'étape de rééchantillonnage ne permet plus une parallélisation du filtre selon chaque particule.

### 2.3.3.3 Conclusion

L'ajout de l'étape de rééchantillonnage a permis d'endiguer le problème de dégénérescence rencontré dans les filtres de Monté Carlo, faisant ainsi du filtre particulaire avec interaction l'un des meilleurs filtres en matière d'estimation pour systèmes non-linéaires. Cependant, ce filtre n'est pas à l'abri du phénomène de dégénérescence notamment lorsque les bruits d'observations et/ou les bruits du système dynamique sont faibles ou inexistant.

Selon certains auteurs dont Oudjane [Oudjane, 2000], le caractère discret du filtre est déterminant dans ce phénomène de divergence et propose de réaliser l'estimation à partir d'une approximation continue de la densité de probabilité de la mesure. Cette approximation est effectuée par l'introduction d'une étape de régularisation dans le filtre particulaire avec interaction. Ce nouveau filtre se nomme alors le filtre particulaire régularisé. D'autres auteurs dont Higuchi [Higuchi, 1997], proposent de résoudre le phénomène de dégénérescence en combinant la théorie de l'évolution génétique à l'étape de prédiction du filtre particulaire avec interaction. Cette technique permet une meilleure couverture de l'espace d'état de la part des particules et diminue fortement le risque d'appauvrissement des états ainsi que la dégénérescence des particules. On parle alors de filtre particulaire génétique. Ces deux filtres sont maintenant présentés en détails dans les deux prochaines sections.

## 2.3.4 Filtre particulaire régularisé RPF

### 2.3.4.1 Description du filtre

L'étape de rééchantillonnage proposée dans les filtres particuliers avec interaction est une méthode commode pour diminuer le problème de dégénérescence du jeu de particules. Cependant lorsque le système dynamique est mal posé, c'est-à-dire le rapport signal sur bruit est soit très grand soit très petit, les filtres particuliers rencontrent des problèmes d'appauvrissement des particules. Ce phénomène est d'autant plus accentué que dans l'étape de rééchantillonnage, les particules représentent une distribution discrète au lieu d'une distribution continue. Si ce problème n'est pas pris en compte correctement, il peut conduire à un appauvrissement sévère des particules où toutes les particules occupent le même point de l'espace d'état donnant ainsi une représentation erronée de la distribution de probabilité a posteriori.

Une solution possible pour endiguer ce problème a été proposée dans [Oudjane, 2000] et [Musso *et al.*, 2001], sous le nom de filtre particulaire régularisé. Ce filtre est identique

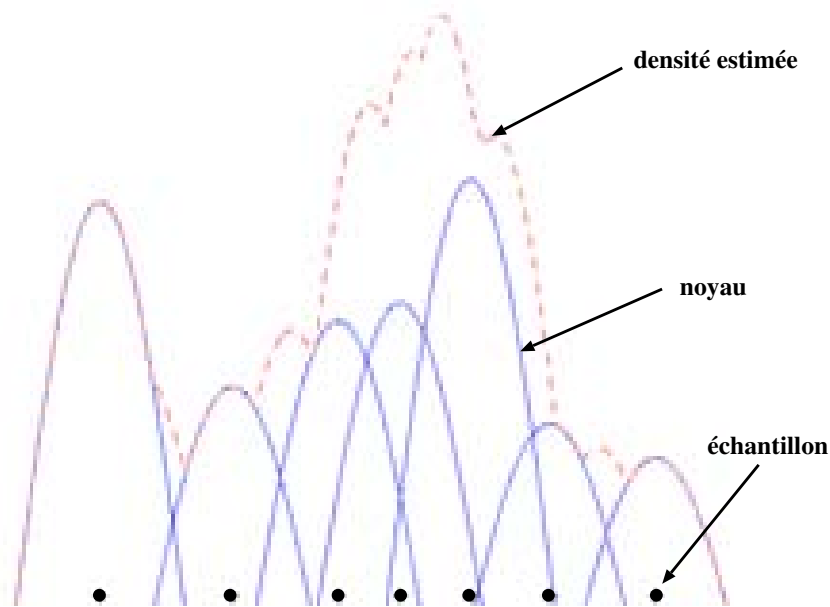


FIG. 2.2: Approximation continue de la distribution de probabilité

au filtre particulaire avec interaction excepté l'étape de rééchantillonnage qui ne s'effectue plus par rapport à une représentation discrète de la distribution de probabilité a posteriori mais à partir d'une approximation continue de la distribution de probabilité  $p(\underline{X}_k | \underline{Y}_{1:k})$  (cf. figure 2.2). Ainsi le filtre particulaire régularisé approche la distribution de probabilité a posteriori selon la relation suivante :

$$p(\underline{X}_k | \underline{Y}_{1:k}) \approx \sum_{n=1}^N w_k^n K_h(\underline{X}_k - \underline{X}_k^n) \quad (2.71)$$

où  $K_h$  est le noyau de régularisation défini par :

$$K_h(\underline{X}) = \frac{1}{h^{nx}} K\left(\frac{\underline{X}}{h}\right) \quad (2.72)$$

avec

- $h$  : le facteur de dilatation qui définit la taille de la fenêtre du noyau de régularisation, sous la contrainte que  $h > 0$ .
- $nx$  : la dimension du vecteur d'état.

Le paramètre  $h$  joue un rôle de lissage de la densité de probabilité a posteriori. Ainsi si  $h$  est petit, le noyau de régularisation approche une impulsion de Dirac et donc on obtient une représentation discrète de la distribution de probabilité. Au contraire si  $h$  est grand on s'approche d'une représentation continue de celle-ci.



Le noyau de régularisation  $K$  et le facteur de dilatation  $h$  sont choisis de manière à minimiser l'erreur quadratique moyenne (notée MISE) entre la vraie distribution de probabilité et la représentation empirique fournie par le filtre particulaire régularisé :

$$MISE(h, K) = E \left[ \int [\hat{p}(\underline{X}_k | \underline{Y}_{1:k}) - p(\underline{X}_k | \underline{Y}_{1:k})]^2 d\underline{X}_k \right] \quad (2.73)$$

Une description du filtre particulaire régularisé est donnée par l'algorithme 2.6.

---

Algorithme 2.6 : Filtre particulaire régularisé RPF

---

- **Pour**  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = \frac{1}{N})$  selon la loi initiale  $\mathcal{L}_0$ .
- **Pour**  $k \geq 1$ 
  - **Evolution** : générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k^n | \underline{X}_{k-1}^n)$ .
  - **Correction** :
    - pour toutes les particules calculer les poids non normalisés :
 
$$\tilde{w}_k^n = p(\underline{Y}_k | \underline{X}_k^n) \tilde{w}_{k-1}^n$$
    - puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$ .
  - **Rééchantillonnage** :
    - si le critère de rééchantillonnage est vérifié alors :
      - calculer  $\mathbf{S}_k$  la matrice de covariance empirique du jeux de particules  $\mathcal{S}_k^N$
      - calculer  $\mathbf{D}_k$  tel que  $\mathbf{D}_k \mathbf{D}_k^T = \mathbf{S}_k$
      - effectuer un tirage aléatoire avec remise parmi  $\mathcal{S}_k^N$  et proportionnellement aux poids  $w_k^n$
      - mettre les poids  $w_k^n$  à la valeur  $\frac{1}{N}$
      - générer  $N$  variables aléatoires  $\epsilon^n$  à partir du noyau d'Epanechnikov
      - enfin réaliser  $\underline{X}_k^n = \underline{X}_k^n + h_{opt} \mathbf{D}_k \epsilon^n$
  - **Estimation** : le nuage  $\mathcal{S}_{k|k}^N = (s_k^k, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_{n=1}^N w_k^n \delta_{\underline{X}_k^n}$$

---

En terme de complexité, le filtre particulaire régularisé est comparable au filtre particulaire avec interaction, demandant seulement en plus la génération de  $N$  particules selon

le noyau de régularisation à chaque étape de rééchantillonnage. Dans la pratique, il obtient de meilleures performances que le filtre particulaire avec interaction, en particulier il est capable d'endiguer les cas d'appauvrissement sévères du jeu de particules lorsque les bruits du système dynamique sont nuls. Néanmoins dans le cas où les bruits de mesures sont faibles la quasi-totalité des particules ont une vraisemblance nulle (figure 2.3) et le filtre particulaire régularisé peut alors diverger malgré l'étape de régularisation.

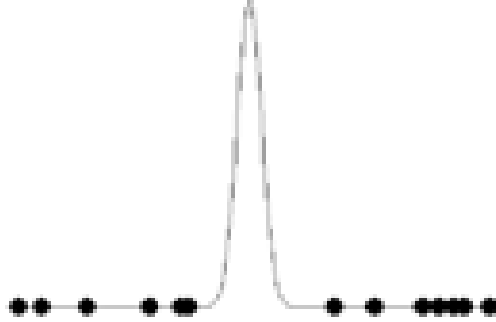


FIG. 2.3: Dégénérescence du filtre particulaire régularisé : faible bruit d'observation

### 2.3.4.2 Choix optimal du noyau

Dans le cas où toutes les particules ont le même poids, le choix optimal du noyau de régularisation est un noyau d'Epanechnikov [Oudjane, 2000] [Rossi, 2004] ainsi :

$$K_{opt} = \begin{cases} \frac{nx+2}{2C_{nx}} (1-||x||) & \text{si } ||x|| < 1 \\ 0 & \text{sinon} \end{cases} \quad (2.74)$$

Pour ce même cas, le paramètre de dilatation optimal est :

$$h_{opt} = A(K)N^{\frac{1}{nx+4}} \quad (2.75)$$

où

$$A = \left[ 8C_{nx}^{-1} (nx+4)(2\sqrt{\pi})^{nx} \right]^{\frac{1}{nx+4}} \quad (2.76)$$

où  $C_{nx}$  est le volume de l'hypersphère défini dans  $\mathcal{R}^{nx}$ . Il est important de noter que le noyau optimal  $K_{opt}$  est totalement indépendant de la distribution de probabilité  $p(\underline{x})$  et qu'en revanche le facteur de dilatation en est fortement dépendant. Son choix est donc primordial pour le comportement de l'estimée [Delaigle, 1999].

### 2.3.5 Le filtre particulaire génétique

Le filtre particulaire génétique proposé initialement par Higuchi dans [Higuchi, 1997] est une méthode originale pour résoudre le problème de dégénérescence du filtre particulaire avec interaction combinant la théorie de l'évolution génétique à celui-ci. Les auteurs proposent d'introduire une étape de mutation génétique dans l'étape traditionnelle de prédiction du filtre particulaire avec interaction, ainsi les particules évoluent selon l'équation dynamique du système d'état puis subissent des mutations de telle façon qu'elles occupent au mieux possible l'espace d'état. Dans cette section nous allons rappeler le principe de l'évolution génétique, comment l'appliquer à un système d'espace d'état et enfin comment l'intégrer au filtre particulaire.

#### 2.3.5.1 Evolution génétique et système d'état

**Le principe de l'évolution en génétique :** Les algorithmes d'évolution génétique permettent de faire évoluer dans le temps une population  $\mathcal{P}_k = (\mathcal{I}_k^1, \mathcal{I}_k^2, \dots, \mathcal{I}_k^N)$  de  $N$  individus (ou chromosomes) où chacun de ces individus représente une solution potentielle au problème posé. L'évolution dans le temps d'une telle approche se décompose en quatre étapes :

1. la création de la population initiale,
2. l'évaluation à l'instant  $k$  de chaque solution fournissant ainsi une mesure de leur importance dans la population,
3. la sélection génétique permettant de former une nouvelle population  $\mathcal{P}_{k+1}$  à partir de la population  $\mathcal{P}_k$  et en fonction des gènes recherchés (c'est-à-dire de leur importance dans la population)
4. l'évolution proprement dite effectuée par la mutation de la population, une nouvelle population  $\mathcal{P}_{k+1}$  est ainsi formée en appliquant des opérateurs génétiques sur la population  $\mathcal{P}_k$

On peut noter qu'il y a deux familles d'opérateurs génétiques différents :

- les opérateurs de mutation<sup>1</sup> : qui permettent de créer de nouveaux individus en appliquant de faible changement à un unique individu,
- les opérateurs de mélange<sup>2</sup> : qui permettent de créer des nouveaux individus en combinant certains gènes de plusieurs individus.

En prenant en compte cette description, on peut définir l'évolution génétique par l'algorithme suivant :

**Evolution génétique d'un système d'espace d'état :** Maintenant, appliquons ce principe à une population dont l'évolution est régie par un modèle d'espace d'état. La

---

<sup>1</sup>mutation operators

<sup>2</sup>crossover operators

---

**Algorithme 2.7 : L'évolution génétique**


---

- Pour  $k = 0$ 
    - **Initialisation** : générer  $\mathcal{P}_0$
  - Pour  $k \leq 1$ 
    - **Evaluation** : calculer  $w_{k-1}^n$  de la population  $\mathcal{P}_{k-1}$ ,
    - **Sélection génétique** : générer  $\mathcal{P}_k$  en fonction de  $\mathcal{P}_{k-1}$
    - **Mutation génétique** : muter  $\mathcal{P}_k$  avec les opérateurs génétiques.
- 

population est alors substituée par un ensemble de vecteurs d'état :

$$\mathcal{P}_k = \underline{X}_k^1, \underline{X}_k^2, \dots, \underline{X}_k^N \quad (2.77)$$

et les différentes étapes de l'évolution deviennent :

1. l'initialisation : doit générer l'ensemble des états  $\underline{X}_k^n$  en fonction de la distribution de probabilité  $p(\underline{X}_0)$  et évaluer l'importance de chacun de ces états selon la distribution de probabilité  $p(\underline{X}_0|\underline{Y}_0)$ .
  - Pour  $n = 1 \dots N$  calculer  $\underline{X}_0^n = p(\underline{X}_0)$  et  $w_0^n = p(\underline{X}_0^n|\underline{Y}_0)$
2. l'évaluation : doit permettre de calculer l'importance de chaque état au sein de la population selon une fonction de vraisemblance  $w_k^n = f(\underline{X}_k^n)$  qui peut être définie heuristiquement ou en suivant les fondements mathématiques du problème posé. Ces fonctions heuristiques peuvent être  $w_k^n = |\underline{Y} - f(\underline{X}_k^n)|$ ,  $w_k^n = (\underline{Y} - f(\underline{X}_k^n))^2$  ou encore la loi normale  $w_k^n = e^{-\frac{(\underline{Y} - \underline{X}_k^n)^2}{2\mathbf{R}}}$ .
  - Pour  $n = 1 \dots N$  calculer  $w_k^n = f(\underline{X}_k^n)$
3. la sélection : permet de créer une nouvelle population en sélectionnant les individus de plus grandes importances, il existe de nombreuses techniques de sélection, entre autre toutes les techniques de rééchantillonnage des filtres particulières avec interactions sont ici utilisables.
4. la mutation génétique : doit générer une nouvelle population en lui appliquant les opérateurs génétiques et permet ainsi de faire évoluer la population dans le temps.
  - Pour  $n = 1 \dots N$  générer un nombre aléatoire  $r \in [0, 1]$ 
    - Si  $r < p_c$  alors sélectionner aléatoirement deux individus  $\{\underline{X}_k^A, \underline{X}_k^B\}$  dans la population de  $\mathcal{P}_k$  et effectuer la mutation suivante :

$$\begin{cases} \underline{X}_k^{newA} &= \gamma \underline{X}_k^A + (1 - \gamma) \underline{X}_k^B \\ \underline{X}_k^{newB} &= \gamma \underline{X}_k^B + (1 - \gamma) \underline{X}_k^A \end{cases} \quad (2.78)$$

où  $\gamma \in [0, 1]$  est choisi par l'utilisateur selon le problème à résoudre. Les parents  $\{\underline{X}_k^A, \underline{X}_k^B\}$  sont alors remplacés par leurs descendants  $\{\underline{X}_k^{newA}, \underline{X}_k^{newB}\}$ .

- Pour  $n = 1 \dots N$  générer un nombre aléatoire  $r \in [0, 1]$ 
  - si  $r < p_m$  alors sélectionner aléatoirement l'individu  $\underline{X}_k^A$  dans la population  $\mathcal{P}_k$  et effectuer la mutation suivante :

$$\underline{X}_k^{newA} = \underline{X}_k^A + \epsilon_m \quad (2.79)$$

où  $\epsilon_m$  est régi par la loi normale  $\mathcal{N}(0, \mathbf{Q}_m)$  définissant la variation possible autour de l'individu parent.

$p_c$  et  $p_m$  représentent la probabilité d'effectuer soit un brassage génétique soit une mutation génétique.

Avec une telle approche, il est possible d'estimer dans les états successifs  $\underline{X}_{0:k} = (\underline{X}_0, \underline{X}_1 \dots \underline{X}_k)$  d'un problème régi par un modèle d'espace d'état, mais probablement avec une mauvaise précision. En effet, l'évolution de la population est effectuée par l'étape de mutation et ne prend pas en compte le modèle dynamique du système d'état. Ainsi la nouvelle population  $\mathcal{P}_k$  est le résultat d'une diffusion des états autour de l'ancienne population  $\mathcal{P}_{k-1}$  (figure 2.4(a)) et n'est donc pas orientée dans l'espace d'état comme le ferait l'étape de prédiction des filtres particulaires (figure 2.4(b)). Intuitivement on comprend que cette méthode rencontre des problèmes de convergence dus au caractère diffus de l'évolution et impose une mauvaise précision sur l'estimée.

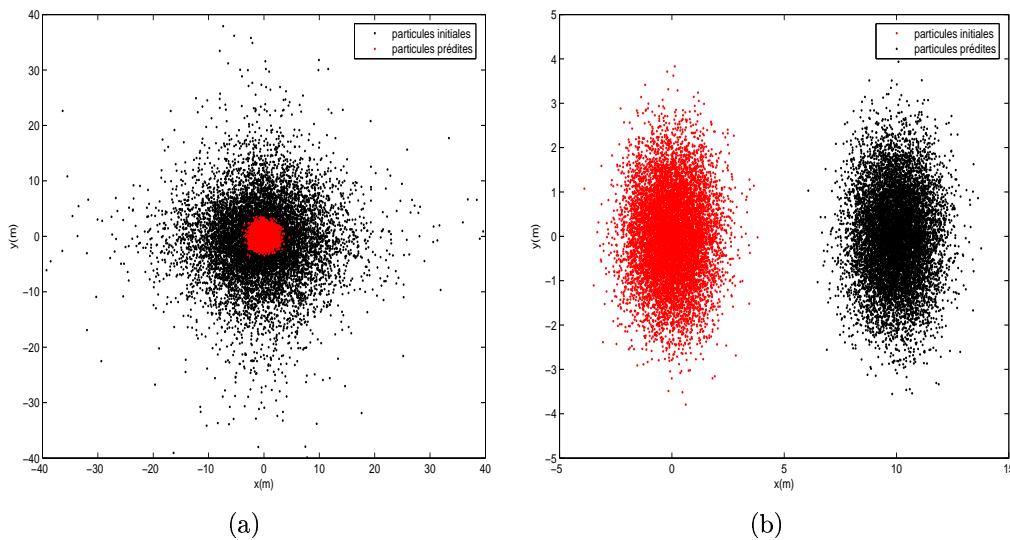


FIG. 2.4: Evolution génétique/filtre particulaire : figure (a) génétique, figure (b) filtre particulaire

### 2.3.5.2 Principe du filtre particulaire génétique

La structure algorithmique d'un filtre particulaire avec interaction et l'approche d'évolution génétique ont en réalité de nombreuses similarités :

- l'association population  $\mathcal{P}^n, w^n$  est similaire au jeu de particules  $S^N = (\underline{X}^n, w^n)$ ,
- l'étape d'évaluation est similaire à l'étape de correction,
- l'étape de sélection est similaire à l'étape de rééchantillonnage.

En effet, seul les étapes d'évolution des états sont différentes dans ces deux approches. Or, chacune de ces étapes présente un défaut :

1. l'étape de mutation impose une convergence lente de la population vers la vraie solution en raison d'une diffusion excessive des particules dans l'espace d'état.
2. lorsque les bruits des équations dynamiques sont faibles, l'étape de prédiction du filtre particulaire introduit un phénomène de dégénérescence du en partie au manque de diffusion des particules dans l'espace d'état.

A priori, leurs défauts semblent être complémentaires et ces deux étapes peuvent être combinées afin de pallier le phénomène de dégénérescence rencontré dans les filtres particulaires avec interaction. Le filtre ainsi obtenu s'appelle le filtre particulaire génétique et peut être décrit par l'algorithme 2.8.

Si l'on compare le filtre particulaire génétique au filtre particulaire avec interaction traditionnel, seule l'étape de prédiction est différente. Elle réalise, dans un premier temps, une mutation du jeux de particules  $\mathcal{S}_{k-1}^n$  selon les différents opérateurs génétiques afin que les particules couvrent le mieux possible l'espace d'état puis, dans un deuxième temps, elle effectue l'évolution des particules selon les équations dynamiques du système d'état. Néanmoins pour que la couverture de l'espace d'état soit optimale, il est nécessaire d'orienter le processus de mutation du jeux de particules et donc de superviser les opérateurs génétiques. Pour ceci, l'algorithme de Métropolis-Hasting [Metropolis *et al.*, 1953] [Hastings, 1970] est utilisé. Les nouveaux opérateurs génétiques résultants sont alors appelés opérateurs génétiques "métropolisés".

### 2.3.5.3 Les opérateurs génétiques "métropolisés"

Avec les opérateurs génétiques traditionnels, le jeu de particules subit des mutations diffuses tout autour de l'ancien jeu de particules. Cependant, pour que les particules couvrent correctement l'espace d'état il est nécessaire que cette mutation soit orientée. Or, lors de la dernière correction, le jeu de particules est orienté dans l'espace d'état, selon la densité d'importance  $p(\underline{Y}_{k-1}, \underline{X}_{k-1}^n)$ . Il est donc obligatoire que les opérateurs génétiques prennent en compte cette densité d'importance pour réaliser la mutation [Liu, 1996]. Pour ceci les opérateurs génétiques traditionnels sont modifiés en introduisant l'algorithme de Métropolis-Hastings dans leur structure. Ces nouveaux opérateurs sont présentés dans les algorithmes 2.9 et 2.10.

---

**Algorithme 2.8 : Filtre particulaire génétique GPF**


---

- Pour  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = 1/n)$  selon la loi initiale  $\mathcal{L}_0$
- Pour  $k \neq 1$ 
  - **Evolution :**
    - appliquer l'opérateur de mélange génétique à  $\mathcal{S}_{k-1}^N$
    - appliquer l'opérateur de mutation génétique à  $\mathcal{S}_{k-1}^N$
    - générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k^n, \underline{X}_{k-1}^n)$

- **Correction :**

- pour toutes les particules calculer les poids non normalisés

$$\tilde{w}_k^n = p(\underline{Y}_k | \underline{X}_k^n) w_k^n$$

- puis normaliser les poids  $w_k^n = \tilde{w}_k^n / \sum_{n=1}^N \tilde{w}_k^n$
- **Rééchantillonnage :**
  - effectuer un tirage aléatoire avec remise parmi  $\mathcal{S}_k^N$  et proportionnellement aux poids  $w_k^n$
  - mettre les poids à la valeur  $\frac{1}{N}$
- **Estimation :** le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_{n=1}^N w_k^n \delta_{s_k^n}$$


---

On peut noter que les probabilités  $p_c$  et  $p_m$  n'apparaissent plus dans les opérateurs génétiques "métropolisés", car elles sont remplacées par les critères de Métropolis.

#### 2.3.5.4 Conclusion

D'après les résultats expérimentaux présentés par Tito dans [Tito *et al.*, 2003a] et [Tito *et al.*, 2003b], le comportement du filtre particulaire génétique est bien meilleur que celui du filtre particulaire avec interaction. La mutation génétique lors de l'étape de prédiction permet bien une meilleure couverture des particules de l'espace d'état et endigue le phénomène de dégénérescence du filtre SIR. De plus l'utilisation de l'algorithme de Métropolis-Hasting dans les opérateurs génétiques assure une couverture optimale de l'espace d'état au vu des observations et provoque une convergence sûre et rapide de ce filtre, même avec un nombre de particules réduit. L'introduction de mutations génétiques dans l'étape d'évolution permet aussi un gain en temps de calcul, en mémoire, en précision du filtre particulaire avec interaction classique.

---

**Algorithme 2.9 : L'opérateur de brassage génétique "métropolisé"**


---

- Pour  $n \dots N$  sélectionner aléatoirement deux particules  $\{\mathcal{S}_{k-1}^A, \mathcal{S}_{k-1}^B\}$  et appliquer leur l'opérateur de brassage génétique traditionnel

$$\begin{cases} \underline{X}_{k-1}^{newA} &= \gamma \underline{X}_{k-1}^A + (1 - \gamma) \underline{X}_{k-1}^B \\ \underline{X}_{k-1}^{newB} &= \gamma \underline{X}_{k-1}^B + (1 - \gamma) \underline{X}_{k-1}^A \end{cases}$$

où  $\gamma$  est régi par la loi uniforme  $\mathcal{U}[0, 1]$ .

- Remplacer les parents par leurs descendants en prenant en compte le critère de Métropolis :
  - $\mathcal{S}_{k-1}^{new}$  est acceptée si

$$p(\underline{Y}_{k-1} | \underline{X}_{k-1}^{new}) > \max\{p(\underline{Y}_{k-1}, \underline{X}_{k-1}^A), p(\underline{Y}_{k-1}, \underline{X}_{k-1}^B)\}$$

- sinon elle est acceptée selon probabilité

$$\frac{p(\underline{Y}_{k-1} | \underline{X}_{k-1}^{new})}{\max\{p(\underline{Y}_{k-1}, \underline{X}_{k-1}^A), p(\underline{Y}_{k-1}, \underline{X}_{k-1}^B)\}}$$


---

---

**Algorithme 2.10 : L'opérateur de mutation génétique "métropolisé"**


---

- Pour  $n \dots N$  sélectionner aléatoirement une particule  $\mathcal{S}_{k-1}^A$  et appliquer lui l'opérateur de mutation génétique traditionnel

$$\underline{X}_{k-1}^{newA} = \underline{X}_{k-1}^A + \epsilon_m$$

où  $\epsilon_m$  régi par  $\mathcal{N}(0, \sigma_m^2)$

- Remplacer le parent par son descendant en prenant en compte le critère de Métropolis
  - $\mathcal{S}_{k-1}^{newA}$  est acceptée si

$$p(\underline{Y}_{k-1} | \underline{X}_{k-1}^{new}) > p(\underline{Y}_{k-1}, \underline{X}_{k-1}^A)$$

- sinon elle est acceptée selon probabilité

$$\frac{p(\underline{Y}_{k-1} | \underline{X}_{k-1}^{new})}{p(\underline{Y}_{k-1}, \underline{X}_{k-1}^A)}$$


---

## 2.4 Conclusion

Dans ce chapitre nous avons décrit les deux principales méthodes de filtrage à savoir : les méthodes analytiques avec le filtre de Kalman et les méthodes numériques avec le



filtre de Monté Carlo et ses variantes. Dans les méthodes analytiques, le filtre de Kalman étendu est le filtre par excellence donnant des résultats rapides et satisfaisants dans le cadre de systèmes faiblement non-linéaires. Cependant lorsque le système devient fortement non-linéaire ses performances se dégradent. L'Unscented Kalman Filter permet de résoudre ce problème mais seulement sous la contrainte que les distributions de probabilité a priori et a posteriori soient de natures gaussiennes. En dehors de ces hypothèses, il fournit lui aussi de mauvais résultats. Il semble donc peu utilisable dans le contexte de la robotique mobile où de nombreux biais sont introduits dans l'étape de prédiction (biais du gyromètre, connaissance approximative de la géométrie du robot etc...) et où les distributions de probabilité ont peu de chance d'être gaussiennes.

Lorsque les conditions de non-linéarités sont fortes et que les méthodes analytiques échouent, les méthodes numériques (en particulier les filtres de Monté Carlo) sont alors utilisées. Cependant les filtres de Monté Carlo présentent eux aussi quelques faiblesses. Ainsi, pour les filtres de Monté Carlo basés sur la procédure SIS, et pour un nombre constant de particules dans le temps, il est quasi-inévitable que l'estimation diverge à long terme. Pour les filtres de Monté Carlo basés sur la procédure SIS-R, en particulier le filtre particulière avec interaction, ce phénomène est atténué par l'ajout d'une étape de rééchantillonnage. Néanmoins ces filtres peuvent encore diverger lorsque le système d'état est faiblement bruité.

Il existe deux méthodes pour diminuer le phénomène de divergence, le filtrage particulière régularisé et le filtrage particulière génétique :

1. La première méthode permet de conserver la diversité du jeu de particules grâce à une étape de régularisation par noyau de convolution et ainsi d'effectuer une estimation correcte lorsque les bruits du système dynamique sont faibles, ou même voire inexistantes. Cependant, ce filtre peut tout de même être mis en défaut lorsque le bruit de mesure est très faible.
2. La seconde méthode permet elle aussi de conserver la diversité du jeu de particules mais cette fois en améliorant l'étape d'évolution du filtre SIR à l'aide d'opérateurs génétiques. Cette technique permet une estimation correcte lorsque les bruits du système dynamique sont faibles. De plus l'introduction de l'algorithme acceptation/rejet de Métropolis-Hasting dans les opérateurs génétiques permet une convergence rapide de ce filtre et une diminution des particules nécessaires à son bon fonctionnement.

Enfin, pour un bon fonctionnement de ces méthodes de filtrage, certaines contraintes sur les informations doivent être respectées. Ainsi pour les divers filtres de Kalman, l'estimation de l'état du système se base sur les deux premiers moments des informations, il est alors préférable que ces informations soient de nature gaussienne et soient nécessairement blanches. En revanche comme les filtres particuliers traitent directement la densité de probabilité, la nature des informations importe peu (ex : uniforme, gaussienne, multimodale etc...). Par contre, comme pour le filtre de Kalman, ces informations doivent être

blanches. Si cette dernière contrainte n'est pas respectée, cela signifie que ces informations comportent une partie stochastique qui est elle-même régie par un système d'état. Pour obtenir un fonctionnement optimal des méthodes de filtrage présentées dans ce chapitre, il est alors nécessaire d'identifier ce système d'état et de l'introduire dans le système d'état du filtre de Kalman ou du filtre particulaire. C'est l'objet du prochain chapitre.



# Chapitre 3

## Modélisation des erreurs

Comme on l'a vu à la fin du chapitre précédent, les méthodes de filtrage requièrent certaines contraintes sur les informations qu'elles traitent dont la plus importante est leur blancheur. Si cette contrainte n'est pas respectée cela signifie que les informations ne sont pas uniquement caractérisées par leurs propriétés statistiques mais sont aussi par leur comportement temporel. Dans ce cas, il est nécessaire de le modéliser et de le prendre en compte dans le processus de fusion afin d'obtenir un filtrage optimal. Considérons  $\tilde{\chi}_k$  une information non blanche, celle-ci peut être décrite de la manière suivante :

$$\tilde{\chi}_k = \chi_k + b_{\chi_k} + \epsilon_{\chi_k} \quad (3.1)$$

avec

- $\tilde{\chi}$  : l'information fournie par un capteur,
- $\chi$  : la valeur réelle de l'information ou la partie systématique de l'information  $\tilde{\chi}$
- $b_{\chi}$  : la partie non-blanche de l'information  $\tilde{\chi}$  que l'on nommera par la suite "partie stochastique",
- $\epsilon_{\chi}$  : un bruit blanc résiduel.

où la partie stochastique  $b_{\chi}$  de l'information  $\tilde{\chi}$  peut être considérée comme un processus stochastique dont le comportement dans le temps est régi par le système d'état suivant :

$$\begin{cases} b_{\chi_k} = \mathbf{F}b_{\chi_{k-1}} + \Phi w_k \\ y_k = \mathbf{H}b_{\chi_k} + \Psi v_k \end{cases} \quad (3.2)$$

Ce système d'état décrit donc le comportement dans le temps de la partie stochastique de l'information  $\tilde{\chi}_k$ , lorsqu'une telle information est prise en compte dans une méthode filtrage. Il est alors nécessaire de caractériser ce système d'état (en particulier le système dynamique) et de l'intégrer au système d'état de la méthode de filtrage.

Dans ce chapitre, nous allons tout d'abord rappeler les divers outils mathématiques nécessaires à la caractérisation des processus stochastiques notamment la fonction d'auto-corrélation qui permet de décrire leur comportement temporel, puis présenter les différents modèles de base permettant une caractérisation simple et rapide de nombreux processus

stochastiques. Ce sont les processus de constante aléatoire, les processus de marche aléatoire ou encore les processus de Gauss-Markov. Enfin lorsque ces derniers ne peuvent approcher les véritables processus stochastiques, les processus autorégressifs sont souvent utilisés. Ceux-ci ont la particularité de pouvoir utiliser un grand nombre de paramètres afin d'obtenir une description très fidèle du processus à modéliser. Ces paramètres sont le plus le souvent déterminés par des méthodes de régression linéaire telle que la méthode de Yule-Walker, la méthode de la Covariance ou encore la méthode de Burg (cf section 3.3).

## 3.1 Fondement mathématiques

### 3.1.1 Fonction d'autocorrélation et densité spectrale

Considérons un processus stochastique stationnaire au second ordre, c'est-à-dire un processus qui à n'importe quel instant du temps à la même moyenne et la même variance. D'après [Brown *et al.*, 1997], pour caractériser entièrement un processus stochastique, il est nécessaire de connaître les dépendances dans le temps de toutes les valeurs du processus et propose d'utiliser les caractéristiques de la fonction d'autocorrélation pour définir le processus.

La fonction d'autocorrélation d'un processus  $b(t)$  est définie comme la moyenne du produit  $b(t).b(t - \tau)$  :

$$\bar{R}_{bb}(\tau) = E[b(t).b(t - \tau)] = \int_{-\infty}^{+\infty} b(t).b(t - \tau)d\tau \quad (3.3)$$

où  $\tau$  est le décalage temporel entre les échantillons.

Si le problème est traité dans le domaine discret, la fonction d'autocorrélation est remplacée par la séquence d'autocorrélation qui est définie de la manière suivante :

$$\bar{R}_{bb}(m) = E[b(k).b(k - m)] = \text{mean} \left[ \sum_{k=-\infty}^{\infty} b(k).b(k - m) \right] \quad (3.4)$$

où

- $k$  représente les instants d'échantillonnage du processus, tels que  $t = k\Delta_T$  avec  $\Delta_T$  la période d'échantillonnage du processus discret,
- $m$  représente le décalage temporel entre les échantillons.

D'après les équations (3.3) et (3.4), les valeurs de  $\bar{R}_{bb}$  sont connues si le nombre de mesures expérimentales du processus aléatoire  $b$  est infini. Cependant, dans la pratique, la fonction d'autocorrélation est calculée à partir d'un nombre de mesures fini. Dans ce cas  $\bar{R}_{bb}(m)$  est alors remplacée par  $R_{bb}(m)$ . Ainsi pour une série de mesures discrètes du

processus  $b(k)$  avec  $k = 1 \dots N$ , la séquence d'autocorrélation est :

$$R_{bb}(m) = E[b(k).b(k - m)] = \frac{1}{N - |m|} \sum_{k=1}^{N-|m|} b(k).b(k - m) \quad (3.5)$$

Pour le décalage  $m = 0$ , la valeur de la séquence d'autocorrélation est donnée par :

$$R_{bb}(0) = E[b^2(k)] = \frac{1}{N} \sum_{k=1}^N b^2(k) = \sigma_b^2 + \mu_b^2 \quad (3.6)$$

où

- $\sigma_b$  est l'écart-type du processus aléatoire  $b(k)$ ,
- $\mu_b$  est la moyenne du processus aléatoire  $b(k)$ .

D'après l'équation (3.6), si le processus étudié est non biaisé (processus à moyenne nulle), pour  $m = 0$  la séquence d'autocorrélation fournit directement la variance de ce processus. De plus, si la fonction d'autocorrélation décroît rapidement dans le temps, le processus varie rapidement dans le temps. Au contraire, si la fonction d'autocorrélation décroît doucement alors le processus varie lentement. De par ce constat, on peut en déduire qu'un processus peut être aussi décrit par ses caractéristiques fréquentielles et notamment par sa densité spectrale de puissance. La densité spectrale de puissance est calculée par la transformée de Fourier de la fonction d'autocorrélation du processus étudié. Les densités spectrales de puissance d'un signal continu et d'un signal discret sont décrites respectivement par les équations (3.7) et (3.8).

$$S_{bb}(w) = \int_{-\infty}^{+\infty} R_{bb}(\tau) e^{-jw\tau} d\tau \quad (3.7)$$

$$S_{bb}(e^{jw}) = \sum_{m=-\infty}^{+\infty} R_{bb}(m) e^{-jwm} \quad (3.8)$$

La densité spectrale de puissance décrit comment la puissance (ou plutôt dans notre cas la variance) d'une série de mesures est distribuée dans le domaine fréquentiel. Réciproquement, la fonction d'autocorrélation ou la séquence d'autocorrélation peuvent être calculées par la transformée inverse de Fourier de la densité spectrale de puissance et sont définies respectivement par les équations (3.9) et (3.10) :

$$R_{bb}(\tau) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S_{bb}(w) e^{jw\tau} dw \quad (3.9)$$

$$R_{bb}(m) = \frac{1}{2\pi} \int_{-\pi}^{+\pi} S_{bb}(e^{jw}) e^{jwm} dw \quad (3.10)$$

### 3.1.2 Le bruit blanc

Le bruit blanc [Anderson *et al.*, 1979], lorsqu'il est stationnaire est un processus aléatoire de moyenne nulle ( $\mu_b = 0$ ) et de densité spectrale de puissance constante ( $S_{bb}(\cdot) = S_{bb}(0)$ ). D'après les équations (3.9) et (3.10), la fonction d'autocorrélation et la séquence d'autocorrélation d'un bruit blanc sont données respectivement par les équations (3.11) et (3.12) :

$$R_{bb}(\tau) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S_{bb}(0) e^{jw\tau} dw = \frac{S_{bb}(0)}{2\pi} \int_{-\infty}^{+\infty} e^{jw\tau} dw = S_{bb}(0) \delta(\tau) \quad (3.11)$$

$$R_{bb}(m) = \frac{1}{2\pi} \int_{-\pi}^{+\pi} S_{bb}(0) e^{jwm} dw = \frac{S_{bb}(0)}{2\pi} \int_{-\pi}^{+\pi} e^{jwm} dw = S_{bb}(0) \delta(m) \quad (3.12)$$

où  $\delta(\tau)$  est la fonction de Dirac pour la représentation continue et  $\delta(m)$  est l'impulsion unitaire pour la représentation discrète et sont définies de la manière suivante :

$$\delta(\tau) = \begin{cases} 0 & \text{pour } \tau \neq 0 \\ \infty & \text{pour } \tau = 0 \end{cases} \quad \text{et} \quad \delta(m) = \begin{cases} 0 & \text{pour } m \neq 0 \\ 1 & \text{pour } m = 0 \end{cases} \quad (3.13)$$

Ainsi en considérant l'équation (3.12) et les propriétés de l'impulsion unitaire  $\delta(m)$ , on peut en déduire que la fonction d'autocorrélation  $R_{bb}(m)$  est égale à  $R_{bb}(0)$  :

$$R_{bb}(0) = \sigma_b^2 = S_{bb}(0) \delta(0) \Leftrightarrow S_{bb}(m) \delta(m) \quad (3.14)$$

Les figures 3.1(a) et 3.1(b) montrent respectivement la fonction d'autocorrélation et la densité spectrale de puissance d'un bruit blanc. D'après la fonction d'autocorrélation, il n'existe aucune autocorrélation d'un bruit blanc en dehors de  $m=0$ . De plus, en prenant en compte les caractéristiques de l'impulsion de Dirac, la variance d'un bruit blanc est infinie. Ainsi le bruit blanc (souvent appelé processus aléatoire pur) est seulement un concept théorique et n'est pas physiquement réalisable [Gelb, 1974]. En dépit de ceci, dans la pratique, le bruit blanc peut être utilisé avec succès pour approcher certains processus aléatoires ou bien encore pour générer d'autres processus aléatoires en le passant au travers de filtres formeurs (shaping filters en anglais).

## 3.2 Processus générés à partir d'un bruit blanc

La plupart des processus stochastiques ne peuvent être définis analytiquement car à tout instant ils sont gouvernés par le hasard. Cependant ils peuvent être approchés par des processus dont les caractéristiques (moyenne, variance, fonction d'autocorrélation etc...) se rapprochent le plus possible de leurs caractéristiques. Ces processus sont généralement générés en passant un bruit blanc à travers un filtre formeur (généralement un système dynamique linéaire), ce filtre modifie donc les caractéristiques de la séquence d'entrée pour qu'elle approche le plus possible les caractéristiques du processus stochastique à modéliser. Dans ces conditions, les différents paramètres du filtre formeur sont optimalement estimés en minimisant au sens des moindres carrés la différence entre le processus et la

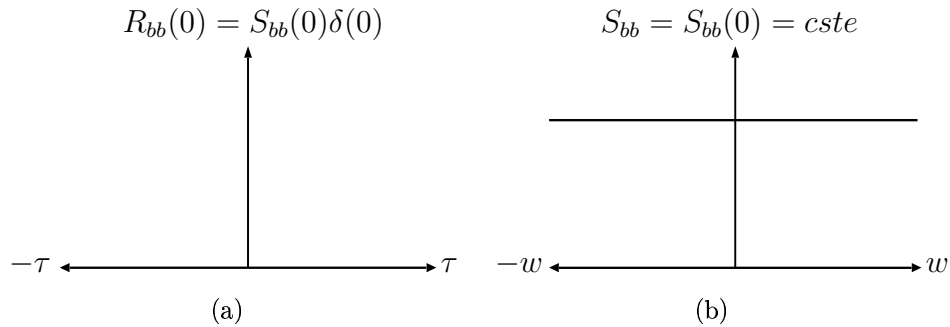


FIG. 3.1: Caractéristiques d'un bruit blanc, figure (a) fonction d'autocorrélation figure (b) densité spectrale

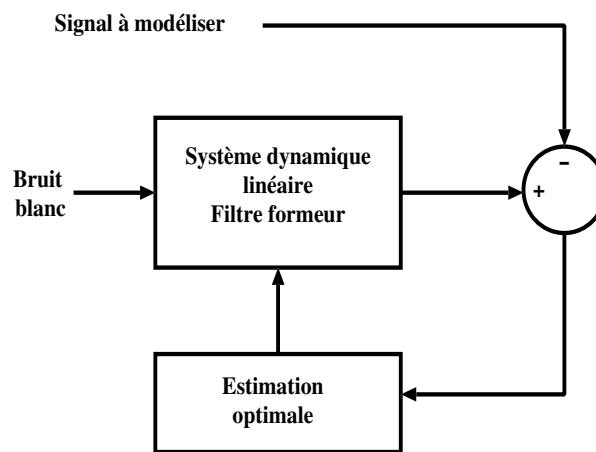


FIG. 3.2: Détermination du filtre formeur

sortie du filtre formeur (figure 3.2).

Dans la suite de cette section nous allons présenter les modèles de processus stochastiques les plus utilisés dans le domaine de la robotique mobile. De plus, pour plus de lisibilité,  $b(k)$  est désormais écrit  $b_k$ .

### 3.2.1 La constante aléatoire

La constante aléatoire appelée en anglais random bias ou random constant, est une quantité aléatoire impossible à prédire et de valeur constante. Dans ce cas, ce processus est défini par l'équation différentielle :

$$\dot{b}(t) = 0 \quad (3.15)$$

Dans le domaine discret, ce processus est alors défini par l'équation de récurrence :

$$b_{k+1} = b_k \quad (3.16)$$



En substituant l'équation (3.16) dans l'équation (3.4) on obtient alors la fonction d'auto-corrélation d'un processus de constante aléatoire :

$$R_{bb}(m) = E[b_k^2] = R_{bb}(0) = \text{cste} \quad (3.17)$$

Ainsi, le filtre formeur nécessaire à l'obtention d'un tel processus est un filtre qui a pour particularité : une condition initiale aléatoire, aucune entrée et réalise une intégration de sa sortie d'un instant à l'autre.

### 3.2.2 La marche aléatoire

Pour un processus de type de marche aléatoire (appelé en anglais random walk process ou Winner process ou encore Brownian-motion process), sa prédiction est entièrement définie par un processus aléatoire pur, c'est-à-dire un bruit blanc. Dans le domaine continu ce processus est donc décrit par l'équation différentielle :

$$\dot{b}(t) = \epsilon(t) \quad (3.18)$$

et dans le domaine discret par l'équation de récurrence :

$$b_{k+1} = b_k + \epsilon_k \quad (3.19)$$

Ainsi, le processus de marche aléatoire est généré en intégrant une séquence aléatoire décorrélée dans le temps. D'ailleurs en prenant en compte la théorie de la limite centrale et pour un nombre de mesures très important, l'équation (3.19) converge vers :

$$b_{k+1} = \sum_{i=1}^k \epsilon_i \quad (3.20)$$

D'après cette équation, il est facile de déterminer la moyenne et la variance d'un tel processus :

$$\mu_b = E[b_{k+1}] = E\left[\sum_{i=1}^k \epsilon_i\right] = \sum_{i=1}^k E[\epsilon_i] = 0 \quad (3.21)$$

$$\sigma_b^2 = E[b_{k+1}^2] - \mu_b^2 = E[b_{k+1}^2] = \sum_{i=1}^k E[\epsilon_i^2] = k\sigma_\epsilon^2 \quad (3.22)$$

D'après le calcul de la variance (cf. équation (3.22)), le processus de marche aléatoire est un processus non stationnaire car sa variance change linéairement dans le temps. Par conséquent les caractéristiques de la fonction d'autocorrélation ne sont pas utilisables pour définir ce processus. Cependant d'après [Mohamed, 1999], il peut être considéré stationnaire pendant un faible intervalle de temps. L'appellation de marche aléatoire est une analogie au comportement d'une personne effectuant des déplacements successifs dans des directions complètement arbitraires.

### 3.2.3 Les processus de Gauss-Markov

Les processus de Gauss-Markov sont des processus stationnaires dont leur fonction d'autocorrélation sont représentées par une exponentielle. Par exemple, pour le processus de Gauss-Markov du premier ordre la fonction d'autocorrélation est définie par :

$$R_{bb}(\tau) = \sigma_b^2 e^{-\beta_1 |\tau|} \quad (3.23)$$

où

- $\sigma_b$  est l'écart-type du processus,
- $\beta_1$  est l'inverse du temps de corrélation  $\tau_{c1}$  du processus pour lequel la fonction d'autocorrélation vaut  $R_{bb}(\tau_{c1}) = 1/e \sigma_b^2$ .

Dans ce cas, le filtre formeur est le système à boucle fermée du premier ordre que l'on peut voir dans la figure 3.3. Comme le montre la figure 3.4, la corrélation entre les mesures d'un processus de Gauss-Markov du premier ordre diminuent lorsque le décalage temporel séparant les mesures augmente et converge vers 0 pour  $\tau = \infty$ . Le processus de Gauss-Markov du 1er ordre est très souvent utilisé pour modéliser les processus stochastiques car sa description mathématique est vraiment simple. Dans la robotique mobile il est utilisé souvent pour représenter les parties stochastiques des informations inertielles [Park, 2004] [Abuhadrous, 2005] .

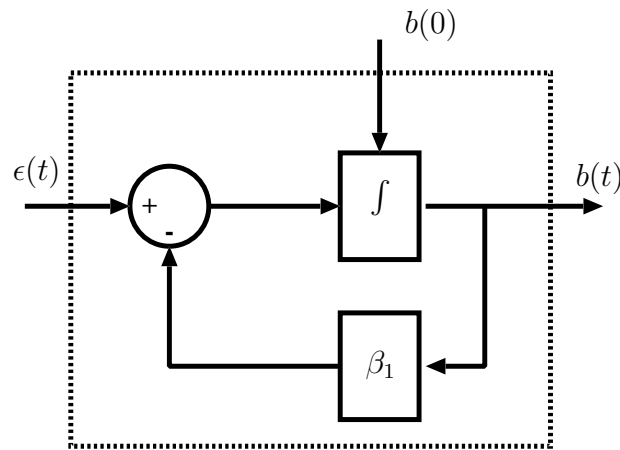


FIG. 3.3: Filtre formeur pour un processus de Gauss-Markov du 1<sup>er</sup> ordre

Des processus de Gauss-Markov d'ordre plus élevé peuvent être générés en utilisant la formulation générale de l'autocorrélation de ces processus [Gelb, 1974]. Ainsi pour un ordre  $p$ , la fonction d'autocorrélation d'un processus de Gauss-Markov est donnée par la relation :

$$R_{bb}(\tau) = \sigma_b^2 e^{\beta_p |\tau|} \sum_n^{p-1} \frac{(p-1)! (2\beta_p |\tau|)^{p-n-1} (p+n-1)}{(2p-2)! n! (p-n-1)!} \quad (3.24)$$

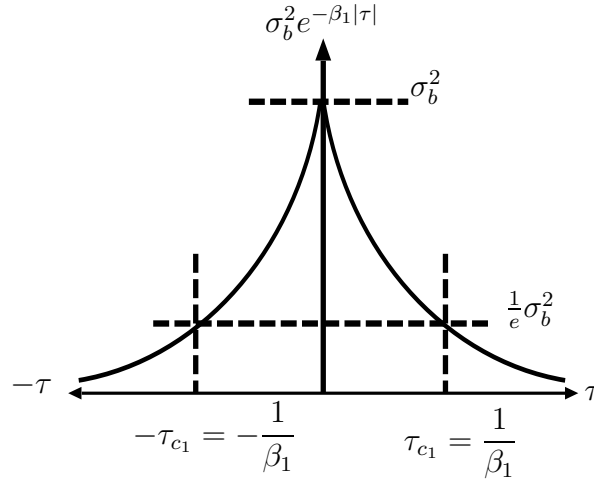


FIG. 3.4: Fonction d'autocorrélation d'un processus de Gauss-Markov du 1<sup>er</sup> ordre

où le temps de corrélation  $\tau_{c_p}$  et par conséquent  $\beta_p$  sont trouvés analytiquement en résolvant l'équation :

$$R_{bb}(\tau_{c_p}) = \frac{1}{e} \sigma_b^2 \quad (3.25)$$

Ainsi, par exemple, pour obtenir la fonction d'autocorrélation d'un processus de Gauss-Markov du second ordre il suffit de poser  $p = 2$  et  $n = \{0, 1\}$ . Alors la fonction d'autocorrélation de ce processus est décrite par :

$$R_{bb}(\tau) = \sigma_b^2 e^{-\beta_2 |\tau|} (1 + \beta_2 |\tau|) \quad (3.26)$$

où  $\beta_2$  est calculé en résolvant l'équation  $R_{bb}(\tau_{c_2}) = \frac{1}{e} \sigma_b^2$ , c'est-à-dire :

$$\begin{aligned} \sigma_b^2 e^{-\beta_2 |\tau_{c_2}|} (1 + \beta_2 |\tau_{c_2}|) &= \sigma_b^2 e^{-1} \\ -\beta_2 |\tau_{c_2}| + \ln(1 + \beta_2 |\tau_{c_2}|) &= -1 \\ \beta_2 &= \frac{2.146194}{\tau_{c_2}} \end{aligned} \quad (3.27)$$

Les formules des fonctions d'autocorrélation des processus de Gauss-Markov d'ordre 1 à 5, ainsi que leur temps de corrélation, sont présentés dans le tableau 3.1.

Il existe deux cas particuliers pour les processus de Gauss-Markov :

1. le cas où l'ordre  $p$  du processus de Gauss-Markov est égal à zéro. Ce processus ne présente alors aucune corrélation dans le temps des données et tend vers un bruit blanc.
2. le cas où l'ordre  $p$  l'ordre du processus tend vers l'infini. L'équation (3.24) tend alors vers  $e^{-\beta_p |\tau|}$  et la fonction d'autocorrélation vers une constante de valeur  $\sigma_b^2$ . Le processus de Gauss-Markov d'ordre  $p \rightarrow \infty$  peut être alors considéré comme un processus de constante aléatoire.

Ordre	Autocorrélation fonction $R_{bb_p}(\tau)$	Temps de corrélation
1	$\sigma_b^2 e^{-\beta_1}  \tau $	$\frac{1}{\beta_1}$
2	$\sigma_b^2 e^{-\beta_2}  \tau  (1 + \beta_2)  \tau $	$\frac{2.146194}{\beta_2}$
3	$\sigma_b^3 e^{-\beta_3}  \tau  (1 + \beta_3  \tau  + \frac{1}{3} \beta_3^2  \tau ^2)$	$\frac{2.904629}{\beta_3}$
4	$\sigma_b^4 e^{-\beta_4}  \tau  (1 + \beta_4  \tau  + \frac{3}{5} \beta_4^2  \tau ^2 + \frac{1}{15} \beta_4^3  \tau ^3)$	$\frac{3.512648}{\beta_4}$
5	$\sigma_b^5 e^{-\beta_5}  \tau  (1 + \beta_5)  \tau  + \frac{3}{7} \beta_5^2  \tau ^2 + \frac{2}{21} \beta_5^3  \tau ^3 + \frac{1}{105} \beta_5^4  \tau ^4)$	$\frac{4.034225}{\beta_5}$
⋮	⋮	⋮
p	$\sigma_b^2 e^{\beta_p  \tau } \sum_n^{p-1} \frac{(p-1)! (2\beta_p  \tau )^{p-n-1} (p+n-1)}{(2p-2)! n! (p-n-1)!}$	résoudre $R_{bb}(\tau_{c_p}) = \frac{1}{e} \sigma_b^2$

TAB. 3.1: Fonctions d'autocorrélation et temps de corrélation des processus de Gauss-Markov d'ordre 1 à n

Pour les processus de Gauss-Markov les équations différentielles pour le domaine continu et les équations de récurrence pour le domaine discret régissant leurs comportements dans le temps sont, pour le premier ordre :

$$\dot{b}(t) = -\beta_1 b(t) + \sqrt{2\beta_1 \sigma_b^2} w(t) \quad (3.28)$$

et

$$b_{k+1} = (1 - \beta_1 \Delta_T) b_k + \sqrt{2\beta_1 \sigma_b^2 \Delta_T} w(k) \quad (3.29)$$

pour le second ordre :

$$\ddot{b}(t) = -\beta_2 \dot{b} + -\beta_2^2 b(t) + w(t) \quad (3.30)$$

et

$$b_{1_{k+1}} = b_{1_k} + b_{2_k} \Delta_D \quad (3.31)$$

$$b_{2_{k+1}} = -\beta_2^2 \Delta_T b_{1_k} + (1 - 2\beta_2 \Delta_T) b_{2_k} + \Delta_T w_k$$

La figure 3.5 montre les fonctions d'autocorrélation des processus de Gauss-Markov d'ordre 1 à 5 et ce pour deux différents temps de corrélation  $\tau_{c_1}$ . Les paramètres  $\beta_p$  pour chaque ordre sont calculés en fonction des formules du temps de corrélation données dans le tableau 3.1.

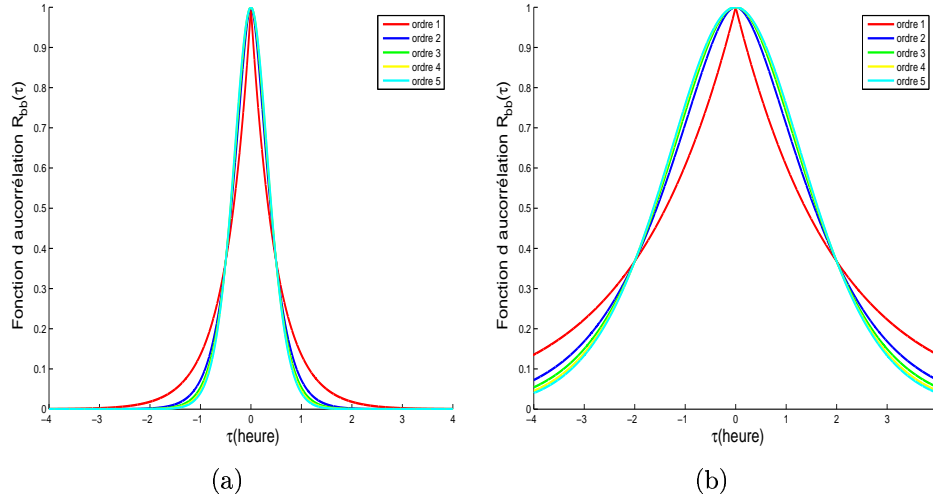


FIG. 3.5: Fonctions d'autocorrélation de processus Gauss-Markov d'ordre 1 à 5 pour un temps de corrélation de (a) 1/2 heure (b) 2 heures

Enfin, les processus aléatoires de Gauss-Markov sont souvent utilisés dans de nombreuses applications de la physique car leur modélisation est très simple, permet d'approcher de nombreux processus aléatoires avec une bonne précision et sont facilement implantables dans un système.

### 3.2.4 Précision de la modélisation des processus

D'après les sections précédentes, la modélisation d'un processus aléatoire stochastique repose en partie sur les différentes caractéristiques de la fonction d'autocorrélation empirique du processus à modéliser (à partir des données expérimentales). Cependant, dans de telles conditions, le nombre limité des mesures induit des erreurs dans cette fonction d'autocorrélation et les valeurs des paramètres des modèles dépendent donc du niveau d'incertitude de la fonction d'autocorrélation et peuvent changer selon le nombre de mesures expérimentales. Un autre problème plus complexe a été pointé dans [Brown *et al.*, 1997]. En effet, pour un processus blanc de moyenne nulle, la relation suivante est satisfaite :

$$\sigma^2 R_{bb}(\tau) \approx \frac{4}{T} \int_0^\infty \bar{R}_{bb}^2(\tau) d\tau \quad (3.32)$$

où :

- $\sigma^2 R_{bb}(\tau)$  : est la variance de la fonction d'autocorrélation du processus déterminé à partir d'un nombre fini de mesures,
- $T$  : est la durée totale de l'acquisition des mesures expérimentales,
- $\bar{R}_{bb}^2(\tau)$  : est la vraie fonction d'autocorrélation du processus.

Cette équation peut être utilisée pour estimer le nombre de mesures nécessaires pour déterminer avec une bonne précision la fonction d'autocorrélation du processus à modé-

liser. Prenons par exemple un processus de Gauss-Markov du premier ordre. En substituant la vraie fonction d'autocorrélation  $\bar{R}_{bb}^2(\tau)$  par la fonction d'autocorrélation théorique  $\sigma_b^2 e^{-\beta 1|\tau|}$  alors :

$$\sigma^2 R_{bb}(\tau) \approx \frac{4}{T} \int_0^\infty \sigma_b^2 e^{-\beta 1|\tau|} d\tau \approx \frac{2\sigma_b^4}{T\beta 1} \approx \frac{2\sigma_b^4 \tau_{c1}}{T} \quad (3.33)$$

où  $\tau_{c1}$  est le temps de corrélation d'un processus de Gauss-Markov du premier ordre.

D'après l'équation (3.33), la précision  $acc_R$  (ou le niveau d'incertitude) de la fonction d'autocorrélation  $R_{bb}(\tau)$  peut être alors décrite par le rapport entre l'écart-type de la fonction d'autocorrélation théorique  $\bar{R}_{bb}(\tau)$  et la variance du processus à modéliser :

$$acc_R \approx \sqrt{\frac{\sigma^2 R_{bb}(\tau)}{\sigma_b^4}} \approx \sqrt{\frac{2\tau_{c1}}{T}} \quad (3.34)$$

Par conséquent si le niveau d'incertitude envisagé est de 5%, la durée totale  $T$  d'acquisition des mesures expérimentales doit être approximativement égale à  $2\tau_{c1}/acc_R^2 = 400\tau_{c1}$ , c'est-à-dire 400 fois le temps de corrélation du processus aléatoire. En prenant un cas où le processus a un temps de corrélation d'une heure, il est alors nécessaire d'acquérir des données pendant 400 heures. Du point de vue expérimental, une caractérisation précise d'un processus stochastique demande donc de grandes ressources en terme de temps et en terme de stockage des informations et est quasi-irréalisable dans la pratique.

D'un autre point de vue, l'équation (3.34) peut être utilisée pour estimer la précision de la fonction d'autocorrélation obtenue à partir d'un nombre fini de mesures expérimentales (de durée totale d'acquisition  $T$ ). Dans ce cas, les paramètres  $\alpha$  et/ou  $\beta$  du modèle et donc le processus modélisé sont déduits de  $R_{bb}(\tau)$  avec une précision définie par  $acc_R$ .

Finalement, comme le montre Gelb dans [Gelb, 1974], à cause des limitations expérimentales, la fonction autocorrélation ne peut être déterminée correctement et la plupart des modélisations de processus aléatoire peuvent être satisfaites par le modèle de Gauss-Markov du premier ordre.

### 3.3 Modélisation autorégressive

Lorsque le niveau d'incertitude de la fonction d'autocorrélation est grand, la modélisation du processus par un processus de Gauss-Markov est peu recommandée. Pour résoudre ce problème la modélisation autorégressive du processus peut être alors utilisée. Introduite dans les années 1950, cette méthode est utilisée dans de nombreux domaines comme la reconnaissance de la parole, la musique etc... N'étant pas limitée à un ou deux paramètres pour caractériser le processus, elle permet une modélisation beaucoup plus flexible et fidèle du processus. Cette modélisation est un cas particulier de la modélisation ARMA<sup>1</sup> pour lequel il n'est pas nécessaire de résoudre un jeu d'équations non-linéaires

---

<sup>1</sup>Autoregressive Moving Average

pour déterminer les paramètres  $\alpha_n$  du modèle. De plus en imposant des contraintes sur les paramètres  $\alpha_n$ , il est possible d'obtenir n'importe lequel des processus précédemment présentés.

Dans cette section, sont présentés les aspects fondamentaux de la modélisation AR ainsi que trois différentes méthodes de régression linéaire nécessaires à l'estimation des paramètres  $\alpha_n$  du modèle AR : la méthode de Yule-Walker, la méthode de la covariance et la méthode de Burg.

### 3.3.1 Processus autorégressif AR

Considérons dans un premier temps un processus autorégressif à moyenne mobile. La modélisation ARMA est basée sur le principe mathématique suivant : chaque mesure  $x_k$  à l'instant  $k$  d'une série de mesures est dépendante : 1) d'une somme pondérée du passé de ces mesures (partie AR) et 2) d'une somme pondérée du présent et du passé d'un processus aléatoire décorréolé des mesures (partie MA). Le processus ARMA peut alors être décrit par la fonction de transfert en  $z$  suivante :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{n=0}^q \beta_n z^{-n}}{1 + \sum_{n=1}^p \alpha_n z^{-n}} \quad (3.35)$$

où

- $X(z)$  est la transformée en  $z$  de l'entrée  $x_k$ ,
- $Y(z)$  est la transformée en  $z$  de la sortie  $y_k$ ,
- $p$  est l'ordre du processus AR,
- $\alpha_1 \dots \alpha_p$  sont les paramètres (pondérations) du processus AR
- $q$  est l'ordre du processus MA,
- $\beta_0 \dots \beta_q$  sont les paramètres (pondérations) du processus MA.

Le processus AR est un cas particulier du processus ARMA où l'ordre de la partie MA est nul. Ce processus est décrit par la fonction de transfert en  $z$  :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\beta_0}{1 + \sum_{n=1}^p \alpha_n z^{-n}} \quad (3.36)$$

Ainsi, l'adjectif autorégressif vient du fait que l'échantillon du présent d'un processus autorégressif est seulement prédit par les échantillons de son propre passé. Dans le domaine discret, l'équation de récurrence régissant ce processus peut être obtenue par la transformée inverse en  $z$  de l'équation (3.36) et est donnée par le relation :

$$y_k = - \sum_{n=1}^p \alpha_n y_{k-n} + \beta_0 x_k \quad (3.37)$$

c'est-à-dire :

$$y_k = -\alpha_1 y_{k-1} - \alpha_2 y_{k-2} \dots - \alpha_p y_{k-p} + \beta_0 x_k \quad (3.38)$$

Comme pour les processus générés à partir de filtres formeurs (marche aléatoire, processus de Gauss-Markov etc...), l'entrée  $x_k$  du modèle autorégressif est une séquence aléatoire de moyenne nulle et non corrélée dans le temps, c'est-à-dire un bruit blanc. Le principal problème de cette modélisation est d'estimer optimalement les valeurs des paramètres  $\alpha_n$  du processus AR. Il est en général résolu en minimisant l'erreur de prédiction  $e_k$  entre le signal original  $b_k$  et le signal estimé  $\hat{b}_k$  déterminé par le modèle AR de la forme suivante :

$$\hat{b}_k = -\sum_{n=1}^p \alpha_n y_{k-n} \quad (3.39)$$

La fonction de coût utilisée pour résoudre le problème de minimisation est alors la somme des erreurs quadratiques  $\mathcal{E}_k$  de  $e_k$  :

$$\begin{aligned} \mathcal{E}_k &= \sum_k e_k^2 \\ &= \sum_k \left[ y_k - \hat{y}_k \right]^2 \\ &= \sum_k \left[ y_k + \sum_{n=1}^p \alpha_n y_{k-n} \right]^2 \\ &= \sum_k \left[ -\sum_{n=1}^p \alpha_n y_{k-n} + \beta_0 \epsilon_k + \sum_{n=1}^p \alpha_n y_{k-n} \right]^2 \\ &= \sum_k \beta_0^2 \epsilon_k^2 \end{aligned} \quad (3.40)$$

Dans ce cas, en supposant que  $\epsilon_k$  est un bruit blanc de variance unitaire, la moyenne des erreurs quadratiques (ou bien la variance si la moyenne de  $e_k$  est nulle)  $\sigma_e^2$  est alors égale à  $\beta_0^2$ . Où  $\beta_0^2$  représente la variance ou encore la puissance estimée du bruit blanc en entrée du modèle AR.

### 3.3.2 Détermination des paramètres des processus AR

Ils existent de nombreuses méthodes permettant de déterminer les paramètres  $\alpha_n$  du modèle AR. Dans cette section sont seulement présentées les trois plus connues de ces méthodes à savoir la méthode de Yule-Walker, la méthode de la covariance et la méthode de Burg.

#### 3.3.2.1 La méthode de Yule-Walker

La méthode de Yule-Walker, aussi connue sous le nom de la méthode d'autocorrélation, se déroule de la manière suivante :



1. la séquence d'autocorrélation  $R_{bb}(m)$  du processus à modéliser est calculée à partir de données expérimentales,
2. les paramètres du modèle AR sont déterminés en résolvant un jeu d'équations linéaires qui permet de minimiser au sens des moindres carrés  $\mathcal{E}_k$  par rapport aux différents paramètres  $\alpha_n$

La condition de minimisation du jeu d'équations est définie par la relation suivante :

$$\frac{\partial \mathcal{E}_k}{\partial \alpha_n} = 0, \quad 1 \leq n \leq p. \quad (3.41)$$

En prenant comme point de départ la troisième ligne de l'équation (3.40), nous obtenons :

$$\mathcal{E}_k = \sum_k \left[ b_k + \sum_{n=1}^p \alpha_n b_{k-n} \right]^2 = \sum_k \left[ b_k^2 + 2b_k \sum_{n=1}^p \alpha_n b_{k-n} + \sum_{n=1}^p \alpha_n b_{k-n} \sum_{i=1}^p \alpha_i b_{k-i} \right] \quad (3.42)$$

En appliquant (3.41) à (3.42) nous obtenons :

$$\sum_k \left[ 2b_k b_{k-n} + 2b_{k-n} b_{k-i} \sum_{n=1}^p \alpha_n \right] = 0 \quad (3.43)$$

soit :

$$\sum_{n=1}^p \alpha_n \sum_k b_{k-n} b_{k-i} = - \sum_k b_k b_{k-n} \quad (3.44)$$

Si la somme  $\sum_k$  est effectuée à partir d'un nombre infini de mesures, alors l'équation (3.44) devient :

$$\sum_{n=1}^p \alpha_n \bar{R}_{bb}(n-i) = \bar{R}_{bb}(n), \quad 1 \leq i \leq p, \quad (3.45)$$

où  $\bar{R}_{bb}(n)$  est l'ensemble des séquences d'autocorrélation et  $\bar{R}_{bb}(n-i)$  les coefficients de la matrice d'autocorrélation de l'ensemble des mesures. Encore une fois, dans la pratique il est impossible d'avoir un nombre infini de mesures pour caractériser le processus et  $\sum_k e_k^2$

ne peut être connue au delà  $k > N$  avec  $N$  le nombre fini de mesures. Pour résoudre ce problème, dans la méthode de Yule-Walker toutes les mesures pour  $k > N$  sont mises à zéro en appliquant un pré-fenêtrage des mesures. Ces séquences d'autocorrélation sont seulement calculées pour chaque mesure appartenant à la fenêtre d'observation. Ainsi dans le jeu d'équations linéaires de la relation (3.45),  $\bar{R}_{bb}(n)$  est remplacée par  $R_{bb}(n)$  et les paramètres  $\alpha_n$  du modèle AR peuvent être déterminés en résolvant l'expression matricielle :

$$R_{bb} \cdot \alpha = -\mathbf{r}_{bb} \Rightarrow \alpha = -R_{bb}^{-1} \cdot \mathbf{r}_{bb} \quad (3.46)$$

où :

$$\alpha = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{pmatrix}, \quad \mathbf{r}_{bb} = \begin{pmatrix} R_{bb}(1) \\ R_{bb}(2) \\ R_{bb}(3) \\ \vdots \\ R_{bb}(p) \end{pmatrix} \quad (3.47)$$

et où :

$$R_{bb} = \begin{pmatrix} R_{bb}(0) & R_{bb}(1) & R_{bb}(2) & \dots & R_{bb}(p-1) \\ R_{bb}(1) & R_{bb}(0) & R_{bb}(1) & \dots & R_{bb}(p-2) \\ R_{bb}(2) & R_{bb}(1) & R_{bb}(0) & \dots & R_{bb}(p-3) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ R_{bb}(p-1) & R_{bb}(p-2) & R_{bb}(p-3) & \dots & R_{bb}(0) \end{pmatrix} \quad (3.48)$$

Finalement la moyenne des erreurs quadratiques  $\sigma_e^2$  est déterminée en résolvant les équations suivantes :

$$\begin{pmatrix} R_{bb}(0) & R_{bb}(1) & R_{bb}(2) & \dots & R_{bb}(p) \\ R_{bb}(1) & R_{bb}(0) & R_{bb}(1) & \dots & R_{bb}(p-1) \\ R_{bb}(2) & R_{bb}(1) & R_{bb}(0) & \dots & R_{bb}(p-2) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ R_{bb}(p-1) & R_{bb}(p-2) & R_{bb}(p-3) & \dots & R_{bb}(0) \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{pmatrix} = \begin{pmatrix} \sigma_e^2 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (3.49)$$

L'équation (3.49) est plus connue sous le nom d'équation de Yule-Walker. On peut noter que la matrice  $R_{bb}$  est symétrique et a une structure de Toeplitz (matrice dont tous les éléments de chaque diagonale sont égaux). Ainsi, au lieu de calculer directement  $R_{bb}^{-1}$  pour résoudre les équations de Yule-Walker, il est plus efficace d'appliquer l'algorithme de Levinson-Durbin. Cet algorithme est une technique itérative qui permet de calculer récursivement les coefficients de prédiction  $\alpha_n$  (paramètres du modèle AR) à partir de leurs coefficients précédents  $\alpha_{n-1}$ . Cette procédure peut être décrite par les équations suivantes :

$$\begin{aligned} \mathcal{E}_{k_0} &= R_{bb}(0) \\ \gamma_n &= \frac{R_{bb}(n) + \sum_{i=1}^{n-1} \alpha_{i,n-1} R_{bb}(n-i)}{\epsilon_{k-1}} \quad 1 \leq n \leq p \\ \alpha_{n,n} &= \gamma_n \end{aligned} \quad (3.50)$$

$$\alpha_{1,n} \alpha_{i,n-1} + \gamma_n \alpha_{n-i,n-1} \quad 1 \leq i \leq n-1$$

$$\epsilon_{k_n} = (1 - \gamma_n^2) \epsilon_{k_{n-1}}$$

Les équations sont résolues récursivement pour  $n = 1, 2 \dots p$  et la solution finale pour les paramètres AR est fournie par la relation :

$$\alpha_{i,p} \quad 1 \leq i \leq p \quad (3.51)$$

Ainsi, les valeurs des paramètres du modèle AR sont déterminées par minimisation de l'erreur de prédiction au sens des moindres carrés. Les quantités intermédiaires  $\gamma_n$  dans l'équation (3.50) sont connues sous le nom de coefficient de réflexions et doivent être inférieurs à 1 pour garantir la stabilité de tous les pôles du filtre AR. D'après [Brown *et al.*, 1997]

cette méthode présente de bonnes performances lorsque les mesures expérimentales sont très nombreuses. En revanche dans le cas où l'acquisition des données expérimentales est de courte durée, les performances se dégradent à cause du processus de pré-fenêtrage. De plus, d'après [Erkelens *et al.*, 1997], cette méthode peut introduire un biais dans l'estimation des paramètres AR et peut entraîner une instabilité du modèle.

### 3.3.2.2 La méthode de la Covariance

La méthode de la Covariance est similaire à la méthode Yule-Walker dans le sens où elle minimise l'erreur de prédiction au sens des moindres carrés. Cependant, contrairement à la méthode de Yule-Walker, elle ne réalise aucun pré-fenêtrage des mesures et utilise donc toutes les mesures disponibles pour minimiser l'erreur de prédiction. En prenant en compte cette hypothèse dans l'équation (3.46), le jeu d'équations linéaires à résoudre est alors :

$$\sum_{n=1}^p \alpha_n \sum_k b_{k-n} b_{k-i} = - \sum_k b_k b_{k-n} \quad (3.52)$$

$$\sum_{n=1}^p \alpha_n C_{bb}(n-i) = C_{bb}(0, n), \quad 1 \leq i \leq p$$

où  $C_{bb}(0, n)$  sont les variances ou les covariances des mesures et  $C_{bb}(n-i)$  sont les coefficients de la matrice de covariance de l'ensemble des mesures. Dans ce cas, et par analogie avec les équations de Yule-Walker, nous pouvons obtenir le jeu d'équations linéaires suivant pour déterminer les paramètres du modèle AR :

$$\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_4 \end{pmatrix} = - \begin{pmatrix} C_{bb}(1,1) & R_{bb}(1,2) & R_{bb}(1,3) & \dots & R_{bb}(1,p) \\ C_{bb}(2,1) & R_{bb}(2,2) & R_{bb}(2,3) & \dots & R_{bb}(2,p) \\ C_{bb}(3,1) & R_{bb}(3,2) & R_{bb}(3,3) & \dots & R_{bb}(3,p) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ C_{bb}(p,1) & R_{bb}(p,2) & R_{bb}(p,3) & \dots & R_{bb}(p,p) \end{pmatrix}^{-1} \begin{pmatrix} C_{bb}(0,1) \\ C_{bb}(0,2) \\ C_{bb}(0,3) \\ \vdots \\ C_{bb}(0,p) \end{pmatrix} \quad (3.53)$$

D'après [Makhoul, 1975], bien que la méthode de la covariance n'utilise pas de pré-fenêtrage des mesures expérimentales, les paramètres du modèle AR sont déterminés avec une meilleure précision qu'avec la méthode de Yule-Walker. Comme le montre l'équation (3.53), cette méthode utilise la séquence de covariance à la place de la séquence d'autocorrélation. Ici  $C_{bb}$  est symétrique mais n'a pas une structure de Toeplitz. Il est donc impossible d'utiliser l'algorithme de Levinson-Dublin pour résoudre le jeu d'équations linéaires. Dans ce cas, pour effectuer efficacement l'inversion de  $C_{bb}$ , la factorisation de Cholesky de la matrice  $C_{bb}$  est très souvent utilisée.

En général, la méthode de la Covariance permet une estimation plus précise des paramètres du modèle AR, surtout lorsque la durée d'acquisition des mesures est courte. Cependant, le fait de ne pas utiliser l'algorithme de Levinson-Durbin pour résoudre le jeu d'équations peut entraîner une instabilité des modèles AR fournis par la méthode de la Covariance [Hoon *et al.*, 1996].

### 3.3.2.3 La méthode de Burg

La méthode de Burg [Burg, 1975] a été introduite pour résoudre les inconvénients des deux précédentes méthodes. Elle fournit à la fois des modèles stables et des paramètres AR de grande précision et ce pour un nombre restreint de données expérimentales. Cette méthode essaye avant tout d'utiliser au maximum les mesures expérimentales en introduisant une "backward" équation  $e_f(k)$  et une "forward" équation  $e_b(k)$  de la prédiction des termes d'erreur. Dans ce cas, et par analogie avec la méthode Yule-Walker, cette méthode minimise non plus  $\mathcal{E}_k$ , mais  $\mathcal{E}_{k_{Burg}}$  définie par la somme de ces deux équations :

$$\mathcal{E}_{k_{Burg}} = \sum_{k=1}^N [e_f^2(k) + e_b^2(k)] = \min \quad (3.54)$$

où  $e_f(k)$  et  $e_b(k)$  sont :

$$e_f(k) = b(k) + \alpha_1 b(k-1) + \alpha_2 b(k-2) + \dots + \alpha_{p-1} b(k-p+1) \quad (3.55)$$

$$e_b(k) = b(k-p) + \alpha_1 b(k-p+1) + \alpha_2 b(k-p+2) + \dots + \alpha_{p-1} b(k-1)$$

En substituant  $\mathcal{E}_k$  par  $\mathcal{E}_{k_{Burg}}$  dans la cinquième équation de la relation (3.50), les erreurs prédites par ces deux équations peuvent être décrites par les équations suivantes :

$$e_{f_n}(k) = e_{f_{n-1}}(k) + \gamma_n e_{b_{n-1}}(k-1) \quad (3.56)$$

$$e_{b_n}(k) = e_{b_{n-1}}(k-1) + \gamma_n e_{f_{n-1}}(k)$$

Cette forme récursive de la prédiction des erreurs est réalisée suivant une structure en treillis (appelée lattice structure en anglais) d'un filtre récursif (cf figure 3.6).

Comme pour la méthode de Yule-Walker, la précision des paramètres du modèle AR  $\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_p$  et  $\sigma_e^2$  devrait dépendre du niveau d'incertitude de la séquence d'auto-corrélation  $R_{bb}(\tau)$ . Cependant pour résoudre ce problème, Burg propose de déterminer en premier les coefficient de réflexion  $\gamma_n$  sans utiliser la dernière relation de l'équation (3.50) mais en intégrant l'équation (3.56) dans l'équation (3.54). Ainsi en fixant la condition de linéarité du jeu d'équations par rapport aux coefficients de réflexion  $\gamma_n$  (au lieu des paramètres  $\alpha_n$  pour la méthode Yule-Walker) est :

$$\frac{\partial \mathcal{E}_{k_{Burg}}}{\partial \gamma_n} = 0 \quad (3.57)$$

Les coefficients de réflexion sont décrits par :

$$\gamma_n = \frac{-2 \sum_{k=n}^N [e_{f_n}(k) + e_{b_n}(k-1)]}{\sum_{k=n}^N e_{f_{n-1}}^2(k) + \sum_{k=n}^N e_{b_{n-1}}^2(k-1)} \quad (3.58)$$

Cette relation montre clairement que la valeur des coefficients de réflexion est toujours inférieure à 1 et donc que le modèle AR ainsi déduit est toujours stable. Finalement, les paramètres  $\alpha_n$  du modèle AR sont obtenus en satisfaisant la contrainte décrite par la quatrième équation de la relation (3.50).

Enfin, d'après [Rezeck *et al.*, 1997], l'utilisation des équations "backward" et "forward" pour la minimisation des erreurs de prédiction entraîne une meilleure estimation des paramètres  $\alpha_n$  du modèle AR.

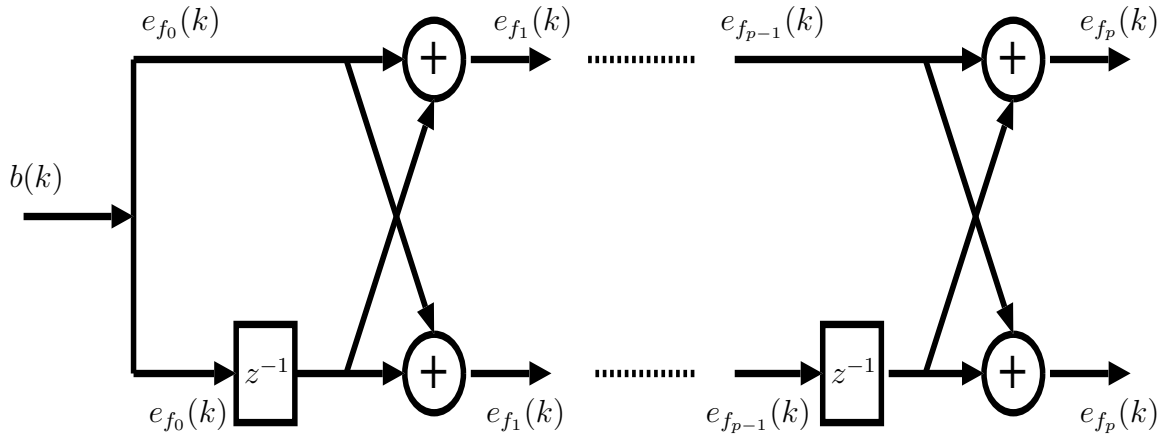


FIG. 3.6: Filtre récursif en forme de treillis

## 3.4 Filtrage et informations stochastiques

### 3.4.1 Modification du système d'état

Comme on l'a vu au début de ce chapitre, lorsque les informations fournies par les différents capteurs ne sont pas blanches, il est nécessaire de caractériser leurs parties systématiques et stochastiques et de les prendre en compte dans la méthode de filtrage. Dans ce cas, ces parties deviennent de nouvelles inconnues (ou variables du système), et doivent être intégrées au vecteur d'état :

$$\underline{X} = \begin{pmatrix} \underline{X}_l \\ \underline{X}_s \\ \underline{X}_b \end{pmatrix} \quad (3.59)$$

où dans notre situation :

- $\underline{X}_l$  est le vecteur décrivant l'attitude du robot,
- $\underline{X}_s$  est le vecteur contenant les différentes variables représentant les parties systématiques des diverses informations non blanches,
- $\underline{X}_b$  est le vecteur contenant les différentes variables permettant de définir le comportement de la partie stochastique des diverses informations non blanches. Ces

variables sont bien sûr définies en fonction de la méthode de modélisation utilisée (marche aléatoire, Gauss-Markov, AR etc...).

Nous rappelons que pour tout filtre bayésien, que ce soit le filtre de Kalman ou le filtre particulière, celui-ci se décompose en deux étapes : l'étape de prédiction et l'étape de correction.

**Etape de prédiction :** lorsqu'il existe des informations stochastiques en entrée d'un filtre bayésien, cette étape se décompose alors en trois parties, 1) la prédiction de l'attitude du véhicule, 2) la prédiction des parties systématiques des informations non blanches et 3) la prédiction des parties stochastiques des informations non blanches :

1. La prédiction de l'attitude du véhicule est réalisée en prenant en compte son modèle cinématique et les informations proprioceptives. Si ces informations sont blanches, elles sont fournies directement par les capteurs proprioceptifs, sinon ces informations sont fournies par l'estimation de leur partie systématique,
2. La prédiction des parties systématiques est réalisée en prenant en compte leurs variations possibles d'un instant à l'autre, ces variations sont souvent déduites selon les contraintes de l'application.
3. La prédiction des parties stochastiques est, quant à elle, complètement définie par les différents paramètres déterminés lors de leur modélisation.

**Etape de correction :** lors de cette étape, la partie systématique et la partie stochastique de toutes les informations non-blanches doivent être estimées. Pour pouvoir estimer ces deux inconnues, il est nécessaire d'avoir au moins deux équations, où plutôt deux observations. Dans ce cas, il est nécessaire d'avoir au moins une deuxième source d'information fournissant une observation soit de la partie systématique soit de la partie stochastique des informations non blanches.

### 3.4.2 Exemple

Prenons l'exemple d'un système de localisation 2D hybride, où : 1) la localisation à l'estime est fonction d'informations proprioceptives provenant d'un odomètre et d'un gyromètre, 2) la localisation absolue est fonction d'informations fournies par un GPS et 3) la fusion de données est effectuée par un filtre de Kalman. Si la totalité des informations fournies par les différents capteurs sont considérées blanches, le système sera régi par le système d'état suivant :

$$\begin{cases} \underline{X}_k &= f(\underline{X}_{k-1}, \Delta_D, \omega_{gyro}) \\ \underline{Y}_{gps} &= \mathbf{H}\underline{X}_k + v_{gps} \end{cases} \quad (3.60)$$

où :

- $\underline{X}_k = (x_k, y_k, \theta_k)^T$  est le vecteur d'état à l'instant  $k$ ,
- $\Delta_D$  est l'information odométrique,
- $\omega_{gyro}$  est l'information gyrométrique,

- $\underline{Y}_{gps} = (x_{gps}, y_{gps})^T$  est la position fournie par le GPS,
- $\mathbf{H}$  est la matrice d'observation fonction du vecteur  $\underline{X}_k$  et des informations GPS,
- et  $v_{gps}$  sont les incertitudes sur la position GPS.

Cependant, comme pour tout capteur inertiel, l'information fournie par le gyromètre n'est pas blanche (cf. equation 1.13), mais se décompose en deux parties :

1. une partie blanche donnée par  $K_\omega \omega + \epsilon_\omega$ , où :
  - $K_\omega$  est le facteur d'échelle de mesure,
  - $\omega$  la mesure proprement dite du gyromètre,
  - et  $\epsilon_\omega$  est le bruit de mesure.
2. une partie stochastique  $b_\omega$  (aussi appelé biais du gyromètre) qui est une variable aléatoire non-observable et dépendante de la température du capteur.

Ce biais, ne pouvant être observé, est inconnu et doit être estimé. Pour ceci, son comportement temporel est caractérisé et pris en compte dans le système de localisation. Ainsi en considérant que, par exemple,  $b_\omega$  peut être modélisé par un processus autorégressif d'ordre 2, c'est-à-dire un processus aléatoire régi par le système dynamique suivant :

$$b_{\omega_k} = -\alpha_1 b_{\omega_{k-1}} - \alpha_2 b_{\omega_{k-2}} + \beta_0 \epsilon_{b_k} \quad (3.61)$$

où :

- $\alpha_1, \alpha_2, \beta_0$  sont les paramètres de la modélisation autorégressive,
- $\epsilon_b$  une variable aléatoire pure.

Le vecteur d'état du système de localisation doit être augmenté de la manière suivante :

$$\underline{X}_k^a = (x_k, y_k, \theta_k, \dot{\theta}_k, b_{\omega_k}, b_{\omega_{k-1}})^T \quad (3.62)$$

Le système d'état est modifié de la manière suivante , où :

- l'étape de prédiction est alors :

$$\underline{X}_k^a = f^a(\underline{X}_{k-1}^a, \Delta_D, \alpha_1, \alpha_2, \beta_0) \quad (3.63)$$

c'est-à-dire :

$$\begin{aligned} x_k &= x_{k-1} + \Delta_D \cos(\theta_k) \\ y_k &= y_{k-1} + \Delta_D \sin(\theta_k) \\ \theta_k &= \theta_{k-1} + \Delta_T \dot{\theta}_k \\ \dot{\theta}_k &= \dot{\theta}_{k-1} + \epsilon_{\dot{\theta}} \\ b_{\omega_k} &= -\alpha_1 b_{\omega_{k-1}} - \alpha_2 b_{\omega_{k-2}} + \beta_0 \epsilon_b \\ b_{\omega_{k-1}} &= b_{\omega_{k-1}} \end{aligned} \quad (3.64)$$

où

- $\Delta_D$  est la distance parcourue par l'odomètre,
- $\Delta_T$  le temps écoulé entre les instants  $k$  et  $k-1$ ,
- $\epsilon_{\dot{\theta}}$  est un bruit blanc gaussien définissant les variations possibles de la vitesse de rotation du véhicule entre les instants  $k$  et  $k-1$ .

– l'étape de correction est maintenant :

$$\begin{cases} \omega_{gyro} &= \dot{\theta} + b_\omega \\ \omega_s &= \dot{\theta} + \epsilon_s \\ \underline{Y}_{gps} &= \mathbf{H}^a \underline{X}_k^a + v_{gps} \end{cases} \quad (3.65)$$

où :

- $\omega_s$  est une information blanche permettant d'observer la partie systématique de l'information gyrométrique et donc d'estimer la vitesse angulaire du véhicule. Information, qui par exemple, peut être fournie par le GPS,
- $\epsilon_s$  incertitude sur  $\omega_s$ ,
- $H^a$  est la matrice d'observation fonction du vecteur d'état augmenté  $\underline{X}_k^a$  et des informations GPS.

Ainsi, d'après la description de ce nouveau système d'état, les diverses modifications présentées dans la partie 3.4.1 sont retrouvées et nécessaires à la prise en compte d'informations non-blanches dans un système d'état, en effet :

1. le vecteur d'état a été augmenté, prenant ainsi en compte les parties systématique et stochastique de l'information gyrométrique,
2. dans l'étape de prédiction, le calcul de la pose du véhicule est réalisée en fonction de son modèle cinématique, de l'information odométrique et de la partie systématique  $\dot{\theta}$  de l'information gyrométrique. La prédiction de la partie stochastique de l'information gyrométrique est réalisée en fonction des paramètres autorégressifs et la prédiction de la partie systématique en fonction de ses variations possibles définies par le bruit blanc gaussien  $\epsilon_{\dot{\theta}}$ .
3. enfin l'étape de correction, en plus de corriger la pose du véhicule en fonction des informations GPS, permet d'estimer les parties stochastique et systématique de l'information gyrométrique. Dans ce cas, l'information gyrométrique elle-même et une seconde source d'information (observant ici la partie systématique de l'information gyrométrique) sont utilisées pour effectuer cette estimation. Ainsi l'information provenant du gyromètre permet d'estimer à la fois ses parties stochastique et systématique à condition que sa partie systématique soit contrainte par les informations provenant du second capteur.

## 3.5 Conclusion

Dans ce chapitre diverses méthodes de modélisation des processus stochastiques ont été présentées. Les modèles de processus, tels que la constante aléatoire, la marche aléatoire ou encore les processus de Gauss-Markov permettent une description simple de ces processus. Dans ce cas, leur détermination repose entièrement sur les caractéristiques de



la fonction d'autocorrélation du véritable processus. Cependant, en pratique le nombre limité de mesures induit des erreurs dans cette fonction, et les processus peuvent être mal modélisés.

Dans ce cas, la modélisation autorégressive des processus stochastiques est souvent préférée. Elle permet de décrire leur comportement en fonction de leur propre passé et d'un nombre illimité de paramètres. Ces différents paramètres sont déterminés par des méthodes de régression linéaire telles que les méthodes de Yule-Walker, la méthode de la Covariance ou encore la méthode de Burg. Cette dernière méthode semble la plus appropriée car elle permet de déterminer ces paramètres avec une bonne précision et assure une stabilité du modèle dans toutes les conditions.

Enfin, si les informations d'un système de fusion (filtre de Kalman ou filtre particulaire) comportent une partie stochastique, il est nécessaire de la caractériser puis de la prendre en compte dans les diverses étapes du filtrage. Dans ce cas, les parties systématiques et stochastiques de ces informations deviennent de nouvelles inconnues du système de fusion et sont introduites dans le vecteur d'état. Elles sont alors estimées au même titre que les autres variables du système et nécessitent de nouvelles informations pour leur estimation.

Deuxième partie

Applications



# Chapitre 4

## Choix et caractérisation des capteurs

Dans les deux premières sections de ce chapitre, va être présenté le choix des différents capteurs proprioceptifs et extéroceptifs nécessaires au bon fonctionnement de notre système de localisation hybride. Le choix de ces capteurs s'effectuera en fonction des critères suivants :

- faible coût d'intégration,
- capteurs proprioceptifs entraînant une faible divergence de la méthode de positionnement relatif,
- capteurs extéroceptifs les plus adéquats pour le positionnement absolu en environnement routier, c'est-à-dire sans aménagement de celui-ci.

Pour chaque capteur ainsi choisi, trois caractéristiques seront définies :

1. les données fournies par le capteur,
2. les incertitudes des données,
3. la datation des données (date d'acquisition de celles-ci par le système de localisation).

Enfin la troisième section de ce chapitre montrera l'influence de la latence de certains capteurs sur le processus de fusion de données. Une algorithmie sous-jacente au problème de filtrage en sera déduite, elle permettra de recalculer les informations en fonction de leur date d'acquisition pour pouvoir effectuer la fusion de données séquentiellement sans avoir à imposer une synchronisation de celles-ci.

### 4.1 Capteurs proprioceptifs

#### 4.1.1 Choix des capteurs

Avant de choisir les capteurs proprioceptifs, il est nécessaire de déterminer quelle technique de localisation relative est la plus appropriée pour notre approche. D'après l'état de l'art sur les méthodes positionnement (cf chapitre 1), de nombreuses techniques existent :

la navigation inertielle (1), la navigation par odométrie (2) et la navigation hybride couplant différents types de capteurs (3).

1. **La navigation inertielle** : de par son coût élevé elle ne semble pas très adéquate pour notre système qui ce veut avant tout un système de positionnement avec un faible coût d'intégration.
2. **La navigation par odométrie** : grâce au système ABS équipant la plupart des véhicules de dernières générations, la navigation par odométrie utilisant le modèle cinématique de type char semble facile à mettre en œuvre. Néanmoins, notre véhicule expérimental étant relativement ancien nous n'avons pas à disposition de système ABS et donc d'informations odométriques sur les deux roues. Cette technique de navigation n'a donc pas été retenue dans notre cas.
3. **La navigation hybride** :
  - *Odomètre + gyromètre* : une solution possible de navigation hybride est de combiner les informations provenant d'un odomètre et d'un gyromètre et d'utiliser le modèle cinématique de type char (cf section 1.1.1.1), la vitesse angulaire de lacet n'est plus déduite des deux odomètres mais directement fournie par le gyromètre. Un compromis prix et précision pour les deux capteurs peut être ici facilement réalisé, cette technique semble viable pour notre système.
  - *Odomètre + angle au volant* : une deuxième solution consiste à coupler les informations provenant d'un odomètre et d'un capteur d'angle du volant. Si l'angle de braquage des roues peut être calculé en fonction de l'angle de braquage du volant c'est-à-dire si  $\delta_r = f(\delta_v)$  est connue, alors le positionnement relatif du véhicule peut être estimé à l'aide d'un modèle cinématique de type tricycle (cf section 1.1.1.2). Pour réaliser un capteur d'angle au volant, un simple potentiomètre fixé sur l'axe du volant peut être utilisé, ce capteur étant peu coûteux cette technique semble être elle aussi utilisable.
  - *Radar à effet Doppler* : le radar à effet Doppler peut être utilisé pour remplacer l'odométrie dans les systèmes de positionnement relatif. Cependant comme pour la navigation inertielle, le prix élevé du radar à effet Doppler n'autorise pas son utilisation dans notre système.

Donc deux types de techniques de localisation relative sont a priori utilisables dans notre système de localisation, la navigation par odométrie couplée avec un gyromètre (modèle cinématique de type char) et la navigation par odométrie couplée avec le capteur d'angle au volant (modèle cinématique de type tricycle). Pour faire le choix entre l'une de ces deux méthodes, nous allons réaliser la localisation de notre véhicule sur un parcours de plusieurs kilomètres avec chacune de ces deux méthodes et les comparer aux positions fournies par un DGPS. Nous rappelons brièvement les équations de chacun des modèles cinématiques associés à ces deux techniques.

#### 4.1.1.1 Equations du modèle de type char

D'après les équations (1.3) et en prenant compte le fait la vitesse angulaire de lacet  $\dot{\theta}$  est directement fournie par le gyromètre, le déplacement relatif du véhicule est donné par les équations suivantes :

$$\begin{aligned}x_k &= x_{k-1} + \Delta_{D_k} \cos(\theta_k) \\y_k &= y_{k-1} + \Delta_{D_k} \sin(\theta_k) \\ \theta_k &= \theta_{k-1} + \Delta_T w_{gyro}\end{aligned}\tag{4.1}$$

avec :

- $\Delta_{D_k} = (Top\_odo_k - Top\_odo_{k-1}) \times MPTO$  la distance parcourue entre les instants  $k - 1$  et  $k$ ,
- $\Delta_T$  la durée séparant ces deux instants.

#### 4.1.1.2 Equations du modèle de type tricycle

D'après les équations (1.6) et en considérant que l'orientation des roues est donnée par la fonction  $\delta_r = f(\delta_v)$  où  $\delta_v$  est l'angle au volant, le déplacement relatif du véhicule est décrit par le jeu d'équations :

$$\begin{aligned}x_k &= x_{k-1} + \Delta_D \cos(\theta_k) \\y_k &= y_{k-1} + \Delta_D \sin(\theta_k) \\ \theta_k &= \theta_{k-1} + \frac{\Delta_D}{l} \tan \delta_r\end{aligned}\tag{4.2}$$

avec :

- $\Delta_{D_k}$  la distance parcourue entre les instants  $k - 1$  et  $k$ ,
- $l$  la distance entre essieux.

Il faut noter que l'angle au volant, dont la mesure est disponible dans notre véhicule a été calibré en statique et que la fonction  $\delta_r = f(\delta_v)$  ainsi trouvée est une fonction affine.

D'après la figure 4.1, il est aisé de voir que la technique de localisation couplant l'odométrie et le gyromètre est beaucoup plus précise que celle couplant odométrie et angle au volant. On peut expliquer ces résultats par le fait que la fonction  $\delta_r = f(\delta_v)$  est sujette à de fortes contraintes mécaniques lors du déplacement du véhicule (suspension, pression des pneus, etc...) et donc que la transformation affine entre l'angle au volant et l'angle des roues n'est plus vérifiée. Pour obtenir de bons résultats avec cette technique, il faudrait estimer l'angle des roues en fonction de l'angle au volant mais aussi de l'ensemble de ces contraintes mécaniques ce qui semble difficile à obtenir. La technique de positionnement couplant l'odométrie et un gyromètre est donc choisie pour effectuer la localisation relative et par conséquent les capteurs proprioceptifs embarqués dans notre véhicule expérimental sont un odomètre et un gyromètre.

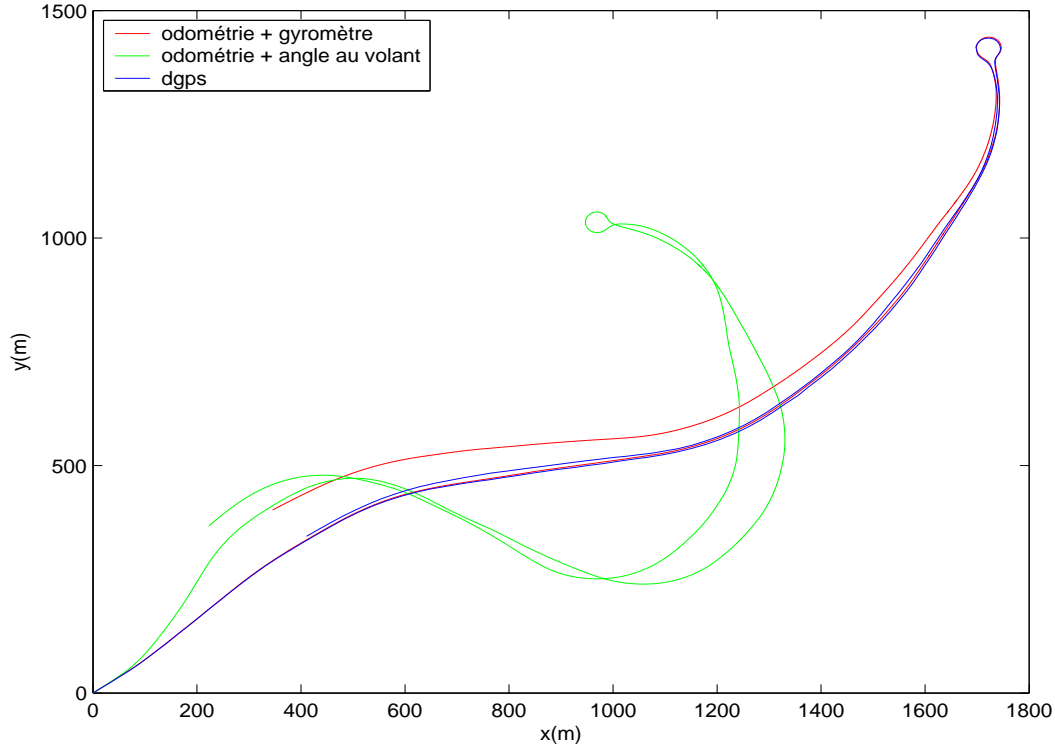


FIG. 4.1: Résultats comparatifs odomètre+gyromètre/odomètre+angle au volant

## 4.1.2 L'odomètre

### 4.1.2.1 La donnée odométrique

En raison du caractère récursif des méthodes de fusions de données (filtre de Kalman et filtre particulaire dans notre cas), l'odomètre doit permettre d'évaluer la distance parcourue par le véhicule entre deux étapes de correction. Deux cas de figure peuvent être rencontrés (cf figure 4.2), soit aucune information extéroceptive n'est traitée par le système de fusion entre deux informations odométriques, auquel cas la distance parcourue par le véhicule est simplement :

$$\Delta_D = q_o + v_{\Delta_D} \quad (4.3)$$

où

- $q_o$  est le pas de quantification de l'odomètre en mètre par top odomètre,
- $v_{\Delta_D}$  est l'incertitude de la donnée odométrique,

soit une ou plusieurs informations extéroceptives sont reçues et, dans ce cas, il est nécessaire d'effectuer une interpolation des données odométriques pour déterminer la distance a priori parcourue par le véhicule pour chacune de ces informations. Ainsi en considérant une interpolation linéaire :

$$\Delta_D = v_o |t_{o_k} - t_i| + v'_{\Delta_D} \quad (4.4)$$

avec

- $t_o$  la date d'acquisition de l'information odométrique,
- $t_i$  la date d'acquisition + latence de l'information extéroceptive,
- et  $v_o$  la vitesse du véhicule déduites des informations odométriques antérieures à l'instant  $k$ , c'est-à-dire :

$$v_o = q_o / (t_{o_k} - t_{o_{k-1}}) \quad \text{si } t_{o_k} < t_i$$

$$q_o / (t_{o_{k-1}} - t_{o_{k-2}}) \quad \text{sinon}$$

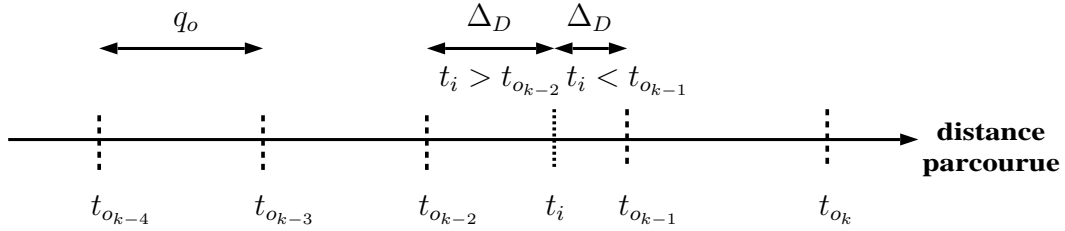


FIG. 4.2: Distance parcourue par le véhicule

#### 4.1.2.2 Incertitude de la donnée odométrique

D'après la section 1.1.1.3, les sources classiques d'erreurs de l'odométrie peuvent être divisées en deux catégories, les erreurs systématiques et les erreurs non systématiques. Dans la solution choisie pour le positionnement relatif, un seul odomètre est utilisé, donc les erreurs systématiques peuvent être résumées par les erreurs de résolution du codeur et de fréquence d'échantillonnage tandis que les erreurs non systématiques restent inchangées, à savoir glissement du véhicule, navigation sur terrain accidenté etc... De plus, comme l'accélération du véhicule n'est pas prise en compte lors des différentes interpolations (cf équation 4.4), celle-ci est une nouvelle source d'erreurs dans le calcul du déplacement du véhicule. Cependant comme notre véhicule ne peut subir des accélérations et des décélérations supérieures à  $1g$  et que le pas de quantification de l'odomètre est relativement petit ( $q_o \simeq 19,6cm$ ) on peut considérer qu'elle est négligeable par rapport aux autres sources d'erreurs. Ainsi en sachant que les résultats expérimentaux ont montré que l'erreur sur la distance parcourue par le véhicule est toujours inférieure à 1%, nous supposons que les incertitudes  $v_{\Delta_D}$  et  $v'_{\Delta_D}$  sur la donnée odométrique peuvent être décrites par la même loi normale :

$$v_{\Delta_D} = v'_{\Delta_D} \sim \mathcal{N}(0, \sigma_{\Delta_D}) \quad (4.5)$$

avec

$$\sigma_{\Delta_D} = \frac{\Delta_D}{100} \quad (4.6)$$



### 4.1.2.3 Datation de la donnée odométrique

Si l'on considère que la durée de transmission de l'information odométrique au système de traitement est très faible, alors la datation de la donnée odométrique est donnée par la date d'acquisition de l'information  $Top\_odo_k$ .

### 4.1.3 Le gyromètre

Comme nous l'avons vu dans la section 1.1.2, le gyromètre permet la mesure d'une vitesse angulaire. Placé sur l'axe vertical du véhicule, il permet de mesurer la vitesse de rotation de celui-ci. D'après l'équation (1.13), cette information peut être décomposée en trois composantes : (1) le produit facteur d'échelle  $\times$  donnée gyrométrique, (2) le biais et (3) le bruit de mesure inhérent à tout capteur.

#### 4.1.3.1 La donnée gyrométrique

Nous allons considérer que seule la première composante représente cette donnée. Ainsi la donnée gyrométrique est décrite par :

$$w_{gyro} = K_g w + v_{gyro} \quad (4.7)$$

où

- $K_g$  est le facteur d'échelle,
- $w$  est la mesure gyrométrique,
- $v_{gyro}$  incertitude de la donnée gyrométrique.

Pour calibrer le facteur d'échelle, le gyromètre a été installé sur un bras robotique capable de réaliser des déplacements circulaires de  $360^\circ$  et ce à différentes vitesses de rotation allant de  $-1$  rad/s à  $1$  rad/s. Comme le montre la figure 4.3 le facteur d'échelle de notre gyromètre peut être considéré comme constant dans le domaine d'utilisation  $[-1 \ +1]$  rad/s.

#### 4.1.3.2 Incertitudes sur la donnée gyrométrique

Les deux composantes restantes (1.13) représentent donc les incertitudes sur la donnée gyrométrique. Ainsi :

$$v_{gyro} = b_w + \epsilon_w \quad (4.8)$$

où

- $b_w$  est le biais
- $\epsilon_w$  est le bruit de mesure

D'après la section 1.1.2, le biais dépend de la température. Cependant n'ayant pas de banc de test approprié, en particulier d'enceintes thermostatées couvrant le domaine de température que peut subir un véhicule, il nous est impossible d'établir une relation entre le biais et la température. Néanmoins nous avons pu quand même réaliser quelques tests

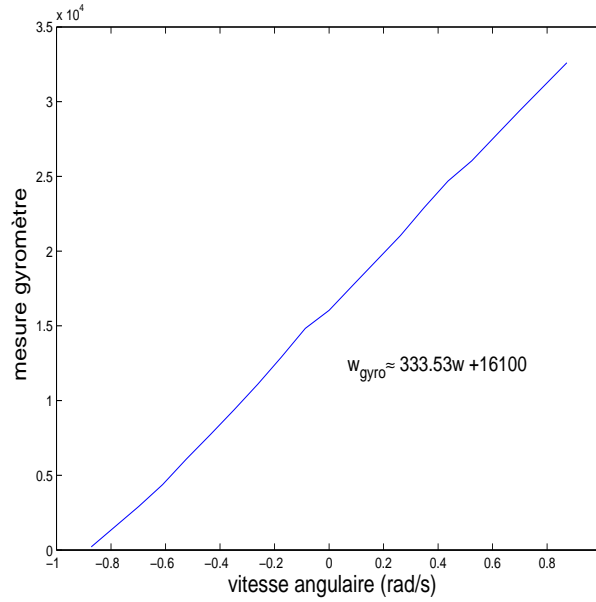


FIG. 4.3: Facteur d'échelle du gyromètre

en température et en statique du gyromètre. Les résultats sont présentés dans la figure 4.4. Au vu des courbes, le comportement décrit dans l'état de l'art est bien retrouvé. Ainsi le biais induit une dérive de la mesure pendant les dix premières minutes de fonctionnement du gyromètre. Une fois que l'échauffement des composants électroniques du gyromètre s'est stabilisé, la mesure est seulement entachée par le bruit de mesure.

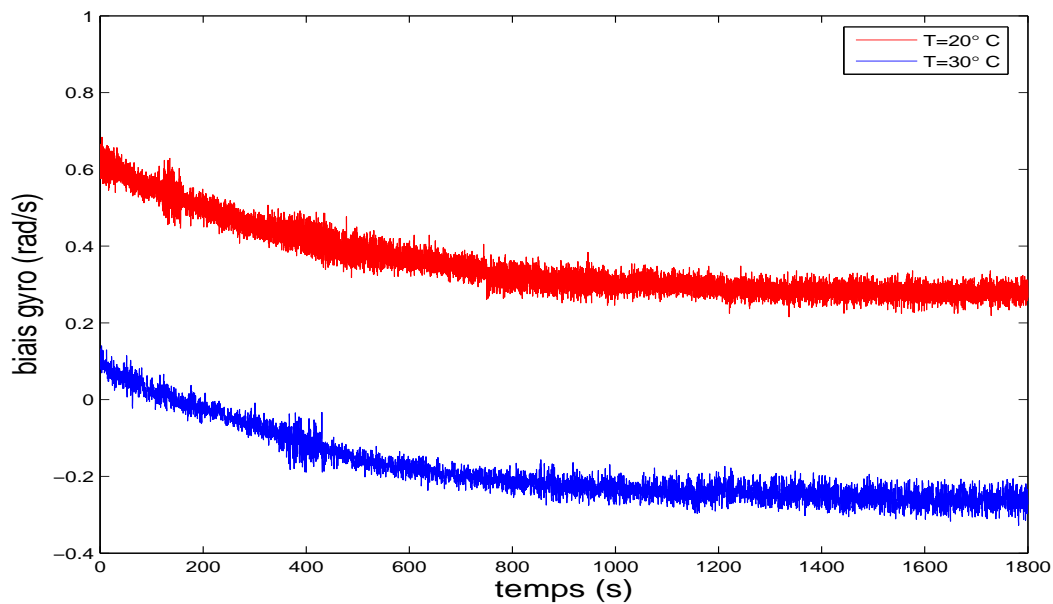


FIG. 4.4: Comportement en situation statique du gyromètre en fonction de la température.

En général, ce bruit est considéré comme gaussien centré blanc. Pour vérifier cette hypothèse, une fois le gyromètre stabilisé en température, la densité de probabilité des mesures gyrométriques a été comparée à la densité de probabilité théorique de la loi normale  $\mathcal{N}(b_w, \sigma_{gyro}^2)$  (figure 4.5(b)) et un test de blancheur de ces mesures (figure 4.5(a)) a été réalisé. Ainsi, d'après la figure 4.5(b), la densité de probabilité déduite des mesures est très similaire à la densité de probabilité théorique donc le bruit de mesure sera supposé de nature gaussienne. La figure 4.5(a) présente la fonction d'autocorrélation des mesures, celle-ci s'approchant d'une impulsion de Dirac, le bruit de mesure est un bruit blanc. Donc le bruit de mesure peut être considéré comme un bruit gaussien centré blanc, stationnaire une fois la température du gyromètre stabilisée, d'où la notation  $\epsilon_w$  dans l'équation (4.8).

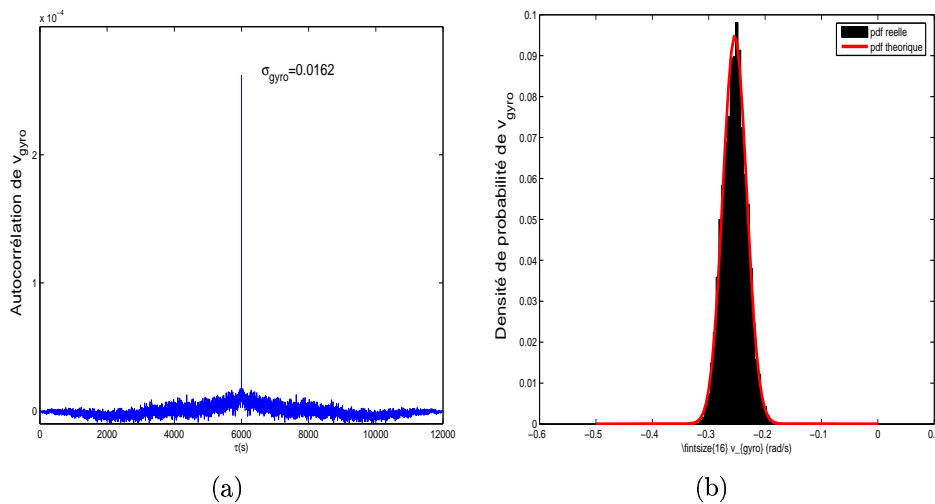


FIG. 4.5: Caractérisation du bruit de mesure du gyromètre, figure (a) test de blancheur, figure (b) densité de probabilité

Le principal problème du modèle d'incertitude décrit dans l'équation (4.8) est que le biais est inconnu. Nous savons qu'il est fonction de la température, cependant nous ne pouvons établir cette relation. Plusieurs méthodes existent pour déterminer ce paramètre, la première est de placer le gyromètre dans une enceinte thermostatée, la température est alors constante et par la même occasion le biais est lui-même constant et connu. Cependant un tel système demande une longue phase de stabilisation en température et donc une longue attente avant de pouvoir se servir du véhicule, ce qui est inenvisageable dans le domaine de l'automobile. La deuxième méthode est d'estimer la valeur du biais lors du processus de localisation. D'ailleurs dans [Park, 2004] [Abuhadrous, 2005] les auteurs estiment les biais des différents gyromètres d'une centrale inertielle en utilisant les observations fournies par un GPS différentiel et en considérant que les biais peuvent être décrits par des processus markoviens. Cependant, dans notre système, la localisation relative est seulement réalisée par l'association d'un gyromètre et de l'odométrie, il est donc impossible d'utiliser cette méthode pour estimer le biais du gyromètre.

Dans notre approche, le biais sur les données gyrométriques est donc a priori inconnu. Néanmoins pour limiter les erreurs de positionnement relatif induites par celui-ci, nous avons considéré qu'il pouvait être décrit par un processus de constante aléatoire. Cette constante est alors déterminée avant chaque expérimentation par une procédure d'apprentissage. Ainsi sous les contraintes que la stabilisation en température des composants du gyromètre soit effectuée et que le véhicule ne présente aucun mouvement, le biais (ou la constante) est estimé en effectuant la moyenne des données fournies par le gyromètre pendant quelques minutes :

$$\hat{b}_w = \frac{1}{N} \sum_{n=1}^N K_g w - K_g w_0 \quad (4.9)$$

où  $N$  est le nombre de données gyrométriques acquises pendant l'apprentissage et  $w_0$  la donnée fournie par le gyromètre pour une vitesse angulaire nulle lors de la calibration.

Ainsi, après la phase d'apprentissage, le biais est considéré connu et constant pendant une courte durée, mais qui est assez longue pour réaliser nos expérimentations. Les incertitudes sur les données gyrométriques sont dans ce cas seulement le bruit de mesure :

$$v_{gyro} = \epsilon_w \quad (4.10)$$

De ce fait, la donnée gyrométrique devient :

$$w_{gyro} = K_g w - \hat{b}_w \quad (4.11)$$

### 4.1.3.3 Datation

Comme pour la donnée odométrique, si l'on considère que la durée de transmission de l'information gyrométrique au système de traitement est très faible, alors la datation de la donnée gyrométrique  $w_{gyro}$  est donnée par la date d'acquisition de l'information gyrométrique.

## 4.2 Capteurs extéroceptifs

### 4.2.1 Choix des capteurs

Le choix des capteurs extéroceptifs est beaucoup plus aisé que le choix des capteurs proprioceptifs. En effet il découle des méthodes de positionnement absolu potentiellement utilisables dans les contextes routiers (en sachant qu'aucun aménagement du réseau routier n'est envisageable). D'après la section 1.2, il n'en existe que trois dans notre cas : le positionnement par satellites, le positionnement par vision et le positionnement par téléphone cellulaire. Cependant le manque de maturité de cette dernière technique la rend peu viable dans la réalité et seules les deux premières techniques semblent envisageables dans notre application. Comme le positionnement par satellites est la technique par excellence pour localiser un robot mobile ou un véhicule en environnement extérieur, il est impératif

d'intégrer ce système dans notre approche et notre choix se porte donc vers un GPS. Cependant la contrainte du faible coût d'intégration de notre approche impose l'utilisation d'un GPS bas coût et donc de faible précision. Ainsi pour améliorer la précision de notre système de localisation, des informations redondantes aux informations GPS doivent être prises en compte et seront dans notre cas obligatoirement fournies par un système de positionnement par vision. Dans la suite de ce mémoire, ce système est appelé capteur "vision" et est donc le deuxième capteur extéroceptif choisi pour notre application.

## 4.2.2 Le GPS

### 4.2.2.1 Les données GPS

D'après la section 1.2.2, les principales informations fournies par le GPS sont les coordonnées géographiques du récepteur à savoir sa longitude, sa latitude et sa hauteur ellipsoïdale donnée dans le système de coordonnées WGS84 et le GDOP permettant de calculer l'incertitude sur ces coordonnées. Pour que ces informations soient compatibles avec les données fournies par les autres capteurs et le système d'informations géographiques (carte plane dans le système de coordonnées LAMBERT 93), les coordonnées géodésiques du récepteur GPS doivent être converties en coordonnées géocentriques puis projetées dans le référentiel LAMBERT93 (cf annexe B). Les données GPS sont alors :

$$\underline{Y}_{gps} = \begin{pmatrix} x_{gps} \\ y_{gps} \end{pmatrix} + v_{gps} \quad (4.12)$$

où

- $(x_{gps}, y_{gps})^T$  est une position cartésienne dans le référentiel LAMBERT 93 décrivant la position du récepteur GPS sur la carte,
- $v_{gps}$  incertitude sur la position fournie par le GPS.

### 4.2.2.2 Incertitudes sur les données GPS

Dans la plupart des travaux de localisation de véhicule utilisant un GPS bas coût les incertitudes sur les données GPS sont approchées par un bruit blanc gaussien centré qui suit la loi normale :

$$v_{gps} \sim \mathcal{N}(0, \mathbf{R}_{gps}) \quad (4.13)$$

où  $\mathbf{R}_{gps}$  est la matrice de covariance associée aux données GPS. En général cette matrice est décrite de la manière suivante :

$$\mathbf{R}_{gps} = \begin{pmatrix} \sigma_{x_{gps}}^2 & 0 \\ 0 & \sigma_{y_{gps}}^2 \end{pmatrix} \quad \text{avec} \quad \sigma_{x_{gps}} = \sigma_{y_{gps}} = EURE_{gps} \times GDOP_{gps} \quad (4.14)$$

Ici l' $EURE_{gps}$  est l'écart-type sur la position fournie par le GPS (ce paramètre est fourni par le constructeur du récepteur GPS) et le  $GDOP_{gps}$  donne la confiance sur cette position en fonction de la position des satellites dans le ciel. Le GDOP est en général compris entre 1 et 5, au dessus de 5 la position fournie par le GPS est complètement aberrante et ne doit pas être prise en compte par le système. Enfin on peut noter que les paramètres

de corrélation ne sont pas fournis par le récepteur.

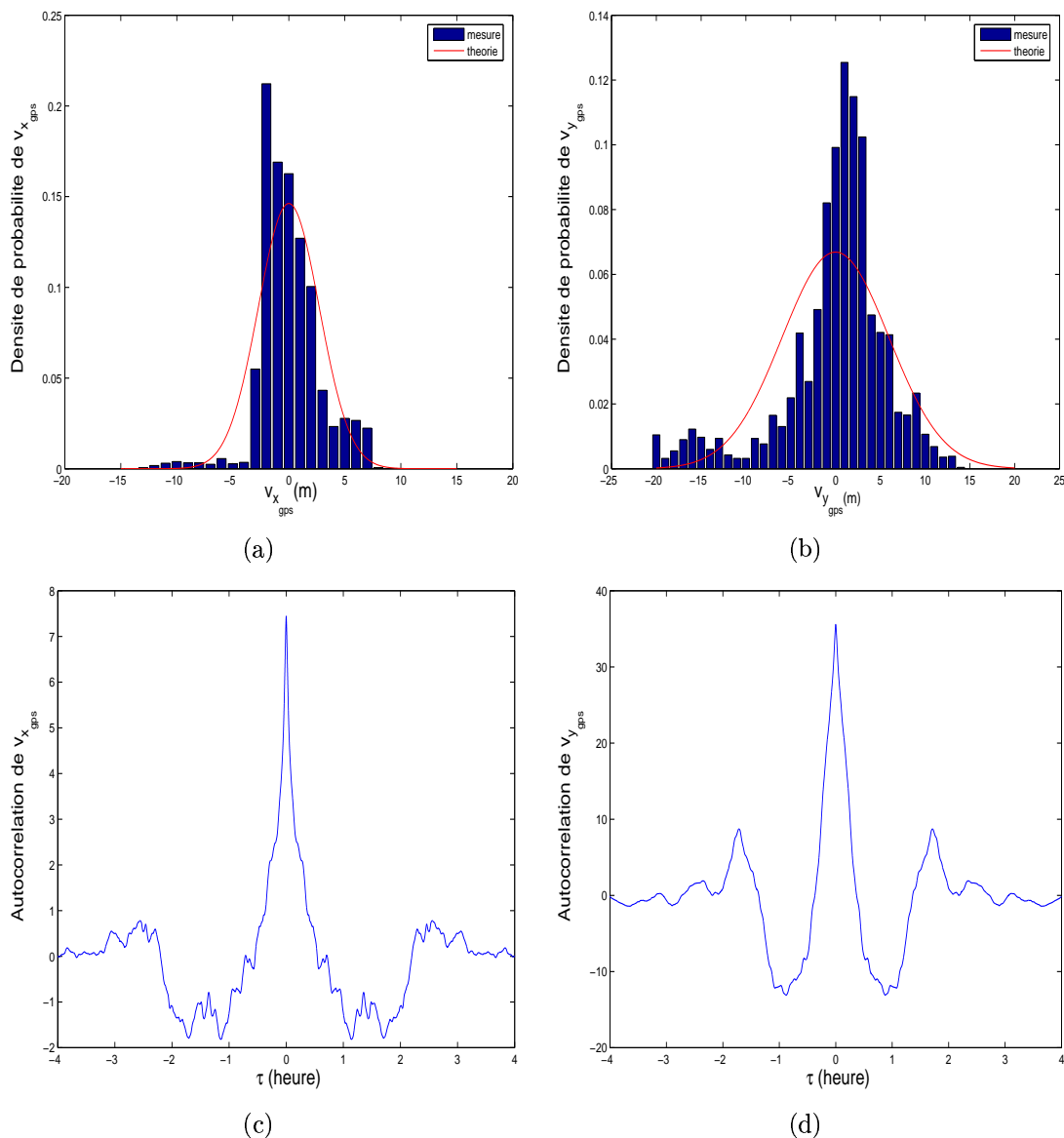


FIG. 4.6: Caractéristiques des incertitudes gps : figure (a) et (b) densités de probabilité de  $v_{x_{gps}}$  et de  $v_{y_{gps}}$ , figure (c) et (d) fonctions d'autocorrélation de  $v_{x_{gps}}$  et de  $v_{y_{gps}}$

Pour corroborer ces affirmations, nous avons effectué le test expérimental suivant : les informations de position  $x_{gps}$  et  $y_{gps}$  fournies par le GPS ont été acquises pendant plusieurs heures sachant que le GPS a été disposé à un emplacement dont la position était parfaitement connue. A partir des données recueillies, nous avons pu obtenir une approximation de la distribution de probabilité (figures 4.6(a) et 4.6(b)) et effectuer un test de blancheur (figures 4.6(c) et 4.6(d)) des incertitudes de chacun de ces paramètres. En comparant les densités de probabilité de  $v_{x_{gps}}$  et  $v_{y_{gps}}$  ainsi obtenues à leurs distributions

de probabilité théoriques (distributions décrites par les lois  $\mathcal{N}(0, \sigma_{v_{x_{gps}}}^2$  ou  $\sigma_{v_{y_{gps}}}^2$ ) où les écarts types  $\sigma_{v_{x_{gps}}}$  et  $\sigma_{v_{y_{gps}}}$  sont ceux trouvés expérimentalement). Il est aisé de voir que les distributions de probabilité de  $v_{x_{gps}}$  et  $v_{y_{gps}}$  ne sont pas de nature gaussienne. De plus, les tests de blancheur réalisés par les fonctions d'autocorrélations montrent que les incertitudes ne sont pas blanches mais décrivent plutôt des processus qui varient lentement et corrélés dans le temps. Ces résultats démontrent que les incertitudes des informations GPS ne peuvent être décrites par des bruits blancs gaussiens centrés rendant ainsi la relation (4.13) caduque. Dans ce cas, les informations GPS ne peuvent alimenter le processus de fusion (filtre de Kalman ou filtre particulaire) sans une modélisation au préalable de leurs parties stochastiques et doivent être considérées comme étant :

$$\underline{Y}_{gps} = \underline{Y}'_{gps} + \underline{b}_{gps} + v'_{gps} \quad (4.15)$$

où

- $\underline{Y}'_{gps}$  est la partie systématique des informations GPS qui représente la vraie position du récepteur GPS,
- $\underline{b}_{gps}$  est la partie stochastique des informations GPS,
- $v'_{gps}$  est un bruit blanc résiduel entachant les mesures.

D'après le chapitre 3, il existe de nombreux processus pouvant représenter la partie stochastique d'une information non-blanche : processus de marche aléatoire, processus de Gauss-Markov, processus périodique aléatoire ou encore processus autorégressif. Pour déterminer quelle est la modélisation la plus adéquate pour les informations GPS, il suffit de comparer la moyenne des erreurs quadratiques entre la prédiction des erreurs GPS (selon les différents processus proposés) et la vraie mesure des erreurs GPS, c'est-à-dire :

$$\frac{1}{N} \sum_{n=1}^N [b_k - f_M(b_{k-1} \dots b_{k-m}, \epsilon_k)]^2 \quad (4.16)$$

où

- $b_k$  est la mesure des erreurs GPS à l'instant  $k$ ,
- $N$  est le nombre de mesures acquises expérimentalement,
- $f_M$  sont les équations de récurrences définissant le filtre formeur du processus stochastique,
- $m$  est le nombre de mesures du passé du processus pris en compte par le modèle,
- $\epsilon_k$  est le signal blanc injecté dans le modèle à chaque instant  $k$ .

Ainsi, le modèle obtenant la moyenne des erreurs quadratiques de prédiction la plus faible, et ce dans les situations statique et dynamique, sera utilisé dans le filtrage.

**Erreurs de prédiction des différents modèles en statique :** Le tableau 4.1 présente les statistiques sur les erreurs de prédiction (moyenne, maximum, moyenne des erreurs quadratiques) selon différents modèles de processus aléatoires à savoir : bruit blanc, marche aléatoire, Gauss-Markov d'ordre 1 et AR d'ordre 1 à 5 définis par la méthode de

Burg.

D'après la première ligne du tableau 4.1, il est clair que la modélisation des incertitudes

Modèle de processus	Erreurs en x (m)			Erreur en y (m)		
	moy	max	RMS	moy	max	RMS
Bruit blanc gaussien	1.994	8.495	2.559	3.691	20.320	4.6
Rand Walk	0.038	0.649	0.077	0.104	1.298	0.171
Gauss-Markov ordre 1	0.091	0.6028	0.118	0.179	1.352	0.236
AR ordre 1	0.038	0.6484	0.077	0.0994	1.295	0.171
AR ordre 2	0.038	0.6484	0.076	0.1034	1.262	0.168
AR ordre 3	0.041	0.7166	0.074	0.1041	1.253	0.168
AR ordre 4	0.043	0.7173	0.071	0.1057	1.281	0.165
AR ordre 5	0.043	0.6724	0.069	0.1053	1.255	0.163

TAB. 4.1: Erreur de prédiction en statique selon les modélisations stochastiques

GPS par un bruit blanc gaussien n'est pas adéquate. D'après les fonctions d'autocorrélation de la figure 4.6, les incertitudes du GPS sont des processus aléatoires qui varient très lentement avec un temps de corrélation proche de 900s ( $t_c \approx 1/4h$ ). Comme la période d'échantillonnage des informations GPS est de 1s, alors leurs incertitudes ne varient quasiment pas d'un instant à l'autre, ce qui explique que la modélisation des incertitudes par un processus de marche aléatoire donne quasiment les mêmes résultats qu'un processus autorégressif d'ordre 5. A priori d'après les résultats du tableau 4.1, à part le bruit blanc et le processus de Gauss-Markov n'importe quel autre processus peut être utilisé pour modéliser les incertitudes du GPS en statique.

La figure 4.7 présente la moyenne des erreurs quadratiques de prédiction des modèles AR d'ordre 1 à 5, et ce pour les méthodes de régression linéaire de Yule-Walker, de Burg et de la Covariance. D'après cette figure on retrouve les propos énoncés dans la section 3.5, à savoir que la méthode Yule-Walker fournit les modèles autorégressifs les moins précis, la méthode de la Covariance les modèles les plus précis et que la méthode de Burg donne des résultats similaires à la méthode de la Covariance. Enfin, comme la méthode de Burg est la seule à garantir la stabilité du modèle, il est préférable d'utiliser cette méthode pour définir la partie stochastique des informations GPS.

**Erreurs de prédiction des différents modèles en dynamique.** Le tableau 4.2 présente les statistiques des erreurs de prédiction (moyenne, maximum, moyenne des erreurs quadratiques) selon les processus aléatoires utilisés dans le paragraphe précédent.

Là encore, d'après la première ligne du tableau 4.2 on peut voir que le bruit blanc gaussien n'est pas adéquat pour modéliser les incertitudes GPS. Sinon tous les autres processus donnent des résultats similaires à part peut être les modèles AR d'ordre élevé qui améliorent sensiblement la précision de la prédiction. On peut donc tout simplement



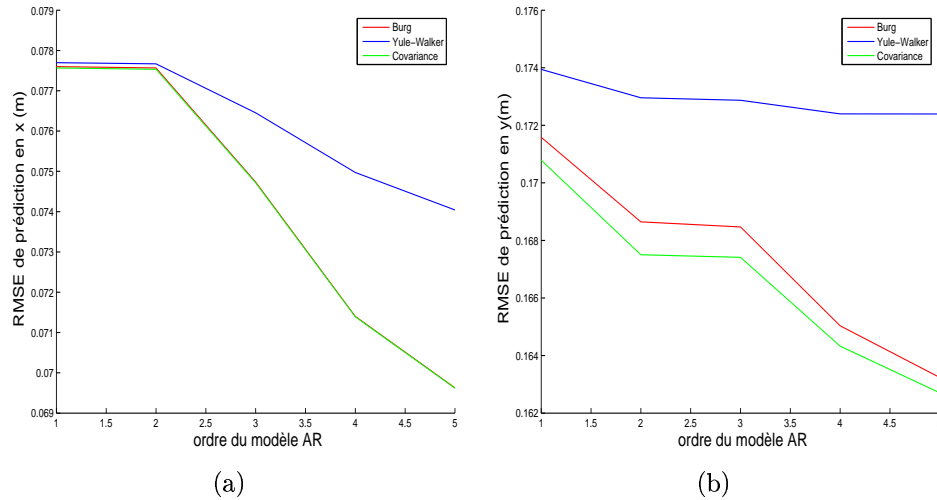


FIG. 4.7: RMSE de prédiction en statique de  $b_x$  et  $b_y$  pour les méthodes de Yule-Walker, de Burg et de covariance

Modèle de processus	Erreurs en x (m)			Erreur en y (m)		
	moy	max	RMS	moy	max	RMS
Bruit blanc gaussien	1.530	6.903	1.935	2.582	10.704	3.227
Rand Walk	0.5554	5.0738	0.8491	0.4304	5.0576	0.6518
Gauss-Markov ordre 1	0.5591	5.1599	0.8521	0.4413	4.9312	0.6583
AR ordre 1	0.5533	5.0318	0.8477	0.4280	5.0604	0.6488
AR ordre 2	0.5543	5.2363	0.7860	0.4206	4.9791	0.6239
AR ordre 3	0.5536	5.2379	0.7600	0.4219	4.9336	0.6183
AR ordre 4	0.5533	5.2528	0.7454	0.4222	4.9153	0.6176
AR ordre 5	0.5536	5.2592	0.7413	0.4224	4.9063	0.6175

TAB. 4.2: Erreur de prédiction en dynamique selon les modélisations stochastiques

utiliser n'importe lequel de ces processus pour modéliser les incertitudes GPS.

La figure 4.8 présente la moyenne des erreurs quadratiques de prédiction en dynamique des modèles AR d'ordre 1 à 5 fournis par les trois différentes méthodes de régression linéaire. D'après cette figure, il apparaît que les trois méthodes fournissent des modèles quasi-équivalents, cependant comme la méthode de Burg garantit la stabilité de ces modèles, il est préférable d'utiliser cette méthode.

**Etape du filtrage.** D'après les deux paragraphes précédents, de nombreuses méthodes peuvent être utilisées pour modéliser la partie stochastique des informations GPS (marche aléatoire, Gauss-Markov, AR). La plus précise de toutes est bien sûr la modélisation AR d'ordre élevé. Cependant celle-ci représente un inconvénient pour le filtrage et notamment

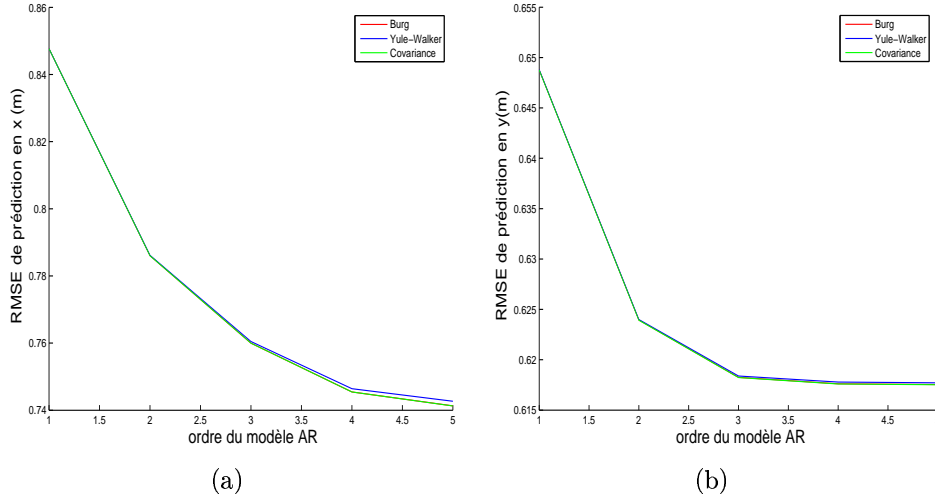


FIG. 4.8: RMSE de prédiction en dynamique de  $b_x$  et  $b_y$  pour les méthodes de Yule-Walker, de Burg et de covariance

pour le filtrage particulière. En effet, pour toute information non blanche, la partie systématique et la partie stochastique sont des inconnues et doivent être intégrées au vecteur d'état pour être estimées (cf section 3.4). Si l'on considère un modèle AR d'ordre 5, il est alors nécessaire d'ajouter douze variables au vecteur d'état, c'est-à-dire 5 variables pour la partie systématique et 1 variable pour la partie stochastique des informations  $x$  et  $y$  du GPS. Dans cette situation l'espace d'état à couvrir par la méthode de filtrage devient très grand et l'estimation de la position du véhicule par un filtre particulière demande un très grand nombre de particules qui peut dépasser la puissance de calcul des ordinateurs actuels. Ainsi, pour réaliser le filtrage, nous avons donc préféré utiliser une modélisation AR du premier ordre. Les informations GPS peuvent être alors décrites par :

$$\begin{aligned} x_{gps_k} &= x_k - \alpha_x b_{x_{k-1}} + \beta_x \epsilon_{b_{x_k}} + v'_{x_{gps}} \\ y_{gps_k} &= y_k - \alpha_y b_{y_{k-1}} + \beta_y \epsilon_{b_{y_k}} + v'_{y_{gps}} \end{aligned} \quad (4.17)$$

où

- $x_k$  et  $y_k$  sont les parties systématiques des informations GPS,
- $b_{x_k}$  et  $b_{y_k}$  sont les parties stochastiques des informations GPS que l'on nommera les paramètres de biais du GPS,
- $\alpha_x$ ,  $\beta_x$ ,  $\alpha_y$  et  $\beta_y$  les paramètres des modèles AR du premier ordre avec  $\alpha_x$  et  $\alpha_x$  de l'ordre de 0.99 et  $\beta_x$  et  $\beta_y$  de l'ordre de 0.2,

En considérant que les parties systématiques  $x_k$  et  $y_k$  des informations GPS sont en réalité la vraie position du véhicule et donc la position que l'on veut estimer, il est alors seulement nécessaire d'intégrer les variables de la partie stochastique dans le vecteur d'état :

$$\underline{X} = (x \ y \ \theta \ b_x \ b_y)^T \quad (4.18)$$

Alors les incertitudes sur les informations GPS sont renommées  $v'_{gps} = \mathcal{N}(0, \mathbf{R}'_{gps})$  et sont déterminées en fonction des erreurs résiduelles de prédiction du modèle AR du premier ordre, tel que :

$$\mathbf{R}'_{gps} \sim \begin{pmatrix} \sigma_{AR_x}^2 & 0 \\ 0 & \sigma_{AR_y}^2 \end{pmatrix} \quad (4.19)$$

où  $\sigma_{AR_x}$  et  $\sigma_{AR_y}$  sont les écarts-types des erreurs résiduelles de prédiction.

#### 4.2.2.3 Datation des données GPS

Comme pour tout récepteur GPS, les informations GPS sont fournies au système avec un retard. Ce retard est communément appelé la latence GPS et correspond principalement au temps de traitement des informations reçues de chacun des satellites visibles. Cette latence est variable et dépend fortement de la position et du nombre des satellites visibles. Généralement de l'ordre de quelques centaines de millisecondes, elle ne peut être négligée par le système de fusion. En effet, si l'on considère un véhicule se déplaçant le long d'une trajectoire rectiligne à la vitesse de  $90km/h$ , une latence de  $200ms$  induit une erreur sur la position du véhicule d'environ  $5m$ , si la mise à jour est réalisée en fonction de cette observation alors l'estimation de la position sera obligatoirement erronée. Avant de considérer les observations GPS, il est donc nécessaire de recalibrer dans le temps ces informations en fonction de leur latence, ceci est l'objet du paragraphe 4.2.3.3. Dans la suite de cette section nous allons décrire la méthode utilisée pour quantifier cette latence.

D'après le paragraphe précédent, les erreurs sur les informations GPS varient pour l'essentiel très lentement dans le temps. On peut donc supposer que le déplacement relatif du véhicule peut être calculé avec une bonne précision en fonction de deux informations GPS à condition que le temps séparant ces informations soit relativement faible par rapport au temps de corrélation des incertitudes GPS (de l'ordre de la minute). Ce même déplacement peut être aussi calculé en fonction des capteurs proprioceptifs du véhicule sachant que là aussi une bonne précision n'est obtenue que si le temps écoulé entre le début et la fin du déplacement reste court. La latence peut être alors identifiée par une minimisation de la somme des erreurs quadratiques entre le déplacement calculé avec deux informations GPS et le déplacement calculé avec les informations proprioceptives :

$$lat_g = \text{Arg min} \left( \sum_t \left| \Delta_{D_{gps}}(t) - \Delta_{D_{pro}}(t - \phi) \right| \right) \quad \phi \in [0, \phi_{max}] \quad (4.20)$$

avec

- $\Delta_{D_{gps}}$  la distance calculée à partir des informations GPS.
- $\Delta_{D_{pro}}$  la distance déduite des informations proprioceptives.

La minimisation est bornée entre 0 et  $\phi_{max} = 1s$  dans le temps car on considère que la latence est obligatoirement comprise dans ce même intervalle, ce qui est réaliste pour un GPS.

Pour réaliser ce calcul, nous nous sommes imposés les conditions expérimentales suivantes :

- le véhicule se déplace selon une trajectoire rectiligne, ainsi le déplacement se résume à la distance curviligne parcourue par le véhicule :
- pour les informations GPS ce déplacement est décrit par

$$\Delta_{D_{gps}}(t) = \sqrt{(x_{gps}(t) - x_{gps}(0))^2 + (y_{gps}(t) - y_{gps}(0))^2} \quad (4.21)$$

où  $(x_{gps}(0), y_{gps}(0))$  est la position de départ du véhicule et  $(x_{gps}(t), y_{gps}(t))$  la position du véhicule à l'instant  $t$

- pour les informations proprioceptives, ce déplacement peut être déduit simplement à partir de l'odométrie,

$$\Delta_{D_{pro}}(t) = \sum_{t=t_0}^{t_n} \Delta_D(t) \quad (4.22)$$

avec  $\Delta_D(t_n)$  le déplacement curviligne du véhicule mesuré par l'odométrie entre deux instants  $t_n$  et  $t_{n-1}$ ,

- la distance à parcourir est d'environ  $1500m$ ,
- le véhicule doit évoluer à vitesse constante le plus longtemps possible, cette vitesse étant comprise entre  $50$  et  $100km/h$ .

La figure 4.9(a) montre que les distances calculées  $\Delta_{D_{gps}}$  et  $\Delta_{D_{pro}}$  en fonction des deux différentes sources d'informations décrivent bien le même déplacement curviligne du véhicule et ce pour différentes vitesses d'évolution. Cependant dans la partie linéaire des courbes (figure 4.9(b)) c'est-à-dire quand le véhicule évolue à vitesse constante, la courbe calculée en fonction des informations GPS est en retard par rapport à celle calculée en fonction des données odométriques, démontrant ainsi la présence d'une latence pour les informations GPS. Les sommes des erreurs quadratiques pour  $\phi$  allant de 0 à 1s et pour les vitesses  $60$  et  $70km/h$  sont montrées dans les figures 4.9(c) et 4.9(d). Il est aisé de voir que ces fonctions présentent un minimum global autour de  $250ms$ . Ce minimum est donc la latence du GPS. D'ailleurs, on peut trouver les valeurs de latences pour des vitesses d'évolution du véhicule comprises entre  $50km/h$  à  $100km/h$  dans le tableau 4.3. Au vu de ce tableau on peut en déduire que la latence du GPS n'est pas forcément constante. Mais, d'après les tests expérimentaux que nous avons effectués, celle-ci est toujours comprise entre  $200ms$  et  $260ms$ . Dans la suite de ce mémoire, nous considérons que la latence du GPS est constante et égale à l'espérance de toutes les latences mesurées expérimentalement soit environ  $220ms$  et qui par exemple, engendra une erreur maximale de  $1.44m$  à  $130km/h$  de  $1m$  à  $90km/h$  et de  $0.55m$  à  $50km/h$  sur la localisation du véhicule.

### 4.2.3 Le capteur "Vision"

Comme énoncé dans le début de cette partie, nous voulons effectuer la localisation de notre véhicule avec des capteurs bas coûts. Dans ce cas, les informations provenant du GPS sont peu précises et il est donc nécessaire d'avoir une deuxième source d'informations extéroceptives plus précises pour obtenir une localisation correcte du véhicule. De plus, d'après la section 4.2.2.2, les informations GPS ne sont pas blanches et introduisent deux

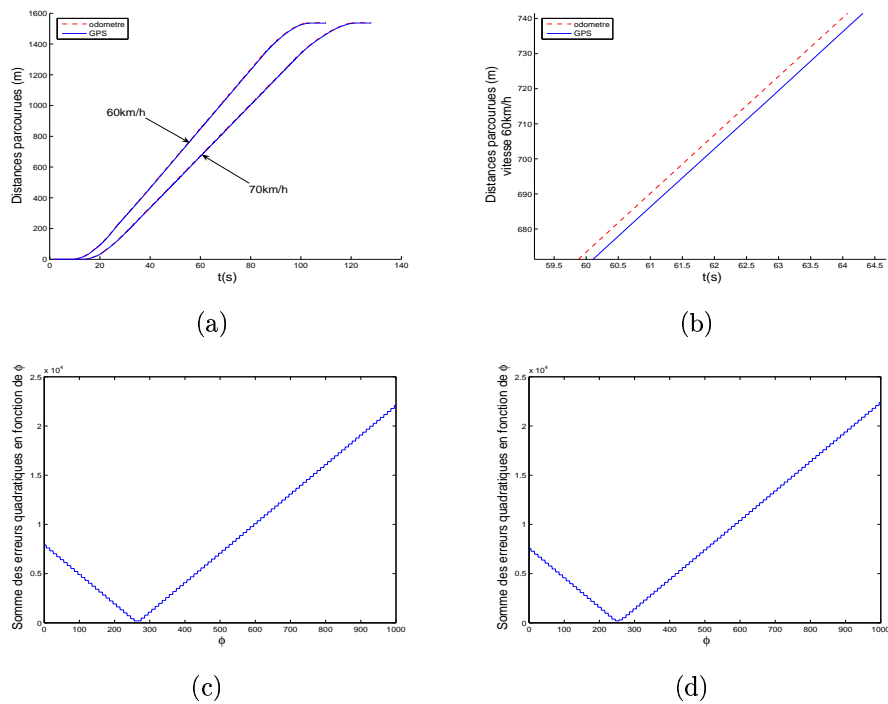


FIG. 4.9: Caractérisation de la latence du GPS, figure(a) déplacement relatif pour des vitesses d'évolution de  $60\text{km/h}$  et  $70\text{km/h}$ , figure (b) zone linéaire du déplacement, figure (c) et (d) : somme des erreurs quadratiques pour  $60\text{km/h}$  et  $70\text{km/h}$

vitesse (km/h)	50	60	70	80	90	100
latence (s)	0.200	0.260	0.250	0.220	0.230	0.220

TAB. 4.3: Latences estimées pour différentes vitesses d'évolution du véhicule

variables supplémentaires  $b_x$  et  $b_y$  (parties stochastiques des informations GPS) dans le vecteur d'état. Les informations fournies par le capteur "vision" devront donc permettre une estimation de ces paramètres et améliorer ainsi la localisation du véhicule.

Le capteur "vision" est un capteur hybride couplant les informations provenant d'un algorithme de détection de voies par vision monoculaire et les informations provenant des capteurs proprioceptifs du véhicule. L'algorithme de détection de voies a été conçu dans notre laboratoire par Romuald Aufrère [Aufrère, 2001]. Il permet entre autre de détecter les marquages au sol, de les mettre en correspondance avec un modèle général de route et d'estimer la position latérale et l'orientation du véhicule sur la chaussée ainsi que certains paramètres géométriques de la route, comme sa largeur ou encore sa courbure. Dans sa version initiale cet algorithme connaissait quelques limitations :

1. l'utilisation d'un modèle paramétrique à courbure constante pour représenter la géométrie de la route qui, nous le rappelons, ne peut décrire toutes les situations routières et notamment les zones de transitions (cf. section 1.2.4.3),

2. l'impossibilité de réaliser un suivi temporel en fonction du comportement cinématique du véhicule car, à l'époque, notre véhicule expérimental n'était pas équipé de capteurs proprioceptifs.

Dans la suite de cette section, nous allons rappeler le principe de cet algorithme auquel nous avons ajouté une étape de prédiction prenant en compte le comportement cinématique et substitué un modèle clothoïdal au modèle à courbure constante. Enfin, le principe de la détection des amers visuels ne sera pas présenté en détails. Néanmoins, de nombreuses informations peuvent être trouvées dans [Aufrère *et al.*, 2000a] [Aufrère *et al.*, 2000b].

#### 4.2.3.1 Le principe de l'algorithme de vision

Dans la plupart des méthodes traditionnelles de localisation basées sur la vision, le processus est divisé en deux étapes : 1) la détection de primitives dans l'image, 2) l'estimation de la position à partir des primitives images. De par le manque d'informations sur la cohérence entre les primitives images et les paramètres 3D à estimer, l'estimation de la localisation du véhicule est très sensible à la vraisemblance des primitives détectées. Pour pallier ce défaut, l'approche utilisée dans le capteur "vision" propose de guider la détection des primitives par un modèle statistique qui fournit les dépendances entre les informations images et les paramètres de localisation.

Ainsi les deux étapes traditionnelles de localisation sont réunies en une seule, le vecteur d'état  $\underline{X}_v$  à estimer est constitué par les primitives images, les paramètres de localisation du véhicule et les paramètres de géométrie de la route. La cohérence entre les primitives et les paramètres 3D est régie par la distribution de probabilité normale  $\mathcal{N}(\underline{X}_v(0), \mathbf{P}_{X_v}(0))$ . Cette loi est apprise pendant une phase d'apprentissage, en considérant que l'ensemble des primitives images sont reliées aux paramètres 3D de la scène routière par un jeu d'équations considérées linéaires autour du point de fonctionnement défini par la loi normale  $\mathcal{N}(\underline{X}_l(0), \mathbf{P}_{X_l}(0))$ . Où  $\underline{X}_l(0)$  est le vecteur moyen des paramètres de localisation et de géométrie de la chaussée et  $\mathbf{P}_{X_l}(0)$  une matrice de covariance décrivant les variations possibles de ces paramètres en considérant toutes les situations possibles de scènes routières.

Ainsi, de cette manière, la localisation est toujours contrainte par le modèle statistique général et ne peut être aberrante. De plus une estimation de sa qualité est fournie par l'algorithme de vision et permet de savoir si la localisation doit être prise en compte ou non. La figure 4.10 présente l'organigramme de l'algorithme de vision.

### A Les paramètres 2D et 3D et l'apprentissage

Dans cette partie l'ensemble des paramètres de localisation, de géométrie de la route ainsi que le modèle utilisé pour les primitives vont être présentés. Une fois ces paramètres énoncés, l'apprentissage nécessaire au fonctionnement de l'algorithme va être décrit en

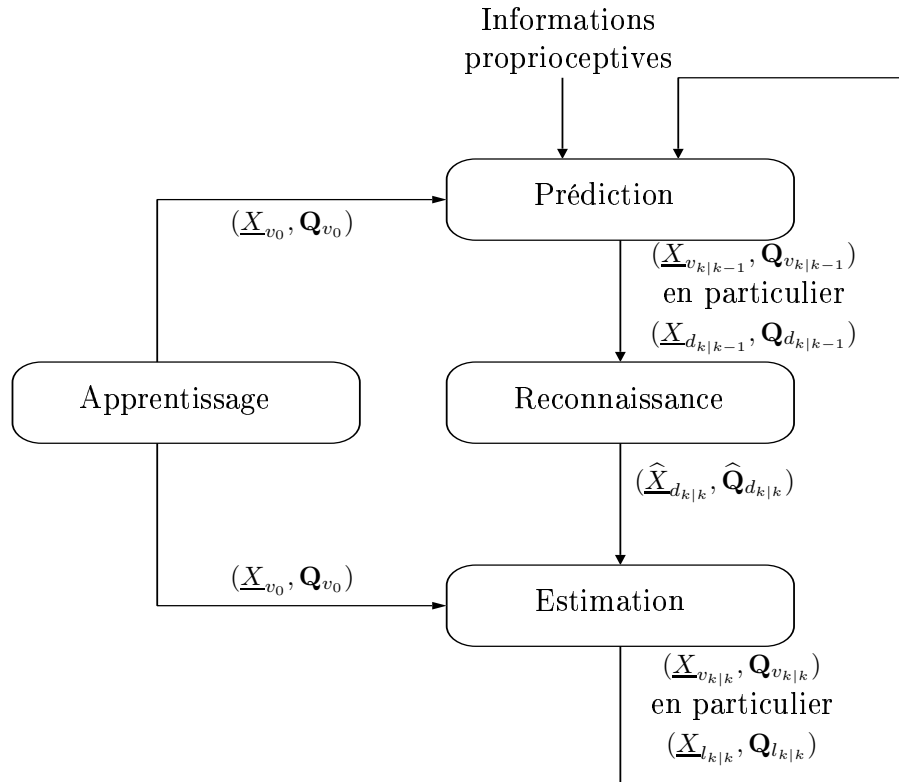


FIG. 4.10: Organigramme de l'algorithme de vision

détail.

**Paramètres de localisation et de géométrie de la route.** Les paramètres décrivant la position latérale et l'orientation du véhicule sur la chaussée ainsi que les paramètres géométriques de la route sont définis par le vecteur suivant :

$$\underline{X}_l = (L, X_0, \Psi_0, \alpha, C_h, C_l)^T \quad (4.23)$$

avec :

- $L$  : la largeur de la voie de circulation ou de la chaussée,
- $X_0$  : la position latérale du véhicule sur la chaussée,
- $\Psi_0$  : l'orientation du véhicule sur la chaussée,
- $\alpha$  : l'inclinaison de la caméra,
- $C_h$  : la courbure latérale de la route,
- $C_l$  : la variation clothoïdale de la courbure.

**Les primitives images :** Les primitives images permettent de représenter au mieux les bords de la route dans l'image. Dans de nombreuses applications, les bords de la route sont représentés par des fonctions d'interpolation (polynômes d'ordre élevé, splines etc..). Dans cette approche, les bords de la route sont représentés par un ensemble de  $2(n - 1)$

segments (cf figure 4.11), dont les ordonnées de leurs extrémités sont fixes et définies une fois pour toutes lors de la phase d'apprentissage. Les primitives images sont alors décrites par le vecteur  $\underline{X}_d$  composé des abscisses des extrémités de chacun de ces segments.

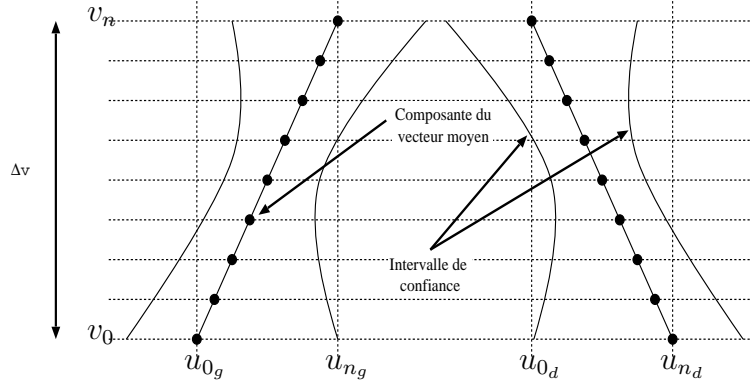


FIG. 4.11: Primitives images de l'algorithme de vision

$$\underline{X}_d = (u_{1_g}, \dots, u_{n_g}, u_{1_d}, \dots, u_{n_d})^T \quad (4.24)$$

où :

- $u_{1_g}, \dots, u_{n_g}$  représentent les abscisses du bord gauche de la route,
- $u_{1_d}, \dots, u_{n_d}$  représentent les abscisses du bord droit de la route.

Une matrice de covariance  $\mathbf{P}_d$  associée au vecteur  $\underline{X}_d$  est définie et permet de connaître les variations possibles de la position de chacun des segments dans l'image. Ces variations sont régies soit par la loi de distribution de probabilité  $\mathcal{N}(\underline{X}_{d_0}, \mathbf{P}_{d_0})$  définie par la phase d'apprentissage pour la première itération de l'algorithme, soit par la loi  $\mathcal{N}(\underline{X}_{d_{k|k-1}}, \mathbf{P}_{d_{k|k-1}})$  déduite de la phase de suivi pour les itérations suivantes. Cette loi permet, en outre, de définir les zones de recherche dans l'image et donc de diriger la détection des primitives.

**La phase d'apprentissage.** Comme décrit précédemment, le but de cette phase est d'apprendre la loi de distribution de probabilité  $\mathcal{N}(\underline{X}_{v_0}, \mathbf{P}_{v_0})$  régissant l'ensemble des paramètres du vecteur d'état. Cette loi doit être représentative de l'ensemble des scènes routières existantes et apprise une fois pour toutes. Elle est définie comme étant une fonction linéaire des paramètres de la loi  $\mathcal{N}(\underline{X}_{l_0}, \mathbf{P}_{l_0})$  et des paramètres intrinsèques et extrinsèques de la caméra sous l'hypothèse que la route est localement plane et que la courbure de la route peut être définie par un modèle clothoïdal.

$$\mathcal{N}(\underline{X}_{v_0}, \mathbf{P}_{v_0}) = f(e_u, e_v, Z_0, \mathcal{N}(\underline{X}_{l_0}, \mathbf{P}_{l_0})) \quad (4.25)$$

avec  $e_u = \frac{f}{d_u}$  et  $e_v = \frac{f}{d_v}$  les paramètres intrinsèques de la caméra numérique dont :

- $f$  est la focale de la caméra,
- $d_u$  et  $d_v$  la largeur et la hauteur d'un pixel de la matrice CCD.



et  $Z_0$  la hauteur de la caméra par rapport au sol.

Les calculs du vecteur  $\underline{X}_{v_0}$  et de la matrice de covariance  $\mathbf{P}_{v_0}$  sont présentés dans l'annexe A.

## B Localisation du véhicule

La localisation nécessite deux étapes : (1) l'algorithme doit rechercher l'ensemble des primitives  $\hat{\underline{X}}_{d_k|k}$  représentant a priori la route pour l'image  $k$ . (2) si cette étape est réussie, l'estimation des paramètres de localisation peut être effectuée à l'aide d'un filtre de Kalman sachant les dépendances entre les primitives images et les paramètres de localisation apprises au moment de l'apprentissage. Ces deux étapes sont décrites ci-dessous.

**(1) Reconnaissance des bords de voies dans l'image.** Cette phase repose sur l'exploration de l'espace des primitives images par un algorithme de type "branch and bound". La figure 4.12 présente l'organigramme de cette procédure de recherche. Par un souci de clarté, l'aspect "bracktracking" nécessaire à l'algorithme de "branch and bound" n'est pas décrit dans cet organigramme.

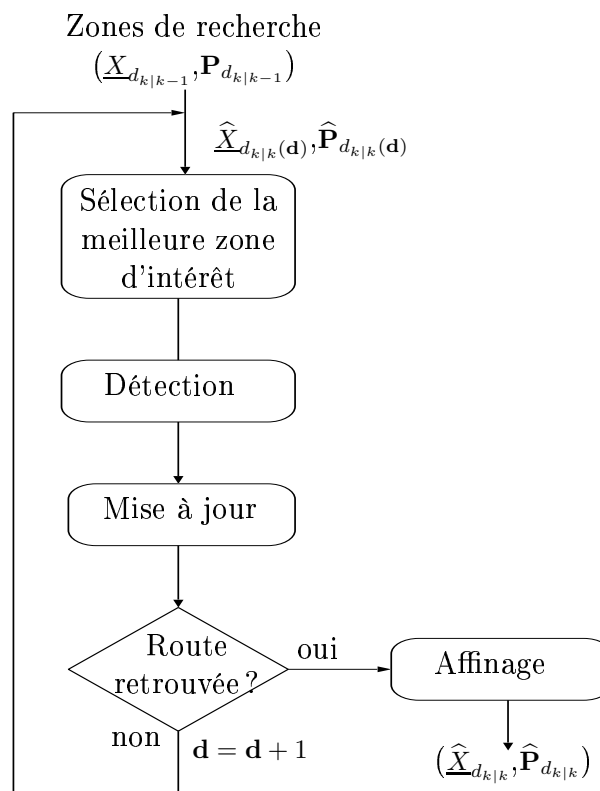


FIG. 4.12: Organigramme de la recherche des primitives images

Pour chaque profondeur d'analyse  $\mathbf{d}$  de l'arbre de recherche, la reconnaissance est constituée des trois étapes suivantes : d'abord l'algorithme recherche la zone d'intérêt la

mieux positionnée dans l'image, un segment censé représenter le bord de la voie de circulation dans cette zone est alors détecté et enfin le vecteur d'état  $\hat{\underline{X}}_{d_k|k}(\mathbf{d})$  des primitives images est mis à jour en fonction du segment précédemment détecté.

Pour chaque estimation de  $\hat{\underline{X}}_{d_k|k}(\mathbf{d})$ , un critère de reconnaissance de la route est calculé. Ce critère quantifie la probabilité que la route soit retrouvée conditionnellement aux segments détectés dans l'image et peut être écrit  $P(R_r|S_r, \mathbf{d})$ . Ainsi à chaque profondeur  $\mathbf{d}$  de l'arbre de recherche de la méthode "Branch and Bound", celui-ci peut être calculé de la manière suivante :

$$\begin{cases} P(R_r|S_r, \mathbf{d}) = P(R_r|S_r, \mathbf{d} - 1) + \frac{v_n - v_{n-1}}{\Delta_v} & \text{si le segment est détecté} \\ P(R_r|S_r, \mathbf{d}) = P(R_r|S_r, \mathbf{d} - 1) & \text{sinon.} \end{cases} \quad (4.26)$$

où  $\Delta_v$  est la hauteur totale de la zone d'analyse dans l'image et  $v_n$  et  $v_{n-1}$  sont les ordonnées de zone de recherche dans laquelle a été recherché le segment pour la profondeur  $\mathbf{d}$  de l'arbre de recherche.

Ainsi la route est retrouvée si  $P(R_r|S_r, \mathbf{d}) \geq R_{Th}$ , avec  $R_{Th} \in [0, 1]$  un seuil que l'utilisateur se fixe selon les diverses situations routières. Par exemple pour un environnement routier signalisé avec des marquages au sol,  $R_{Th}$  peut être fixé 0.4 alors que pour un environnement rural tel que des chemins il peut être fixé à 0.2.

Enfin la figure 4.13 présente un exemple simplifié d'évolution de l'algorithme de vision pour une image donnée.

**(2) Estimation de la position du véhicule.** Une fois la recherche des primitives dans l'image effectuée à l'instant  $k$ , le vecteur  $\hat{\underline{X}}_{d_k|k}$  ainsi que la matrice  $\hat{\mathbf{P}}_{d_k|k}$  sont à notre disposition. Une estimation du vecteur d'état  $\underline{X}_{v_k|k}$  et de la matrice de covariance  $\mathbf{P}_{v_k|k}$  est possible par l'intermédiaire d'un filtre de Kalman, ce filtre prenant en compte la loi  $\mathcal{N}(\underline{X}_{v_0}, \mathbf{P}_{v_0})$  déduite de l'apprentissage :

$$\begin{cases} \underline{X}_{v_k|k} = \underline{X}_{v_0} + \mathbf{K}_d[\hat{\underline{X}}_{d_k|k-1} - \mathbf{H}_d\underline{X}_{v_0}] \\ \mathbf{P}_{v_k|k} = \mathbf{P}_{v_0} - \mathbf{K}_d\mathbf{H}_d\mathbf{P}_{v_0} \end{cases} \quad (4.27)$$

avec

$$\begin{aligned} - \mathbf{K}_d &= \mathbf{P}_{v_0}\mathbf{H}_d^T \left[ \mathbf{H}_d\mathbf{P}_{v_0}\mathbf{H}_d^T + \hat{\mathbf{P}}_{d_k|k} \right]^{-1} \\ - \mathbf{H}_d &\text{ est tel que } \hat{\underline{X}}_d = \mathbf{H}_d\underline{X}_v + \underline{v}_d \text{ avec } \mathbf{E}[\underline{v}_d\underline{v}_d^T] = \hat{\mathbf{P}}_d \end{aligned}$$

Par conséquent une estimation de l'attitude du véhicule sur la chaussée est donnée par le vecteur  $\underline{X}_{l_k|k}$  inclus dans le vecteur  $\underline{X}_{v_k|k}$  et les incertitudes sur chacun des paramètres sont caractérisées par la matrice de covariance  $\mathbf{P}_{l_k|k}$  incluse dans la matrice  $\mathbf{P}_{v_k|k}$ .

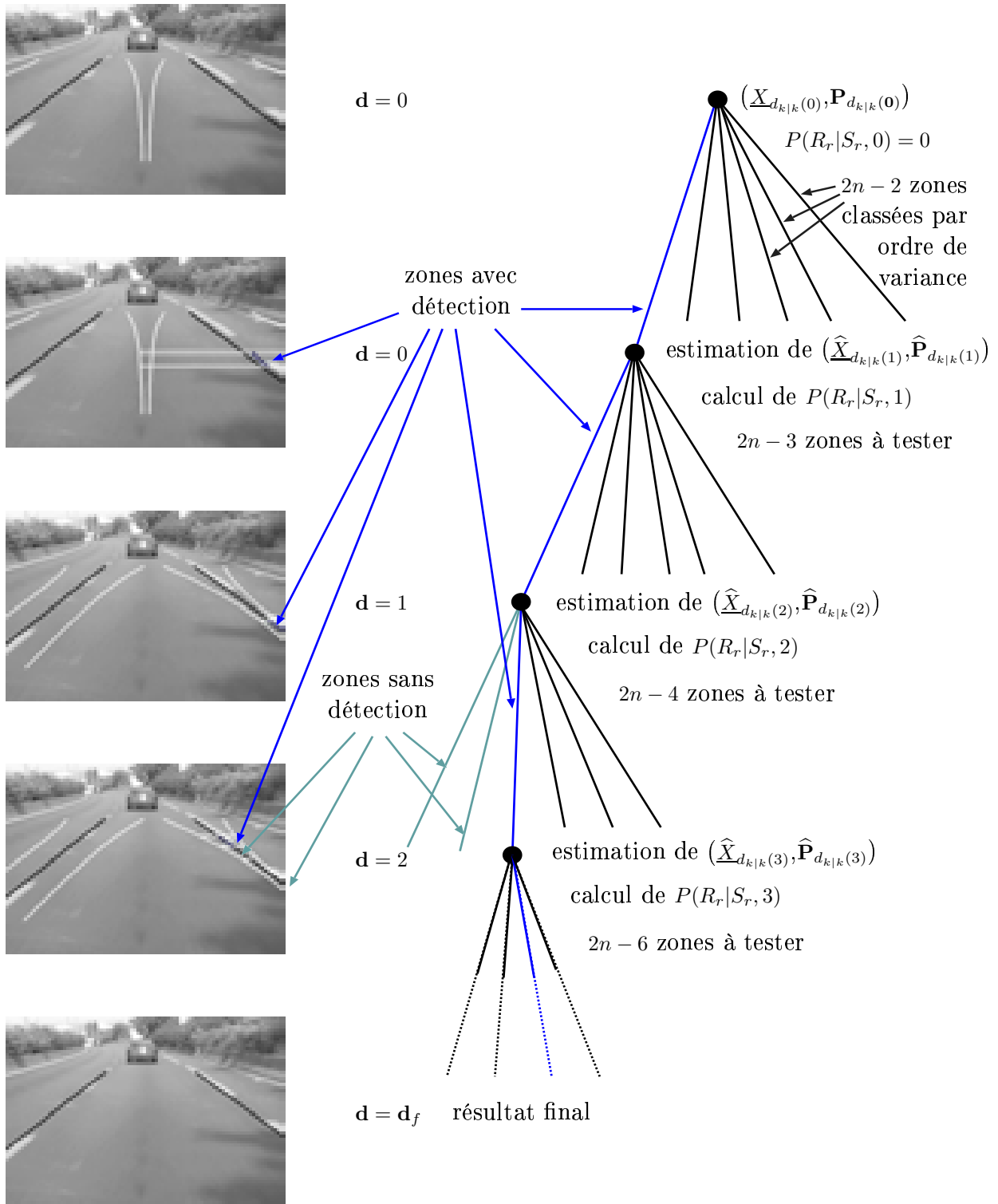


FIG. 4.13: Evolution de l'algorithme de vision sur une image

### C L'étape de prédiction

Avant de lancer une nouvelle reconnaissance des bords de voies à l'instant  $k + 1$ , une connaissance a priori des zones de recherche dans l'image est nécessaire pour guider la recherche. Cette étape va être décomposée en deux parties 1) l'évolution des paramètres de localisation selon les informations proprioceptives et l'évolution des paramètres géométriques de la route entre l'image  $k - 1$  et l'image  $k$ , 2) l'estimation de nouvelles zones de recherche en fonction des nouveaux paramètres de localisation et de géométrie de la route.

**(1) évolution des paramètres de localisation et de géométrie de route.** Entre deux images, il est possible de connaître le comportement cinématique du véhicule sur la chaussée en prenant en compte des informations proprioceptives fournies par l'odométrie et le gyromètre. Ainsi en utilisant le modèle cinématique char (paragraphe 1.1.1.1) il est possible de déterminer les nouvelles position et orientation du véhicule sur la route ainsi que les paramètres géométriques de la route à l'instant  $k$  par la relation suivante :

$$\underline{X}_{l_k|k-1} = f(\underline{X}_{l_{k-1}|k-1}, \Delta_D, w_{gyro}) \quad (4.28)$$

où

$$\begin{cases} L_{k|k-1} &= L_{k-1|k-1} + \epsilon_L \\ X_{0_k|k-1} &= X_{0_{k-1}|k-1} - \Delta_{D_k} \sin(\Psi_{0_{k-1}|k-1}) \\ \Psi_{0_k|k-1} &= \Psi_{0_{k-1}|k-1} + \Delta_T w_{gyro} \\ \alpha_{k|k-1} &= \alpha_{k-1|k-1} + \epsilon_\alpha \\ C_{h_k|k-1} &= C_{h_{k-1}|k-1} + \Delta_{D_k} C_{l_{k-1}|k-1} \\ C_{l_k|k-1} &= C_{l_{k-1}|k-1} + \epsilon_{C_l} \end{cases} \quad (4.29)$$

avec :

- $\Delta_D$  le déplacement longitudinal du véhicule entre les instants  $k$  et  $k - 1$ , cette information est fournie par l'odomètre,
- $w_{gyro}$  la vitesse de rotation du véhicule fournie par le gyromètre,
- $\Delta_T$  le temps écoulé entre l'image  $k - 1$  et l'image  $k$ ,
- $\epsilon_L, \epsilon_\alpha, \epsilon_{C_l}$  des bruits blancs gaussiens décrivant respectivement les variations possibles des paramètres  $L, \alpha$  et  $C_l$  entre deux images.

La nouvelle matrice de covariance  $\mathbf{P}_{l_k|k-1}$  associée au vecteur  $\underline{X}_{l_k|k-1}$  peut être établie de la manière suivante :

$$\mathbf{P}_{l_k|k-1} = \mathbf{J}_{f|X_l} \mathbf{P}_{l_{k-1}|k-1} \mathbf{J}_{f|X_l}^T + \mathbf{J}_{f|P} \mathbf{S}_p \mathbf{J}_{f|P}^T + \mathbf{M}_g \mathbf{Q}_g \mathbf{M}_g^T \quad (4.30)$$

avec la matrice de covariance  $\mathbf{S}_p$  représentant les incertitudes sur les données proprioceptives :

$$\mathbf{S}_p = \begin{pmatrix} \sigma_{\Delta_D}^2 & 0 \\ 0 & \sigma_{w_{gyro}}^2 \end{pmatrix} \quad (4.31)$$

la matrice de covariance  $\mathbf{Q}_g$  représentant les variations de paramètres géométriques de la route  $L$ ,  $\alpha$  et  $C_l$  :

$$\mathbf{Q}_p = \begin{pmatrix} \sigma_L^2 & 0 & 0 \\ 0 & \sigma_\alpha^2 & 0 \\ 0 & 0 & \sigma_{C_l} \end{pmatrix} \quad (4.32)$$

et la matrice  $\mathbf{M}_g$  :

$$\mathbf{M}_g = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.33)$$

Enfin les jacobiens  $\mathbf{J}_{f|X}$  et  $\mathbf{J}_{f|P}$ , gradients de la fonction d'évolution suivant les paramètres de localisation et de géométrie de la route et les informations proprioceptives sont donnés par :

$$\mathbf{J}_{f|X} = \frac{\partial f}{\partial \underline{X}_{k-1|k-1}} \quad \mathbf{J}_{f|P} = \frac{\partial f}{\partial (\Delta_D, w_{gyro})} \quad (4.34)$$

**(2) estimation des nouvelles zones de recherche.** Les paramètres a priori de localisation et de géométrie de la route sont maintenant connus. Connaissant la corrélation entre ces paramètres et les primitives images, il est possible d'estimer le vecteur d'état a priori  $\underline{X}_{v_k|k-1}$  et sa matrice de covariance associée  $\mathbf{P}_{v_k|k-1}$  en fonction du vecteur  $\underline{X}_{l_k|k-1}$  et la matrice de covariance  $\mathbf{P}_{l_k|k-1}$  sachant que le vecteur d'état est régi par la loi  $\mathcal{N}(\underline{X}_{v_0}, \mathbf{P}_{v_0})$ . Le filtre de Kalman suivant permet de réaliser cette estimation :

$$\begin{cases} \underline{X}_{v_k|k-1} &= \underline{X}_{v_0} + \mathbf{K}_l [\underline{X}_{l_k|k-1} - \mathbf{H}_l \underline{X}_{v_0}] \\ \mathbf{P}_{v_k|k-1} &= \mathbf{P}_{v_0} - \mathbf{K}_l \mathbf{H}_l \mathbf{P}_{v_0} \end{cases} \quad (4.35)$$

avec

$$\begin{aligned} - \mathbf{K}_l &= \mathbf{P}_{v_0} \mathbf{H}_l^T \left[ \mathbf{H}_l \mathbf{P}_{v_0} \mathbf{H}_l^T + \mathbf{P}_{l_k|k-1} \right]^{-1} \\ - \mathbf{H}_l &\text{ est tel que } \underline{X}_l = \mathbf{H}_l \underline{X}_v + \underline{v}_l \text{ avec } \mathbf{E}[\underline{v}_l \underline{v}_l^T] = \mathbf{P}_l \end{aligned}$$

Les nouvelles zones de recherche dans l'image à l'instant  $k$  sont définies par le vecteur  $\underline{X}_{d_k|k-1}$  et la matrice de covariance  $\mathbf{Q}_{d_k|k-1}$ , contenus dans le le vecteur d'état  $\underline{X}_{v_k|k-1}$  et sa matrice de covariance  $\mathbf{Q}_{v_k|k-1}$ .

#### 4.2.3.2 Les données "vision"

Lorsque l'algorithme de vision a réalisé la localisation du véhicule sur la chaussée, il fournit l'estimation d'un ensemble de paramètres représentant à la fois la position et l'orientation du véhicule sur la chaussée ainsi qu'une description géométrique de celle-ci. Dans la suite de ce paragraphe nous allons déterminer quels sont les paramètres les plus adéquats pour notre approche de localisation et donc définir ce que l'on nommera les données "vision".

- *la largeur de la route  $L$*  : nos travaux sont focalisés sur la localisation de véhicule dans un contexte routier péri-urbain où les routes sont en général de largeur connue et normalisée. Donc ce paramètre ne peut discriminer de manière efficace différentes sections de la carte du réseau routier et semble donc peu intéressant pour notre approche. En revanche si l'on envisageait d'effectuer la localisation d'un véhicule dans un environnement non structuré et si la carte était à même de fournir la largeur des routes, des chemins... alors ce paramètre deviendrait discriminant pour la localisation du véhicule et serait particulièrement intéressant à utiliser.
- *la courbure de la route  $C_h$*  : ce paramètre peut potentiellement permettre de sélectionner les sections de la carte où la courbure de la route est similaire à  $C_h$  et donc d'émettre une hypothèse sur la position du véhicule. Ce paramètre semble tout à fait adéquat pour notre approche, cependant notre système d'informations géographiques ne dispose pas d'une telle information, rendant donc ce paramètre inutilisable dans notre situation.
- *la courbe longitudinale de la route  $C_l$*  : en général aucune carte n'est à même de fournir une telle information, la notre ne manquant pas à la règle ce paramètre est lui aussi inutilisable.
- *l'angle de tangage de la caméra  $\alpha$*  : ce paramètre est nécessaire au bon fonctionnement de l'algorithme de vision mais ne fournit aucune information relative à la localisation du véhicule sur la route ainsi que sur sa géométrie. Ce paramètre peut donc être complètement ignoré.
- *les paramètres  $X_0$  et  $\Psi_0$*  sont des paramètres de localisation à part entière, représentant la position latérale ainsi que l'orientation du véhicule sur la chaussée. Ils sont indispensables pour espérer réaliser la localisation du véhicule sur la chaussée et seront donc utilisés dans notre approche.

En résumant les différents propos énoncés ci-avant, seuls les paramètres de localisation  $X_0$  et  $\Psi_0$  fournis par l'algorithme de vision sont a priori utilisables pour notre approche de localisation. Ainsi les données "vision" peuvent être définies par :

$$\underline{Y}_{vision} = \mathbf{M}_l \underline{X}_{l|k} = \begin{pmatrix} X_0 \\ \Psi_0 \end{pmatrix} + v_{vision} \quad (4.36)$$

avec

$$\mathbf{M}_l = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (4.37)$$

$$\underline{X}_l = (L, X_0, \Psi_0, \alpha, C_h, C_l)^T$$

### 4.2.3.3 Incertitudes sur les données "vision"

L'algorithme de "vision" fournit une estimation des incertitudes sur les paramètres de localisation du véhicule sur la chaussée ainsi que sur la géométrie de la route, ces incertitudes sont décrites par la loi normale  $\mathcal{N}(0, \mathbf{P}_{l_k|k})$ . Ainsi pour obtenir les incertitudes  $v_{vision}$  des données "vision", il suffit d'extraire de cette loi de probabilité, les parties concernant les paramètres  $X_0$  et  $\Psi_0$ . Ainsi celles-ci peuvent être formulées de la manière suivante :

$$v_{vision} \sim \mathcal{N}(0, \mathbf{R}_{vision}) \quad (4.38)$$

où

$$\mathbf{R}_{vision} = \mathbf{M}_l \mathbf{P}_{l_k|k} \mathbf{M}_l^T \quad (4.39)$$

### 4.2.3.4 Datation des données "vision"

Pour que la localisation soit la plus optimale possible, la correction de la position selon les données "vision" pour le filtre de Kalman ou pour le filtre particulaire devrait être obligatoirement réalisée lorsque la caméra fournit l'image à l'algorithme de vision. En effet cet algorithme ne peut effectuer instantanément une estimation des paramètres de localisation du véhicule sur la chaussée. Il fournit donc ces informations avec une latence correspondant au temps de traitement nécessaire à l'estimation de ces paramètres. Cette latence peut varier d'une centaine de millisecondes pour la première image traitée à une dizaine de millisecondes lorsque l'étape de suivi est réalisée. Donc, comme pour les données GPS, il est nécessaire de recalculer dans le temps les données "vision" sachant que la latence introduite par l'algorithme de vision peut être calculée de la manière suivante :

$$lat_v = t_{image} - t_{vision} \quad (4.40)$$

où  $t_{image}$  est la date à laquelle l'image a été fournie par la caméra au système informatique et  $t_{vision}$  la date à laquelle l'algorithme de vision a réalisé l'estimation des paramètres de localisation du véhicule sur la chaussée.

## 4.3 Prise en compte du temps

Les capteurs que l'on se propose d'utiliser dans notre application ont tous une fréquence d'acquisition différente, à savoir pour :

- le gyromètre : une fréquence d'acquisition de 10Hz,
- l'odomètre : une fréquence d'acquisition variable en fonction de la vitesse du véhicule,
- le GPS : une fréquence d'acquisition de 1Hz,
- le capteur "vision" : une fréquence qui dépend du temps nécessaire à l'estimation de la position du véhicule sur la chaussée, si celui-ci est inférieur à 33ms alors la cadence est de 30hz sinon elle est variable et dépend de ce temps.

De plus, deux de ces capteurs (le capteur "vision" et le GPS) fournissent leurs informations avec une latence, c'est-à-dire que leurs informations à l'instant  $t$  ne sont en fait valides qu'à l'instant  $t - \text{latence}$ . Ce phénomène représente un problème pour le filtrage car les méthodes de filtrage telles que le filtre de Kalman ou encore le filtre particulaire traitent les informations de manière séquentielle et en aucun cas il ne peuvent revenir dans le passé et donc prendre en compte la latence des informations.

Pour résoudre ce problème il est nécessaire d'ajouter un traitement de bas niveau qui classe les informations reçues par le système de fusion en fonction de leurs dates d'acquisition et de leurs latences pour que le filtrage puisse se dérouler séquentiellement. En réalité, ce traitement de bas niveau stocke le passé du filtre c'est-à-dire les états a priori et les états a posteriori estimés ainsi que toutes les informations permettant ces estimations. Puis, pour chaque nouvelle information reçue par le système de fusion, celle-ci est stockée et classée en fonction du temps. Ainsi les diverses étapes du filtrage peuvent être effectuées séquentiellement à partir de cette dernière information, les nouveaux états a priori et a posteriori ainsi obtenus écrasent les anciens.

Prenons l'exemple de la figure 4.14. Le filtre a estimé séquentiellement les états a priori en fonction des diverses informations proprioceptives jusqu'à l'instant  $t_{gps}$ . A cet instant, une information GPS est reçue par le système, mais celle-ci est en réalité valide pour l'instant  $t'_{gps} = t_{gps} - \text{lat}_g$ . Pour remettre à jour l'état du système il ne suffit plus d'effectuer l'étape de correction en fonction des données GPS, mais il faut tout d'abord les recaler en fonction de la latence du GPS, réinitialiser le filtre avec l'état a priori du système précédent l'instant  $t'_{gps}$  puis effectuer à nouveau séquentiellement les diverses étapes de prédiction et de correction du filtre à partir de cet état et en fonction des informations stockées entre  $t'_{gps}$  et  $t_{gps}$ . Les états stockés correspondant à chacune de ces informations sont alors remplacés par la nouvelle estimation. La figure 4.15 montre le même exemple auquel sont ajoutées des informations "vision", le principe reste exactement le même sauf que dans ce cas il est aussi nécessaire de recaler les informations "vision".

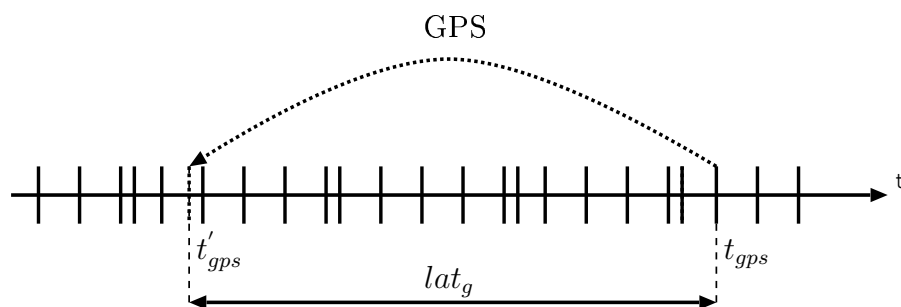


FIG. 4.14: Recalage des données GPS

De ces deux exemples peut être déduite une structure algorithmique sous-jacente au filtrage permettant de réaliser le traitement de bas niveau :



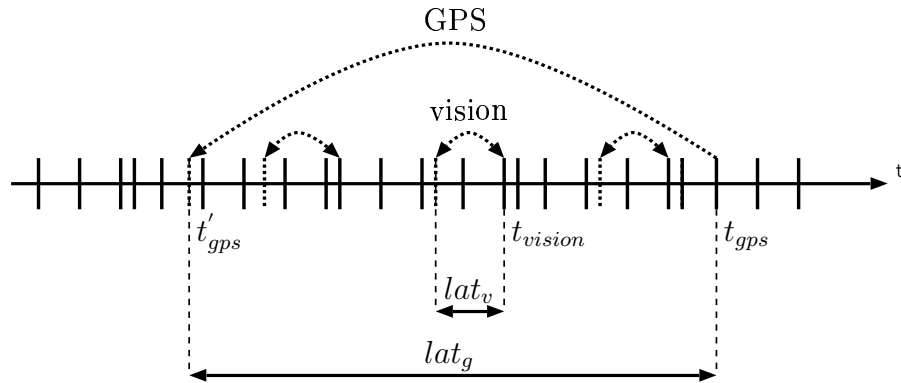


FIG. 4.15: Recalage des données GPS et "vision"

---

**Algorithme 4.1 : Structure algorithmique sous-jacente au filtrage**


---

- information reçue
    - si information proprioceptive :
      - effectuer l'étape de prédiction du filtre
    - si information extéroceptive :
      - recalcr l'information à l'instant  $k - i$  en fonction de la latence
      - effectuer les diverses étapes du filtre selon les informations  $k - i$  à  $k$
  - attente d'une nouvelle information
- 

Le problème de datation et donc de recalage dans le temps des observations peut être considéré comme un traitement de bas niveau de notre approche. Il ne sera énoncé dans aucun des algorithmes présentés dans la suite de ce manuscrit, cependant il faut bien garder en mémoire qu'avant chaque étape de filtrage de notre méthode, cet algorithme est effectué par le système informatique.

## 4.4 Conclusion

Le choix des capteurs est porté :

- pour les capteurs proprioceptifs : sur un odomètre et un gyromètre. Ici l'odomètre fournit des informations sur le déplacement curviligne du véhicule et le gyromètre fournit des informations sur la vitesse de rotation du véhicule. Une fois couplés à un modèle cinématique du véhicule, ces capteurs permettent de réaliser la localisation à l'estime du véhicule, et donc d'effectuer l'étape de prédiction de la méthode de filtrage.
- pour les capteurs extéroceptifs : sur un GPS bas coût et un capteur hybride couplant une caméra numérique, les capteurs proprioceptifs et un algorithme de vision. Ici le GPS fournit la position du véhicule dans le référentiel du monde et le capteur

hybride "vision" fournit la position et l'orientation du véhicule sur la chaussée. Ces dernières informations sont données dans un référentiel local à la route où circule le véhicule. Donc ces informations n'évoluent pas dans le même espace d'état que le système de localisation (position et orientation estimées dans le référentiel du monde) et ainsi lors de l'étape de correction du filtrage, une carte du réseau routier devra être nécessairement utilisée pour relier les informations "vision" au vecteur d'état.

Du point de vue de la caractérisation de ces capteurs et notamment de leurs incertitudes, deux capteurs posent un problème :

1. le gyromètre : en effet les incertitudes sur les informations fournies par le gyromètre dépendent de la température. N'ayant aucun moyen de caractériser convenablement cette dépendance, nous avons considéré que ces incertitudes peuvent être décrites par un processus de type "constante aléatoire" et un bruit blanc gaussien. Dans ce cas le processus constante aléatoire est appris avant chaque nouvelle expérimentation.
2. le GPS : les incertitudes sur la position  $(x_{gps}, y_{gps})^T$  fournie par le GPS ne sont ni blanches, ni gaussiennes comme le considèrent de nombreux travaux mais peuvent être décrites par des processus stochastiques qui varient très lentement dans le temps. Nous avons donc choisi de modéliser ces incertitudes par des processus auto-régressifs du premier ordre. Cette modélisation impose l'introduction de deux nouvelles variables  $b_x$  et  $b_y$  dans le vecteur d'état, ces deux variables en fin de compte représentent l'estimation des erreurs de position du GPS, celles-ci au même titre que les paramètres de localisation doivent être estimées et prédites (en fonction du modèle AR) à chaque étape du filtrage.

Enfin le phénomène de latence rencontré pour les informations extéroceptives ne peut être pris en compte par les méthodes de filtrage traditionnelles. Un traitement bas niveau des informations est alors nécessaire avant d'effectuer les différentes étapes du filtre. Ce traitement bas niveau se résume en trois étapes : 1) stocker les informations, 2) classer les informations en fonction de leurs dates d'acquisition et de leurs latences 3) effectuer séquentiellement les diverses étapes du filtre depuis la dernière information reçue.



# Chapitre 5

## Localisation avec filtre de Kalman

Ce chapitre présente notre approche de localisation lorsque la fusion de données est réalisée par un filtre de Kalman. Il peut être décomposé en deux parties, une première qui décrit en détails les diverses étapes du filtre de Kalman et une deuxième qui présente les résultats obtenus avec une telle approche.

La première partie décrit donc l'initialisation du filtre de Kalman. L'étape de prédiction et notamment l'évolution des paramètres de localisation en fonction des informations proprioceptives et du modèle cinématique char ainsi que l'évolution des paramètres de biais selon la modélisation autorégressive de la partie stochastique des informations GPS est tout d'abord présentée. Ensuite, les étapes de correction sont décrites, en particulier la mise à jour de l'état du système en fonction des données GPS où toute donnée GPS aberrante est éliminée grâce à la prise en compte de la modélisation de la partie stochastique des informations GPS. Enfin, est exposée la mise à jour en fonction des informations "vision" où la prise en compte de la carte du réseau routier permet à la fois de déterminer l'équation d'observation reliant les informations "vision" au vecteur d'état et de résoudre le problème de multimodalité rencontré avec de telles informations.

La deuxième partie présente, tout d'abord le comportement de cette approche en fonction de la disponibilité des capteurs extéroceptifs et la précision ainsi obtenue. Ensuite seront exposés la nécessité d'intégrer la modélisation de la partie stochastique des informations GPS au système de fusion et enfin les problèmes rencontrés avec le filtre Kalman lorsque les informations "vision" sont multimodales.

### 5.1 Vecteur d'état

#### 5.1.1 Description

Le vecteur d'état peut être décomposé en deux types de paramètres, tout d'abord les paramètres de position et d'orientation et les paramètres de biais des informations GPS déduits de la modélisation AR des incertitudes GPS. Le vecteur d'état utilisé dans notre

application est donc :

$$\underline{X} = (x, y, \theta, b_x, b_y, \dot{\theta})^T \quad (5.1)$$

avec :

- $x$  et  $y$  position du véhicule dans le référentiel Lambert93,
- $\theta$  orientation du véhicule dans ce même référentiel,
- $b_x$  et  $b_y$  biais sur les données GPS,
- $\dot{\theta}$  vitesse de rotation du véhicule.

Il faut noter que la vitesse de rotation  $\dot{\theta}$ , utile dans le modèle d'évolution du véhicule et donc nécessaire à l'étape de prédiction du filtre de Kalman, est en général directement fournie par le gyromètre. Cependant la mesure gyrométrique est bruitée, afin de réaliser un filtrage de cette information le paramètre  $\dot{\theta}$  est introduit dans le vecteur d'état. Ainsi à chaque donnée gyrométrique  $w_{gyro}$  ce n'est plus une étape de prédiction qui est réalisée mais une étape de correction en fonction de celle-ci.

### 5.1.2 Vecteur d'état initial et matrice de covariance associée

Le vecteur initial est défini lorsque la première information GPS est reçue par le système. Ainsi les paramètres de position sont  $x_0 = x_{gps_1}$  et  $y_0 = y_{gps_1}$ . Comme l'information GPS ne fournit aucune hypothèse sur les paramètres d'orientation et de biais, ceux-ci sont donc considérés nuls. Le vecteur initial est le suivant :

$$\underline{X}_{0|0} = (x_{gps_1}, y_{gps_1}, 0, 0, 0, 0)^T \quad (5.2)$$

Pour effectuer correctement l'initialisation du filtre de Kalman, il est nécessaire de définir la matrice de covariance  $P_{0|0}$  associée au vecteur initial et, avant tout, de définir les variances des paramètres initiaux :

1. Variances de  $b_{x_0}$  et  $b_{y_0}$  : dépendent des variations possibles des erreurs GPS et sont définis par les lois normales  $\mathcal{N}(0, \sigma_{x_{gps}})$  et  $\mathcal{N}(0, \sigma_{y_{gps}})$ , leurs variances sont :

$$var(b_{x_0}) = \sigma_{x_{gps}}^2 \quad \text{et} \quad var(b_{y_0}) = \sigma_{y_{gps}}^2 \quad (5.3)$$

2. Variances de  $x_0$  et  $y_0$  : dépendent de la modélisation de la partie stochastique des informations GPS. D'après l'équation (4.17) à l'instant  $k = 0$  la position du véhicule est décrite par :

$$\begin{aligned} x_0 &= x_{gps_1} + b_{x_0} \\ y_0 &= y_{gps_1} + b_{y_0} \end{aligned} \quad (5.4)$$

par conséquent les variances sur chacun de ces paramètres de position sont égales à la somme des variances des données GPS et des variances des biais, et sont :

$$var(x_0) = \sigma_{AR_x}^2 + \sigma_{x_{gps}}^2 \quad \text{et} \quad var(y_0) = \sigma_{AR_y}^2 + \sigma_{y_{gps}}^2 \quad (5.5)$$

3. Variances des paramètres d'orientation : la variance de l'orientation  $\theta$  dépend des variations possibles de ce paramètres dans l'espace d'état à couvrir lors de l'initialisation, à savoir l'ensemble des directions appartenant à l'intervalle  $[-\pi, \pi]$ . La

variance de la vitesse de rotation dépend des variations possibles des mesures du gyromètre à savoir  $\dot{\theta} \in [-1, 1]$  (section 4.1.3.2). Ainsi les variances sur ces paramètres peuvent être considérées comme :

$$\text{var}(\theta) = \pi^2 \quad \text{et} \quad \text{var}(\dot{\theta}) = 1 \quad (5.6)$$

Enfin en prenant en compte le fait que les paramètres sont décorrélés la matrice de covariance initiale est :

$$P_{0|0} = \begin{pmatrix} \sigma_{AR_x}^2 + \sigma_{x_{gps}}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{AR_y}^2 + \sigma_{y_{gps}}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \pi^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{x_{gps}}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{y_{gps}}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.7)$$

## 5.2 Etape de prédiction

Comme pour tout filtre de Kalman, la prédiction peut se décomposer en deux parties : la prédiction du vecteur d'état et la prédiction de sa matrice de covariance associée. Ces parties sont décrites dans les deux prochains paragraphes.

### 5.2.1 Prédiction du vecteur d'état

#### 5.2.1.1 Equation de prédiction

La prédiction de l'état est réalisée en prenant en compte le fait que :

- l'évolution des paramètres de localisation  $x$ ,  $y$  et  $\theta$  est réalisée par un modèle de type char présenté dans la section 1.1.1.1,
- les paramètres de biais  $b_x$  et  $b_y$  sont régis par des processus autorégressifs du premier ordre (cf section 4.2.2.2),
- la vitesse de rotation du véhicule  $\dot{\theta}$  est décrite par un processus de marche aléatoire, c'est-à-dire que pour chaque étape de prédiction, celui-ci n'évolue pas mais il est entaché d'un bruit centré blanc gaussien additif.

L'évolution du véhicule est alors décrite par la fonction :

$$\underline{X}_{k|k-1} = f(\underline{X}_{k-1|k-1}, \underbrace{\alpha_{x_k}, \beta_{x_k}, \alpha_{y_k}, \beta_{y_k}}_{\text{paramètres AR}}, \underbrace{\Delta_D, \epsilon_{b_x}, \epsilon_{b_y}, \epsilon_{\dot{\theta}}}_{\text{vecteur d'entrée } U_k}) \quad (5.8)$$

où

$$\begin{cases} x_{k|k-1} &= x_{k-1|k-1} - \Delta_D \sin(\theta_{k-1|k-1} + \dot{\theta}_{k-1|k-1} \Delta_T) \\ y_{k|k-1} &= y_{k-1|k-1} + \Delta_D \cos(\theta_{k-1|k-1} + \dot{\theta}_{k-1|k-1} \Delta_T) \\ \theta_{k|k-1} &= \theta_{k-1|k-1} + \dot{\theta}_{k-1|k-1} \Delta_T \\ b_{x_{k|k-1}} &= -\alpha_{x_k} b_{x_{k-1|k-1}} + \beta_{x_k} \epsilon_{b_x} \\ b_{y_{k|k-1}} &= -\alpha_{y_k} b_{y_{k-1|k-1}} + \beta_{y_k} \epsilon_{b_y} \\ \dot{\theta}_{k|k-1} &= \dot{\theta}_{k-1|k-1} + \epsilon_{\dot{\theta}} \end{cases} \quad (5.9)$$

avec :

- $\Delta_T$  : le temps écoulé entre les instant  $k - 1$  et  $k$ ,
- $\Delta_D$  : la distance parcourue entre les instants  $k-1$  et  $k$ , sachant que  $\Delta_D \sim \mathcal{N}(\bar{\Delta}_D, \sigma_{\Delta_D}^2)$ ,
- $\alpha_{x_k}$   $\alpha_{y_k}$   $\beta_{x_k}$   $\beta_{y_k}$  : les paramètres AR des processus stochastiques  $b_x$  et  $b_y$ . Ces paramètres doivent être recalculés à chaque instant,
- $\epsilon_{b_x}$  et  $\epsilon_{b_y}$  : les bruits gaussiens  $\mathcal{N}(0, 1)$  alimentant chacun des processus autorégressifs,
- $\epsilon_{\dot{\theta}}$  : le bruit gaussien  $\mathcal{N}(0, \sigma_{\dot{\theta}}^2)$  représentant les variations possibles de la vitesse angulaire du véhicule d'un instant à l'autre.

D'après la modélisation AR des incertitudes GPS (section 4.2.2.2), les paramètres  $\alpha_x$ ,  $\alpha_y$ ,  $\beta_x$  et  $\beta_y$  sont calculés une fois pour toute par la méthode de régression linéaire (dans notre cas, la méthode de Burg). Ainsi ces paramètres sont définis pour la fréquence d'échantillonnage du GPS, c'est-à-dire 1Hz. Or l'étape de prédiction du filtre de Kalman s'effectue lors de la réception des données odométriques, sa fréquence d'échantillonnage est alors dépendante de la vitesse du véhicule, variable dans le temps et donc incompatible avec celle du GPS. De ce fait, les paramètres  $\alpha_x$ ,  $\alpha_y$ ,  $\beta_x$  et  $\beta_y$  ne peuvent être directement utilisés dans les équations de prédiction et sont remplacés par les paramètres  $\alpha_{x_k}$ ,  $\alpha_{y_k}$ ,  $\beta_{x_k}$  et  $\beta_{y_k}$  tels que :

$$\alpha_{x/y_k} = f_{\alpha}(\Delta_T, \tau_{b_{x/y}}) \quad \text{et} \quad \beta_{x/y_k} = f_{\beta}(\Delta_T, \tau_{b_{x/y}}) \quad (5.10)$$

où  $\tau_{b_{x/y}}$  est la constante de temps des processus markoviens  $b_x$  et  $b_y$ .

Le prochain paragraphe présente la démarche nécessaire pour calculer ces différents paramètres à chaque étape de prédiction.

### 5.2.1.2 Calcul des paramètres $\alpha_{x_k}$ $\alpha_{y_k}$ $\beta_{x_k}$ $\beta_{y_k}$ des processus Markovien

Considérons les variables  $b_x$  et  $b_y$  régies par des processus AR du premier ordre. Il est possible d'établir les équations de récurrence suivantes pour ces deux variables :

$$b_{x_k} + \alpha_{1_x} b_{x_{k-1}} = \beta_{0_x} \epsilon_k \quad (5.11)$$

$$b_{y_k} + \alpha_{1_y} b_{y_{k-1}} = \beta_{0_y} \epsilon_k \quad (5.12)$$

Avec  $\alpha_{1_x}$ ,  $\alpha_{1_y}$ ,  $\beta_{0_x}$  et  $\beta_{0_y}$  les paramètres des modèles AR du premier ordre estimés par régression linéaire. Si nous appliquons la transformée en  $z$  en prenant en compte le théorème du retard, alors nous pouvons obtenir les fonctions de transfert caractérisant ces processus :

$$H_{b_x}(z) = \frac{\beta_{0_x}}{1 - \alpha_{1_x} z^{-1}} \quad (5.13)$$

$$H_{b_y}(z) = \frac{\beta_{0_y}}{1 - \alpha_{1_y} z^{-1}} \quad (5.14)$$

où  $\alpha_{1_x}$  et  $\alpha_{1_y}$  sont de la forme  $e^{-\frac{Te}{\tau}}$ .  $Te$  est la période d'échantillonnage et  $\tau$  la constante de temps du processus. Dans notre situation, les paramètres AR ont été estimés pour la fréquence d'échantillonnage du GPS et donc  $1Hz$ . Or l'étape de prédiction n'est pas échantillonnée selon la fréquence du GPS mais selon celle de l'odomètre. De plus, cette fréquence d'échantillonnage varie en fonction de la vitesse du véhicule, il est donc nécessaire à chaque nouvelle prédiction de recalculer les paramètres du modèle AR de la manière suivante :

- prenons la fonction de transfert en  $z$  d'un modèle AR sous la forme générale :

$$\frac{\beta}{1 - e^{-\frac{Te}{\tau}} z^{-1}} = \frac{\beta}{1 - e^{-\frac{Te}{\tau}}} \underbrace{\frac{1 - e^{-\frac{Te}{\tau}}}{1 - e^{-\frac{Te}{\tau}} z^{-1}}}_{\text{fonction de transfert normalisée}} \quad (5.15)$$

- changeons la période d'échantillonnage  $Te$  par la nouvelle période d'échantillonnage  $Te'$  dans la fonction de transfert normalisée, tout en conservant la constante de temps. La nouvelle fonction de transfert est alors :

$$H'(z) = \frac{\beta'}{1 - \alpha' z^{-1}} \quad (5.16)$$

avec :

$$\beta' = \frac{1 + e^{-\frac{Te'}{\tau}}}{1 + e^{-\frac{Te}{\tau}}} \quad \alpha' = -e^{-\frac{Te'}{\tau}} \quad (5.17)$$

Donc en considérant que  $Te'$  est équivalent à  $\Delta_T$ , le temps écoulé entre les instants  $k$  et  $k - 1$ , les paramètres  $\alpha_{x_k}$ ,  $\alpha_{y_k}$ ,  $\beta_{x_k}$  et  $\beta_{y_k}$  du modèle AR à chaque nouvelle prédiction du vecteur d'état (équations (5.9)) sont donnés par les relations suivantes :

$$\beta_{x_k} = \frac{1 + e^{-\frac{\Delta_T}{\tau_{bx}}}}{1 + e^{-\frac{1}{\tau_{bx}}}} \quad \alpha_{x_k} = -e^{-\frac{\Delta_T}{\tau_{bx}}} \quad (5.18)$$

$$\beta_{y_k} = \frac{1 + e^{-\frac{\Delta_T}{\tau_{by}}}}{1 + e^{-\frac{1}{\tau_{by}}}} \quad \alpha_{y_k} = -e^{-\frac{\Delta_T}{\tau_{by}}} \quad (5.19)$$

### 5.2.2 Prédiction de la covariance

La prédiction de la covariance est décrite par l'équation suivante :

$$\mathbf{P}_{k|k-1} = \mathbf{J}_e \mathbf{P}_{k-1|k-1} \mathbf{J}_e^T + \mathbf{J}_u \mathbf{S}_u \mathbf{J}_u^T \quad (5.20)$$

avec :



- le jacobien  $\mathbf{J}_e$ , constitué des dérivées partielles des équations de prédiction de l'état selon chacun des paramètres du vecteur d'état sera donné par :

$$\mathbf{J}_e = \begin{pmatrix} 1 & 0 & -\Delta_D \cos(\theta_{k|k-1}) & 0 & 0 & -\Delta_D \Delta_T \cos(\theta_{k|k-1}) \\ 0 & 1 & -\Delta_D \sin(\theta_{k|k-1}) & 0 & 0 & -\Delta_D \Delta_T \sin(\theta_{k|k-1}) \\ 0 & 0 & 1 & 0 & 0 & \Delta_T \\ 0 & 0 & 0 & -\alpha_{x_k} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\alpha_{y_k} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.21)$$

- la matrice covariance  $\mathbf{S}_u$  représentant les incertitudes sur les variables d'entrées de l'étape prédiction :

$$\mathbf{S}_u = \begin{pmatrix} \sigma_{\Delta_D}^2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \sigma_{\dot{\theta}}^2 \end{pmatrix} \quad (5.22)$$

- le jacobien  $\mathbf{J}_u$ , constitué des dérivées partielles des équations de prédiction de l'état selon chacun des paramètres du vecteur d'entrée :

$$\mathbf{J}_u = \begin{pmatrix} -\sin(\theta_{k|k-1}) & 0 & 0 & 0 \\ \cos(\theta_{k|k-1}) & 0 & 0 & 0 \\ 0 & \beta_{x_k} & 0 & 0 \\ 0 & 0 & \beta_{y_k} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.23)$$

### 5.3 Mise à jour par information gyromètre

Comme indiqué dans le paragraphe 5.1.1, les informations fournies par le gyromètre sont fortement bruitées. Pour limiter l'influence du bruit sur l'étape de prédiction du filtre de Kalman il est nécessaire au préalable de filtrer ces données. Le filtre de Kalman étant capable de remplir cette fonction, le paramètre  $\dot{\theta}$  représentant la vitesse de rotation du véhicule a été ajouté au vecteur d'état. Ainsi pour estimer et donc filtrer par la même occasion ce paramètre, il est nécessaire de remettre à jour le vecteur d'état et en particulier  $\dot{\theta}$  pour chaque mesure fournie par le gyromètre.

L'équation d'observation à l'instant  $k$ , reliant la vitesse angulaire de rotation du véhicule fournie par le gyromètre au vecteur d'état est décrite par l'équation suivante :

$$w_{gyro} = \mathbf{H}_{gyro} \underline{X}_{k|k} + v_{w_{gyro}} \quad (5.24)$$

avec :

- la matrice d'observation :

$$\mathbf{H}_{gyro} = (0 \ 0 \ 0 \ 0 \ 0 \ 1) \quad (5.25)$$

– l’incertitude sur la mesure, un bruit gaussien centré blanc :

$$v_{w_{gyro}} \sim \mathcal{N}(0, \sigma_{w_{gyro}}^2) \quad (5.26)$$

Alors les équations de mise à jour de l’état sont :

$$\begin{cases} \underline{X}_{k|k} &= \underline{X}_{k|k-1} + \mathbf{K}_{gyro}(w_{gyro} - \mathbf{H}_{gyro}\underline{X}_{k|k-1}) \\ \mathbf{P}_{k|k} &= (\mathbf{1} - \mathbf{K}_{gyro}\mathbf{H}_{gyro})\mathbf{P}_{k|k-1} \end{cases} \quad (5.27)$$

avec :

$$\mathbf{K}_{gyro} = \mathbf{P}_{k|k-1}\mathbf{H}_{gyro}^T[\mathbf{H}_{gyro}\mathbf{P}_{k|k-1}\mathbf{H}_{gyro}^T + \sigma_{w_{gyro}}^2]^{-1} \quad (5.28)$$

## 5.4 Mise à jour par informations GPS

La phase de mise à jour en fonction des données GPS peut être décomposée en deux étapes : une première étape où la cohérence des informations GPS avec l’état a priori est vérifiée, éliminant ainsi toutes les informations aberrantes dues aux occultations des satellites par les bâtiments, aux multi-trajets, ou encore au manque de satellites dans la constellation. Puis une deuxième étape où la mise à jour en fonction des ces informations ainsi filtrées est effectuée .

### 5.4.1 Cohérence des informations GPS

Avant de mettre à jour le vecteur d’état, il est nécessaire de vérifier la cohérence de la position fournie par le GPS avec la prédiction du filtre de Kalman. En effet, pour des raisons de trajets multiples ou de faible visibilité des satellites, le GPS peut fournir des informations aberrantes. Il est donc nécessaire de ne pas remettre à jour le vecteur d’état en fonction de ces informations. La vérification de cette cohérence peut être aisément réalisée par un test de similarité entre la position a priori GPS déduite du vecteur d’état a priori  $\underline{X}_{k|k-1}$  et la nouvelle donnée GPS (cf figure 5.1).

En reprenant la définition des informations GPS (cf équation (4.17)), on peut en déduire la position a priori  $\underline{Y}_{gps_{k|k-1}} = (x_{gps_{k|k-1}}, y_{gps_{k|k-1}})^T$  que devrait fournir le GPS :

$$\begin{cases} x_{gps_{k|k-1}} &= x_{k|k-1} + b_{x_{k|k-1}} \\ y_{gps_{k|k-1}} &= y_{k|k-1} + b_{y_{k|k-1}} \end{cases} \quad (5.29)$$

ainsi que son incertitude :

$$\mathbf{R}_{gps_{k|k-1}} = \mathbf{H}_{gps}\mathbf{P}_{k|k-1}\mathbf{H}_{gps}^T \quad (5.30)$$

avec

$$\mathbf{H}_{gps} \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix} \quad (5.31)$$

où  $\mathbf{H}_{gps}$  est la matrice d’observation reliant les données GPS au vecteur d’état, elle est aussi utilisée dans les équations de mise à jour du filtre de Kalman décrites dans le prochain paragraphe.

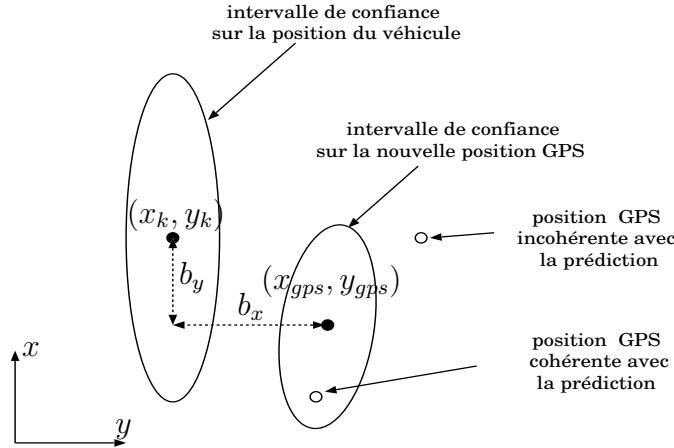


FIG. 5.1: Cohérence des informations GPS

Afin de vérifier la cohérence entre la position fournie par le GPS à l'instant  $k$  et sa position a priori  $\underline{Y}_{gps_k|k-1}$  déduite du vecteur d'état  $X_{k|k-1}$ , un test de similarité défini par le calcul de la distance de Mahalanobis suivant est réalisé :

$$d_{gps} = (\underline{Y}_{gps} - \underline{Y}_{gps_k|k-1})(\mathbf{R}_{gps_k|k-1} + \mathbf{R}'_{gps})^{-1}(\underline{Y}_{gps} - \underline{Y}_{gps_k|k-1})^T \quad (5.32)$$

Si  $d_{gps} > 1$  la nouvelle position GPS n'est pas dans l'intervalle de confiance décrit par l'ellipse définie par  $\underline{Y}_{gps_k|k-1}$  et  $\mathbf{R}_{gps_k|k-1}$ . Cette position est donc aberrante et le vecteur ne doit pas être remis à jour en fonction de cette observation. En revanche si  $d < 1$ , la nouvelle position est cohérente avec le vecteur d'état a priori et la mise à jour peut être effectuée.

### 5.4.2 Mise à jour

Une fois le test de cohérence effectué, il est possible de mettre à jour le nouvel état du véhicule en fonction des informations GPS. L'équation d'observation à l'instant  $k$  reliant les informations GPS au vecteur d'état est la suivante :

$$\underline{Y}_{gps} = \mathbf{H}_{gps}\underline{X}_{k|k} + v'_{gps} \quad (5.33)$$

avec :

- $\mathbf{H}_{gps}$  la matrice d'observation, cette matrice est définie dans l'équation (5.31) du paragraphe précédent.
- les incertitudes sur la mesure GPS sont régies par la loi normale :

$$v_{gps} \sim \mathcal{N}(0, \mathbf{R}'_{gps}) \quad (5.34)$$

en sachant que la matrice de covariance associée aux informations GPS est décrite par :

$$\mathbf{R}'_{gps} = \begin{pmatrix} \sigma_{AR_x}^2 & 0 \\ 0 & \sigma_{AR_y}^2 \end{pmatrix} \quad (5.35)$$

Alors les équations de mise à jour de l'état sont :

$$\begin{cases} \underline{X}_{k|k} &= \underline{X}_{k|k-1} + \mathbf{K}_{gps}(\underline{Y}_{gps} - \mathbf{H}_{gps}\underline{X}_{k|k-1}) \\ \mathbf{P}_{k|k} &= (1 - \mathbf{K}_{gps}\mathbf{H}_{gps})\mathbf{P}_{k|k-1} \end{cases} \quad (5.36)$$

avec :

$$\mathbf{K}_{gps} = \mathbf{P}_{k|k-1}\mathbf{H}_{gps}^T[\mathbf{H}_{gps}\mathbf{P}_{k|k-1}\mathbf{H}_{gps}^T + \mathbf{R}'_{gps}]^{-1} \quad (5.37)$$

## 5.5 Mise à jour par informations "vision"

Les informations "vision" à savoir la position latérale et l'orientation du véhicule sur la chaussée sont fournies pour une position précise du véhicule dans le réseau routier. Ainsi pour réaliser l'étape de correction du filtre de Kalman, il est nécessaire de déterminer l'équation d'observation en fonction des informations "vision" et des attributs de la carte du réseau routier. De plus, lorsque le réseau routier comporte plusieurs voies de circulation ces informations peuvent devenir potentiellement multimodales. Le filtre de Kalman ne fonctionnant pas avec des observations multimodales, il est important de choisir le mode le plus probable pour réaliser la mise à jour.

### 5.5.1 Fonction d'observation

Les informations fournies par le capteur "vision"  $\underline{Y}_{vision} = (X_0, \Psi_0)^t$  sont données dans un repère tangentiel à la route (figure 5.2). Avant de mettre à jour le vecteur d'état, il est nécessaire d'établir une relation entre l'attitude du véhicule et celui-ci. Cette relation doit décrire le passage du référentiel tangentiel à la route au référentiel du monde. Pour la déterminer, une carte précise du réseau routier va être utilisée. L'équation d'observation reliant les informations "vision" au vecteur d'état est ainsi déduite de cette relation.

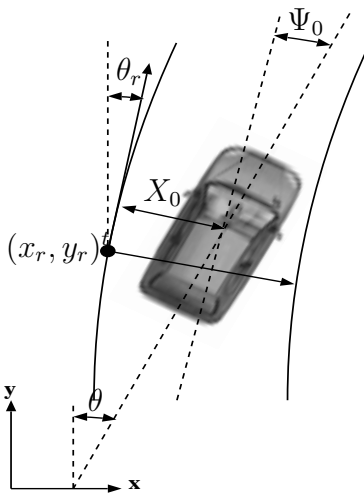


FIG. 5.2: Passage référentiel route au référentiel monde

La carte de l'environnement routier utilisée dans notre application a été créée à partir de relevés topographiques effectués par la D.D.E <sup>1</sup> du Puy-de-Dôme, et ne concerne qu'une partie du réseaux routier du département (Avenue Bingen dans le centre de Clermont Ferrand). Elle ne possède que très peu d'attributs par rapport aux cartes que l'on peut trouver dans le commerce (ex : Navtech), en revanche elle est très précise et fournit des attributs spécifiques à notre application comme la position et l'orientation du repère local tangentiel à la route où évolue le véhicule :

- $x_r$  et  $y_r$  l'origine du repère local de la route dans le référentiel du monde,
- $\theta_r$  l'orientation de ce même repère dans le monde.

Ainsi considérons l'attitude d'un véhicule  $(x, y, \theta)$  définie dans le référentiel du monde, son attitude  $(X_r, Y_r, \Psi_r)^T$  dans le référentiel de la route peut être alors traduite par les équations suivantes :

$$\begin{cases} X_r = \cos(\theta_r)(x - x_r) + \sin(\theta_r)(y - y_r) \\ Y_r = -\sin(\theta_r)(x - x_r) + \cos(\theta_r)(y - y_r) \\ \Psi_r = \theta - \theta_r \end{cases} \quad (5.38)$$

avec :

- $X_r$  la position latérale du véhicule sur la chaussée,
- $Y_r$  la position longitudinale du véhicule sur la chaussée,
- $\Psi_r$  l'orientation du véhicule.
- $x_r, y_r$  et  $\theta_r$  les paramètres fournis par la carte numérique pour une attitude donnée du véhicule  $(x, y, \theta)^t$ .

Comme le capteur "vision" ne fournit aucune information longitudinale, l'équation d'observation pour les informations "vision" est décrite par le jeu d'équations (5.38) où la relation concernant la position longitudinale  $Y_r$  est éliminée. Ainsi à l'instant  $k$ , l'équation d'observation est définie par la fonction non linéaire :

$$\underline{Y}_{vision} = h_{vision}(\underline{X}_{k|k}, x_r, y_r, \theta_r, v_{vision}) \quad (5.39)$$

où

$$h_{vision}(\underline{X}_{k|k}, x_r, y_r, \theta_r, 0) = \begin{cases} X0 = \cos(\theta_r)(x_{k|k} - x_r) + \sin(\theta_r)(y_{k|k} - y_r) \\ \Psi0 = \theta_{k|k-1} - \theta_r \end{cases} \quad (5.40)$$

Dans cette situation, l'équation d'observation est donc linéarisée autour du point de fonctionnement  $(X_0, 0, \Psi_0)$ . Cependant d'après l'état a priori il existe un ensemble de points de fonctionnement autour desquels cette observation pourrait être linéarisée (cf figure 5.4(a)). D'ailleurs ceux-ci peuvent être définis par la relation :

$$(X_i, Y_i, \Psi_i) = f(X_0, \psi_0, \text{géométrie de la route}) \in \mathcal{N}(X_{k|k-1}, P_{k|k-1}) \quad (5.41)$$

---

<sup>1</sup>Direction Départementale de l'Equipement

C'est-à-dire l'ensemble des positions incluses dans l'intervalle de confiance défini par l'état a priori sachant que la position latérale et l'orientation du véhicule sur la chaussée sont respectivement  $X_0$  et  $\Psi_0$ . Cependant, de telles hypothèses ne peuvent être gérées par le filtre de Kalman et donc, pour tenir compte des erreurs de linéarisation de l'équation d'observation, il est nécessaire d'augmenter artificiellement les incertitudes sur les informations "vision".

### 5.5.2 Incertitudes des informations "vision"

Pour tenir compte de erreurs de linéarisation du filtre de Kalman, les incertitudes des informations "vision" vont être artificiellement augmentées en prenant en compte toutes les attitudes possibles du véhicule sur la chaussée au vu de l'état a priori. En réalité toutes ces attitudes sont rassemblées sur une trajectoire parallèle et translatée de  $X_0$  par rapport au bord gauche de la route. Comme le capteur "vision" peut fournir une approximation de la géométrie de la route (figure 5.3), il est possible de déterminer cette trajectoire et donc de définir l'ensemble de ces attitudes.

Ainsi en considérant que localement la route peut être décrite comme un cercle de rayon  $R = 1/C_h$ , alors pour une ordonnée  $Y_{r_j}$  connue dans le référentiel de la route, l'abscisse du bord gauche de la route est  $X_{r_j} \approx \frac{C_h}{2} Y_{r_j}^2$ , l'orientation locale de la route en ce point est  $\Psi_{r_j} \approx C_h Y_{r_j}$  et les attitudes possibles du véhicule sur la chaussée sont définies par :

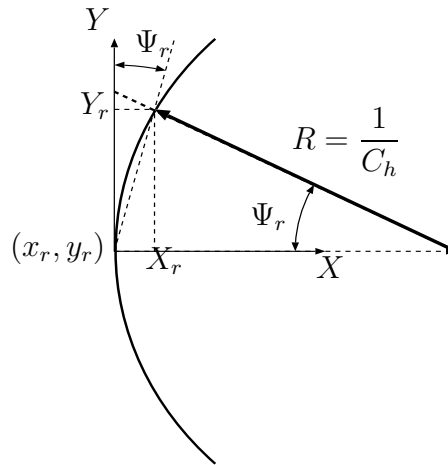


FIG. 5.3: Modélisation locale de la route

$$\begin{cases} X_{r_j} = \frac{C_h}{2} Y_{r_j}^2 + X_0 \\ \Psi_{r_j} = C_h Y_{r_j} + \Psi_0 \end{cases} \quad Y_{r_j} \in [-\sqrt{\sigma_{Y_r}^2}, \sqrt{\sigma_{Y_r}^2}] \quad (5.42)$$

Où  $\sigma_{Y_r}^2$  est définie en fonction de la matrice de covariance associée à l'état a priori et de l'orientation du repère de la route :

$$\sigma_{Y_r}^2 = \mathbf{M}_{P/Y_r} \mathbf{P}_{k|k-1} \mathbf{M}_{P/Y_r}^T \quad (5.43)$$

avec  $\mathbf{M}_{P/Y_r}$  est le jacobien composé des dérivées partielles selon les paramètres de l'état de l'équation régissant  $Y_r$  (cf équations (5.38)) :

$$\mathbf{M}_{Q/Y_r} = (\cos(\theta) \quad \sin(\theta) \quad 0 \quad 0 \quad 0 \quad 0) \quad (5.44)$$

Il est alors possible de caractériser les incertitudes supplémentaires pour les informations "vision" en calculant la variance de toutes ces attitudes. Ainsi les nouvelles incertitudes des informations "vision" sont décrites par la matrice de covariance  $\mathbf{R}'_{vision}$  telle que :

$$\mathbf{R}'_{vision} = \frac{1}{N} \sum_{j=1}^N \underbrace{\begin{pmatrix} X_{r_j} - X_r \\ \Psi_{r_j} - \Psi_r \end{pmatrix} \begin{pmatrix} X_{r_j} - X_r \\ \Psi_{r_j} - \Psi_r \end{pmatrix}^T}_{\text{covariance additionnelle}} + \mathbf{R}_{vision} \quad (5.45)$$

Ainsi comme le montre la figure 5.4(c), l'observation "vision" bien qu'elle soit linéarisée autour du point de fonctionnement  $(X_0, 0, \Psi_0)$  englobe toutes les attitudes possibles du véhicule sur la chaussée et permet au filtre de Kalman de converger vers une solution viable.

### 5.5.3 Gestion de la multimodalité

Jusqu'à maintenant nous avons toujours fait l'hypothèse que la route où le véhicule évolue comporte une seule voie de circulation. Pourtant ce n'est pas le cas dans des environnements routiers tels que les autoroutes où les voies express qui comportent au minimum 2 voies de circulation. Dans cette situation, l'algorithme de vision fournit une information qui potentiellement peut se situer sur n'importe laquelle des voies de circulation, ainsi l'observation devient multimodale. Le filtre de Kalman ne sachant pas gérer la multimodalité, il est nécessaire de déterminer qu'elle est l'attitude la plus probable du véhicule dans le réseau routier avant d'effectuer la mise à jour.

Cette situation est notamment rencontrée lors des premières itérations de l'algorithme, où les incertitudes sur l'état sont grandes et la véritable position du véhicule peut appartenir à plusieurs voies de circulation (figure 5.5(a)). On peut être aussi confronté à ce problème lors des changement de voie où l'état estimé ne permet pas de déterminer sur quelle voie de circulation se situe véritablement le véhicule (figure 5.5(b)).

Pour réaliser ce choix, les  $N$  attitudes possibles du véhicule selon les  $N$  voies de circulation sont comparées avec la prédiction de l'observation, l'attitude la plus proche du point de vue statique de l'observation a priori est retenue pour mettre à jour le vecteur d'état. La distance de Mahalanobis suivante permet d'effectuer cette comparaison :

$$d_i = (\underline{Y}_{vision} - \underline{Y}_{k|k-1})(\mathbf{R}_{vision} + \mathbf{P}_{Y_{k|k-1}})(\underline{Y}_{vision} - \hat{\underline{Y}}_{k|k-1})^T \quad i = 1 \dots N \quad (5.46)$$

où l'observation a priori est :

$$\underline{Y}_{k|k-1} = h_{vision}(\underline{X}_{k|k-1}, x_r, y_r, \theta_r, 0) \quad (5.47)$$

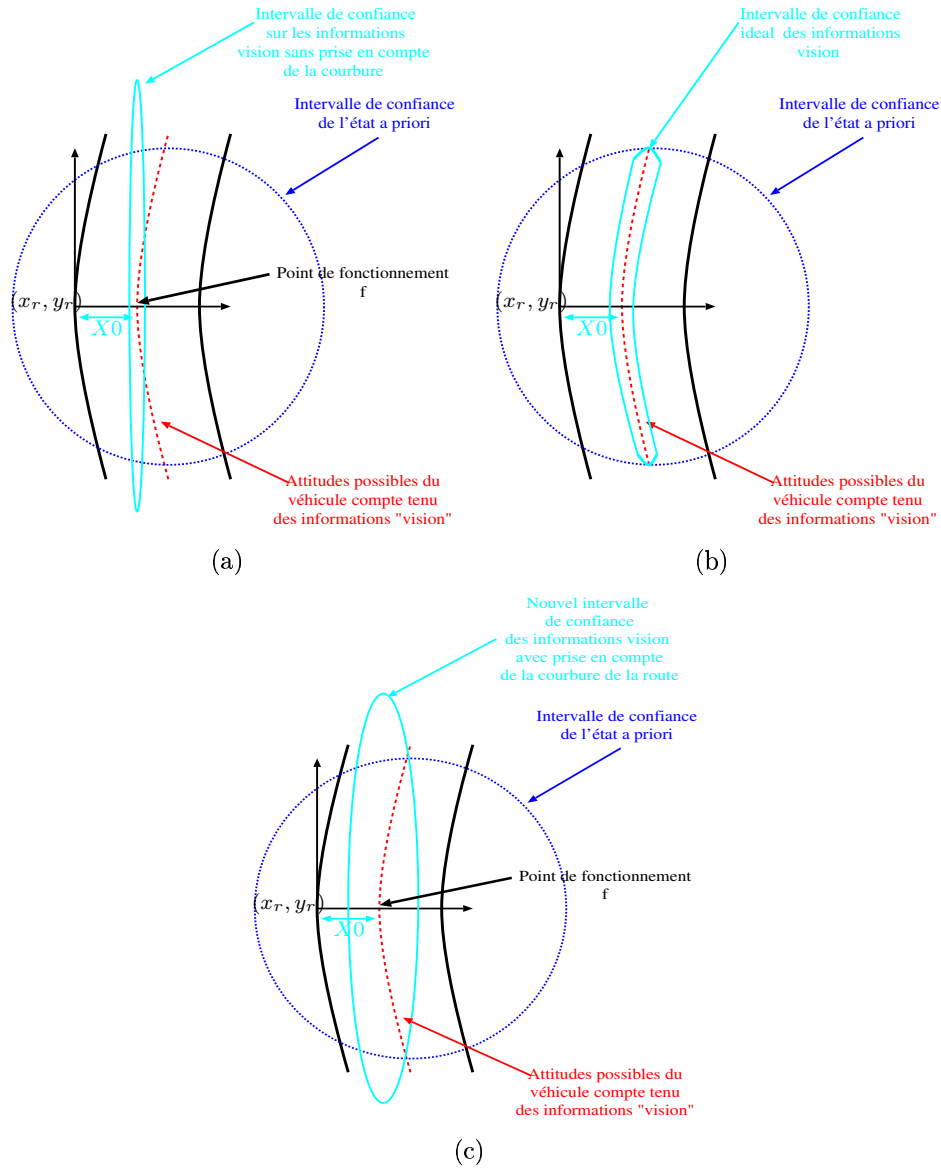


FIG. 5.4: Observations "vision" : (a) cas où la linéarisation s'effectue au point de fonctionnement  $(X_0, 0, \Psi_0)$ , (b) cas idéal, (c) cas où les incertitudes "vision" sont augmentées artificiellement

et sa matrice de covariance associée est :

$$\mathbf{R}_{Y_{k|k-1}} = \mathbf{H}_{vision} \mathbf{P}_{k|k-1} \mathbf{H}_{vision}^T \quad (5.48)$$

avec la matrice  $\mathbf{H}_{vision}$  est le Jacobien composé des dérivées partielles selon les paramètres du vecteur d'état de l'équation d'observation :

$$\mathbf{H}_{vision} = \begin{pmatrix} \cos(\theta) & \sin(\theta) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (5.49)$$



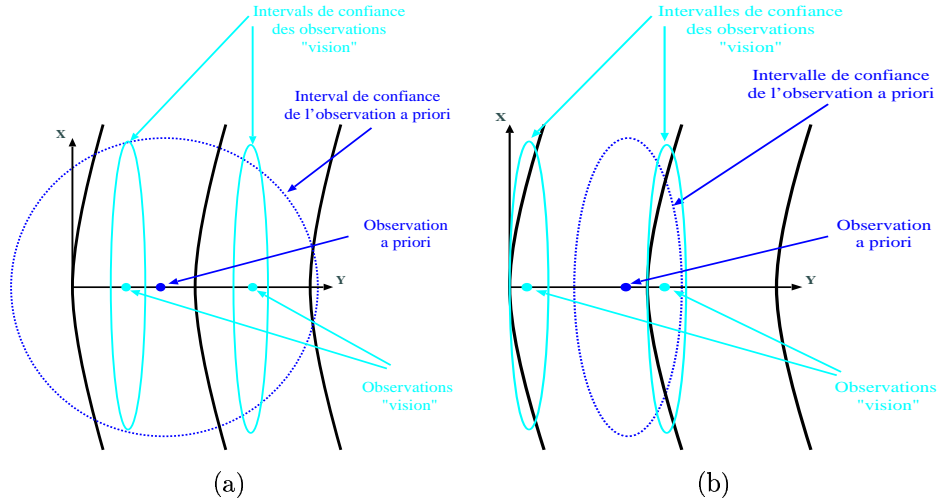


FIG. 5.5: Mise à jour en fonction d'informations "vision" multimodales, figure (a) état a priori, figure (b) état a posteriori

#### 5.5.4 Mise à jour

En tenant compte de l'ensemble des remarques présentées dans les chapitres précédents, la mise à jour en fonction des informations "vision" peut être définie par l'équation d'observation suivante :

$$\underline{Y}_{vision} = h_{vision}(\underline{X}_{k|k}, x_r, y_r, \theta_r, v'_{vision}) \quad (5.50)$$

avec  $v'_{vision}$ , le vecteur d'incertitudes sur la mesure "vision" régi par la loi normale :

$$v'_{vision} \sim \mathcal{N}(0, \mathbf{R}'_{vision}) \quad (5.51)$$

Alors les équations de mise à jour de l'état sont :

$$\begin{aligned} \underline{X}_{k|k} &= \underline{X}_{k|k-1} + \mathbf{K}_{vision}(\underline{Y}_{vision} - h_{vision}(\underline{X}_{k|k-1}, x_r, y_r, \theta_r, 0)) \\ \mathbf{P}_{k|k} &= (1 - \mathbf{K}_{vision}\mathbf{H}_{vision})\mathbf{P}_{k|k-1} \end{aligned} \quad (5.52)$$

avec

$$\mathbf{K}_{dgps} = \mathbf{P}_{k|k-1}\mathbf{H}_{vision}^T[\mathbf{H}_{vision}\mathbf{P}_{k|k-1}\mathbf{H}_{vision}^T + \mathbf{R}'_{vision}]^{-1}$$

## 5.6 Résultats

L'ensemble de nos expérimentations ont été réalisées dans un environnement péri-urbain proche du centre ville de Clermont-Ferrand ayant pour particularité d'être doté de différents contextes routiers :  $2 \times 2$  voies,  $1 \times 1$  voie, rond point et d'avoir été cartographié de manière très précise par la DDE (cf figure 5.6).

Les résultats présentés dans cette section sont décomposés en trois parties : une première partie où est montrée l'influence de la disponibilité des capteurs extéroceptifs (GPS et capteur "vision") sur la localisation du véhicule et ceci dans les trois contextes différents :

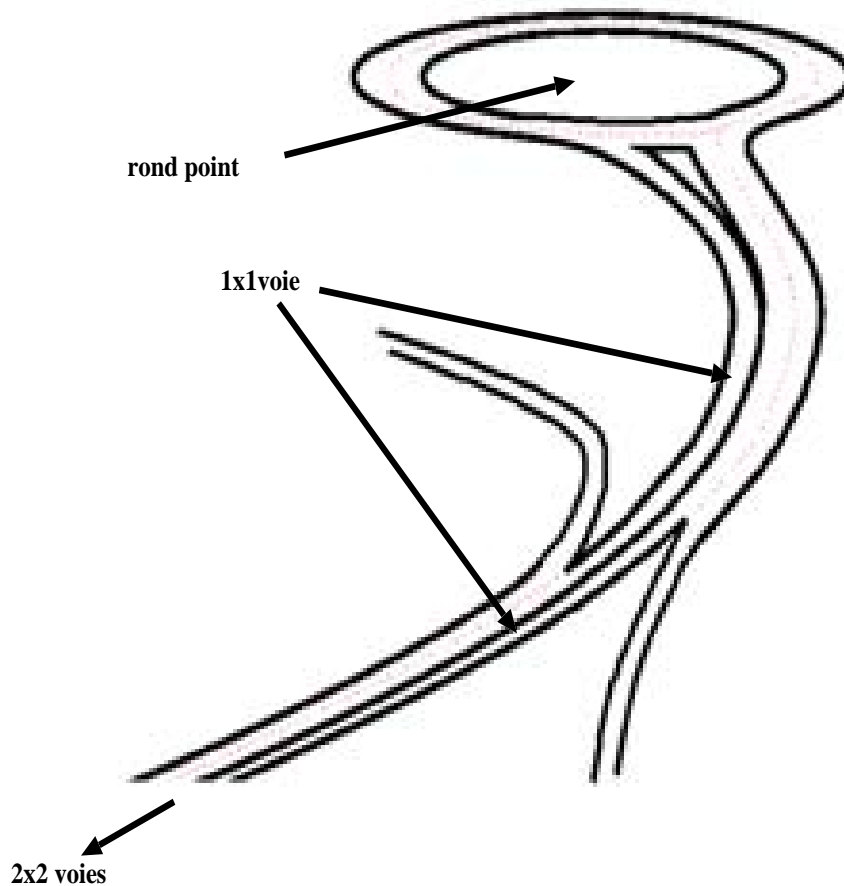


FIG. 5.6: Carte de l'avenue Bingen à Clermont-Ferrand

1. le GPS et le capteur "vision" sont disponibles,
2. seul le GPS est disponible,
3. le GPS et le capteur "vision" subissent des pertes de données.

Dans une seconde partie il est démontré que la modélisation AR des incertitudes GPS est absolument nécessaire pour réaliser correctement la localisation du véhicule. Et enfin, une troisième et dernière partie, les effets de la multimodalité des informations "vision" sur la localisation et en particulier lors de changement de voies sont présentés.

Pour une meilleure compréhension des résultats, ceux-ci seront fournis dans le référentiel de la route, on parlera alors de position latérale et de position longitudinale du véhicule sur la chaussée. La vérité terrain sera quant à elle fournie par un GPS différentiel Omnistar (système GPS augmenté par satellites) et les erreurs de localisation seront alors définies par :

$$\begin{aligned} \text{erreur latérale} &= X_r - X_{dgps_r} \\ \text{erreur longitudinale} &= Y_r - Y_{dgps_r} \end{aligned} \quad (5.53)$$

avec  $(X_{dgps_r}, Y_{dgps_r})^T$  la position du DGPS dans le référentiel de la route.

### 5.6.1 Localisation selon la disponibilité des capteurs

Afin de pouvoir confronter le processus de la localisation selon la disponibilité des capteurs extéroceptifs, l'ensemble des résultats présentés dans la suite de cette section ont été obtenus à partir de données enregistrées lors d'une seule expérimentation. La disponibilité des capteurs est alors en réalité simulée selon le contexte désiré.

Ainsi pour les trois différents contextes choisis, le trajet du véhicule est toujours le même :

- de  $t = 0$  à  $3s$  le véhicule démarre en dehors de la zone cartographiée,
- de  $t = 3$  à  $140s$  le véhicule évolue sur la  $2 \times 2$  voies puis sur la  $1 \times 1$  voies pour arriver enfin au rond point,
- pour  $t > 140s$  le véhicule fait le tour du rond point puis emprunte la  $2 \times 2$  voies dans l'autre sens de circulation.

#### 5.6.1.1 L'ensemble des capteurs extéroceptifs sont disponibles

La figure 5.7 présente les erreurs de position du véhicule sur la chaussée dans le cas où tous les capteurs extéroceptifs sont disponibles :

- de  $t = 0$  à  $3s$  : les erreurs sont considérées nulles car le véhicule évolue en dehors de la zone cartographiée.
- de  $t = 3$  à  $12s$  : l'algorithme de "vision" considère que la route n'est pas retrouvée, les informations du capteur "vision" ne sont donc pas prises en compte par la fusion de données. La localisation s'effectue alors en fonction du GPS et des capteurs proprioceptifs. Les erreurs latérales et longitudinales sont alors importantes mais inférieures à  $\pm GDO P_{gps} \times EURE_{gps}$ .
- de  $t = 12$  à  $38s$  : le capteur "vision" fournit des informations valides, lorsque la première information "vision" est prise en compte par le système, l'erreur latérale tend brutalement vers 0. Comme les informations "vision" n'émettent aucune hypothèse sur la position longitudinale du véhicule sur la chaussée, celle-ci n'est pas mise à jour et les erreurs longitudinales ne varient pas. En revanche, au fur et à mesure du temps, le recoupement des informations par le filtre de Kalman permet d'estimer la position longitudinale du véhicule et donc de faire tendre l'erreur longitudinale vers 0.
- de  $t = 38$  à  $140s$  : le système a donc convergé vers une solution proche de la véritable solution et aucun changement significatif n'affecte alors la localisation du véhicule.

On notera que lorsque tous les capteurs extéroceptifs sont disponibles, le système converge vers une solution a priori correcte au bout de quelques dizaines de secondes

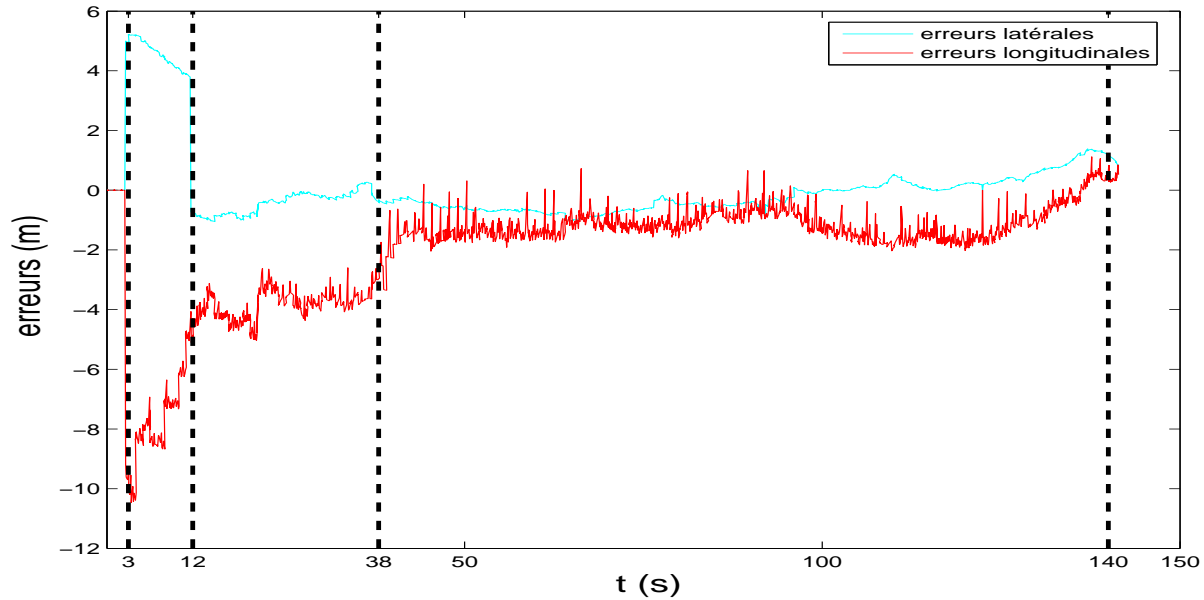


FIG. 5.7: Erreurs latérales et longitudinales

(dans notre cas  $t_c \approx 38 - 12 \approx 26s$ ). L'estimation de la position du véhicule est alors relativement précise comme le montre le tableau suivant :

	Moyenne	Ecart-type
erreurs latérales (m)	0.066	0.287
erreurs longitudinales (m)	-1.018	0.388

TAB. 5.1: Moyennes et écart-type des erreurs de localisation

La figure 5.8 montre l'évolution du véhicule dans le rond point c'est-à-dire pour  $t > 140s$ . Dans cette situation, le rayon de la route est très petit, l'algorithme de "vision" se retrouve en dehors de ses limites de fonctionnement et ne peut estimer la position et l'orientation du véhicule sur la chaussée. Le capteur "vision" est alors dans l'incapacité de fournir des informations au système de fusion et la localisation est réalisée en fonction des informations provenant des capteurs proprioceptifs et du GPS. D'après les fonctions d'autocorrélation des erreurs GPS (cf section 4.2.2.2), celles-ci évoluent très lentement dans le temps ce qui impose qu'à court terme les mises à jour selon les informations GPS ont peu d'influence sur la localisation et celle-ci peut être pratiquement considérée comme une localisation à l'estime. Ainsi au fur et à mesure que le véhicule évolue dans le rond point, les erreurs proprioceptives s'accumulent et la position estimée s'éloigne de la position fournie par le DGPS. Quand le véhicule ressort du rond point, la position estimée se trouve en dehors des voies de circulation mettant en avant le problème de divergence de la localisation à l'estime. Il faut attendre de nouvelles informations "vision" pour que la position estimée tende de nouveau vers la véritable position.

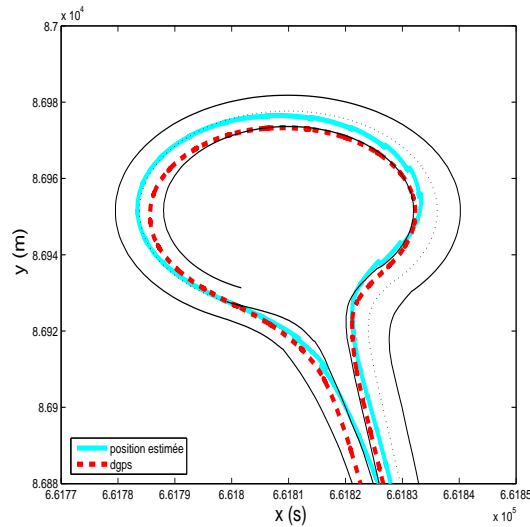


FIG. 5.8: Evolution du véhicule dans le rond point

### 5.6.1.2 Le capteur "vision" est indisponible

La figure 5.9 présente les résultats de localisation lorsque le capteur "vision" n'est pas pris en compte par le système. De  $t = 0$  à  $12s$ , les résultats sont similaires à ceux de la section précédente. Au delà de  $t = 12s$  les erreurs restent importantes car il n'y a aucune information extéroceptive permettant d'apprendre les paramètres de biais  $b_x$  et  $b_y$  et donc de converger vers la véritable position du véhicule. Enfin ces erreurs sont obligatoirement comprises entre  $\pm EURE_{gps} \times GDOP_{gps}$ .

La figure 5.10 montre l'évolution du véhicule dans le rond point. La trajectoire estimée semble similaire à la trajectoire présentée dans la figure 5.8 (lorsque tous les capteurs sont disponibles) mais celle-ci est biaisée selon les axes  $x$  et  $y$  car la position initiale dans le rond point n'est pas estimée correctement. Ainsi pour réaliser une localisation correcte du véhicule, il est obligatoire que des informations extéroceptives autres que celles fournies par le GPS soient prises en compte par le processus de fusion. D'où l'utilisation des informations "vision" dans notre approche.

### 5.6.1.3 Le capteur "vision" et le GPS subissent des pertes d'informations

Dans les deux prochains paragraphes, va être montrée l'influence de pertes d'informations extéroceptives sur la localisation. Ainsi nous allons simuler dans un premier temps la perte d'informations GPS, puis dans un deuxième temps la perte d'informations "vision" pendant un laps de temps allant de  $t \approx 38s$  à  $t \approx 100s$ .

**Perte d'informations GPS** La figure 5.12 présente les résultats de localisation lorsque le système de fusion subit des pertes d'informations GPS :

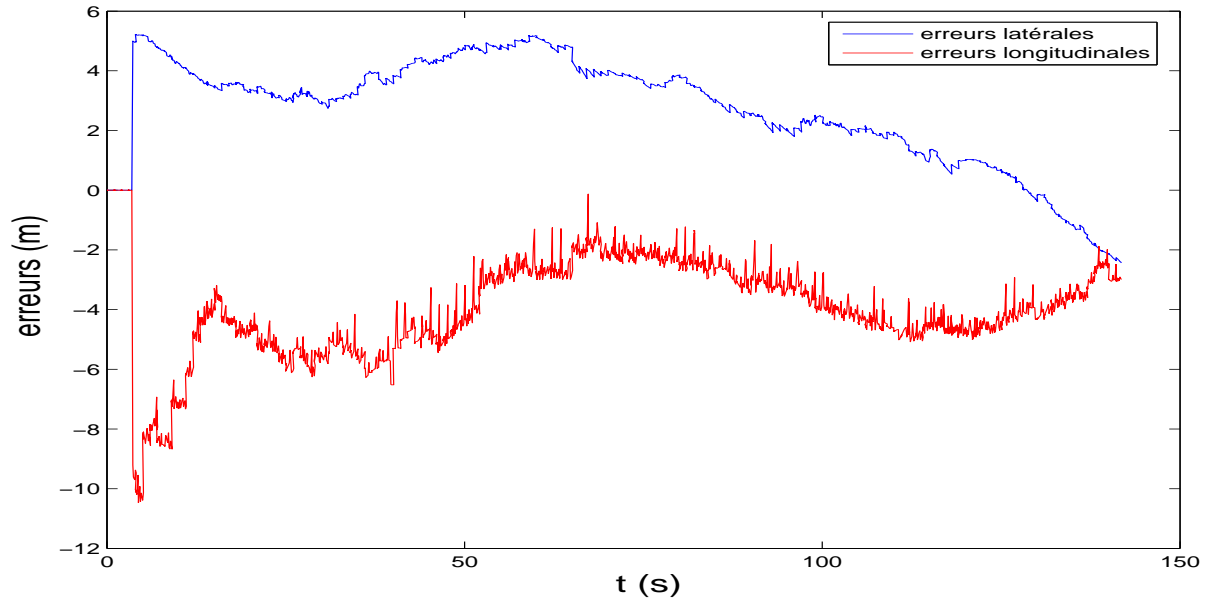


FIG. 5.9: Erreurs latérales et longitudinales : "vision" indisponible

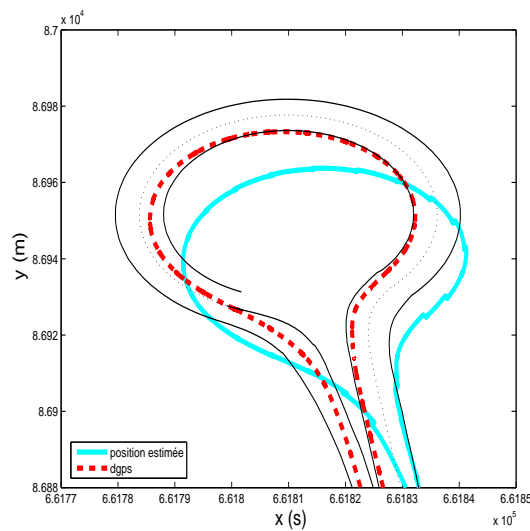


FIG. 5.10: Evolution du véhicule dans le rond point : vision indisponible

- $t = 0$  à  $38s$  les résultats sont similaires aux résultats présentés dans la section 5.6.1.1, ainsi le filtre a quasi convergé vers la vraie solution, l'erreur latérale est alors quasi nulle et l'erreur longitudinale est inférieure à  $2m$ .
- $t = 38$  à  $100s$  : une perte d'informations GPS est simulée, la localisation est alors réalisée en fonction des informations proprioceptives et "vision". Comme les informations "vision" ne fournissent pas d'hypothèse sur la position longitudinale, l'estimation de celle-ci dépend principalement de l'étape de prédiction mais aussi du recouplement des états a priori successifs et des observations "vision" ainsi :

1. si le véhicule évolue dans une ligne droite, le recouplement des états a priori avec les observations "vision" ne permet pas d'estimer la position longitudinale (cf figure 5.11(a)) et donc l'accumulation des erreurs de prédiction peut faire diverger son estimation. Ce phénomène est visible dans les résultats de  $t \approx 38s$  à  $t \approx 75s$ .
  2. au contraire si le véhicule évolue sur route sinueuse le recouplement des états a priori et des observations "vision" permet d'estimer la position longitudinale (cf figure 5.11(b)) et donc de converger vers la vraie solution. Ce phénomène est visible dans les résultats entre  $t \approx 75s$  et  $t \approx 100s$ .
- après  $t = 100s$  le système reconverge de nouveau vers la véritable solution et les résultats redeviennent similaires à ceux de la section 5.6.1.1.

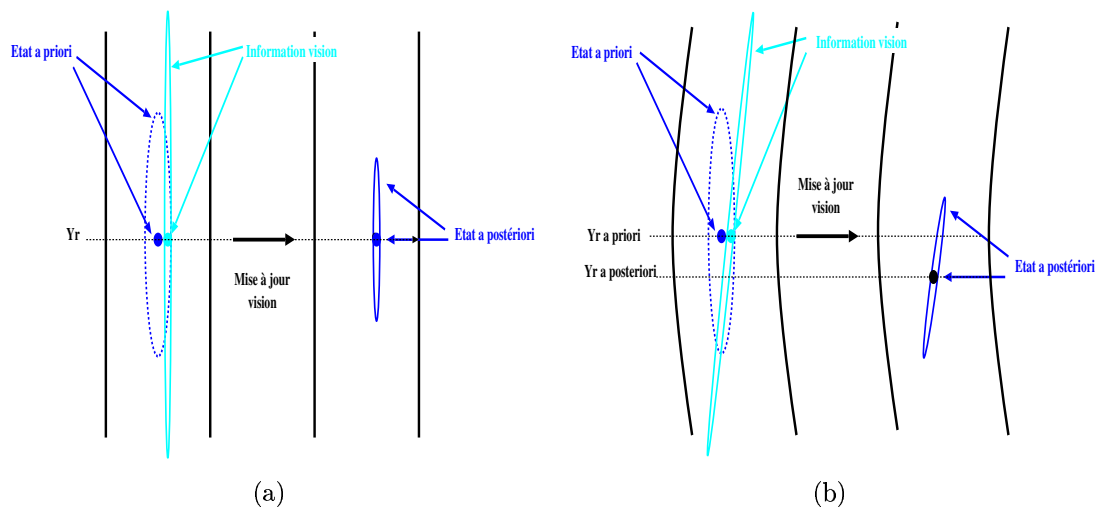


FIG. 5.11: Recouplement des informations "vision" avec l'état a priori : (a) dans une ligne droite (b) dans un virage

**Perte d'informations "vision"** La figure 5.13 présente les résultats de la localisation lors de pertes d'informations "vision" :

- de  $t = 0$  à  $38s$  : les résultats sont similaires à ceux de la section 5.6.1.1,
- de  $t = 38$  à  $100s$  : une perte d'informations "vision" est simulée, la localisation est alors réalisée en fonction des informations proprioceptives et du GPS. A l'instant  $t = 38s$ , le système a convergé vers une solution proche de la réalité et a donc pu estimer la position du véhicule ainsi que les paramètres de biais  $b_x$  et  $b_y$ . Pendant la perte d'informations "vision", ces deux derniers paramètres sont impossibles à estimer et évoluent selon la modélisation AR des incertitudes GPS. Donc à court terme, pour chaque mise à jour de l'état en fonction des informations GPS, les estimées sont pratiquement biaisées de la même façon et les informations GPS ont donc peu d'influence sur la localisation du véhicule. Les erreurs latérales et longitudinales dépendent alors en grande partie des erreurs introduites par l'étape de prédiction

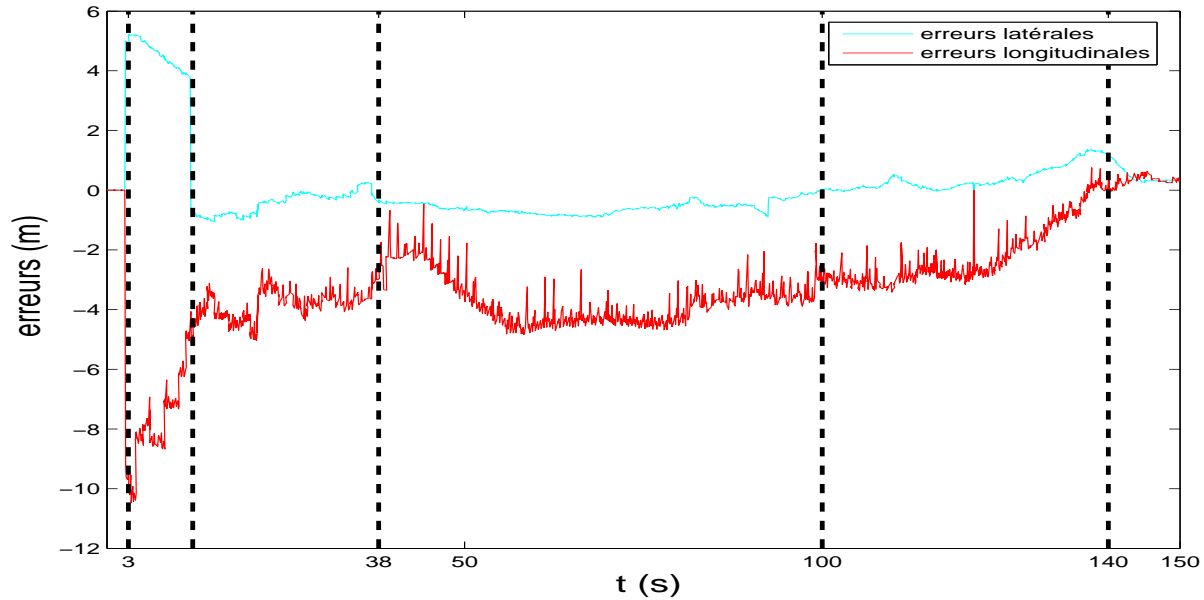


FIG. 5.12: Erreurs latérales et longitudinales : pertes d'informations GPS

et donc des incertitudes des capteurs proprioceptifs. Comme le véhicule évolue sur une portion de route comportant que des virages à grand rayon de courbure et des lignes droites, l'étape de prédiction n'introduit que peu d'erreurs dans l'estimée et les erreurs latérales et longitudinales évoluent très peu.

- après  $t = 100s$  le système reconverge de nouveau vers la véritable solution et les résultats redeviennent similaires à ceux de la section 5.6.1.1.

Si le capteur "vision" ne devait plus fournir d'informations pendant de nombreuses minutes (neige, brouillard etc...), l'estimation des paramètres de biais  $b_x$  et  $b_y$  divergerait dans le temps ainsi que l'estimée de la position du véhicule. De même si le véhicule évoluait sur des portions de routes sinueuses, la prédiction serait moins précise et pourrait introduire des erreurs importantes dans l'estimée. C'est par exemple le cas d'un rond point (cf section 5.7).

De par ces résultats, on peut noter que le système est plus robuste envers les pertes d'informations "vision" qu'envers les pertes d'informations GPS. Ceci est attribué au fait que les informations "vision" ne disposent pas de toutes les données nécessaires à une localisation absolue (position longitudinale du véhicule manquante). Pour pallier ce problème, il est nécessaire d'intégrer dans le processus de fusion des informations complémentaires à celles du capteur "vision". Par exemple on pourrait envisager de détecter d'autres éléments remarquables situés dans l'environnement du véhicule tels que : les panneaux de signalisation, les feux de circulations, passages cloutés et de se servir de leur position relative par rapport au véhicule pour améliorer sa localisation. Bien sûr, dans ce cas le processus de localisation (notamment l'étape de mise en correspondance) nécessite la connaissance de la position de ces nouveaux éléments remarquables dans l'environnement et impose donc



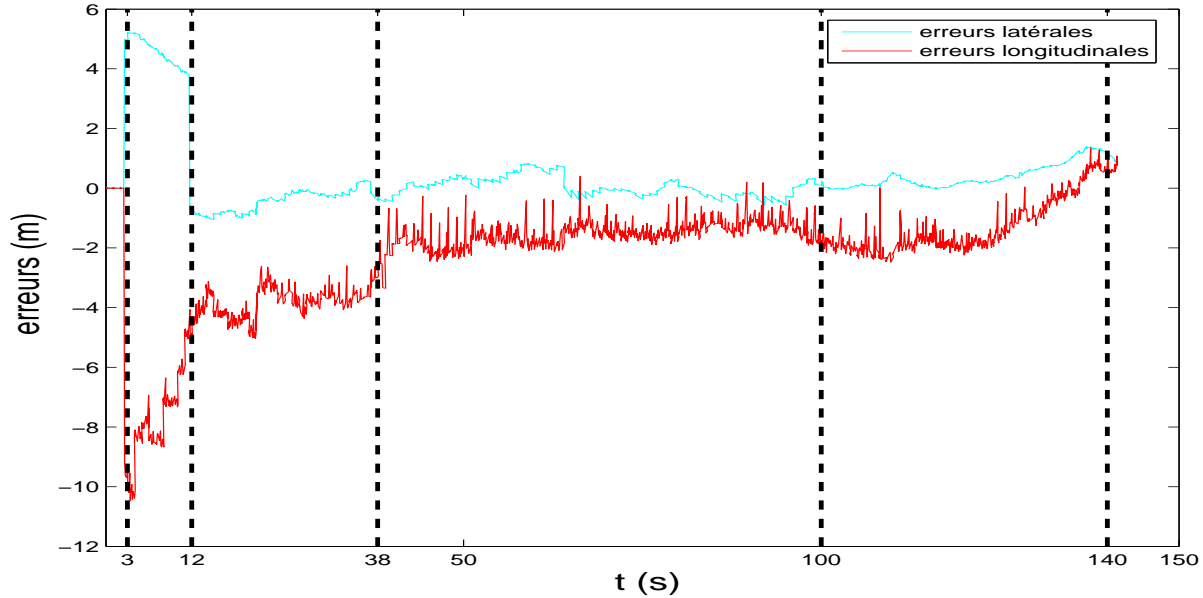


FIG. 5.13: Erreurs latérales et longitudinales : pertes d'informations "vision"

un enrichissement de sa carte.

## 5.6.2 Localisation sans la modélisation AR des incertitudes GPS

Dans cette section, sont présentés les résultats de la localisation lorsque la modélisation de la partie stochastique des informations GPS n'est pas prise en compte dans le processus de fusion. Pour ceci, il suffit de modifier l'équation d'observation reliant les informations GPS à l'état de la manière suivante :

$$\underline{Y}_{gps} = H_{gps} \underline{X}_{k|k} + v_{gps} \quad (5.54)$$

où

$$H_{gps} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (5.55)$$

La figure 5.14 montre les résultats d'un tel système de localisation :

- de  $t = 0$  à  $12s$  : aucune information "vision" n'est valide, les erreurs sont importantes et dépendent directement du fait que les informations GPS sont biaisées.
- après  $t = 12s$  : le capteur vision fournit des informations, dans cette situation les erreurs latérales tendent très rapidement vers 0 et les erreurs longitudinales subissent d'importantes variations dans le temps montrant que le système ne peut converger vers la vraie solution. En effet comme la vision n'apporte aucune information longitudinale et que le système de fusion n'apprend pas les biais des informations GPS, celui-ci est incapable d'estimer correctement la position longitudinale. Les erreurs longitudinales dépendent alors en grande partie des biais sur les informations GPS. Ce comportement nous a donc poussé à rechercher une nouvelle modélisation des

informations GPS pour améliorer la convergence du filtre (cf section 4.2.2.2).

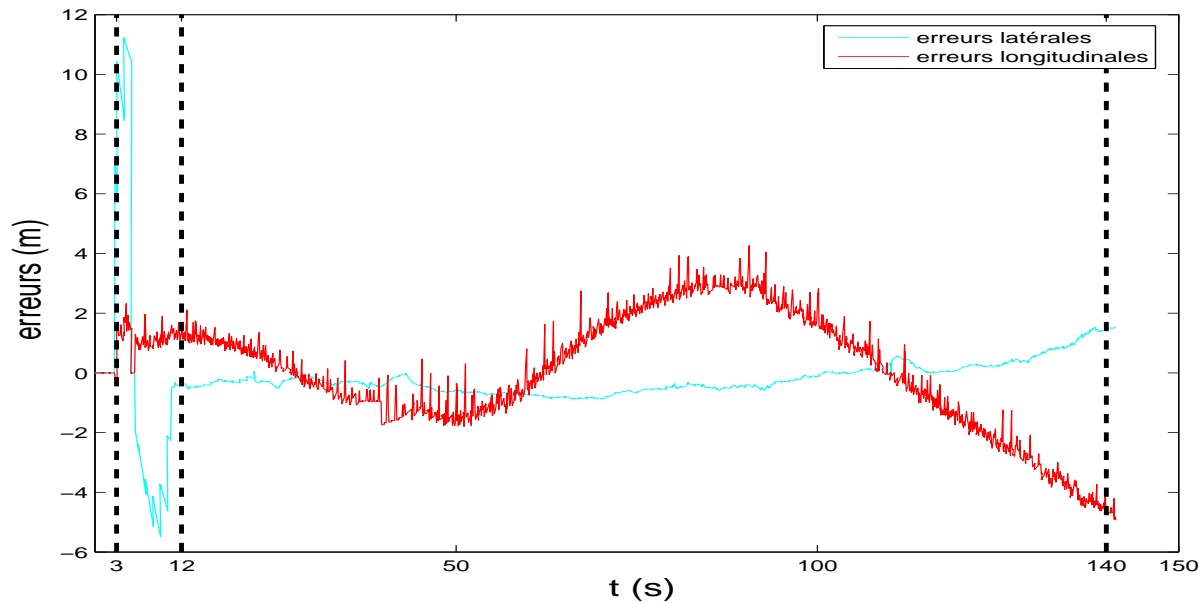


FIG. 5.14: Erreurs latérales et longitudinales : sans modélisation AR des incertitudes GPS

La figure 5.15 montre l'évolution du véhicule dans le rond point. Comme à l'entrée du rond la position longitudinale n'est pas correctement estimée (erreurs longitudinales  $\approx -4m$ ). Le véhicule effectue une trajectoire similaire à celle du DGPS mais biaisée en grande partie par la mauvaise estimation de la position longitudinale.

En comparant ces résultats et les résultats de la section 5.6.1.1, on s'aperçoit qu'il est absolument nécessaire d'intégrer la modélisation de la partie stochastique des informations GPS dans le processus de fusion pour obtenir une localisation correcte du véhicule.

### 5.6.3 Multimodalité et changement de voie

Dans cette section, sont présentés les effets de la multimodalité des informations "vision" sur le comportement du système de localisation. Notamment dans le choix du mode des observations "vision" lorsque la première information "vision" est prise en compte par la fusion et des problèmes qui peuvent être rencontrés lors des changements de voies.

Les figures 5.16(a), 5.16(c) et 5.16(e) montrent les résultats de localisation lorsque le système de fusion choisit le bon mode des informations "vision". D'après la figure 5.16(a), comme la position estimée avant la première information "vision" se situe à droite des voies de circulation, lors de la mise à jour en fonction des informations "vision" le système choisit le mode situé sur la voie de droite et donc converge vers la vraie position. Le premier changement de voie (figure 5.16(c)) s'effectue correctement et d'après la figure

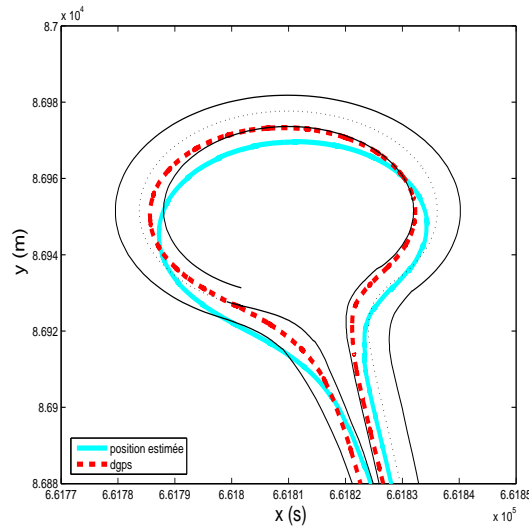


FIG. 5.15: Evolution du véhicule dans le rond point : sans modélisation AR des incertitudes GPS

5.16(e), les erreurs longitudinales diminuent améliorant même la localisation du véhicule.

Les figures 5.16(b), 5.16(d) et 5.16(f) montrent les résultats de localisation lorsque le système de fusion choisit le mauvais mode des informations "vision". D'après la figure 5.16(b), la position estimée avant la première information "vision" se situe sur la mauvaise voie de circulation. Lors de la mise à jour en fonction des informations "vision", le système choisit le mode appartenant à la voie de gauche. Dans ce cas le système converge vers une position erronée et biaisé de 3.5m en latéral. Le premier changement de voie de circulation (figure 5.16(d)) ne peut alors s'effectuer correctement et contrairement à la situation précédente le filtre de Kalman diverge.

Les erreurs longitudinales passent alors de -2m à +8m (figure 5.16(f)). En effet, lors du changement de voie, l'estimée doit varier brusquement de 3.5m selon l'axe latéral de la route. Or d'après les équations de prédiction aucun paramètre ne peut varier d'une telle manière d'un instant à l'autre, on rappelle que :

- les variations des paramètres de position dépendent des incertitudes sur des informations proprioceptives qui sont relativement faibles,
- les variations des paramètres de biais dépendent de la modélisation AR des incertitudes GPS qui, on le sait, ont une variation très lente dans le temps.

Donc l'état a priori est incohérent avec les observations "vision" lors du changement de voie et le filtre de Kalman ne peut estimer correctement la position du véhicule.

D'après ces résultats, lors des changements de voies et si le choix du mode des informations est erroné, la multimodalité des informations "vision" peut entraîner des problèmes de divergence du filtre de Kalman. Pour résoudre ce problème il serait nécessaire que le capteur "vision" puisse lui-même définir sa position sur les différentes voies de circulation

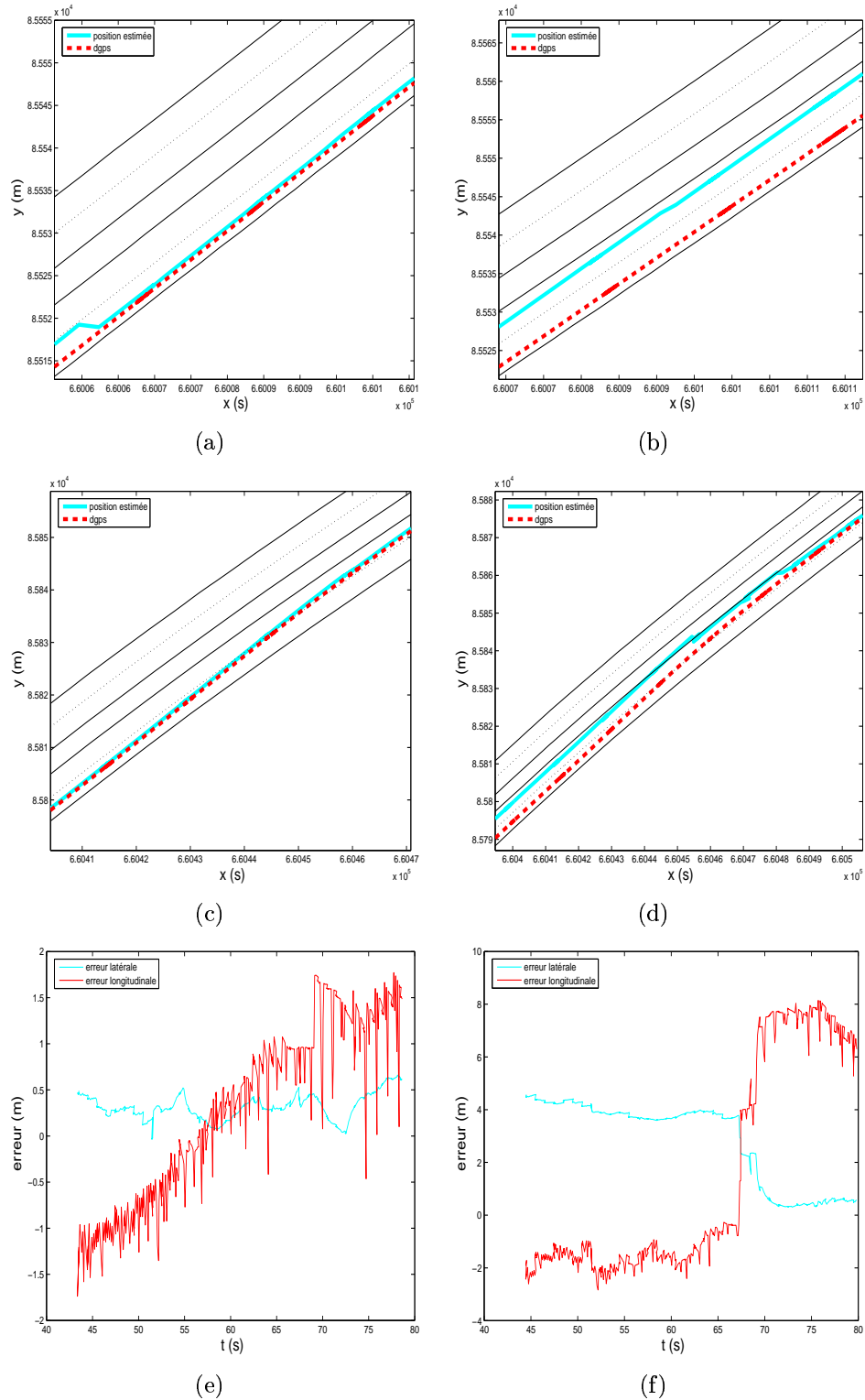


FIG. 5.16: Multimodalité des informations "vision" et changement de voie, figure (a) choix du bon mode, figure (b) choix du mauvais mode, figures (c) et (d) changement de voies et figures (e) et (f) erreurs latérales et longitudinales lors du changement de voies

constituant la route.

## 5.7 Conclusion

L'approche présentée dans ce chapitre permet de localiser le véhicule avec une précision quasi-décimétrique, à condition que tous les capteurs extéroceptifs fournissent régulièrement des informations au processus de fusion et que la partie stochastique des informations GPS soit correctement modélisée (modélisation AR dans notre situation) et intégrée au filtre.

A court terme, cette approche semble plus sensible aux pertes d'informations GPS qu'aux pertes d'informations "vision". En effet, dans cette situation, le filtre de Kalman est dans l'incapacité d'estimer la partie stochastique des informations GPS (les paramètres de biais  $b_x$  et  $b_y$ ). La position estimée se retrouve biaisée et la précision obtenue devient métrique en longitudinal et grâce aux informations "vision" décimétrique en latéral.

A long terme, la perte d'information provenant de l'un des deux ou voire même des deux capteurs extéroceptifs devient catastrophique. En effet, l'accumulation des erreurs proprioceptives lors de l'étape de prédiction entraîne la divergence du filtre de Kalman, la position estimée peut être alors complètement erronée et se retrouver en dehors du réseau routier. Ce phénomène pourrait être endigué si le réseau routier pouvait lui aussi fournir des informations sur la position du véhicule. Cependant sans un aménagement adéquat de celui-ci il semble impossible de fournir de telles informations au système.

Enfin les informations "vision" introduisent deux problèmes dans cette approche 1) un problème de linéarisation des équations de correction et 2) un problème de gestion de leur multimodalité. Le premier problème a été atténué en augmentant artificielle des incertitudes "vision", alors que le second et ce malgré l'ajout d'une étape permettant de discriminer le mode le plus probable ne peut être résolu dans tous les cas de figure. Pour résoudre optimalement ces problèmes, nous avons donc envisagé d'utiliser un filtre particulière à la place du filtre de Kalman car ce filtre traite aisément tous les problèmes d'estimations non-linéaires et gère naturellement les informations ayant une distribution de probabilité quelconque et par conséquent les multimodales. De plus, grâce à ce filtre il est possible de prendre en compte le réseau routier dans la fusion en émettant des hypothèses sur la probabilité que le véhicule appartienne à la route, résolvant par la même occasion les problèmes de pertes d'informations extéroceptives à long terme. Cette étude est l'objet du prochain chapitre.

# Chapitre 6

## Localisation : filtre particulaire

Ce chapitre présente notre approche de localisation lorsque la fusion de données est réalisée par un filtre particulaire. Il peut être décomposé en trois parties, une première qui décrit en détails les diverses étapes du filtre particulaire, une deuxième qui expose le choix du filtre particulaire utilisé dans cette approche et une troisième qui présente les résultats obtenus.

La première partie de ce chapitre décrit l'ensemble des étapes du filtre particulaire c'est-à-dire : l'étape de prédiction et les diverses étapes de correction en fonction des informations GPS, "vision" ainsi que d'une nouvelle source d'information qui n'est d'autre que la carte du réseau routier.

La deuxième partie, présente le choix du filtre particulaire où, notamment, nous montrons que pour notre approche de localisation le filtre particulaire avec interaction traditionnel est soumis à des problèmes de dégénérescence sévères du jeu de particules et ne peut être utilisé pour réaliser la fusion de données. Pour résoudre ce problème, notre choix s'est porté alors vers un filtre particulaire génétique (cf section 2.3.5).

Enfin, comme pour l'approche avec filtre de Kalman, la troisième partie présente les résultats expérimentaux. Elle donne tout d'abord une description comportementale de cette nouvelle approche en fonction de la disponibilité des capteurs extéroceptifs. Puis une justification de l'intégration de la partie stochastique des informations GPS dans le système de fusion et ce même si la méthode de filtrage a été changée. Enfin elle fournit des résultats quantitatifs du comportement du filtre lorsque le système est confronté aux problèmes de multimodalité et confirme la légitimité de l'utilisation d'un filtre particulaire génétique à la place d'un filtre particulaire avec interaction classique.

## 6.1 Le jeu de particules

### 6.1.1 Description

Contrairement à la méthode de localisation par filtre de Kalman où l'on a défini le vecteur d'état, pour le filtre particulière il est nécessaire de définir le jeu de particules  $\mathcal{S}^N$  approchant la loi  $p(\underline{X}_k | \underline{Y}_{1:k})$ . Chaque particule décrit une position, une orientation possible du véhicule dans le référentiel du monde ainsi que les paramètres de biais  $b_x$  et  $b_y$  des informations GPS dans l'espace d'état. Ainsi le vecteur d'état associé à la particule  $n$  sera noté  $\underline{X}^n$  et donné par :

$$\underline{X}^n = (x, y, \theta, b_x, b_y)^T \quad (6.1)$$

Par rapport à la localisation par filtre de Kalman, la vitesse de rotation du véhicule  $\dot{\theta}$  n'apparaît pas dans ce vecteur d'état. En effet avec le filtre de Kalman, cette information était filtrée afin d'obtenir la meilleure prédiction de l'état a priori. Cependant comme nous l'avons présenté dans 2.3.3.1, lorsque les bruits sur les équations dynamiques du système d'état sont faibles, il est possible que le jeu de particules ait tendance à dégénérer et le filtre particulière à diverger. Dans cette approche de localisation, il n'est donc plus nécessaire de filtrer la vitesse de rotation du véhicule et donc de l'introduire dans le vecteur d'état. Sa véritable mesure  $w_{gyro}$  fournie par le gyromètre est alors directement utilisée dans l'étape de prédiction (cf section 6.2).

Le jeu de particules est alors décrit par :

$$\mathcal{S}^N = (\underline{X}^n, w^n) \quad n = 1 \dots N \quad (6.2)$$

avec  $w^n$  le poids de la particule  $n$ .

### 6.1.2 Le jeu de particules initial

Le jeu de particules est défini lorsque la première information GPS est reçue par le système. Les hypothèses sur les variations possibles des paramètres des vecteurs d'état  $\underline{X}^n$  sont les mêmes que pour l'approche par filtre de Kalman. Ainsi en se référant à la section 5.1.2, le vecteur d'état initial de la particule  $n$  peut être défini de la manière suivante :

$$\underline{X}_0^n = \begin{pmatrix} x_{gps_1} + \epsilon_{AR_x}^n - \epsilon_{b_x}^n \\ y_{gps_1} + \epsilon_{AR_y}^n - \epsilon_{b_y}^n \\ \theta^n \\ \epsilon_{b_x}^n \\ \epsilon_{b_y}^n \end{pmatrix} \quad (6.3)$$

où

–  $x_{gps_1}$  et  $y_{gps_1}$  : est la première position fournie par le GPS,

- $\epsilon_{b_x}^n$  et  $\epsilon_{b_y}^n$  : sont des variables aléatoires définissant les variations possibles des paramètres de biais, régies par les lois de probabilité  $\mathcal{N}(0, \sigma_{x_{gps}}^2)$  et  $\mathcal{N}(0, \sigma_{y_{gps}}^2)$ ,
- $\epsilon_{AR_x}^n$  et  $\epsilon_{AR_y}^n$  : sont des variables aléatoires définissant les variations possibles de la position fournie par le GPS, régies par les lois de probabilité  $\mathcal{N}(0, \sigma_{x_{gps}}^2)$  et  $\mathcal{N}(0, \sigma_{y_{gps}}^2)$ ,
- $\theta^n$  : est une variable aléatoire définissant la variation possible de l'orientation du véhicule, régie par la loi de probabilité  $\mathcal{U}[-\pi, \pi]$ .

## 6.2 Etape de prédiction

L'étape de prédiction doit permettre de faire évoluer l'ensemble des particules dans l'espace d'état en sachant que, comme pour le filtre de Kalman :

- le modèle cinématique du véhicule est le modèle de type char décrit dans 1.1.1.1
- les paramètres de biais sont régis par des processus autorégressif du 1<sup>er</sup> ordre.

Ainsi la prédiction de chacune des particules est définie par :

$$p(\underline{X}_k^n, \underline{X}_{k-1}^n) = f(\underline{X}_{k-1}^n, \underbrace{\alpha_{x_k}, \beta_{x_k}, \alpha_{y_k}, \beta_{y_k}}_{\text{paramètres AR}}, \underbrace{\Delta_D^n, \epsilon_{b_x}^n, \epsilon_{b_y}^n, w_{gyro}^n}_{\text{vecteur d'entrée } U_k}) \quad (6.4)$$

où

$$\begin{cases} x_k^n &= x_{k-1}^n - \Delta_D^n \sin(\theta_{k-1}^n + w_{gyro}^n \Delta_T) \\ y_k^n &= y_{k-1}^n + \Delta_D^n \cos(\theta_{k-1}^n + w_{gyro}^n \Delta_T) \\ \theta_k^n &= \theta_{k-1}^n + w_{gyro}^n \Delta_T \\ b_{x_k}^n &= -\alpha_{x_k} b_{x_{k-1}}^n + \beta_{x_k} \epsilon_{b_x}^n \\ b_{y_k}^n &= -\alpha_{y_k} b_{y_{k-1}}^n + \beta_{y_k} \epsilon_{b_y}^n \end{cases} \quad (6.5)$$

avec :

- $\Delta_T$  : le temps écoulé entre les instants  $k-1$  et  $k$ ,
- $\alpha_{x_k}, \beta_{x_k}, \alpha_{y_k}, \beta_{y_k}$  les paramètres du modèle autorégressif régissant les biais  $b_x$  et  $b_y$ , ces paramètres sont recalculés de la même manière que pour le filtre de Kalman.
- $\Delta_D^n$  : une variable aléatoire tirée selon la loi  $\mathcal{N}(\Delta_D, \sigma_{\Delta_D}^2)$ ,
- $w_{gyro}^n$  : une variable aléatoire tirée selon la loi  $\mathcal{N}(w_{gyro}, \sigma_{w_{gyro}}^2)$ ,
- $\epsilon_{b_x}^n$  et  $\epsilon_{b_y}^n$  : des variables aléatoires pures tirées selon la loi  $\mathcal{N}(0, 1)$ .

D'après l'équation (6.5), la prédiction d'une particule est similaire à la prédiction réalisée avec le filtre de Kalman. Là encore, pour que la prédiction des biais soit effectuée correctement, il est nécessaire de recalculer les paramètres  $\alpha_{x_k}, \beta_{x_k}, \alpha_{y_k}$  et  $\beta_{y_k}$  des modèles AR en fonction de la fréquence d'échantillonnage de l'étape de prédiction (section 5.2.1.2).

Les variables d'entrées sont, quant à elles, remplacées par des variables aléatoires discrètes permettant ainsi de balayer l'ensemble de l'espace d'état à partir des trajectoires définies par le nuage de particules  $\mathcal{S}_{k-1}^N$ . Ces variables aléatoires sont tirées selon les



distributions de probabilité des mesures de chacun des capteurs, selon la loi  $\mathcal{N}(w_{gyro}, \sigma_{gyro}^2)$  pour la mesure gyrométrique et  $\mathcal{N}(\Delta_D, \sigma_{\Delta_D}^2)$  pour la mesure odométrique.

### 6.3 Mise à jour par GPS

Pour réaliser la mise à jour en fonction des informations GPS, il est nécessaire de déterminer la vraisemblance de l'observation  $\underline{Y}_{gps}$  conditionnellement à chacune des particules  $\underline{X}_k^n$ , c'est-à-dire de calculer la densité de vraisemblance  $p(\underline{Y}_{gps}|\underline{X}_k^n)$ , puis de calculer les poids non normalisés de chacune des particules selon la relation :

$$\tilde{w}_k^n = p(\underline{Y}_{gps}|\underline{X}_k^n)w_{k-1}^n \quad (6.6)$$

D'après [Doucet *et al.*, 2000], la densité de vraisemblance  $p(\underline{Y}|\underline{X}_k^n)$  peut être optimale à condition que le système d'état soit totalement linéaire gaussien ou que seules les équations dynamiques du système soient non-linéaires. Alors pour le système d'état :

$$\begin{aligned} \underline{X}_k^n &= f(\underline{X}_{k-1}^n) + \underline{W}_k \\ \underline{Y}_k &= H\underline{X}_k^n + \underline{V}_k \end{aligned} \quad (6.7)$$

où  $\underline{W}_k$  et  $\underline{V}_k$  sont des bruits centrés gaussiens indépendants définis respectivement par les lois  $\mathcal{N}(0, \mathbf{Q}_k)$  et  $\mathcal{N}(0, \mathbf{R}_k)$ , la loi  $p(\underline{Y}_k|\underline{X}_k^n)$  optimale est :

$$p(\underline{Y}_k|\underline{X}_k^n) \propto e^{-\frac{1}{2}(\underline{Y}_k - \mathbf{H}f(\underline{X}_{k-1}^n))(\mathbf{R}_k + \mathbf{H}\mathbf{Q}_k\mathbf{H})^{-1}(\underline{Y}_k - \mathbf{H}f(\underline{X}_{k-1}^n))^T} \quad (6.8)$$

Or en tenant compte le fait que l'étape de prédiction est décrite par les équations de prédiction (6.5) et que l'observation  $\underline{Y}_k$  est formée des informations fournies par le GPS, le système d'état pour cette approche de localisation est :

$$\begin{aligned} \underline{X}_k^n &= f(\underline{X}_{k-1}^n) + \underline{W}_k \\ \underline{Y}_{gps} &= H'_{gps}\underline{X}_k^n + \underline{V}_k \end{aligned} \quad (6.9)$$

où

- $\underline{W}_k$  est un vecteur de bruit blanc centré gaussien régi par la loi  $\mathcal{N}(0, \mathbf{Q}_k)$ . D'après les équations de prédiction, le bruit  $\underline{V}_k$  défini par le vecteur  $(0, 0, 0, 0, \epsilon_{b_x}, \epsilon_{b_y})^T$  où n'apparaissent à que les bruits sur les paramètres de biais, ainsi la matrice de covariance  $\mathbf{Q}_k$  est :

$$\mathbf{Q}_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{b_x}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{b_y}^2 \end{pmatrix} \quad (6.10)$$

- $\underline{V}_k$  est le bruit de mesure sur les informations GPS régi par la loi normale  $\mathcal{N}(0, \mathbf{R}'_{gps})$

- La matrice d'observation  $\mathbf{H}$  reliant les observations GPS à l'état de chacune des particules devient la matrice  $\mathbf{H}'_{gps}$  définie par :

$$\mathbf{H}'_{gps} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \quad (6.11)$$

D'après sa description ce système d'état remplit bien les conditions proposées par Doucet et il est donc possible de calculer la densité de vraisemblance optimale  $p(\underline{Y}_{gps} | \underline{X}_k^n)$  pour ce système :

$$p(\underline{Y}_{gps} | \underline{X}_k^n) \propto e^{-\frac{1}{2}(\underline{Y}_{gps} - \mathbf{H}'_{gps} f(\underline{X}_{k-1}^n))(\mathbf{R}'_{gps} + \mathbf{H}'_{gps} \mathbf{Q}_k \mathbf{H}'_{gps})^{-1}(\underline{Y}_{gps} - \mathbf{H}'_{gps} f(\underline{X}_{k-1}^n))^T} \quad (6.12)$$

où

$$\mathbf{R}_{gps} + \mathbf{H} \mathbf{Q}_k \mathbf{H}'^T = \begin{pmatrix} \sigma_{x'_{gps}}^2 + \sigma_{b_x}^2 & 0 \\ 0 & \sigma_{y'_{gps}}^2 + \sigma_{b_y}^2 \end{pmatrix} \quad (6.13)$$

D'après (6.13), il n'y a pas de corrélation entre les différents paramètres rentrant en jeu dans l'équation (6.12) car la matrice  $\mathbf{R}_{gps} + \mathbf{H}'_{gps} \mathbf{Q}_k \mathbf{H}'_{gps}$  ne contient aucun terme de covariance. Ainsi, en prenant en compte la non corrélation des paramètres et en considérant que  $f(\underline{X}_{k-1})$  est remplacée par  $\underline{X}_k^n$  alors la densité de vraisemblance peut être réécrite de la manière suivante :

$$p(\underline{Y}_{gps} | \underline{X}_k^n) \propto e^{-\frac{1}{2} \left[ \frac{(x_{gps} - x_k^n + b_{x_k})^2}{\sigma_{x'_{gps}}^2 + \sigma_{b_x}^2} + \frac{(y_{gps} - y_k^n + b_{y_k})^2}{\sigma_{y'_{gps}}^2 + \sigma_{b_y}^2} \right]} \quad (6.14)$$

## 6.4 Mise à jour par "vision"

### 6.4.1 Mise à jour "vision" unimodale

Comme pour la mise à jour en fonction des informations GPS, il est nécessaire de calculer la densité de vraisemblance  $p(\underline{Y}_{vision} | \underline{X}_k^n)$  puis de mettre à jour les poids selon la relation :

$$w_k^n = p(\underline{Y}_{vision} | \underline{X}_k^n) w_{k-1}^n \quad (6.15)$$

Le principal inconvénient dans cette étape est que les informations "vision" et les états du jeu de particules ne sont pas définis dans le même référentiel. Avant de réaliser la mise à jour des poids, il est nécessaire de redéfinir le jeu de particules dans le même référentiel que les informations "vision" et plus particulièrement dans le même espace d'état où évoluent ces dernières. Alors le nouveau jeu de particule  $\mathcal{S}_{r,k}^N$  obtenu est tel que :

$$\mathcal{S}_{r,k}^N = (\underline{X}_{r,k}^n, w_k^n) \quad n = 1 \dots N \quad (6.16)$$

où  $\underline{X}_{r,k}^n$  décrit la position et l'orientation de la particule  $\mathcal{S}_k^n$  dans le référentiel de la route.

$$\underline{X}_{r,k}^n = \begin{pmatrix} X_{r,k}^n \\ \Psi_{r,k}^n \end{pmatrix} \quad (6.17)$$

avec :

- $X_{r,k}^n$  : la position latérale sur la chaussée de la particule  $n$
- $\Psi_{r,k}^n$  : l'orientation sur la chaussée de la particule  $n$ .

Ainsi en prenant en compte les équations (5.38) nous pouvons définir  $\underline{X}_{r,k}^n$  en fonction de l'état  $\underline{X}_k^n$  et des attributs  $x_r$ ,  $y_r$  et  $\theta_r$  de la carte :

$$\underline{X}_{r,k}^n = \begin{cases} \cos(\theta_r)(x_k^n - x_r) + \sin(\theta_r)(y_k^n - y_r) \\ \theta_k^n - \theta_r \end{cases} \quad (6.18)$$

Lorsque ce nouveau jeu de particules est obtenu, il est possible de calculer la vraisemblance de l'observation "vision" conditionnellement aux vecteurs d'état  $\underline{X}_{r,k}^n$  sachant que l'équation d'observation est :

$$\underline{Y}_{vision} = \underline{X}_{r,k}^n + \underline{W}_k \quad (6.19)$$

L'équation d'observation étant linéaire, la densité de vraisemblance optimale pour la mise à jour "vision" est :

$$p(\underline{Y}_{vision} | \underline{X}_{r,k}^n) \propto e^{-\frac{1}{2}(\underline{Y}_{vision} - \underline{X}_{r,k}^n)(\mathbf{R}_{vision} + \mathbf{H}'_{vision} \mathbf{Q}_k \mathbf{H}'_{vision})^{-1}(\underline{Y}_{vision} - \underline{X}_{r,k}^n)^T} \quad (6.20)$$

avec le jacobien  $\mathbf{H}'_{vision}$  défini par les dérivées partielles des équations (6.18) selon les paramètres du vecteur d'état  $\underline{X}_k^n$  :

$$\mathbf{H}'_{vision} = \begin{pmatrix} \cos \theta_r & \sin \theta_r & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (6.21)$$

Comme il n'y a pas de bruit additif sur les paramètres  $x$ ,  $y$ ,  $\theta$  dans les équations dynamiques du système d'état, la relation  $\mathbf{H}'_{vision} \mathbf{Q}_k \mathbf{H}'_{vision}$  est nulle et donc la densité de vraisemblance  $p(\underline{Y}_{vision} | \underline{X}_{r,k}^n)$  peut être écrite :

$$p(\underline{Y}_{vision} | \underline{X}_{r,k}^n) \propto e^{-\frac{1}{2}(\underline{Y}_{vision} - \underline{X}_{r,k}^n) \mathbf{R}_{vision}^{-1} (\underline{Y}_{vision} - \underline{X}_{r,k}^n)^T} \quad (6.22)$$

Pour éviter de réaliser des calculs matriciels et donc pour gagner du temps, nous allons considérer que les informations "vision" sont décorréliées les unes des autres, ainsi la densité de vraisemblance peut être réécrite :

$$p(\underline{Y}_{vision} | \underline{X}_{r,k}^n) \propto e^{-\frac{1}{2} \left[ \frac{(X_0 - X_{r,k}^n)^2}{\sigma_{X_0}^2} + \frac{(\Psi - \Psi_{r,k}^n)^2}{\sigma_{\Psi}^2} \right]} \quad (6.23)$$

Comme  $\mathcal{S}_k^n$  et  $\mathcal{S}_{r,k}^n$  sont deux représentations différentes du même jeu de particules, on peut considérer que la densité de vraisemblance  $p(\underline{Y}_{vision} | \underline{X}_{r,k}^n)$  peut s'appliquer au jeu de particules  $\mathcal{S}_k^n$  et que par conséquent  $p(\underline{Y}_{vision} | \underline{X}_k^n) = p(\underline{Y}_{vision} | \underline{X}_{r,k}^n)$ . Ainsi la mise à jour des poids du jeu de particules  $\mathcal{S}_k^n$  peut être remise à jour selon :

$$\tilde{w}_k^n = p(\underline{Y}_{vision} | \underline{X}_{r,k}^n) w_{k-1}^n \quad (6.24)$$

### 6.4.2 Multimodalité des observations "vision"

Contrairement au filtre de Kalman, les filtres particulaires savent gérer les informations de densité de probabilité quelconque et notamment les informations multimodales. Ainsi le problème de multimodalité rencontré dans le cadre de réseau routier comportant plusieurs voies de circulation peut être facilement résolu. Il suffit de modifier la densité de vraisemblance  $p(\underline{Y}_{vision} | \underline{X}_{r,k}^n)$  des observations "vision" de telle manière qu'elle prenne en considération cette multimodalité. Cette nouvelle densité de vraisemblance peut être alors définie comme la somme des densités de vraisemblance  $p(\underline{Y}_{vision} | \underline{X}_{ri,k}^n)$  de chacun des modes des informations "vision". Sachant qu'il y a autant de modes que de voies de circulation  $i$ , la densité de vraisemblance tenant compte de la multimodalité des informations "vision" est :

$$p(\underline{Y}_{vision} | \underline{X}_{r,k}^n) = \sum_{i=1}^{N_v} p(\underline{Y}_{vision} | \underline{X}_{ri,k}^n) \quad (6.25)$$

avec :

- $N_v$  le nombre de voies de circulation,
- $\underline{X}_{ri,k}^n$  la position et l'orientation de la particule  $n$  sur la voie de circulation  $i$ ,
- $p(\underline{Y}_{vision} | \underline{X}_{ri,k}^n)$  la densité de vraisemblance des observations "vision" pour la voie de circulation  $i$  définie par l'équation :

$$p(\underline{Y}_{vision} | \underline{X}_{ri,k}^n) \propto e^{-\frac{1}{2} \left[ \frac{(X_0 - X_{ri,k}^n - iL_i)^2}{\sigma_{X_0}^2} + \frac{(\Psi - \Psi_{ri,k}^n)^2}{\sigma_{\Psi}^2} \right]} \quad (6.26)$$

La figure 6.1 présente la densité de vraisemblance de la position latérale du véhicule sur une route comportant deux voies de circulation. Cette densité peut être déduite de  $p(\underline{Y}_{vision} | \underline{X}_{r,k}^n)$ , si l'on ne considère par l'orientation dans l'équation (6.26).

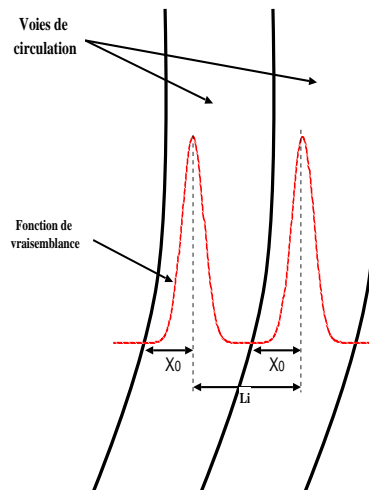


FIG. 6.1: Densité de vraisemblance des informations "vision" (position latérale)

### 6.4.3 Prise en compte de la carte

Contrairement au filtre de Kalman, en l'absence d'information "vision" le filtre particulière peut prendre en compte la carte pour se localiser. Dans ce cas, la carte est considérée comme une observation pour laquelle la fonction de vraisemblance  $p(\text{"carte"} | \underline{X}_{r,k}^n)$  peut être calculée, cette densité définit alors l'appartenance ou non du véhicule à une des voies de circulation du réseau routier. D'ailleurs ce principe est toujours rencontré dans les systèmes de navigation proposés dans le commerce, où la position du véhicule est toujours associée à un des segments de la carte. Potentiellement dans notre situation, il existe différentes manières d'exprimer la densité de vraisemblance  $p(\text{"carte"} | \underline{X}_{r,k}^n)$ . Nous en proposons deux :

1. le véhicule évolue obligatoirement sur une route cartographiée auquel cas on peut décrire  $p(\text{"carte"} | \underline{X}_{r,k}^n)$  par la loi de probabilité (figure 6.2(a)) suivante :

$$p(\text{"carte"} | \underline{X}_{r,k}^n) = \sum_{i=0}^{N_v} p(\text{"voie"} | \underline{X}_{ri,k}^n) \quad (6.27)$$

où  $p(\text{"voie"} | \underline{X}_{ri,k}^n)$  est la probabilité que le véhicule appartienne à la voie de circulation  $i$  :

$$p(\text{"voie"} | \underline{X}_{ri,k}^n) = \begin{cases} 1 & \text{si } \underline{X}_{ri,k}^n \in [iL_i, (i+1)L_i] \\ 0 & \text{sinon} \end{cases} \quad (6.28)$$

Ainsi sous la contrainte que la véritable position du véhicule appartienne à une route cartographiée, toutes les particules en dehors de cette route se voient attribuer un poids nul. Lorsque le rééchantillonnage est effectué, celles-ci sont éliminées du jeu de particules, réorientant celui-ci dans l'espace d'état selon l'orientation de la route. Cependant si le véhicule sort de la zone cartographiée (ex : carte incomplète, chemins etc...), tous les poids des particules deviennent nuls, le filtre particulière est alors mis en échec et est incapable d'estimer  $p(\underline{X}_k | \underline{Y}_{1:k})$ .

2. le véhicule a seulement une forte probabilité d'évoluer sur une route cartographiée et en particulier d'évoluer au centre d'une voie de circulation de la route, auquel cas on peut décrire  $p(\text{"carte"} | \underline{X}_{r,k}^n)$  par la somme de loi normale (figure 6.2(b)) suivante :

$$p(\text{"carte"} | \underline{X}_{r,k}^n) \propto \sum_{i=0}^{N_v} p(\text{"voie"} | \underline{X}_{ri,k}^n) \quad (6.29)$$

où  $p(\text{"voie"} | \underline{X}_{ri,k}^n)$  est la probabilité que la particule appartienne à la voie de circulation  $i$  :

$$p(\text{"voie"} | \underline{X}_{ri,k}^n) \propto e^{-\frac{1}{2} \left[ \frac{(\frac{L_i}{2} - \underline{X}_{ri,k}^n - iL_i)^2}{(\frac{L_i}{2})^2} \right]} \quad (6.30)$$

dont les deux premiers moments sont définis par le centre des voies de circulation  $i$  de la route à savoir  $\frac{L_i}{2}$  et la variance  $(\frac{L_i}{2})^2$ .

Dans ce cas, les poids des particules ne sont plus nuls lorsque le véhicule n'évolue plus sur une route cartographiée mais diminuent en s'éloignant de celle-ci. Cette densité de vraisemblance permet au filtre particulaire d'estimer l'état du véhicule  $p(\underline{X}_k | \underline{Y}_{1:k})$  même si celui-ci sort d'une zone cartographiée.

Le choix entre ces deux densités d'importance est imposé par le problème de localisation. Dans notre situation la carte du réseaux routier est incomplète, on privilégiera donc la deuxième solution pour notre approche.

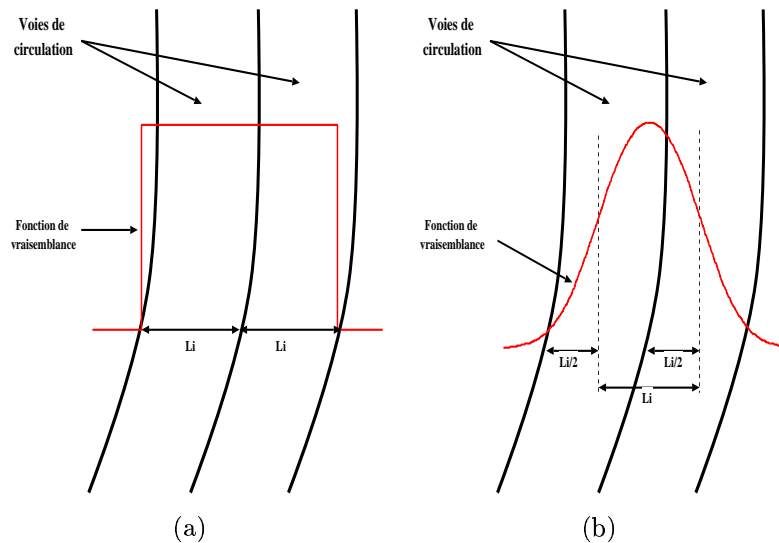


FIG. 6.2: Densités de vraisemblance de l'information carte, figure (a) loi uniforme, figure (b) loi normale

Enfin la mise à jour des poids du jeu de particules est réalisée en appliquant la relation suivante :

$$\tilde{w}_k^n = p(\text{"carte"} | \underline{X}_{r,k}^n) w_{k-1}^n \quad (6.31)$$

## 6.5 Choix du filtre particulaire

Dans les sections précédentes, les étapes de prédiction et de correction ont été définies au vu des différents capteurs choisis pour notre application. Maintenant, il est nécessaire de déterminer le filtre particulaire le plus adéquat pour effectuer la fusion en fonction de ces diverses étapes. D'après la section 2.3, plusieurs filtres particuliers existent : les filtres particuliers avec procédure SIS, les filtres particuliers avec procédure SIS-R, les filtres particuliers régularisés et enfin les filtres particuliers génétiques. Cependant, comme les filtres particuliers avec procédure SIS ont tendance à diverger à long terme notre choix se portera donc sur l'un de ces trois derniers filtres. Commençons tout d'abord par le filtre

particulaire avec procédure SIS-R et en particulier le filtre particulaire avec interaction SIR.

## 6.5.1 Filtre particulaire avec interaction

### 6.5.1.1 Description du filtre

D'après la description du filtre particulaire avec interaction (cf section 2.3.3.1) et de la définition des étapes de prédiction et de correction données dans les sections précédentes, il est possible de décrire le filtre particulaire avec interaction dédié à notre approche par l'algorithme 6.1.

---

Algorithme 6.1 : Filtre particulaire avec interaction lié à notre application

---

- **Pour**  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = \frac{1}{N})$  selon la loi initiale  $\mathcal{L}_0$ .
- **Pour**  $k \geq 1$ 
  - **Evolution** : générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k^n | \underline{X}_{k-1}^n)$  en utilisant (6.5).
  - **Correction** :
    - pour toutes les particules calculer les poids non normalisés
      - si GPS :  $\tilde{w}_k^n = p(\underline{X}_{gps}^n | \underline{X}_k^n) w_{k-1}^n$
      - si "vision"
        - si route retrouvée :  $\tilde{w}_k^n = p(\underline{X}_{vision}^n | \underline{X}_k^n) w_{k-1}^n$
        - sinon :  $\tilde{w}_k^n = p("carte" | s_k^n) w_{k-1}^n$
    - puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=0}^N \tilde{w}_k^n$ .
  - **Rééchantillonnage** :
    - effectuer un tirage aléatoire avec remise parmi  $\mathcal{S}_k^N$  et proportionnellement aux poids  $w_k^n$
    - mettre les poids  $w_k^n$  à la valeur  $\frac{1}{N}$
  - **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approximer la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_k w_k^n \delta_{s_k^n}$$


---

### 6.5.1.2 Dégénérescence du filtre particulaire

Pour étudier le comportement de ce filtre, une série des tests expérimentaux a été réalisée dans les mêmes conditions que dans la section 5.6. Les figures 6.3(a) à 6.3(d)

montrent l'évolution des paramètres d'état  $x$  et  $y$  du jeu de particules obtenue lors de ces tests. Ainsi on peut voir : dans la figure 6.3(a) le jeu de particules à l'initialisation, dans les figures 6.3(b) et 6.3(c) le nuage de particules pour  $t = 7s$  et  $14s$  en sachant que ni les informations "vision" ni la carte sont prises en compte, enfin dans la figure 6.3(d) le nuage de particules pour  $t = 21s$  en sachant que les informations "vision" sont prises en compte.

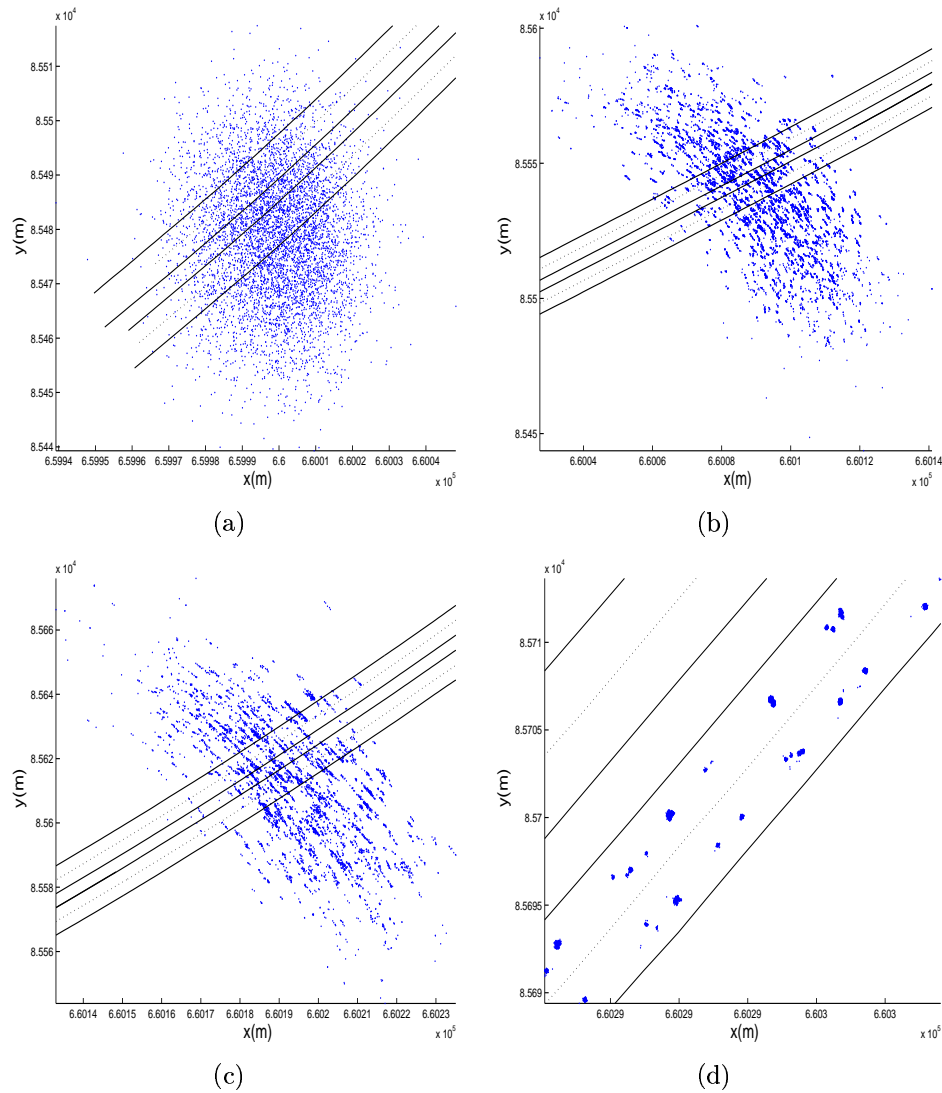


FIG. 6.3: Dégénérescence du filtre particulaire avec interaction

D'après ces figures, au fur et à mesure du temps des disparités apparaissent dans le jeu de particules laissant ainsi des régions de l'espace d'état complètement vide. Ce phénomène est fortement accru lorsque les informations "vision" sont prises en compte. La disparité est alors telle que le jeu de particules est divisé en plusieurs nuages distincts les



uns des autres. On se trouve donc dans un cas sévère de dégénérescence du jeu de particules dont la cause peut être attribuée aux faibles bruits sur les équations dynamiques du système d'état et en particulier aux faibles bruits de mesures des informations gyro-métriques et odométriques.

Plusieurs solutions s'offrent à nous pour régler ce problème, soit :

1. on augmente artificiellement les bruits sur les équations dynamiques du système d'état en rajoutant des bruits additifs sur les équations de position et d'orientation dans la relation (6.5),
2. on augmente le nombre de particules,
3. on utilise un filtre particulaire régularisé ou un filtre particulaire génétique.

La première solution est facile à mettre en œuvre cependant le choix des bruits additifs est totalement heuristique et difficile à quantifier. La seconde solution est elle aussi très simple, en augmentant le nombre de particules, l'espace d'état est mieux recouvert et les effets de dégénérescence sont diminués. Comme la charge de calcul de ce genre de filtre est proportionnelle au nombre de particules (cf section 6.6.4), cette solution n'est applicable qu'au détriment de la rapidité du système de fusion de données, ce qui pose bien sûr des problèmes pour un système de localisation comme le notre qui se veut avant tout un système temps réel. D'après la section 2.3.4, le filtre particulaire régularisé fonctionne correctement lorsque les bruits du système dynamique sont faibles ou même voire quasi-inexistants mais peut potentiellement diverger lorsque les bruits des observations sont faibles. Or, comme les informations fournies par le capteur "vision" sont très précises, ce filtre ne semble pas adéquat pour notre approche. Nous avons donc décidé d'utiliser un filtre particulaire génétique à la place du filtre particulaire avec interaction.

## 6.5.2 Filtre particulaire génétique

### 6.5.2.1 Description du filtre

D'après la description du filtre particulaire génétique (cf section 2.3.5) et les diverses étapes de prédiction et de correction définies dans ce chapitre, il est possible de décrire le filtre génétique par l'algorithme 6.2.

D'après cette description, le filtre particulaire génétique est similaire au filtre particulaire avec interaction. Seule l'étape de prédiction est différente. En effet, le filtre particulaire génétique utilise les propriétés d'évolution génétique (mutation, brassage d'une population, voir section 2.3.5) dans l'étape de prédiction. Ainsi les particules couvrent mieux l'espace d'état, les effets de disparité du jeu de particules rencontrés avec le filtre avec interaction sont atténués et donc ce filtre est moins sujet au phénomène de dégénérescence.

---

Algorithme 6.2 : Filtre particulière génétique lié à notre application

---

- **Pour**  $k = 0$ 
  - générer  $\mathcal{S}_0^N = (\underline{X}_0^n, w_0^n = 1/n)$  selon la loi initiale  $\mathcal{L}_0$
- **Pour**  $k \neq 1$ 
  - **Evolution** :
    - Appliquer l'opérateur de mélange génétique à  $S_{k-1}^N$
    - Appliquer l'opérateur de mutation génétique à  $S_{k-1}^N$
    - Générer  $\underline{X}_k^n$  selon la loi  $p(\underline{X}_k^n, \underline{X}_{k-1}^n)$
  - **Correction** :
    - pour toutes les particules calculer les poids non normalisés
      - si GPS :  $\tilde{w}_k^n = p(\underline{X}_{gps} | \underline{X}_k^n) w_{k-1}^n$
      - si "vision"
        - si route retrouvée :  $\tilde{w}_k^n = p(\underline{X}_{vision} | \underline{X}_k^n) w_{k-1}^n$
        - sinon :  $\tilde{w}_k^n = p("carte" | s_k^n) w_{k-1}^n$
    - puis normaliser les poids :  $w_k^n = \tilde{w}_k^n / \sum_{n=0}^N \tilde{w}_k^n$ .
  - **Rééchantillonnage** :
    - effectuer un tirage aléatoire avec remise parmi  $\mathcal{S}_k^N$  et proportionnellement aux poids  $w_k^n$
    - mettre les poids à la valeur  $\frac{1}{N}$
- **Estimation** : le nuage  $\mathcal{S}_k^N = (\underline{X}_k^n, w_k^n)$  permet d'approcher la loi a posteriori :

$$p(\underline{X}_k | \underline{Y}_{1:k}) = \sum_k w_k^n \delta_{s_k^n}$$


---

### 6.5.2.2 Atténuation du phénomène de dégénérescence

Pour affirmer que le filtre particulière génétique est moins sujet au phénomène de dégénérescence que le filtre particulière avec interaction, une série des tests expérimentaux a été réalisée dans les mêmes conditions que la section 6.5.1. Les figures 6.4(a) à 6.4(d) montrent l'évolution des paramètres d'état  $x$  et  $y$  du jeu de particules ainsi obtenue. On peut donc voir : dans la figure 6.4(a) le jeu de particules à l'initialisation, dans les figures 6.4(b) et 6.4(c) le nuage de particules pour  $t = 7s$  et  $14s$  en sachant que ni les informations "vision" ni la carte sont prises en compte, enfin dans la figure 6.4(d) le nuage de particules pour  $t = 21s$  est observé en sachant que les informations "vision" sont prises en compte.

D'après ces figures, contrairement au filtre particulière avec interaction très peu de

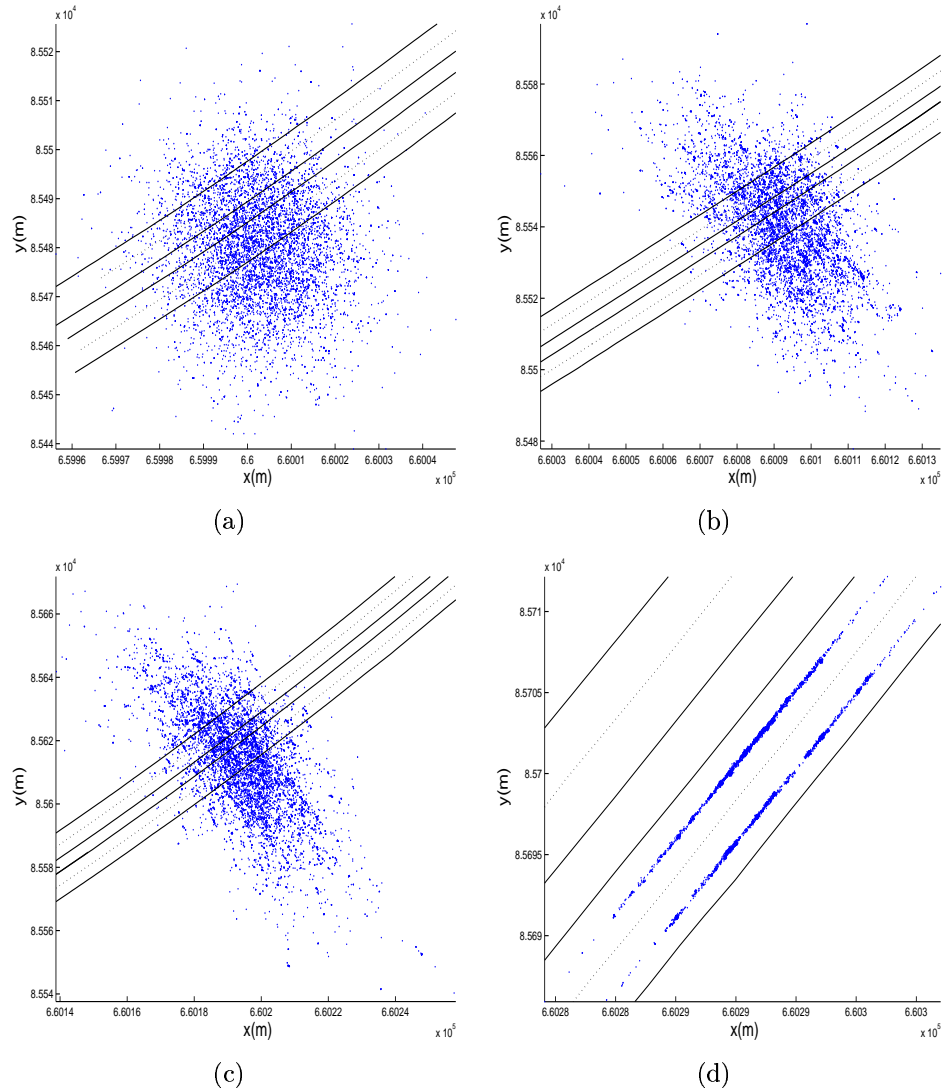


FIG. 6.4: Atténuation du phénomène de dégénérescence avec le GPF

disparités apparaissent dans le jeu de particules au fur et à mesure du temps. D'ailleurs lorsque la vision est prise en compte, le jeu de particules n'est plus divisé en plusieurs nuages distincts les uns des autres mais est divisé en deux nuages représentant bien les deux modes de l'observation "vision". Donc l'introduction des principes d'évolution génétique dans un filtre particulaire permet bien de limiter l'effet de dégénérescence rencontré dans les filtres traditionnels et justifie son utilisation dans notre approche de localisation. Cependant, lorsque le jeu de particules est multimodal ce filtre peut être mis en défaut. En effet, la partie génétique de l'étape de prédiction et, en particulier, les opérateurs de mutation et de brassage ne savent pas gérer cette situation. Nous allons donc présenter une solution à ce problème.

### 6.5.2.3 Jeu de particules multimodal et observation unimodale

Reprenons les résultats de la section précédente. A  $t = 21s$  le jeu de particules est multimodal et peut être considéré comme constitué par deux populations de particules distinctes. Maintenant, considérons qu'une nouvelle correction en fonction des informations GPS soit réalisée et appliquons l'opérateur de brassage génétique au jeu de particules. Comme la densité de vraisemblance  $p(\underline{Y}_{gps} | \underline{X}_k^n)$  est unimodale un nouveau jeu de particules est créé en mélangeant les deux modes de l'ancien jeu de particules recréant ainsi une seule population (figure 6.5). Ainsi le jeu de particules redevient unimodal. Lors de la prochaine correction en fonction des informations "vision", la majorité de ces nouvelles particules apparaissent aberrantes vis à vis de la "vision" et sont éliminées du jeu de particules. Ces particules sont donc créées inutilement et montrent un dysfonctionnement de l'opérateur de brassage génétique lorsque l'observation est unimodale et le jeu de particules est multimodal.

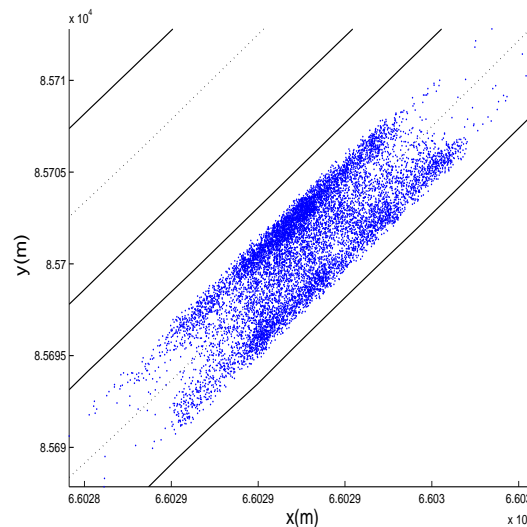


FIG. 6.5: Jeu de particules après correction par une information GPS : opérateur de brassage génétique "métropolisé"

Pour résoudre ce problème, il est nécessaire que cet opérateur soit capable de réaliser un brassage génétique de particules appartenant à une même population et, par conséquent, à un même mode du jeu de particules. Si l'on considère que chaque mode appartient obligatoirement à une seule voie de circulation du réseau routier, il est nécessaire d'utiliser la carte pour améliorer cet opérateur. Ainsi, il suffit de ne plus tirer les particules "parentes" de manière totalement aléatoire mais de réaliser un tirage aléatoire selon la loi probabilité  $p("voie" | \{\mathcal{S}_{k-1}^A, \mathcal{S}_{k-1}^B\})$ . Cette loi représente la probabilité que deux particules appartiennent à la même voie de circulation et par conséquent à la même population. En tenant compte des densités  $p("voie" | \underline{X}_{r,k}^n)$  proposées dans la section 6.4.3 on peut déduire

cette nouvelle loi de probabilité :

$$p("voie" | \{\mathcal{S}_{k-1}^A, \mathcal{S}_{k-1}^B\}) = \sum_{i=0}^{N_v} \prod_{j=A,B} p("voie", \underline{X}_{ri,k}^n) \quad (6.32)$$

Considérons que l'on applique maintenant ce nouvel opérateur de brassage génétique au jeu de particules. D'après la figure 6.6, le brassage génétique ne "détruit" pas les deux populations de particules pour en recréer une seule. Donc la prise en compte des voies de circulation pour le tirage aléatoire des particules parentes permet d'empêcher la création de nouvelles particules aberrantes vis à vis des deux populations et de garder la multimodalité du jeu de particules.

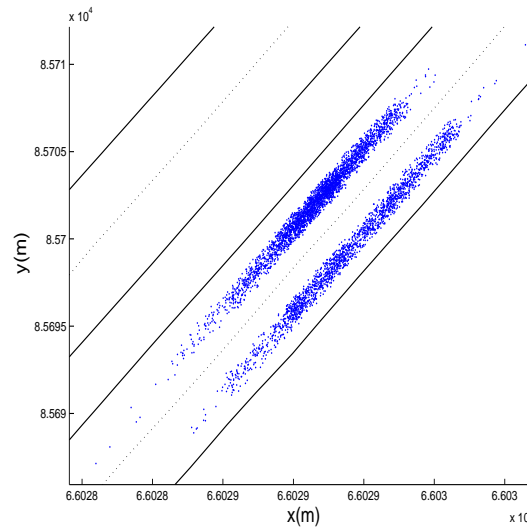


FIG. 6.6: Nouvel opérateur de brassage génétique "métropolisé" : tirage aléatoire selon une (a) loi uniforme (b) loi normale

Enfin, on peut noter que selon le choix de la loi densité de vraisemblance  $p("voie", \underline{X}_{ri,k}^n)$ , à savoir une loi uniforme où une loi normale, le comportement du filtre peut varier lors des changements de voies. En effet pour une loi :

1. uniforme : le brassage génétique ne peut mélanger des particules parentes appartenant à deux voies de circulations différentes, lors des changements de voies une même population se retrouvera donc divisée en deux populations quand celle-ci chevauchera les deux voies de circulation,
2. normale : le brassage génétique peut mélanger des particules parentes appartenant à deux voies de circulation différentes à condition que  $p("voie", \{\mathcal{S}_{k-1}^A, \mathcal{S}_{k-1}^B\})$  soit forte. Selon la densité  $p("voie", \{\mathcal{S}_{k-1}^{A,n}, \mathcal{S}_{k-1}^{B,n}\})$  déduite de la relation (6.32), seules les particules les plus proches du bord commun aux deux voies de circulation ont une chance de subir une mutation. De plus, il est possible d'utiliser un seuil pour contrôler la diffusion des particules entre les deux populations. Intuitivement on

peut considérer que cette solution doit avoir un meilleur comportement lors des changements de voie.

Cependant lors de nos tests, aucune différence notable dans le comportement de ces deux solutions n'a pu être détectée. Notre application n'étant pas un cas général et il est nécessaire de méditer sur le choix de l'une de ces probabilités selon l'approche envisagée par l'utilisateur.

## 6.6 Résultats

L'ensemble de nos expérimentations pour cette approche ont été réalisées dans les mêmes conditions que pour la méthode de localisation avec le filtre de Kalman (cf section 5.6). Les résultats présentés dans cette section sont décomposés en trois parties : une première partie où est montrée l'influence sur la localisation du véhicule de la disponibilité des informations provenant du GPS, du capteur "vision" et la carte et ceci dans les trois contextes suivants :

1. le GPS, le capteur "vision" et la carte sont disponibles,
2. le GPS et la carte sont disponibles,
3. les GPS et le capteur "vision" subissent des pertes de données.

Une seconde partie où est démontré que la modélisation AR des incertitudes GPS est absolument nécessaire pour réaliser correctement la localisation du véhicule. Et enfin, une troisième et dernière partie où est montré comment le filtre particulaire traite la multi-modalité des informations "vision" et en particulier lors de changements de voies.

Pour une meilleure compréhension, les résultats sont fournis dans le référentiel de la route et sont ainsi décrits par la position latérale et la position longitudinale du véhicule sur la chaussée. Nous rappelons que leurs erreurs sur ces variables sont définies par la relation (5.53).

### 6.6.1 Localisation selon la disponibilité des capteurs

Afin de pouvoir à la fois confronter le processus de la localisation selon la disponibilité des informations provenant des différents capteurs et de comparer les résultats de cette approche avec ceux de l'approche précédente, l'ensemble des résultats présentés dans cette section ont été obtenus à partir des mêmes données expérimentales que la section 5.6.1.1. La disponibilité des capteurs est alors encore une fois simulée selon le contexte désiré et le trajet est identique à celui de la section 5.6.1.1 c'est-à-dire :

- de  $t = 0$  à  $3s$  le véhicule démarre en dehors de la zone cartographiée,
- de  $t = 3$  à  $140s$  le véhicule évolue sur la  $2 \times 2$  voies puis sur la  $1 \times 1$  voies pour arriver enfin au rond point,
- pour  $t > 140s$  le véhicule fait le tour du rond point puis emprunte la  $2 \times 2$  voies dans l'autre sens de circulation.

### 6.6.1.1 L'ensemble des capteurs extéroceptifs sont disponibles

La figure 6.7 présente les erreurs de position du véhicule sur la chaussée dans le cas où tous les capteurs extéroceptifs sont disponibles :

- de  $t = 0$  à  $3s$  : les erreurs sont nulles car le véhicule évolue en dehors de la zone cartographiée.
- de  $t = 3$  à  $12s$  : l'algorithme de "vision" considère que la route n'est pas retrouvée, les informations du capteur "vision" ne sont donc pas prises en compte par la fusion de données. Ainsi la localisation s'effectue en fonction des informations provenant du GPS, des capteurs proprioceptifs et de la carte. Dans ce cas, la prise en compte de la carte, permet d'émettre une hypothèse sur la position latérale du véhicule sur la route (cf section 6.4.3) et le filtre particulière positionne le véhicule près du centre de la chaussée. Comme le véhicule circule sur la portion 2x2 voies, les erreurs latérales sont obligatoirement inférieures à la largeur d'une voie (à savoir 3.5m). En revanche, comme aucune information sur la position longitudinale est fournie par la carte, les erreurs longitudinales sont importantes mais inférieures à  $\pm GDO P_{gps} \times EURE_{gps}$  (un exemple de ce comportement peut être vu dans dans la figure 6.14(a))
- de  $t = 12$  à  $38s$  : le capteur "vision" fournit des informations valides, contrairement au filtre de Kalman l'erreur latérale ne tend pas brutalement vers 0 à la première information "vision" valide. En effet, comme le filtre particulière sait prendre compte la multimodalité des informations "vision", il considère que le véhicule peut se situer sur les deux voies de circulation. Le filtre positionne encore une fois le véhicule près du centre de la chaussée et les erreurs latérales sur la position du véhicule restent similaires au cas précédent. Enfin à  $t = 38s$ , le véhicule effectue son premier changement de voie, le filtre particulière peut déterminer la voie de circulation sur laquelle circule le véhicule, l'erreur latérale tend brutalement vers zéro alors que l'erreur longitudinale ne varie pas (un exemple de ce comportement peut être vu dans la figure 6.14(c)).
- de  $t = 38$  à  $140s$  : le système a donc déterminé sur quelle voie de circulation est situé le véhicule, le filtre particulière ne plus prendre en compte la multimodalité des informations "vision" et converge lentement vers la vraie solution.

On notera que lorsque tous les capteurs extéroceptifs sont disponibles, cette approche converge plus lentement que l'approche avec filtre de Kalman car les multimodalités des informations "vision" sont prises en compte. Néanmoins l'estimation de la position du véhicule est elle aussi relativement précise comme le montre le tableau 6.1.

La figure 6.8 montre l'évolution du véhicule dans le rond point c'est-à-dire pour  $t > 140s$ . Dans ce cas, comme pour l'approche avec le filtre de Kalman, l'algorithme de "vision" se retrouve en dehors de ses limites de fonctionnement car le rayon de courbure de la route est trop petit et il ne peut fournir une estimation de la position et de l'orientation du véhicule sur la chaussée au filtre particulière. De plus, d'après la modélisation AR des incertitudes GPS, les erreurs GPS évoluent très lentement dans le temps ce qui à impose qu'à court terme les mises à jour selon les informations GPS n'ont que

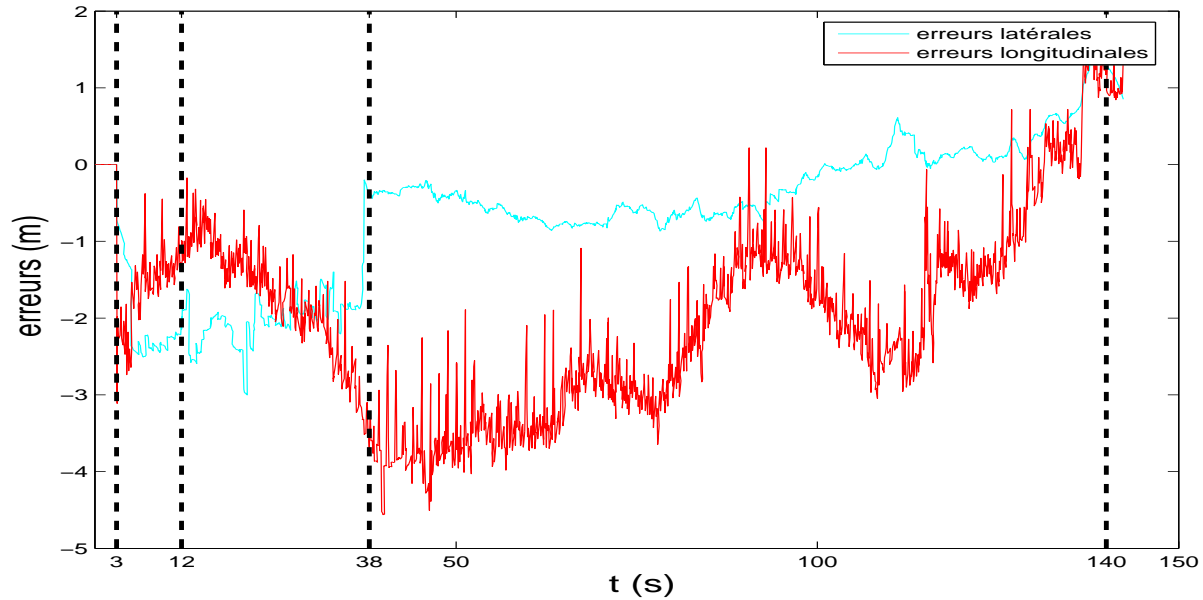


FIG. 6.7: Erreurs latérales et longitudinales

	Moyenne	Ecart-type
erreurs latérales (m)	-0.0171	0.5476
erreurs longitudinales (m)	-1.4239	1.1965

TAB. 6.1: Moyenne et écart-type des erreurs de localisation

peu d'influence sur la localisation, qui est alors en grande partie réalisée en fonction des informations provenant des capteurs proprioceptifs et de la carte. Ainsi au fur et à mesure que le véhicule évolue dans le rond point, les erreurs proprioceptives s'accumulent et la position estimée peut s'éloigner de la véritable position. Mais comme la carte émet l'hypothèse que le véhicule appartient toujours à la route, le phénomène de divergence de la navigation à l'estime est atténué et contrairement au filtre de Kalman, quand le véhicule ressort du rond point, la position estimée appartient toujours à la voie de circulation et est très proche de la véritable position du véhicule.

### 6.6.1.2 Le capteur "vision" est indisponible

La figure 6.9 présente les résultats de localisation lorsque le capteur "vision" n'est pas pris en compte par le système. De  $t = 0$  à  $38s$ , les résultats sont similaires à ceux de la situation précédente. Au delà de  $t = 38s$ , l'hypothèse d'appartenance à la route apportée par la carte permet au filtre particulière de déterminer sur quelle voie de circulation évolue le véhicule, les erreurs latérales restent alors faibles et sont comprises entre  $-1m$  et  $1m$ . En revanche, la carte ne peut émettre une hypothèse précise sur la position longitudinale du véhicule sur la chaussée, les erreurs longitudinales restent importantes mais sont obli-



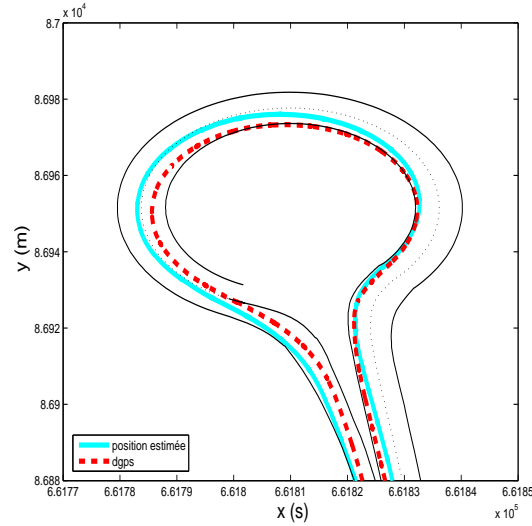


FIG. 6.8: Evolution du véhicule dans le rond point

gatoirement comprises entre  $\pm EURE_{gps} \times GDOP_{gps}$ .

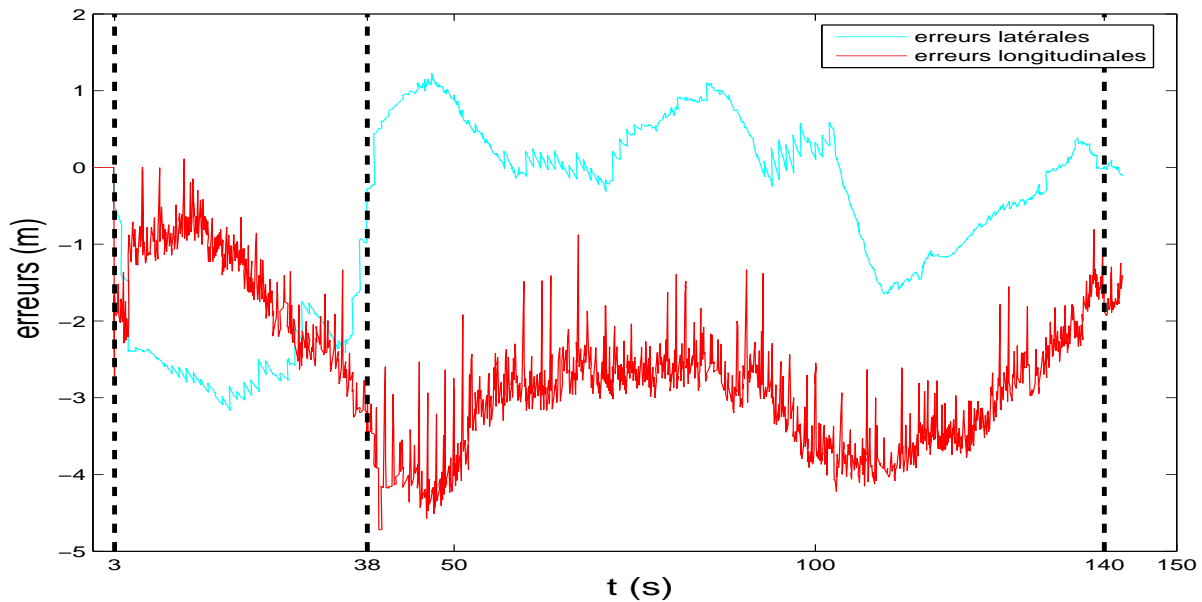


FIG. 6.9: Erreurs latérales et longitudinales : "vision" indisponible

La figure 6.10 montre l'évolution du véhicule dans le rond point. Dans cette situation, la trajectoire estimée est très proche de la trajectoire fournie par le DGPS. En effet la prise en compte de la carte a permis au filtre particulaire de fournir une estimation proche de la véritable position du véhicule à l'entrée du rond point. Le comportement de l'approche de localisation est alors similaire à la situation précédente et l'estimation de la position

du véhicule tout au long du rond point reste correcte.

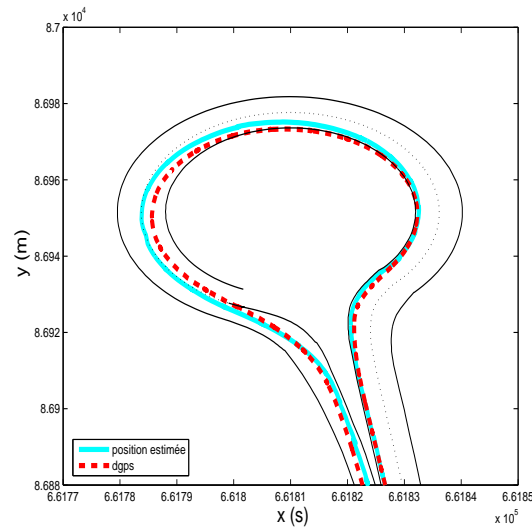


FIG. 6.10: Evolution du véhicule dans le rond point : vision indisponible

Contrairement à l'approche de localisation avec filtre de Kalman, grâce à la prise en compte de la carte cette approche permet une localisation assez précise du véhicule malgré que le capteur "vision" ne fournisse pas d'information.

### 6.6.1.3 Le capteur "vision" et le GPS subissent des pertes d'informations

Dans les deux prochains paragraphes, va être montrée l'influence de pertes d'informations extéroceptives sur la localisation. Ainsi, comme pour la section 5.6.1.3, nous allons simuler dans un premier temps la perte d'informations GPS, puis dans un deuxième temps la perte d'informations "vision" pendant un laps de temps allant de  $t \approx 38s$  à  $t \approx 100s$ .

**Perte d'informations GPS** La figure 6.11 présentent les résultats de localisation lorsque le système de fusion subit des pertes d'informations GPS :

- $t = 0$  à  $38s$  les résultats sont similaires aux résultats présentés dans la section 5.6.1.1, ainsi grâce au recoupement des informations de chacun des capteurs et au changement de voie effectué par le véhicule, le filtre particulier a déterminé précisément la position latérale du véhicule sur la chaussée. En revanche, à cause de la multimodalité des informations "vision", le filtre n'a pas eu le temps de converger vers la bonne position longitudinale et l'erreur longitudinale est grande et proche de  $-3m$ .
- $t = 38$  à  $100s$  : une perte d'informations GPS est simulée, la localisation est alors réalisée en fonction des informations provenant des capteurs proprioceptifs, du capteur "vision" et de la carte. Comme le capteur "vision" et la carte ne fournissent pas d'information sur la position longitudinale, l'estimation de celle-ci dépend principalement de l'étape de prédiction mais aussi du recoupement des jeux de particules a

priori successifs et des densités de vraisemblance  $p(\underline{Y}_k | \text{"vision"})$  et  $p(\underline{Y}_k | \text{"carte"})$ . De la même manière que le recouplement des états a priori et des observations "vision" pour l'approche utilisant le filtre de Kalman (cf section 5.6.1.3), on peut distinguer deux situations de recouplement du jeux de particules et de ces densités de vraisemblance :

1. si le véhicule évolue dans une ligne droite, le recouplement des jeux de particules a priori avec ces deux densités de vraisemblance ne permet pas d'estimer la position longitudinale et donc son estimation diverge en fonction de l'accumulation des erreurs de prédiction. Ce phénomène est visible dans les résultats de  $t \approx 38s$  à  $t \approx 75s$ .
  2. au contraire si le véhicule évolue sur route sinueuse le recouplement des jeux de particules avec ces deux densités de vraisemblance permet d'estimer la position longitudinale et donc de converger vers la vraie solution. Ce phénomène est visible dans les résultats entre  $t \approx 75s$  et  $t \approx 100s$ .
- après  $t = 100s$  le système reconverge de nouveau vers la véritable solution et les résultats redeviennent similaires à ceux de la section 5.6.1.1.

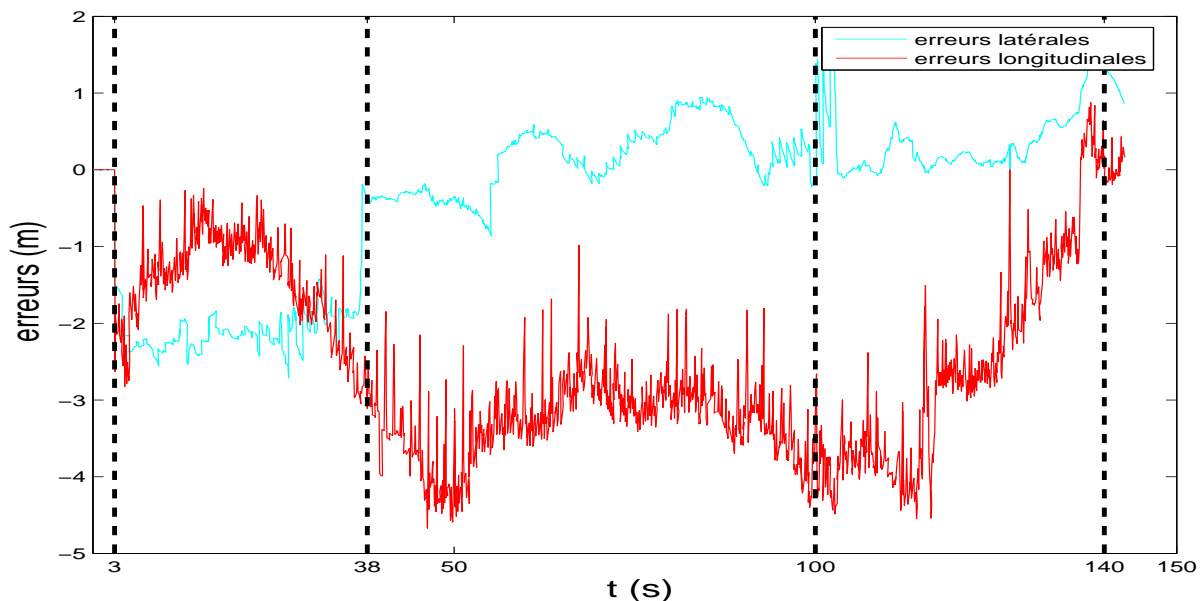


FIG. 6.11: Erreurs latérales et longitudinales : pertes d'informations GPS

**Perte d'informations "vision"** La figure 6.12 présente les résultats de la localisation lors de pertes d'informations "vision" :

- de  $t = 0$  à  $38s$  : les résultats sont similaires à ceux de la section 5.7,
- de  $t = 38$  à  $100s$  : une perte d'information "vision" est simulée, la localisation est alors réalisée en fonction des informations proprioceptives, du GPS et de la carte. A l'instant  $t = 38s$ , le système a convergé vers la même solution que pour la situation précédente c'est-à-dire une erreur latérale proche de 0 et une longitudinale de

- 3m. Pendant la perte d'informations "vision", les paramètres de biais  $b_x$  et  $b_y$  sont impossibles à estimer et évoluent selon la modélisation AR des incertitudes GPS. Donc à court terme, pour chaque mise à jour de l'état en fonction des informations GPS, les estimées sont pratiquement biaisées de la même façon et les informations GPS ont donc peu d'influence sur la localisation du véhicule. Les erreurs latérales et longitudinales dépendent alors en grande partie des erreurs introduites par l'étape de prédiction et des informations provenant de la carte. Comme le véhicule évolue sur une portion de route ne comportant que des virages à grand rayon de courbure et des lignes droites, l'étape de prédiction n'introduit que peu d'erreurs dans l'estimée et la prise en compte de la carte n'apporte que peu d'information sur la position du véhicule. Les erreurs latérales et longitudinales évoluent donc très peu.
- après  $t = 100s$  le système reconverge de nouveau vers la véritable solution et les résultats redeviennent similaires à ceux de la section 5.6.1.1.

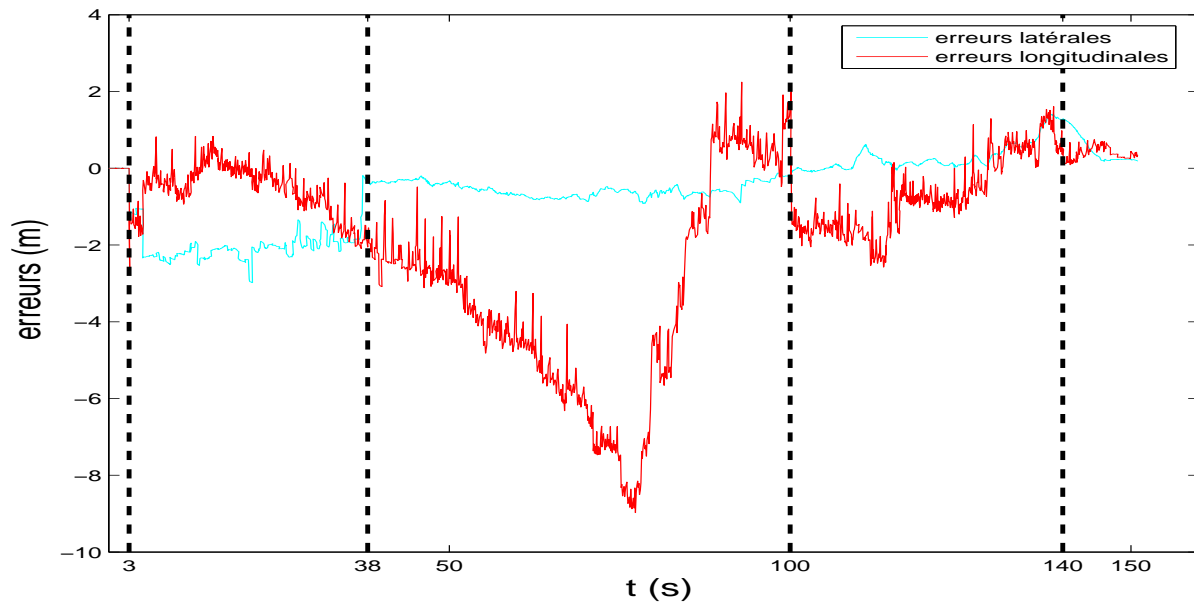


FIG. 6.12: Erreurs latérales et longitudinales : pertes d'informations "vision"

D'après ce dernier résultat, lorsque le capteur "vision" ne fournit plus d'informations, l'estimation de la position du véhicule à court terme diverge en fonction de l'accumulation des erreurs de l'étape de prédiction en revanche à long terme (conditions climatiques difficiles : neige, brouillard etc...), la prise en compte de la carte permet d'obtenir une estimation satisfaisante de la position du véhicule comme le montrent les résultats de la section 6.6.1.2.

De par ces résultats, on peut noter que comme pour l'approche avec filtre de Kalman, le système est plus robuste envers les pertes d'informations "vision" qu'envers les pertes d'informations GPS. Ceci est attribué au fait que les informations provenant du capteur "vision" et de la carte ne disposent pas de toutes les données nécessaires à une localisation

absolue (position longitudinale du véhicule manquant dans les deux cas). Là encore, pour pallier ce problème, il est nécessaire d'intégrer dans le processus de fusion des informations complémentaires à celles du capteur "vision". Ces informations pourraient être obtenues par la détection d'éléments remarquables situés dans l'environnement du véhicule (les panneaux de signalisation, les feux de circulations, passages cloutés).

### 6.6.2 Localisation sans la modélisation AR des incertitudes GPS

Dans cette section, sont présentés les résultats de la localisation lorsque la partie stochastique des informations GPS n'est pas prise en compte dans le processus de fusion. Pour ceci, il suffit de considérer que les incertitudes GPS sont régies par la loi gaussienne  $\mathcal{N}(0, \sigma_{x_{gps}}^2)$  et de supprimer tous les paramètres liés à la modélisation AR des incertitudes GPS dans les diverses étapes du filtre particulière. Enfin, en tenant compte de la relation (6.26), la densité de vraisemblance  $p(\underline{Y}_{gps} | \underline{X}_k^n)$  peut se réécrire :

$$p(\underline{Y}_{gps} | \underline{X}_k^n) \propto e^{-\frac{1}{2} \left[ \frac{(x_{gps} - x_k^n)^2}{\sigma_{x_{gps}}^2} + \frac{(y_{gps} - y_k^n)^2}{\sigma_{x_{gps}}^2} \right]} \quad (6.33)$$

La figure 6.13 montre les résultats d'un tel système de localisation :

- de  $t = 0$  à  $38s$  : l'erreur latérale est comprise entre  $\pm 3.5m$  car les informations "vision" et de la carte sont prises en compte. A  $t = 38s$  le véhicule effectue son premier changement de voie, le filtre particulière peut déterminer sur quelle voie de circulation évolue le véhicule et l'erreur latérale tend brutalement vers 0. En revanche comme le capteur "vision" et la carte ne fournissent pas d'informations sur la position longitudinale, les erreurs longitudinales sont importantes et dépendent directement du fait que les informations GPS sont mal modélisées.
- après  $t = 38s$  : il n'y a pas de changement significatif dans le comportement du filtre, les erreurs latérales varient peu, par contre les erreurs longitudinales évoluent dans le temps et sont fonction du biais introduit par la partie stochastique des informations GPS. Ces erreurs sont par contre toujours comprises entre  $\pm EURE_{gps} \times GDOP$ .

Comme pour l'approche de localisation avec filtre de Kalman, en comparant ces résultats avec les résultats de la section 6.6.1.1, on s'aperçoit qu'il est absolument nécessaire d'intégrer la modélisation AR des incertitudes GPS dans le processus de fusion pour obtenir une localisation correcte du véhicule et notamment une estimation correcte de la position longitudinale.

### 6.6.3 Multimodalité et changement de voie

Dans cette section, sont présentés les effets de la multimodalité des informations "vision" sur le comportement de l'approche de localisation. La figure 6.14(b) présente le jeu de particules lorsque la première information est prise en compte. On peut voir que celui-ci est partagé en deux modes qui sont disposés identiquement sur chacune des voies de circulation. Ainsi comme le montre la figure 6.14(d) l'estimée position latérale fournie par le

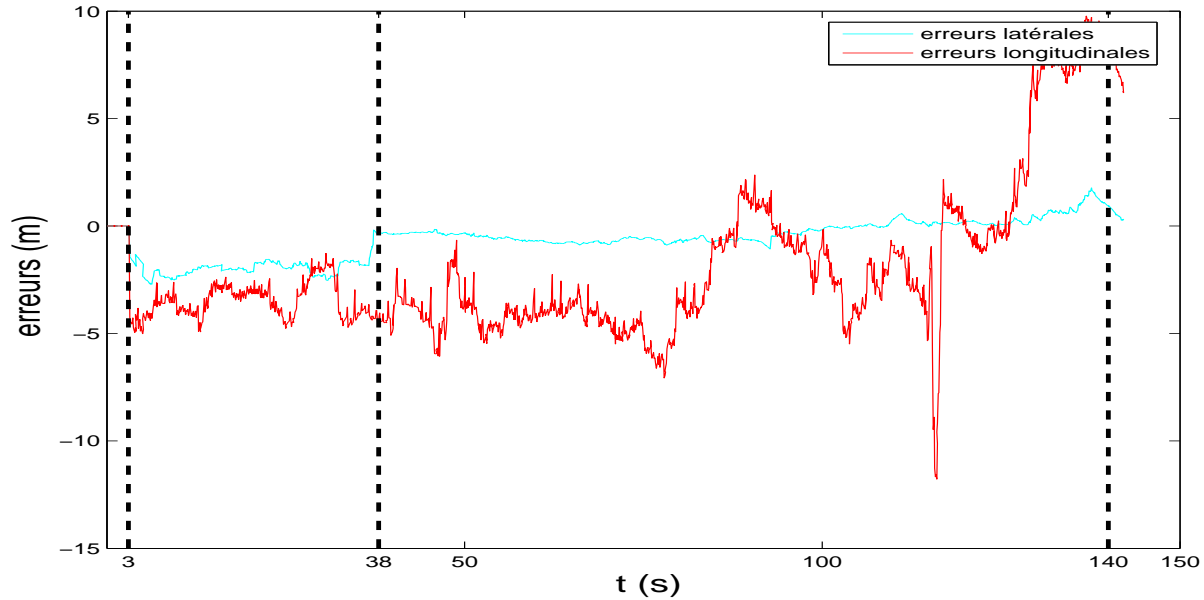


FIG. 6.13: Erreurs latérales et longitudinales : sans modélisation AR des incertitudes GPS

filtre particulière se situe au milieu des deux modes, comme le véhicule est au centre de sa voie de circulation la position latérale est alors le centre de la chaussée. Les figures 6.14(b) et 6.14(c) montrent respectivement l'état du jeu de particules juste avant le changement de voie et juste après le changement de voie. Comme le véhicule est en réalité sur la voie de droite et que le véhicule se déporte sur la voie de gauche pour réaliser son changement de voie, chacun des modes du jeu de particules se déportent eux aussi sur la gauche. Ainsi pendant le changement de voie, le mode de gauche sort petit à petit de la route et est éliminé par le filtre particulière lors des étapes de rééchantillonnage au profit du mode de droite. Lorsque le véhicule a complètement changé de voie, le jeux de particules s'est regroupé en un seul mode et la position latérale du véhicule est correctement estimée comme le montre la figure 6.14(d).

Contrairement à l'approche avec filtre de Kalman, le filtre particulière peut prendre en compte la multimodalité des informations car il traite directement les informations provenant des différents capteurs en fonction de leur distribution de probabilité. Ainsi lorsque la première information "vision" est fournie au système de fusion, il n'est pas obligé de choisir par défaut une voie de circulation comme pour le filtre de Kalman. Ainsi le filtre ne peut converger vers une mauvaise solution. Seuls le déplacement du véhicule sur la chaussée peut éliminer les modes du jeux particules et donc faire converger vers la bonne solution.

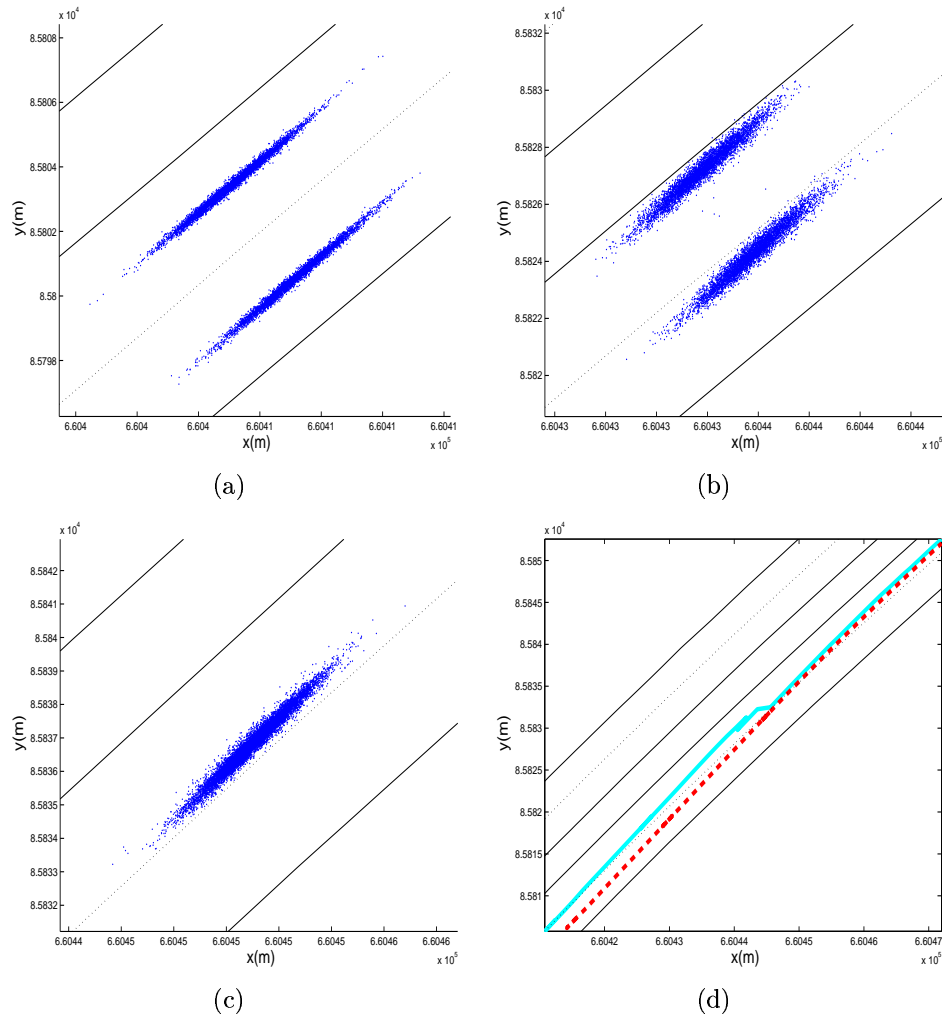


FIG. 6.14: Multimodalité des informations "vision" et changement de voie, figure (a) choix du bon mode, figure (b) choix du mauvais mode, figures (c) et (d) changement de voies

#### 6.6.4 Légitimité du filtre particulaire génétique vis-à-vis du filtre particulaire avec interaction

Cette section présente les performances de notre application de localisation selon que la fusion de données est effectuée par le filtre particulaire génétique ou par le filtre particulaire avec interaction et fournit un avis sur la légitimité de l'utilisation du filtre particulaire génétique dans notre système. L'ensemble des résultats présentés ont été obtenus à partir d'une seule base de données expérimentales et l'algorithme est ici testé sur un ordinateur portable pentium 4 cadencé à 2Ghz.

D'après le tableau 6.2, le filtre particulaire génétique converge vers une solution viable (n'est pas affecté par le phénomène de dégénérescence) lorsque le nombre de particules est supérieur à 500 alors que le filtre particulaire avec interaction traditionnel ne converge

que pour un nombre de particules supérieur ou égal à 5000. Ainsi le filtre particulaire génétique nécessite dix fois moins de particules que le filtre particulaire avec interaction pour assurer la convergence du filtre. Du point de vue de la précision de l'estimée, celle-ci est similaire quel que soit le filtre particulaire et quel que soit le nombre de particules avec un écart-type d'environ 0.56m sur la position latérale et d'environ 1m sur la position longitudinale. Selon ces résultats le filtre particulaire génétique n'améliore pas la précision du filtrage mais par contre nécessite beaucoup moins de particules pour obtenir le même résultat que le filtre particulaire avec interaction. On peut donc s'attendre à ce que ce filtre soit plus rapide que le filtre particulaire avec interaction. Enfin par sécurité nous pouvons considérer que le nombre de particules nominal pour le filtre particulaire génétique est de 1000 particules et que pour le filtre particulaire avec interaction il est de 10000 particules.

Filtre particulaire génétique		
Nombre de particules	Erreurs latérales (m)	Erreurs longitudinales (m)
	écart-type	écart-type
10000	0.568	0.653
5000	0.569	0.975
2000	0.553	0.842
1000	0.558	0.883
500	0.568	1.201
< 500	Le filtre de converge plus	
Filtre particulaire avec interaction		
Nombre de particules	Erreurs latérales en (m)	Erreur longitudinales (m)
	écart-type	écart-type
10000	0.553	1.437
5000	0.546	1.306
< 5000	Le filtre ne converge plus	

TAB. 6.2: Précision du système de localisation en fonction du nombre de particules et du filtre particulaire utilisé

Le tableau 6.3 présente les temps d'exécution de chacune des étapes du système de fusion de données, c'est à dire les temps d'exécution de l'étape de prédiction, des étapes de correction selon les informations "vision" ou GPS et l'étape de rééchantillonnage mis par chacun de ces deux filtres particulaires. Considérons dans un premier temps que ces deux filtres utilisent le même nombre de particules et qu'il soit égal au nombre de particules nominal du filtre particulaire avec interaction (N=10000). Dans ce cas, seuls les temps d'exécution de l'étape de prédiction diffèrent d'un filtre à l'autre. En effet pour le filtre particulaire avec interaction celui-ci est constant quelle que soit la nature de



la dernière information extéroceptive et vaut environ  $t^p = 15ms$ , en revanche pour le filtre particulière génétique, il varie en fonction de cette dernière information. Ainsi, si la dernière information est :

- une information "vision" :

$$t_{vision}^p \approx t^p + t^g + t_{vision}^c \approx 1.453s \quad (6.34)$$

- une information "GPS" :

$$t_{GPS}^p \approx t^p + t^g + t_{GPS}^c \approx 20ms \quad (6.35)$$

avec

- $t^g$  le temps d'exécution de l'étape de mutation génétique,
- $t_{vision}^c$  le temps d'exécution de l'étape de correction selon les informations "vision",
- $t_{GPS}^c$  le temps d'exécution de l'étape de correction selon les informations GPS.

Considérons maintenant notre système de localisation dans un contexte réel, ainsi pour une seconde d'acquisition de données, le système de fusion de données doit traiter 30 informations "vision" et 1 information GPS. Dans ce cas, pour effectuer cette tâche le temps global est pour :

- le filtre particulière génétique :

$$t_{GPF} \approx 30 \times (t_{vision}^c + t_{vision}^p) + t_{GPS}^c + t_{GPS}^p + 31 \times t^r \approx 83s \quad (6.36)$$

- le filtre particulière avec interaction :

$$t_{SIR} \approx 30 \times (t_{vision}^c + t^p) + t_{GPS}^c + t^p + 31 \times t^r \approx 40s \quad (6.37)$$

C'est à dire qu'à nombre de particules égal, le filtre particulière génétique met environ deux fois plus de temps que le filtre particulière avec interaction pour traiter l'ensemble de ces informations et que son utilisation n'est alors pas justifiée. Cependant, d'après les résultats précédents, le nombre nominal de particules pour le filtre particulière génétique est de 1000 particules. Comme l'ensemble des étapes du filtre particulière génétique a une complexité algorithmique en  $O(N)$ , pour ce nombre de particules celui-ci doit être 10 fois plus rapide que pour 10000 particules et donc environ cinq fois plus rapide que le filtre particulière avec interaction. Les résultats présentés dans le tableau 6.3 confirment ces hypothèses et prouve que le filtre particulière génétique est alors préférable au filtre particulière avec interaction dans notre application si l'on utilise 1000 particules.

Néanmoins, malgré le gain en temps de calcul apporté par le filtre particulière génétique, il faut encore 8.4s pour traiter 1s de données. Il est donc impossible d'utiliser cette approche de localisation en temps réel sans une architecture informatique dédiée. Bien sûr on peut parier que l'évolution technologique du matériel informatique devrait pallier très rapidement ce problème.

Etape	SIR (N=10000)	GPF (N=10000)	GPF(N=1000)
Prédiction après GPS	0.015s	0.020s	0.0019s
Prédiction après "vision"	0.015s	1.453s	0.150s
Correction GPS	0.002	0.002s	0.0002s
Correction "vision"	1.300s	1.300s	0.130s
Rééchantillonnage	0.008s	0.008s	0.0007s
Total pour 1s d'acquisition de données	83s	30s	8.4s

TAB. 6.3: Temps d'exécution des différentes étapes du filtre particulaire avec interaction (SIR) et du filtre particulaire génétique (GPF)

## 6.7 Conclusion

L'approche présentée dans ce chapitre permet de localiser le véhicule avec une précision similaire à l'approche de localisation avec filtre de Kalman, à condition qu'encore une fois tous les capteurs extéroceptifs fournissent régulièrement des informations au processus de fusion et que la partie stochastique des informations GPS soit correctement modélisée (modélisation AR dans notre situation) et intégrée au filtre.

Les problèmes rencontrés avec le filtre de Kalman semblent être résolus par l'utilisation d'un filtre particulaire ainsi :

1. la correction en fonction des informations "vision" est ici optimale car le filtre particulaire sait aisément gérer leur multimodalité. Ainsi le système converge vers la bonne solution une fois que les diverses "fausses" hypothèses émises par le capteur "vision" sont éliminées par le système de fusion (changement de voies, changement de situations routières 2x2 voies vers 1x1 voies, etc ...),
2. la prise en compte de la carte dans le processus de fusion permet d'obtenir une localisation du véhicule relativement précise dans n'importe quelle condition et notamment lors des pertes d'informations extéroceptives.

Enfin le système dynamique de notre approche est très faiblement bruité (incertitudes des capteurs proprioceptifs relativement faibles), ce qui pose bien des problèmes dans le choix du filtre particulaire. En effet, dans ce cas le filtre particulaire avec interaction SIR rencontre des problèmes de dégénérescence et ne peut estimer correctement la position du véhicule sans un nombre important de particules (N=10000) et donc avec un temps d'exécution incompatible pour un système temps réel. Pour pallier à la fois le phénomène de dégénérescence et diminuer le nombre de particules, nous avons choisi d'effectuer la fusion de données avec un filtre particulaire génétique. De cette manière notre système de localisation fournit une estimation de la position du véhicule aussi précise qu'avec un filtre particulaire avec interaction mais en utilisant seulement 1000 particules. Dans ce cas, le gain en temps de calcul par rapport au filtre particulaire avec interaction est de 5,

néanmoins malgré cette amélioration ce système ne peut être à l'heure actuelle utilisé en temps réel.

# Chapitre 7

## Conclusion et perspectives

Le travail présenté dans ce mémoire est une contribution pour la conception de systèmes de localisation précis de véhicules routiers répondant aux contraintes de l'industrie de l'automobile, c'est-à-dire un système de faible coût, facile à mettre en œuvre et robuste. La conception de ce système repose sur une approche de fusion de données multisensorielles combinant des informations provenant de capteurs proprioceptifs tels qu'un odomètre et un gyromètre, de capteurs extéroceptifs tels qu'un GPS naturel faible coût et un système de "vision" ainsi que d'une carte précise de l'environnement routier. La principale originalité de cette approche vient de l'utilisation du système de vision (appelé capteur "vision") qui fournit entre autre des informations très précises sur l'attitude du véhicule sur la chaussée telles que sa position latérale et son orientation. Cette attitude, une fois mise en correspondance avec la carte de l'environnement donne des informations de localisation à la fois complémentaires et redondantes aux informations GPS. Ainsi le capteur "vision" permet de suppléer celui-ci lorsque l'intégrité de ses informations est fortement remise en cause (contexte urbain) et d'atteindre une précision submétrique.

Pour réaliser la combinaison d'information nous nous sommes tournés vers les méthodes classiques d'estimation bayésienne comme le filtre de Kalman et ses extensions pour les méthodes analytiques ou les filtres particuliers pour les méthodes numériques. De là deux différentes approches ont été développées. La première approche de localisation utilise le filtre de Kalman étendu pour réaliser la fusion de données, cette technique est la plus usitée dans les problèmes de localisation car elle très simple à mettre en œuvre et peu gourmande en temps de calcul. Cependant, malgré de bons résultats en terme de précision et un fonctionnement temps réel, un certain nombre de problèmes peuvent être attribués à son utilisation :

- problème de linéarisation de l'équation d'observation "vision" car la position longitudinale du véhicule sur la route est absente des données "vision",
- problème de gestion de la multimodalité des informations "vision" (contexte autoroutier) qui n'est pas garantie,
- enfin l'impossibilité de prendre en compte la carte pour atténuer le phénomène de dérive lié au positionnement à l'estime lorsque que les capteurs extéroceptifs sont dans l'incapacité de fournir des informations (conditions climatiques difficiles telles

que le brouillard, la neige pour le capteur "vision" ou contexte urbain pour le GPS).

Pour pallier ces différents inconvénients, une deuxième approche a été proposée. Cette fois-ci, la fusion de données se base sur un filtre particulaire avec interaction. Ce choix se justifie par le fait que cette méthode de filtrage peut 1) combiner des informations de distributions de probabilité quelconques et donc gérer facilement les informations multi-modales fournies par le capteur "vision" dans les contextes autoroutiers, 2) prendre en compte la carte de manière explicite pour contraindre la localisation du véhicule en cas de perte d'informations extéroceptives. Malgré ces avantages et des résultats similaires en terme de précision avec la méthode précédente, un nouveau problème apparaît avec cette solution. En effet, le rapport signal/bruit du système dynamique de notre approche est très grand (incertitudes sur les informations proprioceptives faibles) ce qui impose l'utilisation d'un nombre élevé de particules pour que le filtre particulaire ne connaisse pas de problèmes de dégénérescence. Dans ce cas les temps de calculs des diverses étapes du filtre particulaire deviennent prohibitifs empêchant la localisation temps réel du véhicule. Pour endiguer ce problème, nous proposons l'utilisation d'une méthode de filtrage originale qui combine filtre particulaire et la théorie de l'évolution génétique. D'après nos résultats expérimentaux, ce nouveau filtre souvent appelé dans la littérature filtre particulaire génétique n'apporte que peu d'amélioration en terme de précision par rapport au filtre particulaire avec interaction traditionnel mais en revanche nécessite dix fois moins de particules que celui-ci pour un fonctionnement optimal. Cette diminution du nombre de particules apporte un gain de cinq en temps de calcul par rapport au filtre particulaire avec interaction mais reste pour l'instant insuffisante pour une utilisation temps réel de cette deuxième approche.

Naturellement ces résultats n'ont pu être obtenus sans une caractérisation méticuleuse des informations fournies par chacun des capteurs. Ainsi contrairement à la plupart des systèmes de localisation couplant GPS et capteurs proprioceptifs, la prise en compte du caractère stochastique des informations GPS permet à notre système d'atteindre une précision équivalente à celle d'un GPS différentiel. La modélisation des incertitudes gyrométriques (constante aléatoire + bruit blanc gaussien) ainsi que la prise en compte des caractéristiques temporelles des informations (latences du GPS et du capteur "vision") contribuent elles aussi à la précision du système. En effet elles permettent respectivement d'atténuer le phénomène de dérive de la localisation à l'estime ou encore d'obtenir une fusion de données cohérente dans le temps bien que celles-ci soient traitées de manière asynchrone.

Cependant la modélisation des incertitudes gyrométriques est discutable car elle n'est réaliste que pour une courte durée d'utilisation du système. Ce problème pourrait être résolu en modélisant les incertitudes gyrométriques par un processus markovien (processus de gauss-markov, autorégressif, etc...) plus un bruit blanc gaussien et en estimant le processus markovien (ou le biais du gyromètre) de la même manière que le biais du GPS. Ainsi la phase d'apprentissage permettant de déterminer la valeur initiale de la constante aléatoire ne serait plus nécessaire et le système deviendrait complètement autonome. De

plus si une information de cap (boussole magnétique) est utilisée pour estimer le biais du gyromètre, celle-ci pourrait par la même occasion améliorer le comportement de l'approche avec filtre particulière en réduisant l'intervalle de recherche dans l'espace d'état de l'orientation du véhicule et donc d'assurer une convergence plus rapide du filtre.

Enfin de nombreuses améliorations sont envisageables pour ce système de localisation. La première pourrait être une interaction plus forte entre la carte et l'algorithme de vision. En effet, bien que le modèle paramétrique clothoïdal permette de représenter la géométrie de la plupart des routes, dans le contexte urbain l'algorithme de vision se retrouve limité car les intersections, rond points etc... ne sont plus modélisables par celui-ci et l'estimation de l'attitude du véhicule sur la chaussée devient alors impossible. Dans ce cas, on pourrait envisager d'utiliser la carte pour fournir des informations sur la géométrie de la route et orienter le processus de recherche de l'algorithme de vision selon le contexte routier dans lequel se situe le véhicule. Ainsi le capteur "vision" pourrait localiser le véhicule et donc suppléer le GPS dans pratiquement tous les contextes routiers. De plus, la carte pourrait fournir des informations sur la nature des bords de route au capteur "vision" (lignes blanches, trottoirs, bords naturel) afin de faciliter la détection de primitives dans l'image.

Une autre amélioration de notre système serait la détection soit par vision soit par d'autres capteurs extéroceptifs de nouveaux éléments remarquables de l'environnement routier tel que les panneaux signalétiques, les feux, les passages cloutés etc... Cette solution permettrait d'améliorer la précision et la robustesse du système de localisation et plus particulièrement de pouvoir estimer correctement la position longitudinale du véhicule sur la chaussée palliant ainsi une des limitations les plus importantes de l'utilisation du capteur "vision" dans notre approche. Bien sûr, là encore, la carte pourrait fournir des informations sur la nature de ces éléments remarquables au système de localisation permettant ainsi un choix intelligent des capteurs et des méthodes de détection en fonction de celle-ci.

Dans tous ces cas de figure, un enrichissement de la carte est nécessaire. Elle ne peut plus être alors considérée comme une simple carte vectorielle mais comme une carte "hypermap" contenant une description précise des éléments remarquables de l'environnement routier comme leur position, l'incertitude sur leur position ou encore leur nature. On peut naturellement envisager d'enrichir des cartes préexistantes comme les cartes Navtech auquel cas, chaque élément remarquable serait rattaché à un segment de route de la carte ou encore redéfinir un nouveau format de carte spécialement dédié au système de localisation multisensoriel.



# Troisième partie

## Annexes





# Annexe A

## Apprentissage de l'algorithme de vision

### Définitions des repères

Les figures A.1 et A.2 montrent les repères utilisés dans l'étape d'apprentissage. Le repère  $\mathcal{R}_R : (X, Y, Z)^T$  est un repère lié à la route dont le bord gauche est tangent à l'axe  $Y$ , le repère  $\mathcal{R}_C : (U, V, W)^T$  est lié à la caméra où le plan  $UW$  est parallèle au plan image et l'axe  $V$  est confondu avec l'axe optique.

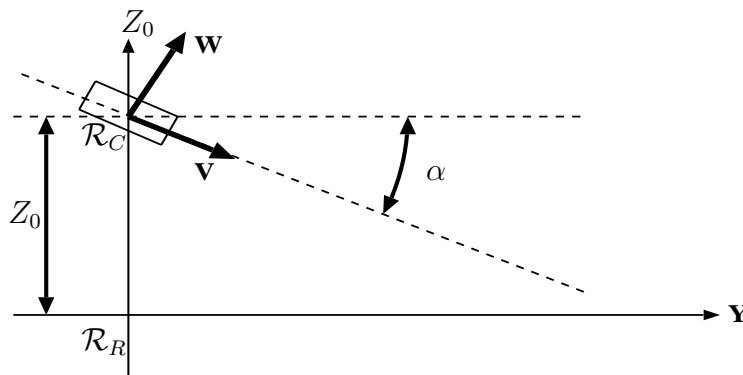


FIG. A.1: Repères utilisés vue de coté

où

- $L$  : la largeur de la route,
- $X_0$  : la position latérale du véhicule sur la route,
- $\Psi_0$  : l'orientation du véhicule sur la route,
- $\alpha$  : l'inclinaison de la caméra,
- $Z_0$  : la hauteur de la caméra par rapport à la route.

Si l'on considère que la route est plane et que sa courbure est représentée par un modèle clothoïdal, alors tout point 3D  $(X, Y, Z)^T$  appartenant à l'un des bord de la route

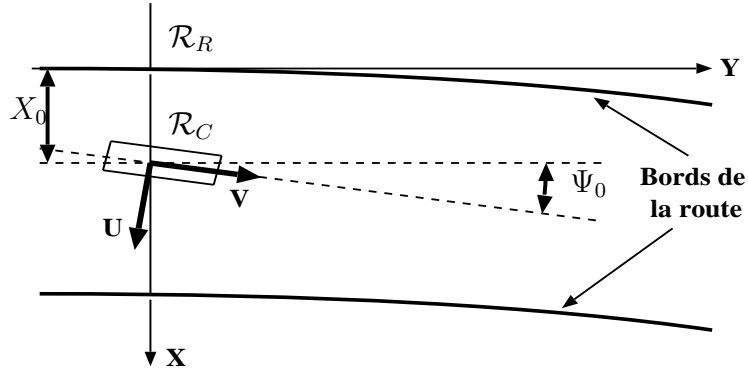


FIG. A.2: Repères utilisés vue de dessus

est totalement défini par les relations :

$$X = \frac{C_h Y^2}{2} + \frac{C_l Y^3}{6} + \lambda L \quad \text{et} \quad Z = 0 \quad (\text{A.1})$$

avec  $\lambda = 0$  pour le bord droit de la route et  $\lambda = -1$  pour le bord gauche de la route.

## Calcul du vecteur moyen

Le vecteur moyen  $\underline{X}_{v_0}$  est constitué par la concaténation des vecteurs  $\underline{X}_{l_0}$  et  $\underline{X}_{d_0}$ .  $\underline{X}_{l_0}$  étant parfaitement connu on peut donc décomposer le calcul du vecteur moyen comme ceci :

$$\underline{X}_{v_0} = \begin{cases} f_l(\underline{X}_{l_0}) \rightarrow f_l(\cdot) = \mathbf{I} & \text{pour } \underline{X}_{l_0} \\ f_d(e_u, e_v, Z_0, \underline{X}_{l_0}) & \text{pour } \underline{X}_{d_0} \end{cases} \quad (\text{A.2})$$

La fonction  $\underline{X}_{d_0} = f_d(e_u, e_v, Z_0, \underline{X}_{l_0})$  permet de calculer les abscisses de chacun des segments dans l'image en fonction des paramètres 3D de la scène routière. Néanmoins, avant de définir cette fonction, il nous faut connaître les équations de projection d'un point 3D appartenant aux bords de la route dans le repère image.

Soit un point 3D  $\underline{P}_R = (X, Y, Z = 0)^T$  défini dans le repère de la route et  $\underline{P}_C = (U, V, W)^T$  sa projection dans le repère caméra en tenant compte du fait que l'axe optique caméra est situé en  $(X_0, 0, Z_0)$  et que la caméra peut subir des rotations selon l'angle de tangage  $\alpha$  et l'angle de lacet  $\Psi_0$  alors :

$$P_C = \begin{pmatrix} \cos(\Psi_0) & \sin(\Psi_0) & 0 \\ -\sin(\Psi_0) & \cos(\Psi_0) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} X - X_0 \\ Y \\ -Z_0 \end{pmatrix} \quad (\text{A.3})$$

$$P_{Rc} = \begin{pmatrix} \cos(\Psi_0)(X - X_0) + \sin(\Psi_0)Y \\ -\cos(\alpha)\sin(\Psi_0)(X - X_0) + \cos(\alpha)\cos(\Psi_0)Y + \sin(\alpha)Z_0 \\ -\sin(\alpha)\sin(\Psi_0)(X - X_0) + \sin(\alpha)\cos(\Psi_0)Y - \cos(\alpha)Z_0 \end{pmatrix} \quad (\text{A.4})$$

Le point 3D  $\underline{P}_R = (X, Y, 0)^T$  a pour projection dans le plan image le point  $\underline{P}_i = (u, v)^T$ . Connaissant les paramètres intrinsèques  $e_u$  et  $e_v$  de la caméra, les coordonnées  $u$  et  $v$  sont données par les équations de projection suivantes :

$$u = \frac{e_u \cos(\Psi_0)(X - X_0) + \sin(\Psi_0)Y}{-\cos(\alpha) \sin(\Psi_0)(X - X_0) + \cos(\alpha) \cos(\Psi_0)Y + \sin(\alpha)Z_0} \quad (\text{A.5})$$

$$v = \frac{e_v(-\sin(\alpha) \sin(\Psi_0)(X - X_0) + \sin(\alpha) \cos(\Psi_0)Y - \cos(\alpha)Z_0)}{-\cos(\alpha) \sin(\Psi_0)(X - X_0) + \cos(\alpha) \cos(\Psi_0)Y + \sin(\alpha)Z_0} \quad (\text{A.6})$$

Maintenant supposons que  $\sin(\alpha)Z_0$  et  $\sin(\Psi_0)(X - X_0)$  soient toujours inférieurs à  $Y$ . Ces hypothèses sont réalistes dans les contextes routiers que l'on a empruntés lors des expérimentations, à savoir  $3m < Y < 100m$ ,  $-0.01 < C_h < 0.01$ ,  $-3.5 < X_0 < 3.5m$ . Les équations de projection deviennent alors :

$$u = \frac{e_u \cos(\Psi_0)(X - X_0) + \sin(\Psi_0)Y}{\cos(\alpha) \cos(\Psi_0)Y} \quad (\text{A.7})$$

$$v = \frac{e_v \sin(\alpha) \cos(\Psi_0)Y - \cos(\alpha)Z_0}{\cos(\alpha) \cos(\Psi_0)Y} \quad (\text{A.8})$$

Revenons sur la fonction  $\underline{X}_{d_0} = f_d(e_u, e_v, Z_0, \underline{X}_{l_0})$ . Cette fonction permet de calculer les abscisses des extrémités des segments représentant les bords de la route dans l'image. Sachant que leurs ordonnées sont fixes et connues, la variable  $Y$  peut être déduite de la relation (A.8) :

$$Y = \frac{e_v \cos(\alpha)Z_0}{\cos(\Psi_0)(-v \cos(\alpha) + e_v \sin(\alpha))} \quad (\text{A.9})$$

Si on réinjecte  $Y$  dans l'équation (A.7) et que nous prenons comme hypothèse que la route suit une trajectoire clothoïdale (à savoir  $X = Ch\frac{Y^2}{2} + Cl\frac{Y^3}{6}$ ) alors les abscisses du bord droit de la route  $u_{i_d}$  et du bord gauche de la route  $u_{i_g}$  sont données par les équations ci-dessous :

$$u_{i_d} = e_u \frac{\left( \cos \Psi_0 \left( \frac{C_h Y}{2} + \frac{C_l Y^2}{6} - X_0 \right) - \sin(\Psi_0) \right)}{\cos(\Psi_0) \cos(\alpha)} \quad (\text{A.10})$$

et

$$u_{i_g} = e_u \frac{\left( \cos \Psi_0 \left( \frac{C_h Y}{2} + \frac{C_l Y^2}{6} - X_0 - \lambda L \right) - \sin(\Psi_0) \right)}{\cos(\Psi_0) \cos(\alpha)} \quad (\text{A.11})$$

avec  $Y_i$  les valeurs de  $Y$  en fonction des différentes valeurs des ordonnées  $v_i$ .

Ainsi la fonction  $\underline{X}_{d_0} = f_d(e_u, e_v, Z_0, \underline{X}_{l_0})$  est caractérisée par le jeu d'équations permettant de calculer les abscisses  $u_{1_d}$  à  $u_{n_d}$  pour les  $n$  segments du bord droit et les abscisses  $u_{1_g}$  à  $u_{n_g}$  pour les  $n$  segments du bord gauche.

## Calcul de la matrice de covariance

La matrice de covariance définissant les variations possibles de l'ensemble des paramètres en fonction des dispersions autorisées autour du vecteur moyen  $\underline{X}_{l_0}$  est donnée par l'équation suivante :

$$\mathbf{P}_{v_0} = \mathbf{J}_C \mathbf{P}_{l_0} \mathbf{J}_C^T \quad (\text{A.12})$$

où le Jacobien  $\mathbf{J}_c$  est le gradient de la fonction  $\underline{X}_{v_0} = f_d(e_u, e_v, Z_0, \underline{X}_{l_0})$  selon chacun des paramètres du vecteur moyen  $\underline{X}_{l_0}$ .

$$\begin{pmatrix} \frac{\partial u_{1g}}{\partial L} & \frac{\partial u_{1g}}{\partial X_0} & \frac{\partial u_{1g}}{\partial \Psi_0} & \frac{\partial u_{1g}}{\partial \alpha} & \frac{\partial u_{1g}}{\partial C_h} & \frac{\partial u_{1g}}{\partial C_l} \\ \frac{\partial \dot{u}_{ng}}{\partial L} & \frac{\partial \dot{u}_{ng}}{\partial X_0} & \frac{\partial \dot{u}_{ng}}{\partial \Psi_0} & \frac{\partial \dot{u}_{ng}}{\partial \alpha} & \frac{\partial \dot{u}_{ng}}{\partial C_h} & \frac{\partial \dot{u}_{ng}}{\partial C_l} \\ \frac{\partial u_{1d}}{\partial L} & \frac{\partial u_{1d}}{\partial X_0} & \frac{\partial u_{1d}}{\partial \Psi_0} & \frac{\partial u_{1d}}{\partial \alpha} & \frac{\partial u_{1d}}{\partial C_h} & \frac{\partial u_{1d}}{\partial C_l} \\ \frac{\partial \dot{u}_{nd}}{\partial L} & \frac{\partial \dot{u}_{nd}}{\partial X_0} & \frac{\partial \dot{u}_{nd}}{\partial \Psi_0} & \frac{\partial \dot{u}_{nd}}{\partial \alpha} & \frac{\partial \dot{u}_{nd}}{\partial C_h} & \frac{\partial \dot{u}_{nd}}{\partial C_l} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{A.13})$$

On notera que cette matrice de covariance peut être mise sous la forme :

$$\mathbf{P}_{v_0} = \begin{pmatrix} \mathbf{P}_{d_0} & \mathbf{P}_{ld_0} \\ \mathbf{P}_{dl_0} & \mathbf{P}_{l_0} \end{pmatrix}. \quad (\text{A.14})$$

# Annexe B

## Systemes géodésiques et projections cartographiques associées

Deux problèmes se posent lorsque l'on souhaite obtenir une localisation absolue d'un objet sur la terre :

1. Notre planète ne peut être réellement décrite par un volume de forme régulière. Elle est en générale assimilée à une géoïde ou à une ellipsoïde :
  - la géoïde est une surface équipotentielle coïncidant avec le niveau moyen de la mer et en chaque point est perpendiculaire à la direction de la pesanteur,
  - l'ellipsoïde est une surface mathématique coïncidant aussi bien que possible avec la géoïde, celle-ci est en générale caractérisée par différents paramètres comme son demi-grand axe, son demi-petit axe, son aplatissement, son excentricité etc... Selon l'endroit où l'on se situe sur le globe terrestre, différents modèles d'ellipsoïdes peuvent être utilisés pour coïncider au mieux localement avec la géoïde.
2. Dans la plupart des applications de localisation en environnement extérieur, le monde est localement représenté sous la forme d'une carte plane. Dans ce cas une loi de conversion entre les coordonnées géodésiques absolues et les coordonnées planes est nécessaire. Là encore selon l'endroit où l'on se situe, différents systèmes de projection peuvent être utilisés.

### Les systèmes géodésiques

#### Description des systèmes de coordonnées

Les coordonnées d'un objet sur le globe terrestre peuvent être exprimées :

1. selon la forme de coordonnées géographiques : dans ce cas les coordonnées sont basées sur une référence d'ellipsoïde de révolution, qui est une approximation de la forme de la Terre et expriment la position de l'objet en termes de latitude, de longitude et de hauteur :

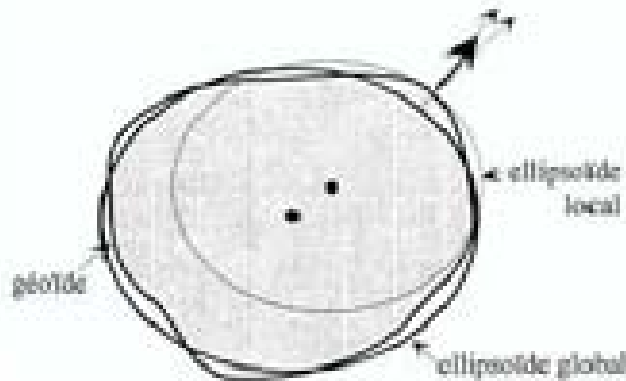


FIG. B.1: Géoïdes et ellipsoïdes de la Terre

- la latitude  $\varphi$  : est l'angle entre le plan équatorial et la normale à la surface de la Terre (l'ellipsoïde) au point en question, est égal à zéro à l'équateur et est compté positif pour l'hémisphère Nord, négatif pour l'hémisphère Sud.
  - la longitude  $\lambda$  : est l'angle entre le méridien de Greenwich et le méridien du point en question. Il est compté positif vers l'est.
  - la hauteur  $h$  : définit la hauteur ellipsoïdale (ne pas confondre avec l'altitude), est la différence en mètre entre le point considéré et l'ellipsoïde de référence mesurée selon la normale à l'ellipsoïde. Cette valeur est définie dans un système géodésique et peut différer de plusieurs dizaines mètres avec l'altitude. Enfin il existe des modèles de géoïde permettant de passer de la hauteur ellipsoïdale à l'altitude mais seulement avec une précision décimétrique.
2. selon les coordonnées cartésiennes géocentriques : dans ce cas, les coordonnées sont décrites par la position cartésienne  $(X, Y, Z)^T$  donnée dans un repère ayant son origine au centre des masses de la Terre. Ces coordonnées ne sont pas fournies par le GPS mais sont souvent utilisées comme intermédiaire lors des calculs de changement de systèmes géodésiques de référence.
  3. selon les coordonnées locales : dans ce cas les coordonnées sont décrites par la position  $(E, N, U)^T$  donnée dans un repère cartésien tangent à la géoïde et orienté selon les directions East, North et Up (cf figure B.3).

### Passage coordonnées géographiques $\Leftrightarrow$ géocentriques

La conversion entre ces deux systèmes de coordonnées requiert une approximation géométrique de la Terre (de la géoïde). En général celle-ci est caractérisée par un ellipsoïde dont la forme est contrôlée par un certain nombre de paramètres tels que son demi-grand axe  $a$ , son demi-petit axe  $b$ , son aplatissement et son excentricité. Ainsi le passage des

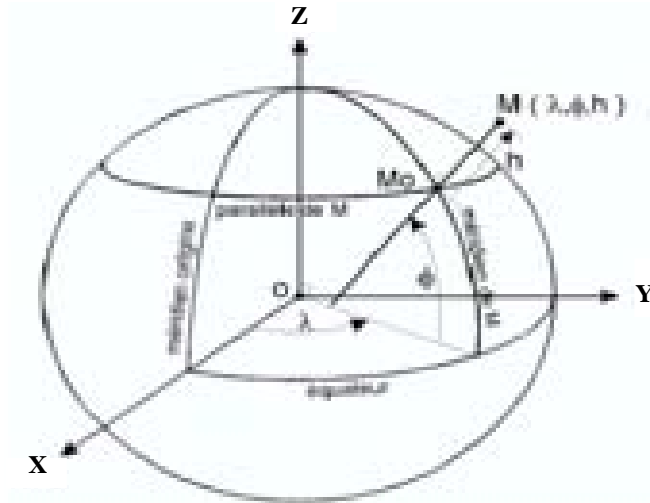


FIG. B.2: Système de coordonnées géographiques et géocentriques

coordonnées géographiques aux coordonnées géocentriques est donné par :

$$\begin{aligned} X &= (N + h) \cos \varphi \cos \lambda \\ Y &= (N + h) \cos \varphi \sin \lambda \\ Z &= [(1 - f)^2 N + h] \sin \varphi \end{aligned} \quad (\text{B.1})$$

et le passage des coordonnées géocentriques aux coordonnées géographiques est donné par :

$$\begin{aligned} \lambda &= \arctan \left( \frac{Y}{X} \right) \\ \varphi &= \arctan \left( \frac{Z}{\sqrt{X^2 + Y^2}} \left( 1 - \frac{(2 - f)fN}{N + h} \right)^{-1} \right) \\ h &= \frac{\sqrt{X^2 + Y^2}}{\cos \varphi} - N \end{aligned} \quad (\text{B.2})$$

où  $f = (a - b)/a$  est le coefficient d'aplatissement de l'ellipsoïde et  $N = a / \sqrt{1 - f(2 - f) \sin^2 \varphi}$  est la distance séparant le point M et l'axe polaire mesurée selon la direction perpendiculaire à l'ellipsoïde passant par le point M.

## Les indicateurs de précision DOP

En général, il est plus instructif pour l'utilisateur d'un GPS de disposer des coordonnées du mobile dans un système de coordonnées locales, ces systèmes sont nombreux (UTM, Lambert, RGF93 etc...) et seront présentés dans les prochaines sections de cette annexe. En ce qui concerne les indicateurs de précision, ils sont toujours calculés dans le référentiel de coordonnées locales ENU (cf figure B.3) et sont déterminés de la manière



suivante : considérons  $\underline{X} = (x, y, z)^T$  le vecteur d'erreur de position dans le système de coordonnées géocentriques, le vecteur  $\underline{X}_L = (e, n, u)$  correspondant dans le référentiel de coordonnées locales est donné par le produit :

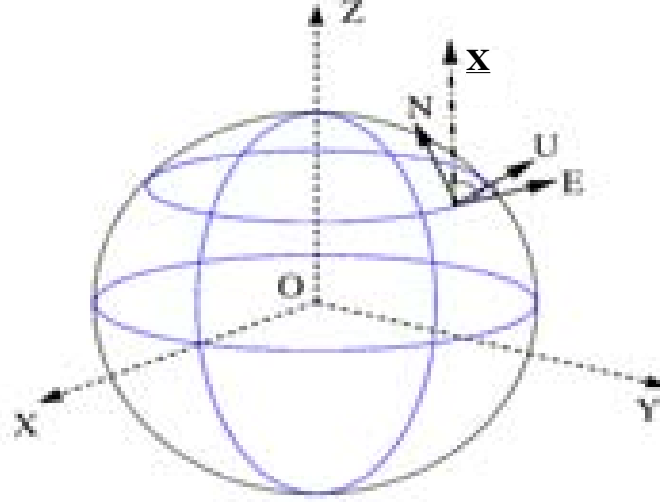


FIG. B.3: Système de coordonnées géocentriques et locales

$$\underline{X}_L = \mathbf{F}^T \underline{X} \quad (\text{B.3})$$

où  $\mathbf{F}$  est la matrice de transformation :

$$\mathbf{F} = \begin{pmatrix} -\cos \lambda & \sin \lambda & 0 \\ -\sin \lambda & -\cos \lambda & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \sin \varphi & 0 & -\cos \varphi \\ 0 & 1 & 0 \\ \cos \varphi & 0 & -\cos \varphi \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{B.4})$$

En tenant compte de cette relation et en connaissant la matrice de covariance associée au vecteur d'erreur dans le référentiel de coordonnées géocentriques, il est possible de déterminer la covariance des mesures dans le référentiel de coordonnées locales ainsi :

$$\text{cov}(e, n, u) = \mathbf{F}^T \text{cov}(x, y, z) \mathbf{F} \quad (\text{B.5})$$

soit :

$$\begin{pmatrix} \sigma_e^2 & \sigma_e \sigma_n & \sigma_e \sigma_h \\ \sigma_n \sigma_e & \sigma_n^2 & \sigma_n \sigma_h \\ \sigma_h \sigma_e & \sigma_h \sigma_n & \sigma_h^2 \end{pmatrix} = \mathbf{F}^T \begin{pmatrix} \sigma_x^2 & \sigma_x \sigma_y & \sigma_x \sigma_z \\ \sigma_y \sigma_x & \sigma_y^2 & \sigma_y \sigma_z \\ \sigma_z \sigma_x & \sigma_z \sigma_y & \sigma_z^2 \end{pmatrix} \mathbf{F} \quad (\text{B.6})$$

Cette matrice de covariance permet de fournir des indicateurs instructifs et fiables sur la précision de la position calculée par le récepteur GPS. Ces indicateurs se nomment DOP<sup>1</sup>

<sup>1</sup>Dilution Of Precision

et sont :

$$\begin{aligned}
 \text{GDOP} &= \sqrt{\frac{\sigma_e^2 + \sigma_n^2 + \sigma_h^2 + \sigma_{c\Delta T}^2}{\sigma}} && : \text{Dilution de précision géométrique} \\
 \text{PDOP} &= \sqrt{\frac{\sigma_e^2 + \sigma_n^2 + \sigma_h^2}{\sigma}} && : \text{Dilution de précision de position} \\
 \text{HDOP} &= \sqrt{\frac{\sigma_e^2 + \sigma_n^2}{\sigma}} && : \text{Dilution de précision horizontale} \\
 \text{VDOP} &= \frac{\sigma_h}{\sigma} && : \text{Dilution de précision verticale}
 \end{aligned} \tag{B.7}$$

où  $\sigma^2$  est la variance de l'erreur globale de position et  $\sigma_{c\Delta T}^2$  la variance de l'erreur de synchronisation du temps. Ces indicateurs permettent de savoir, pour une configuration donnée des satellites visibles comment l'erreur globale de position se distribue selon les directions du système de coordonnées locales, c'est-à-dire que ce soit horizontalement, verticalement, les deux à la fois, ou encore en tenant compte des erreurs de synchronisation.

## Le système WGS84 et projection UTM

Comme ses initiales l'indiquent, le WGS84 est un système de référence global terrestre. Mis au point par l'armée américaine en 1984, il est l'évolution du WGS72 développé une décennie plus tôt. Avec ce système géodésique, la planète est représentée sous la forme d'une ellipsoïde référencée sous le nom IAG-GRS80 dont les propriétés sont décrites dans le tableau B.1

Demi grand-axe $a$	6378137.0m
Demi petit-axe $b$	6356752.314m
Inverse de l'aplatissement $1/f$	298.257222101
Excentricité $e$	0.08181919106

TAB. B.1: Propriétés de l'ellipsoïde IAG-GRS80

La représentation cartographique associée à ce système est réalisée par la projection de l'ellipsoïde IAG-GRS80 sur un cylindre (cf figure B.5(a)), celle-ci est appelée la projection UTM<sup>2</sup>. La carte ainsi obtenue est constituée de 60 fuseaux couvrant 6° en longitude. La France est située sur 3 de ces fuseaux (figure B.4) :

1. UTM Nord fuseau 30 : entre 6 degrés ouest et 0 degré Greenwich,
2. UTM Nord fuseau 31 : entre 0 degré et 6 degrés est Greenwich,
3. UTM Nord fuseau 32 : entre 6 degrés et 12 degrés est Greenwich.

<sup>2</sup>Universal Transversal Mercator

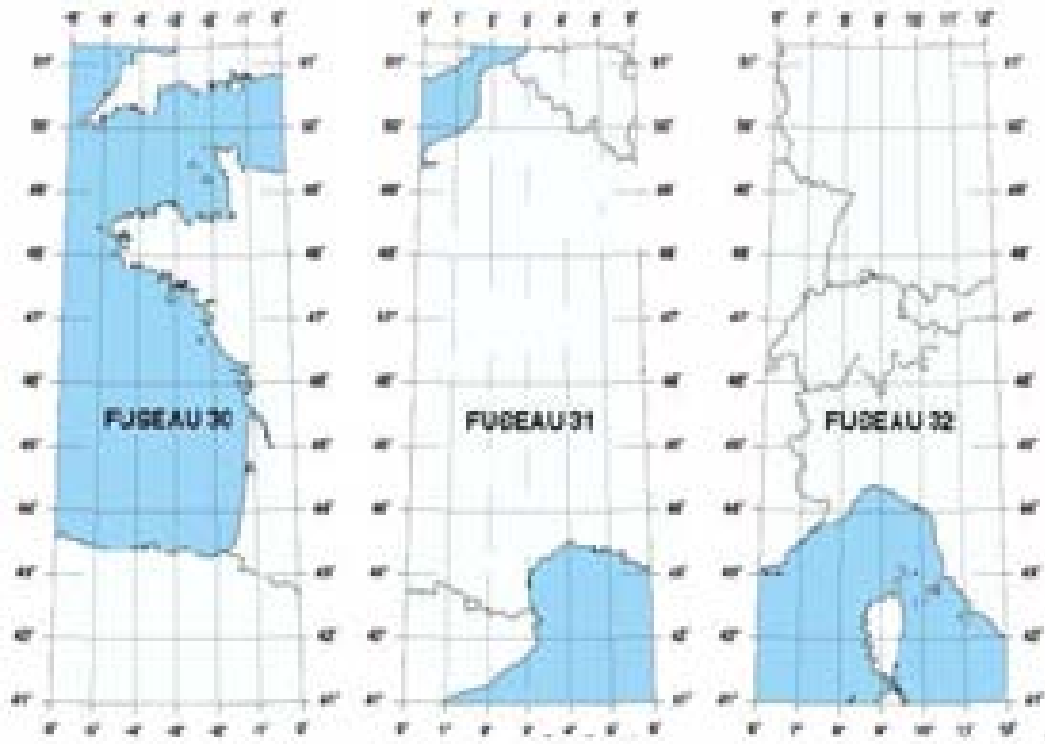


FIG. B.4: La projection UTM

## La Nouvelle Triangulation Française NTF et projection Lambert

Ce système de coordonnées géodésiques est destiné plus particulièrement au positionnement en France. Dans ce cas la Terre est représentée par l'ellipsoïde Clarke 1880 IGN, dont les propriétés sont décrites dans le tableau B.2.

Demi grand-axe $a$	6378388.0m
Demi petit-axe $b$	6356911.9461m
Inverse de l'aplatissement $1/f$	297.00000
Excentricité $e$	0.08199188998

TAB. B.2: Propriétés de l'ellipsoïde Clarke 1880 IGN

La représentation cartographique associée à ce système est réalisée par la projection Lambert, définie comme une conique tangente (cf figure B.5(b)) à l'ellipsoïde Clarke 1880 IGN. Dans le but de minimiser les déformations, la France a été découpée en 4 zones. Dans ce cas la longitude d'origine est toujours le méridien de Paris et les latitudes d'origine<sup>3</sup>

<sup>3</sup>Latitude pour laquelle le cône de projection est tangente à la terre

sont  $49^{\circ}30'$ ,  $46^{\circ}48'$ ,  $46^{\circ}06'$ , et  $42^{\circ}09'54''$  respectivement pour les projections Lambert I, II, III et IV.

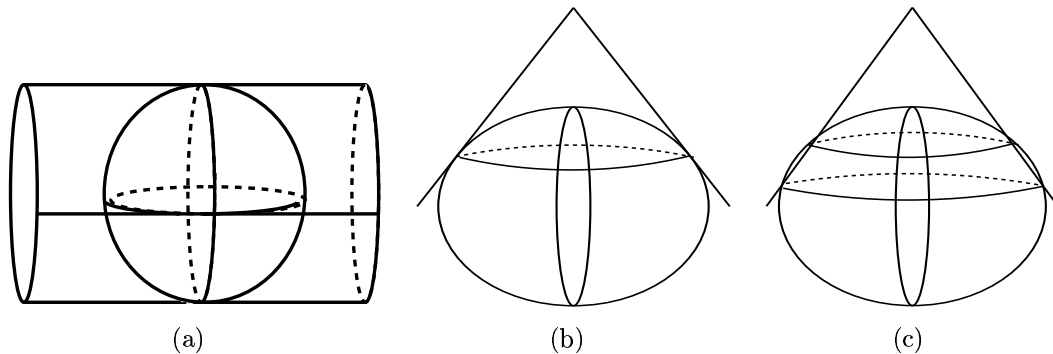


FIG. B.5: Les différents types de projections cartographiques, (a) projection sur un cylindrique, (b) projection sur une conique tangente, (c) projection sur une conique sécante

## Le système RGF93 et projection Lambert 93

Ce système de coordonnées à la fois tridimensionnel et géocentrique sert de base à la création d'un réseau géodésique moderne français, par densification des points européens du réseau mondial géodésique ETRS89<sup>4</sup>. Les coordonnées de ce réseau sont exprimées soit sous la forme tridimensionnelle géodésique (longitude, latitude, hauteur ellipsoïdale), soit sous la forme bidimensionnelle cartésienne selon la projection Lambert 93 unique pour tout le territoire. Cette projection est ici définie par une projection conique sécante à l'ellipsoïde IAG GRS80 utilisée par le système WGS84 (figure B.5(c))

## Les formules de conversions Lambert

Le positionnement par GPS a pour particularité de fournir la position absolue du robot mobile (ou du véhicule) sur un ellipsoïde de référence représentant le globe terrestre. Or, comme dans notre application le mobile est localisé dans une carte référencée dans le système Lambert, une position fournie par le GPS décrite par le couple longitude/latitude

---

<sup>4</sup>European Terrestrial Reference System 1989

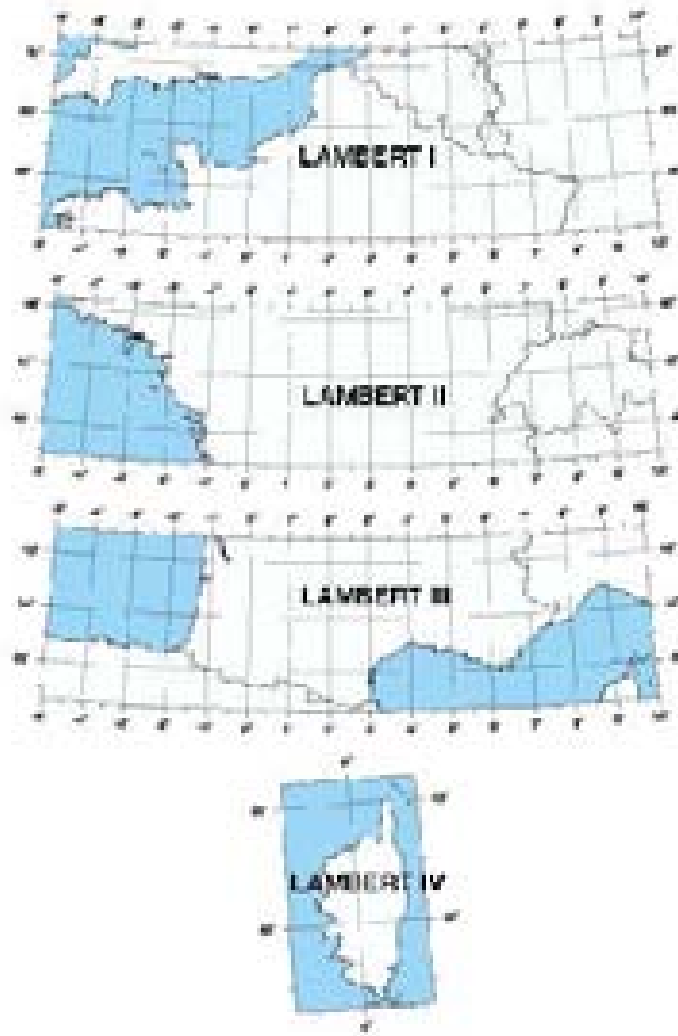


FIG. B.6: Les projections Lambert

doit être convertie dans ce nouveau référentiel :

$$l = \frac{1}{2} \ln \frac{1 + \sin \varphi}{1 - \sin \varphi} - \frac{e}{2} \ln \frac{1 + e \sin \varphi}{1 - e \sin \varphi}$$

$$R = C \exp(-nl)$$

$$\gamma = n(\lambda - \lambda_0) \tag{B.8}$$

$$X = X_s + R \sin \gamma$$

$$Y = Y_s + R \cos \gamma$$

où

- $n$ ,  $C$ ,  $X_s$ ,  $Y_s$  sont des constantes de la projection,
- $\lambda_0$  est la longitude du méridien central,
- $e$  est l'excentricité de l'ellipsoïde.

	Lambert I	Lambert II	Lambert III	Lambert IV	Lambert 93
$N$	0.7604059656	0.7289686274	0.6959127966	0.6712679322	0.7256077650
$C$	11603796.98	11745793.39	11947992.52	12136281.99	11754255.426
$X_s$	600000.000	600000.000	600000.000	234.358	700000.000
$Y_s$	5657616.674	6199695.768	6791905.085	7239161.542	12655612.050

TAB. B.3: Les constantes de projections Lambert

## Passage WGS84 $\Rightarrow$ NTF

Le plus souvent, le système géodésique NTF n'est pas implanté dans les récepteurs GPS, les coordonnées sont alors fournies dans le référentiel WGS84. Avant de réaliser toute projection cartographique Lambert, il convient d'effectuer un changement de repère de ces coordonnées géographiques dans le référentiel géographiques NTF. Ce changement de repère est en fait simplement défini par la translation suivante :

$$\begin{pmatrix} TX = -168m \\ TY = -60m \\ TZ = +320m \end{pmatrix} \quad (\text{B.9})$$

La précision sur ces nouvelles coordonnées est alors de l'ordre de quelques mètres. Si une précision centrimétrique est requise, l'utilisation d'une grille de paramètres<sup>5</sup> dans laquelle les valeurs de translation sont interpolées est nécessaire pour prendre en compte les déformations de la NTF.

---

<sup>5</sup>Grille de paramètre de transformation de coordonnées GR3DF97A



## Annexe C

# Le véhicule expérimental VELAC

Suite à son expérience dans le projet PROMETHEUS et à son envie de développer un programme de recherche ambitieux dans le domaine de la conduite automobile, le LASMEA s'est équipé en 1998 d'un véhicule expérimental nommé VELAC (Véhicule Expérimental du LASMEA pour l'Aide à la Conduite). Ce véhicule de type Citroën Evasion, permet une intégration des différents développements scientifiques réalisés au sein de notre laboratoire. Les différents thèmes de recherche abordés autour de ce démonstrateur sont essentiellement axés sur le thème de la perception, et notamment l'analyse précise de la scène présente aux environs du véhicule. Quelques travaux déjà réalisés dans ce contexte peuvent être cités : la détection de sortie de voie, la détection et suivi d'obstacles sur la chaussée (par vision ou par LIDAR), la localisation du véhicule sur la chaussée. La figure C.1 présente une vue d'ensemble de ce véhicule.



FIG. C.1: Le véhicule expérimental VELAC



## Energie

Une source d'énergie est indispensable pour alimenter l'ensemble du matériel embarqué dans ce véhicule. Actuellement, dans VELAC, cette source d'énergie est assurée par un alternateur spécifique couplé à deux batteries et un onduleur. L'alternateur est entraîné par une courroie placée sur une nouvelle poulie de l'arbre moteur tout spécialement installée pour ce véhicule (cf figure C.2). De plus, afin de garantir une sécurité suffisante, cet alternateur est isolé par rapport à la masse du véhicule.



FIG. C.2: L'alternateur

L'onduleur (24V vers 220V) est associé à deux batteries de 12V montées en série afin de fournir une puissance électrique de l'ordre de 1kW sous 220V. L'ensemble onduleur batterie permettant d'alimenter le matériel en 220V est placé dans un caisson à l'arrière du véhicule (cf figure C.3).



FIG. C.3: L'onduleur et les batteries

## Capteurs proprioceptifs

### L'odomètre

En réalité l'odomètre est réalisé par un capteur de vitesse disposé sur l'arbre de sortie de la boîte de vitesse du véhicule qui délivre un signal carré de fréquence proportionnelle à la vitesse de VELAC. Ce signal est traité par un microcontrôleur de type PIC qui détermine alors le nombre de tops odomètre (nombre de front montant du signal délivré par le capteur de vitesse). Enfin cette information est transmise au système informatique via un bus CAN. D'après la calibration de ce système deux tops odomètre correspondent à un déplacement de 19,6cm.

### Le gyromètre

Le gyromètre embarqué dans notre véhicule est un gyromètre Gyrostar ENVO5F (cf figure C.4) qui appartient à la technologie MEMS. Placé sous les sièges arrières selon l'axe longitudinal du véhicule il fournit des mesures de la vitesse de rotation du véhicule. Ces informations sont tout d'abord traitées et mises en forme par un microcontrôleur avant d'être envoyées au système informatique via un bus CAN.



FIG. C.4: Le gyromètre

## Capteurs extéroceptifs

### Les récepteurs GPS

Deux récepteurs GPS sont embarqués dans notre véhicule expérimental (cf figure C.5). Le premier est un récepteur Motorola Oncore à 8 voies qui permet une localisation n'importe où sur le globe terrestre avec une précision de quelques dizaines de mètres. Le second est un récepteur Omnistar 74LR12 à 12 voies qui se classe dans la classe des récepteurs WADGPS. Grâce aux corrections différentielles envoyées par le réseau satellitaire Omnistar, il permet une localisation avec une précision inférieure à 1m dans les zones couvertes

par ce réseau.



FIG. C.5: Les récepteurs GPS

Les informations fournies par ces deux capteurs sont tout d'abord traitées et mises en forme par un microcontrôleur avant d'être transmises à l'architecture informatique via un bus CAN. Les informations alors reçues par le système informatique sont les coordonnées géographiques du véhicule (dans le référentiel WGS84) délivrées par chacun de ces capteurs à savoir la longitude, la latitude et la hauteur ellipsoïdale ainsi que le facteur de déformation géométrique GDOP. Comme décrit dans l'annexe B, pour que ces informations soit utilisables par le système de localisation, le système informatique réalise alors une conversion des coordonnées géographiques en coordonnées locales (référentiel LAMBERT93)  $(X, Y)^T$  avant tout emploi de celles-ci.

## La caméra numérique

La caméra numérique embarquée dans VELAC est une SONY (DFW-VL500). Placée sur un socle amovible orientable au niveau du rétroviseur intérieur du véhicule, elle fournit une vue de la scène présente à l'avant du véhicule (cf figure C.6). Les images acquises (vitesse d'acquisition maximale de 30 images par seconde) par la caméra sont transmises à l'architecture informatique via un bus numérique de type IEEE 1394. De nombreux formats et tailles d'image sont disponibles avec cette caméra (cf tableau C.1).

De plus, grâce au bus IEEE 1394, l'architecture informatique peut contrôler de nombreux paramètres de la caméra tels que le zoom, le focus, le gain, le shutter ou encore l'iris. On peut noter que les trois derniers paramètres peuvent être mis en mode automatique offrant une aide très précieuse lors du traitement de l'image.

## Le télémètre LASER

Le télémètre laser embarqué dans notre véhicule est un RIEGL LMSZ210-60 (cf figure C.7). Placé dans le pare-chocs avant du véhicule il fournit une image de profondeur de la



FIG. C.6: La caméra numérique

tailles des images	320x240 et 480x640
format des images	YUV411, YUV422 ,YUV444 et RGB
paramètres com- mandables	zoom, focus, gain (auto), shutter (auto), iris (auto), balance des blancs (auto)

TAB. C.1: Caractéristiques de la caméra numérique

scène située à l'avant du véhicule. Les images acquises (vitesse d'acquisition maximale de 4Hz) par le télémètre sont transmises à l'architecture informatique via un bus Ethernet.



FIG. C.7: Le télémètre laser RIEGL LMSZ210-60

Le balayage du faisceau laser nécessaire à l'obtention des images 3D de la scène est réalisé par un jeu des prismes et de miroirs orientables, ainsi avec un angle de balayage horizontal de 60° en horizontal et de 4.3° en vertical, ce télémètre peut fournir une image de profondeur de la scène routière de taille maximale 103x20 avec une portée maximale de 100m (cf figure C.8). Enfin ce capteur est utilisé dans des applications d'anti-collisions

ou encore de SLAM<sup>1</sup>. Il n'est pas utilisé pour notre système de localisation.

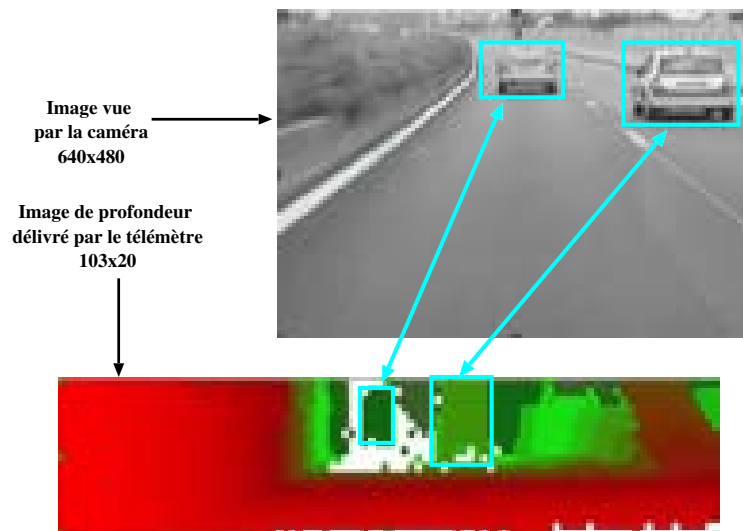


FIG. C.8: Images délivrés par le télémètre et la caméra

## Le RADAR

Le RADAR embarqué dans notre véhicule (cf figure C.9) a été conjointement réalisé par le LASMEA et la société RCS<sup>2</sup> dans le cadre du projet PAROTO<sup>3</sup>. Ce RADAR est un système traditionnel Pulse Doppler avec une longueur d'onde de 3.9mm (c'est-à-dire une fréquence d'émission de l'onde de 77GHz) conforme à la réglementation en vigueur dans l'Union Européenne et aux Etats-Unis. Placé dans le pare-chocs avant du véhicule il permet d'estimer la vitesse relative entre le véhicule porteur et les obstacles potentiels. De plus, ce RADAR a la particularité d'avoir une seule antenne émettrice-réceptrice qui nécessite l'utilisation d'un circulateur (système hyperfréquence) pour transférer les signaux soit entre l'émetteur et l'antenne, soit entre l'antenne et le receptrice.

## Architecture informatique

L'architecture informatique de VELAC est constituée d'un simple ordinateur placé dans un caisson à l'arrière du véhicule (cf figure C.10). Ses caractéristiques sont données dans le tableau C.2.

Cet ordinateur permet 1) l'acquisition des informations provenant des différents capteurs grâce aux différents bus implantés dans le véhicule (bus IEEE1394, bus CAN, bus

<sup>1</sup>Simultaneous Localization And Mapping

<sup>2</sup>Radar Communication Services

<sup>3</sup>Projet Anticollision Radar et Optronique pour l'auTOMobile



FIG. C.9: Radar et Lidar placés dans le pare-chocs

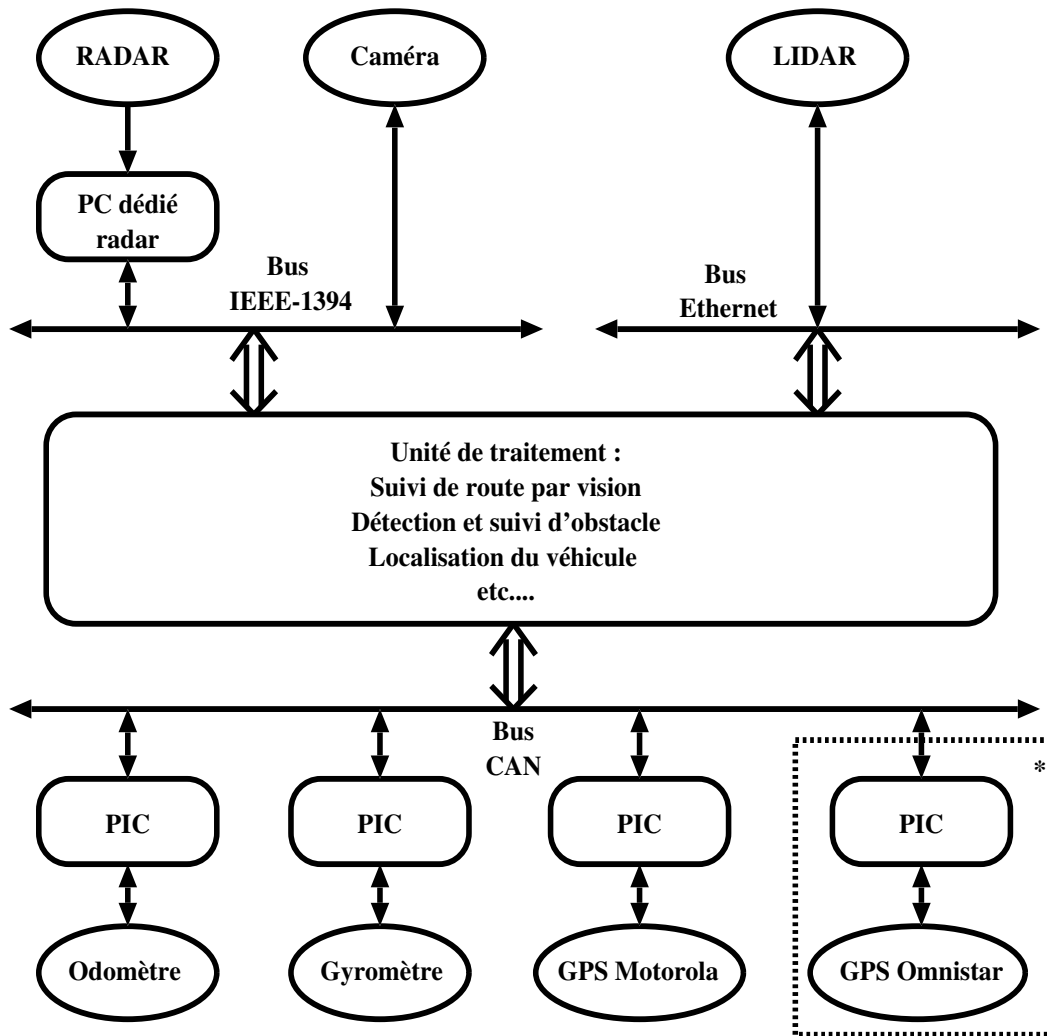
processeur	Intel Pentium 4 2.6 GHz
mémoire	1024 DDR 400 MHz PC3200
carte mère	ASUS P4P800
carte vidéo	Nvidia GF4 FX 5200 128MB
cartes PCI	carte IEEE1394 Texas Instruments, carte CAN PCI EMS
disque durs	UDMA 133 40GB + UDMA 133 120GB
système d'exploitation	SUSE linux

TAB. C.2: Caractéristiques de l'ordinateur embarqué dans VELAC

Ethernet), 2) d'effectuer en temps réel les différents algorithmes d'aide à la conduite développés au laboratoire. Un synoptique de cette architecture est présenté dans la figure C.11



FIG. C.10: Ordinateur embarqué



\*En cours d'intégration, actuellement ce récepteur GPS est relié à l'unité de traitement par le port série.

FIG. C.11: Synoptique de l'architecture informatique

# Bibliographie

- [Abousalem *et al.*, 2000] M. Abousalem, S. Lusin, O. Tubalin., and J. de Sala. Performance and analysis of gps positioning using waas and egnos. In *GNSS Conference*, Edimbourg, Ecosse, 2000.
- [Abuhadrous, 2005] Iyad Abuhadrous. *Système embarqué temps réel de localisation et de modélisation 3D par fusion multi-capteur*. PhD thesis, Ecole des Mines de Paris, Paris, Janvier 2005.
- [Akerblom, 2000] Christer Akerblom. Tracking mobile phones in urban area. Master's thesis, Université de Chalmers et de Göteborg, Suède, Septembre 2000.
- [Alspach *et al.*, 1972] D.L. Alspach and H.W. Sorenson. Non-linear bayesian estimation using gaussian sum approximation. *IEEE Transaction on Automatica Control*, 17 :439–447, 1972.
- [Anderson *et al.*, 1979] Brian. D. O. Anderson and John. B. Moore. *Electrical Engineering, Optimal Filtering*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, USA, 1979.
- [Apostoloff *et al.*, 2003] N. Apostoloff and A. Zelinsky. Robust vision based lane tracking using multiple cues and particle filtering. In *In Proceedings IEEE Intelligent Vehicles Symposium*, Juin 2003.
- [Arulampalam *et al.*, 2002] M.S. Arulampalam, S. Maskel, N. Gordon, and T.Clapp. A tutorial on particles filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transaction on Signal Processing*, 50(2) :174–188, 2002.
- [Aufrère *et al.*, 2000a] R. Aufrère, R. Chapuis, and F. Chausse. A dynamic vision algorithm to locate a vehicle on a non-structured road. *International Journal of Robotics Research*, 19(5) :411–423, 2000.
- [Aufrère *et al.*, 2000b] R. Aufrère, R. Chapuis, and F. Chausse. A fast and robust vision based road following algorithm. In *IEEE International Conference on Intelligent Vehicles*, volume 1, pages 192–197, DearBorn USA, 2000.
- [Aufrère, 2001] R. Aufrère. *Reconnaissance et suivi de route par vision artificielle, application à l'aide à la conduite*. PhD thesis, Université Blaise Pascal, Clermont-Ferrand, France, Juin 2001.
- [Barbour *et al.*, 1998] N. Barbour and G. Schmidt. Inertial sensor technology trends. In *Workshop on Autonomous Underwater Vehicles*, pages 55–62, Cambridge, 20-21 Août 1998.



- [Benedicto *et al.*, 1999] J. Benedicto, P. Michel, and J. Ventura-Traveset. Egnos the first european implementation of gnss : project overview status. *Air & Space Europe*, 1(1) :58–64, Janvier 1999.
- [Benedicto *et al.*, 2001] J. Benedicto and D. Ludwig. Galileo system and architecture. In *Digital Signal Processing Techniques for Space Communication Workshop*, Sessimbra, Protugal, 1-3 Octobre 2001.
- [Bertozzi *et al.*, 1998] M. Bertozzi and A. Broggi. Gold : a parallel real-time stereo vision system for generic obstacle and detection. *IEEE Transactions Image Processing*, 7 :62–81, Janvier 1998.
- [Bertozzi *et al.*, 2002] B. Bertozzi, A. Broggi, M. Cellario, A. Facioli, P. Lombardi, and M. Porta. Artificial vision in road vehicles. *Proceedings of the IEEE*, 90(7) :1258–1271, Juillet 2002.
- [Beucher *et al.*, 1994a] S. Beucher and M. Bilodeau. Road segmentation and obstacle detection by a fast watershed transformation. In *IEEE International Symposium on Intelligent Vehicles*, Paris, France, Octobre 1994.
- [Beucher *et al.*, 1994b] S. Beucher and X. Yu. Road recognition in complex traffic situations. In *IFAC/IFORS Symposium on Transportation Systems : Theory and Application of Advanced Technologie*, pages 413–418, Tianjin, China, Août 1994.
- [Boreinstein *et al.*, 1996a] J. Boreinstein, B. Everett, and L. Feng. Where am i ? Technical report, University of Michigan, Avril 1996.
- [Boreinstein *et al.*, 1996b] J. Boreinstein and L. Feng. Gyrodometry : A new method for combining data from gyros and odometry in mobile robot. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, pages 423–428, Minneapolis, Minnessota, Avril 1996.
- [Brenneke *et al.*, 2003] C. Brenneke, O. Wulf, and B. Wagner. Using 3d laser range data for slam in outdoor environments. In *International Conference on Intelligent Robots and Systems*, Las Vegas, USA, 2003.
- [Broggi *et al.*, 1999] A. Broggi, M. Bertozzi, A. Fascioli, and G. Conte. Automatic vehicle guidance : the experience of the argo vehicle. *World Scientific*, 1999.
- [Brown *et al.*, 1997] R.G. Brown and Y.C Hwang. *Introduction to random signals and applied Kalman filtering with Matlab exercises and solutions*. John Wiley and Sons, 3rd edition edition, 1997.
- [Burg, 1975] J.P. Burg. *Maximum entropy spectral analysis*. PhD thesis, Département de Géophysiques, université de Stanford, Standford, Californie, USA, 1975.
- [Carpenter *et al.*, 1999] J. Carpenter, P. Clifford, and Fearnhead. An improves particle filter for nonlinear problems. *IEE Processing Radar Sonar Navigation*, 146(1), 1999.
- [Cervantes, 2005] J. G. Avina Cervantes. *Navigation visuelle d'un robot mobile dans un environnement d'extérieur semi-structuré*. PhD thesis, Institut National de Polytechnique de Toulouse, 2005.

- [Chapuis *et al.*, 1995] R. Chapuis, A. Potelle, J.L. Brame, and F. Chausse. Realtime vehicle trajectory supervision on highway. *International Journal of Robotics Research*, 14 :531–542, Décembre 1995.
- [Chung *et al.*, 2001] H. Chung, L. Ojeda, and J. Borenstein. Sensor fusion for mobile robot dead-reckoning with a precision-calibrated fiber optic gyroscope. In *IEEE International Conference on Robotics and Automation*, volume 17-1, pages 3588–3593, Korea, Mai 2001.
- [Crisan *et al.*, 1999] D. Crisan, P. Del Moral, and T.J Lyons. Non linear filtering using granching and interacting particle systems. *Markov Processes Related Fields*, 5(3) :293–319, 1999.
- [Crisan *et al.*, 2002] D. Crisan and A. Doucet. Survey of convergence results on particle filtering methods for practitioners. *IEEE Transaction on Signal Processing*, 50(3) :736–746, 2002.
- [Crisman *et al.*, 1991] J.D. Crisman and C.E Thorpe. Unscarf a color vision system for the detection of unstructured roads. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2496–2501, April 1991.
- [Crisman *et al.*, 1993] J.D Crisman and C.E. Thorpe. Scarf : a color vision system that tracks road and intersections. *IEEE Transactions on Robotics and Automation*, 9(1) :49–58, Février 1993.
- [Delaigle, 1999] A. Delaigle. Bandwitch selection in kernel estimation of a density when the data are contaminated by error. Master’s thesis, Université catholique de Louvain, institut de Statistique, 1999.
- [Dellaert *et al.*, 1998] F. Dellaert, D. Pomerleau, and C. Thorpe. Model-based car tracking integated with a road follower. In *International Conference on robotics and automation*, pages 1889–1894, Leuven, Belgique, Mai 1998.
- [Desouza *et al.*, 2002] G.N. Desouza and A.C. Kak. VSION for mobile robot navigztion : a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2) :237–267, Février 2002.
- [Dickmanns *et al.*, 1992] E. Dickmanns and B. Myliwetz. Recusive 3d road and relative ego-state recognition. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 14(2) :199–213, 1992.
- [Dissanayake *et al.*, 2001] G. Dissanayake, P. Newman, Steven Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localisation and map building (slam) problem. *IEEE Transactions on Robotics and Automation*, 17(3) :229–241, June 2001.
- [Doucet *et al.*, 2000] A. Doucet, S. Godsill, and C. Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10 :197–208, 2000.

- [Erkelens *et al.*, 1997] J.S Erkelens and P.M.T. Broersen. Bias propagation in the autocorrelation method of linear prediction. *IEEE Transactions on Speech and Audio Processing*, 5(2) :116–119, Mars 1997.
- [Fuller *et al.*, 1998] R. Fuller, D. Dai, T. Walter, C. Comp, P. Enge, and J.D. Powell. Interoperation and integration of satellite based augmentation systems. In *ION GPS'98*, Nashville, Tennessee, Septembre 1998.
- [Galileo, 2002] Galileo. Mission high level definition version 3.0. Technical report, European Commission and European Space Agency, 2002. [http://europa.eu.int/comm/dgs/energy\\_transport/galileo/documents/technical\\_en.htm](http://europa.eu.int/comm/dgs/energy_transport/galileo/documents/technical_en.htm).
- [Gelb, 1974] A. Gelb. *Applied Optimal estimation*. MIT press, Cambridge Mass, 1974.
- [Glonass, 2002] Glonass. Interface and control document version 5.0. Technical report, Coordinational Scientific Information Center, Russian Federation Ministry of Defence, 2002. [http://www.glonass-center.ru/public\\_e.html](http://www.glonass-center.ru/public_e.html).
- [Gordon *et al.*, 1993] N. Gordon, D. Salmond, and A Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Radar and Signal processing*, 140(2) :107–113, 1993.
- [Gruyer *et al.*, 2004] D. Gruyer, A. Lambert, and B. Mourllion. Etat de l'art des stratégies de fusion de données pour la localisation. Livrable arcos thème 3 t3.2, LIVIC(unité mixte INRET/LCPC), 2004.
- [Guichard *et al.*, 1999] F. Guichard and J. Tarel. Curve extraction combining perceptual grouping and a kalman like fitting. In *IEEE International Conference on Computer Vision*, Kerkyra, Grèce, Septembre 1999.
- [Guivant *et al.*, 2000] J. Guivant, E. Nebot, and S. Baiker. Autonomous navigation and map buibing using laser range sensors in outdoor applications. *Journal of Robotic Systems*, 17(10) :565–583, Octobre 2000.
- [Halbwachs, 1997] E. Halbwachs. *Estimation à erreur bornée pour la génération d'hypothèses multiples de localisation d'un véhicule*. PhD thesis, Université de Technologie de Compiègne, 1997.
- [Handschin, 1970] J.E. Handschin. Monte carlo technics for prediction and filtering of nonlinear stochastique processes. *Automatica*, 6 :555–563, 1970.
- [Hastings, 1970] W.K Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57 :97–109, 1970.
- [Higuchi, 1997] T. Higuchi. Monte carlo methods using genetics algorithm operators. *Journal of Statistical Computation and Simulation*, 59 :1–23, 1997.
- [Hoon *et al.*, 1996] M. De Hoon, T. Van Der Hagen, H. Schoonewlle, and H. Van Dam. Why yule-walker should not be used for autoregressive modeling. In *Annals of Nuclear Energy*. Elsevier Science Ltd, Octobre 1996.

- [Hou, 2004] Haiying Hou. *Modeling inertial sensors errors using allan variance*. PhD thesis, Université de Calgary, Calgary, Alberta, Septembre 2004.
- [Hughes *et al.*, 2005] W.J. Hughes and WAAS Team. Wide aera augmentation system performance analysis. Technical report, Mai 2005.
- [Isard *et al.*, 1998] M. Isard and A. Blake. Condensation conditional density propagation for visual tracking. *International Journal Computer Vidion*, 29(1) :5–28, 1998.
- [Julier *et al.*, 1996] S.J. Julier and J.K. Uhlmann. A general method for approximating nonlinear transformation of probability distribution. Technical report, Université d'Oxford, Novembre 1996.
- [Julier *et al.*, 2001] S.J. Julier and J.K. Uhlman. A new extension of the kalman filter and nonlinear systems. In *SPIE International Symposion on Aerospace/Defence Sensing, Simulation and Controls.*, Orlando, Floride, Septembre 2001.
- [Julier *et al.*, 2004] S.J. Julier and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *IEEE Review*, 92(3), Mars 2004.
- [Julier, 1999] S.J. Julier. The scaled unscented transformation. Technical report, Université d'Oxford, 1999.
- [Kalman *et al.*, 1961] R. E. Kalman and R. Bucy. A new approach to linear filtering and prediction theory. *Trans. ASME, Journal of Basic Engineering*, 83 :95–108, 1961.
- [Kalman., 1960] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME, Journal of Basic Engineering*, 82 :34–45, 1960.
- [Kaske *et al.*, 1997] A. Kaske, D. Wolf, and R. Hudson. Lane boundary detection using statistical criteria. In *International Conference on Quality Control by Artificial Vision*, 1997.
- [Kaske, 1997] A. Kaske. *Contribution à la détection de bords de route imprécis. Implmentation sur le robot ROMANE*. PhD thesis, Institut National de Polytechnique de Lorraine, 1997.
- [Kayton *et al.*, 1997] M. Kayton and W. R. Fried. *Avionics Navigation Systems*. John Wiley and Sons, 2nd edition edition, 1997.
- [Kenue *et al.*, 1992] S. Kenue and S. Bajpayee. Lanelok : robust line and curve fitting of lane boundaries. In *SPIE Conference on Mobile Robots*, pages 491–503, Boston, MA, USA, Novembre 1992.
- [Kenue, 1989] S. Kenue. Detection of lane boundaries and vehicle tracking using image processing techniques. In *SPIE Conference on Mobile Robots*, pages 221–246, Philadelphie, PA, USA, Novembre 1989.
- [Kieffer, 1999] M. Kieffer. *Estimation ensembliste par analyse par intervalles : Application à la localisation de véhicules*. PhD thesis, Université de Paris sud, 1999.

- [Kitawaga, 1996] G. Kitawaga. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1) :1–25, 1996.
- [Kleeman *et al.*, 1995] L. Kleeman and R. Kuc. Mobile robot sonar for target localization and classification. *International Journal of Robotics Research*, 14(4) :295–318, 1995.
- [Kluge *et al.*, 1989] K. Kluge and C. Thorpe. Explicit models for robot road following. In *IEEE International Conference on Robotics and Automation*, pages 1148–1154, Scottsdale, AR, USA, Mai 1989.
- [Kluge *et al.*, 1995a] K. Kluge and C. Thorpe. The yarf system for vision-based road following. *Mathematical and Computer Modelling*, 22(4-7) :213–233, 1995.
- [Kluge *et al.*, 1995b] S. Kluge and K. Lakshmanan. A deformable template approach to lane detection. In *IEEE International Symposium on Intelligent Vehicles*, pages 54–59, Detroit, MI, USA, Septembre 1995.
- [Kong *et al.*, 1994] A. Kong, J.S. Liu, and W.H. Wong. Sequential imputation method and bayesian missing data problems. *Journal American Statistic Association*, 89 :278–288, 1994.
- [Kreusher *et al.*, 1999] C. Kreusher and K. Lakshmanan. Lana : a lane extraction algorithm that uses frequency domain feature. *IEEE Transactions on Robotics and Automation*, 15(2) :343–350, Avril 1999.
- [Kuan *et al.*, 1987] D. Kuan and U.K. Sharma. Model based geometric reasoning for autonomous road following. In *IEEE International Conference on Robotics and Automation*, volume 4, pages 416–423, Mars 1987.
- [Kuan *et al.*, 1988] D. Kuan, G. Phipps, and A.C. Hsueh. Autonomous robotic vehicle road following. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5) :648–658, September 1988.
- [Lakshmanan *et al.*, 1995] S. Lakshmanan and K. Kluge. Lane detection for automotives sensors. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2955–2958, Détroit, USA, Mai 1995.
- [Larder, 2001] S. Larder. E-otd-based location service implementation. In *MLW'01 Mobile Location Workshop*, Espoo, Finland, June 2001. [www.cursor-system.co.uk](http://www.cursor-system.co.uk).
- [Liu *et al.*, 1995] J.S. Liu and R. Chen. Blind deconvolution via sequential imputation. *American Statistic Association*, 90(430) :567–576, 1995.
- [Liu *et al.*, 1998] J.S. Liu and R. Chen. Sequential monte carlo methods for dynamic systems. *American Statistic Association*, 93 :1032–1044, 1998.
- [Liu, 1996] J.S. Liu. Metropolized independent sampling with comparison to rejection sampling and importance resampling. *Statistic and computing*, 6 :113–119, 1996.
- [Macek *et al.*, 2004] K. Macek, B. Williams, S. Kolski, and R. Siegart. A lane detection vision module for driver assistance. In *In Proceeding of the IEEE/APS Conference on Mechatronics and Robotics*, 2004.

- [Makhoul, 1975] J. Makhoul. Linear prediction : a tutorial review. *Proceeding of the IEEE*, 63(4) :561–580, Avril 1975.
- [Marshall, 1956] A.W. Marshall. The use of multi-stage sampling schemes in monte-carlo computations. In M.A. Meyer, editor, *Symposium on Monte Carlo methods*, pages 123–140, 1956.
- [Maurer *et al.*, 1998] M. Maurer, R. Beringer, F. Thomanek, and E. Dickmanns. A compact vision system for road vehicle guidance. In *International Conference on Pattern Recognition*, pages 313–317, Vienna, 1998.
- [Maybeck, 1979] P.S. Maybeck. *Stochastic models, estimation and control*. Academic Press, New York, USA, 1979.
- [Merwe *et al.*, 2001] R. Merwe and E. Wan. The square-root unscented kalman filter for state and parameter estimation. In *IEEE International Conference Acoustics Speech and Signal Processing*, volume 6, pages 701–704, 2001.
- [Metropolis *et al.*, 1953] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21 :1087–1091, 1953.
- [Moeglein *et al.*, 98] M. Moeglein and N. Krasner. An introduction to snaptrack server aided gps technology. In *ION GPS*, pages 333–342, Septembre 98. <http://www.snaptrack.com>.
- [Mohamed, 1999] A.H. Mohamed. *Optimizing the estimation procedure in INS/GPS integration for kinematic applications*. PhD thesis, Departement of Geomatics Engineering, University of Calgary, Calgary, Alberta, Canada, 1999.
- [Monroe *et al.*, 1998] J. Monroe and T. Bushy. *Marine Radionavigation and Communications*. Cornell Maritime Pr/Tidewater Pub, 1998.
- [Moral *et al.*, 1992] P. Del Moral, G. Rigal, and G. Salut. Estimation et commande optimale non linéaire : un cadre unifié pour la résolution particulière. Technical report, LAAS, Toulouse, mars 1992.
- [Moral, 1995] P. Del Moral. Nonlinear filtering using random particles. *Theory Probability and Application*, 40(4), 1995.
- [Moral, 1996] P. Del Moral. Nonlinear filtering : Interacting particle solution. *Markov Processes and Related Fields*, 2(4) :555–580, 1996.
- [Musso *et al.*, 2001] C. Musso, N. Oudjane, and F. Le Gland. *Improved regularised particle filters*, chapter 12, pages 247–271. Springer, 2001.
- [Najjar *et al.*, 2002] M.E El Najjar and P. Bonnifait. A road reduction method using multi-criteria fusion. In *IEEE Intelligent Vehicle Symposium*, Versailles, France, June 2002.
- [Oudjane, 2000] N. Oudjane. *Stabilité et approximations particulières en filtrage non linéaires, application au pistage*. PhD thesis, Université de Rennes I, 2000.

- [Oussalah, 1998] M. Oussalah. *Fusion de données par la théorie des possibilités, Application à la localisation d'un robot mobile*. PhD thesis, Université d'Evry Val d'Essonne, 1998.
- [Park, 2004] Minha Park. *Error analysis and stochastic modelling of MEMS based inertial sensors for land vehicle navigation applications*. PhD thesis, Université de Calgary, Calgary, Alberta, Avril 2004.
- [Pham, 2001] D.T. Pham. Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Monthly Weather Review*, 129(5) :1194–1207, 2001.
- [Retscher *et al.*, 2001] G. Retscher and E. Mok. Integration of mobile phone location services into intelligent gps vehicle navigation systems. In *MMT'01 Mobile Mapping Technology*, Caire, Egypte, 3-5 Janvier 2001.
- [Rezeck *et al.*, 1997] I.A. Rezeck and S.J Roberts. Parametric model order estimation : A brief review. In *IEE Colloquium on the Use of Model Based Digital Signal Processing Techniques in the Analysis of Biomedical Signals*, volume 3, pages 1–6, Londre, Grande Bretagne, 1997.
- [Rives *et al.*, 2001] P. Rives and M.Devy. *La robotique mobile*, chapter Perception pour la localisation. Hermès, 2001.
- [Rossi, 2004] Vivien Rossi. *Filtrage non linéaire par noyaux de convolution Application à un procédé de dépollution biologique*. PhD thesis, Ecole Nationale Supérieur Agronomique de Montpellier, 2004.
- [Royère, 2002] C. Royère. *Contribution à la résolution du conflit dans la théorie de l'évidence. Applications à la perception et la localisation des véhicules intelligents*. PhD thesis, Université de Technologie de compiégne, 2002.
- [Rubin, 1988] D.B. Rubin. *Using the SIR algorithm to simulate posterior distribution*, chapter Bayesian statistics, pages 395–402. Oxford University press, Oxford, 1988.
- [Seigneur *et al.*, 2005] E. Seigneur, M. Kieffer, and A. Lambert. Experimental vehicle localization by bounded-error state estimation using interval analysis. In *International Conference on Robotic and Automation*, Barcelone, Avril 2005.
- [Shaikh *et al.*, ] K. Shaikh, R. Sharif, F. Nagi, H. Jamaluddin, and S. Mansor. Inertial navigation sensors, data processing and integration with gps for mobile mapping.
- [Sorenson *et al.*, 1971] H.W. Sorenson and D.L. Alspach. Recursive bayesian estimation using gaussian sums. *Automatica*, 7 :465–479, 1971.
- [Stentz *et al.*, 2002] A. Stentz, C. Dima, C. Wellington, H. Herman, and D. Stager. A system for semi-autonomous tractor operations. *Autonomous Robots*, 13(1) :87–103, July 2002.
- [Tardos *et al.*, 2002] J.D Tardos, J. Neira, P. Newman, and J.J. Leonard. Robust mapping and localization in indoor environments using sonar data. *International Journal of Robotics Research*, 2002.

- [Thrapp *et al.*, 2001] R. Thrapp and C. Westbrook and D. Subramanian. Robust localization algorithms for an autonomous campus tour guide. In *IEEE International Conference on Robotics and Automation*, pages 2065–2071, Mai 2001.
- [Thurston, 2002] J. Thurston. Galileo glonass and navstar : A report on gps for gis people. <http://www.gisvisionmag.com>, 2002.
- [Tito *et al.*, 2003a] E Tito, H. Lopes, M. Pacheco, and M Vallesco. Convergence of genetic algorithms in continuous search space. Technical report, Department of Electrical Engineering, University of Rio de Janeiro, 2003.
- [Tito *et al.*, 2003b] E Tito, H. Lopes, M. Pacheco, and M Vallesco. Genetic particle filter : an evolutionary perspective of smc methods. Technical report, Department of Electrical Engineering, University of Rio de Janeiro, 2003.
- [Turk *et al.*, 1987] M.A. Turk, .G Morgenthaler and K.D Gremban, and M. Marra. Video road-following for autonomous land vehicle navigation. In *IEEE International Conference on Robotics and Automation*, volume 4, pages 273–280, Mars 1987.
- [Turk *et al.*, 1988] M.A. Turk, .G Morgenthaler and K.D Gremban, and M. Marra. Vits a vision system for autonomous land vehicle. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3) :342–361, Mai 1988.
- [Unser, 1984] M. Unser. *Description statistique de textures : application à l'inspection automatique*. PhD thesis, Ecole Polytechnique de Lausanne, 1984.
- [Vaganay, 1993] J. Vaganay. *Conception d'un système multisensoriel de localisation dynamique 3D pour robot mobile*. PhD thesis, Université Montpellier II, Sciences et Techniques du Languedoc, 1993.
- [Wang *et al.*, 1998] Y. Wang, D. Shen, and E. Teoh. Lane detection using catmull-rom spline. In *IEEE International Conference on Intelligent Vehicles*, volume 1, pages 51–57, Stuttgart, Germany, 1998.
- [Wang *et al.*, 2004] B. Wang, Q. Wei, Q. Tan, S. Yang, and B. Cai. Integration of gis, gps and gsm for the qinghai-tibet railway information management planning. In *International Society for Photogrammetry and Remote Sensing*, Istanbul, Turquie, Juillet 2004.
- [Wölfle *et al.*, 2002] G. Wölfle, R. Hoppe, D. Zimmermann, and F.M. Landnstorfer. Enhanced localisation technique within urban and indoor environments based on accurate and fast propagation models. In *EW'02 European Wireless*, Florence, Italie, 25-28 Février 2002.



# Table des figures

0.1	Architecture fonctionnelle d'un véhicule automatisé . . . . .	14
1.1	Modèle cinématique de type char . . . . .	20
1.2	Modèle cinématique de type tricycle . . . . .	21
1.3	Système inertiel à plate-forme stabilisée . . . . .	25
1.4	Erreurs classiques de mesures inertielles . . . . .	26
1.5	Positionnement absolu par satellites . . . . .	33
1.6	Positionnement relatif par satellites . . . . .	34
1.7	Illustration de la méthode AoA . . . . .	38
1.8	Illustration de la méthode TOA . . . . .	39
1.9	Illustration de la méthode TDOA . . . . .	39
1.10	Organigramme d'un système de localisation basée sur la vision . . . . .	42
1.11	Inverse perspective mapping . . . . .	45
1.12	Classification des pixels dans l'espace de couleurs RB . . . . .	47
1.13	Images binaires prédéfinies . . . . .	48
2.1	Situations mettant en défaut le filtre particulaire avec interaction . . . . .	74
2.2	Approximation continue de la distribution de probabilité . . . . .	77
2.3	Dégénérescence du filtre particulaire régularisé : faible bruit d'observation . . . . .	79
2.4	Evolution génétique/filtre particulaire . . . . .	82
3.1	Caractéristiques d'un bruit blanc . . . . .	93
3.2	Détermination du filtre formeur . . . . .	93
3.3	Filtre formeur pour un processus de Gauss-Markov du 1 <sup>er</sup> ordre . . . . .	95
3.4	Fonction d'autocorrélation d'un processus de Gauss-Markov du 1 <sup>er</sup> ordre . . . . .	96
3.5	Fonctions d'autocorrélation de processus Gauss-Markov d'ordre 1 à 5 . . . . .	98
3.6	Filtre récursif en forme de treillis . . . . .	106
4.1	Résultats comparatifs odomètre+gyromètre/odomètre+angle au volant . . . . .	116
4.2	Distance parcourue par le véhicule . . . . .	117
4.3	Facteur d'échelle du gyromètre . . . . .	119
4.4	Comportement du gyromètre en fonction de la température . . . . .	119
4.5	Caractérisation du bruit de mesure du gyromètre . . . . .	120
4.6	Caractéristiques des incertitudes gps . . . . .	123
4.7	RMSE de prédiction en statique de $b_x$ et $b_y$ pour les méthodes AR . . . . .	126
4.8	RMSE de prédiction en dynamique de $b_x$ et $b_y$ pour les méthodes AR . . . . .	127

4.9	Caractérisation de la latence du GPS . . . . .	130
4.10	Organigramme de l'algorithme de vision . . . . .	132
4.11	Primitives images de l'algorithme de vision . . . . .	133
4.12	Organigramme de la recherche des primitives images . . . . .	134
4.13	Evolution de l'algorithme de vision sur une image . . . . .	136
4.14	Recalage des données GPS . . . . .	141
4.15	Recalage des données GPS et "vision" . . . . .	142
5.1	Cohérence des informations GPS . . . . .	152
5.2	Passage référentiel route au référentiel monde . . . . .	153
5.3	Modélisation locale de la route . . . . .	155
5.4	Observations "vision" . . . . .	157
5.5	Mise à jour en fonction d'informations "vision" multimodales . . . . .	158
5.6	Carte de l'avenue Bingen à Clermont-Ferrand . . . . .	159
5.7	Erreurs latérales et longitudinales . . . . .	161
5.8	Evolution du véhicule dans le rond point . . . . .	162
5.9	Erreurs latérales et longitudinales : "vision" indisponible . . . . .	163
5.10	Evolution du véhicule dans le rond point : vision indisponible . . . . .	163
5.11	Recoupement des informations "vision" avec l'état a priori . . . . .	164
5.12	Erreurs latérales et longitudinales : pertes d'informations GPS . . . . .	165
5.13	Erreurs latérales et longitudinales : pertes d'informations "vision" . . . . .	166
5.14	Erreurs latérales et longitudinales : sans biais GPS . . . . .	167
5.15	Evolution du véhicule dans le rond point : sans biais GPS . . . . .	168
5.16	Multimodalité des informations "vision" et changement de voie . . . . .	169
6.1	Densité de vraisemblance des informations "vision" (position latérale) . . . . .	177
6.2	Densités de vraisemblance de l'information carte . . . . .	179
6.3	Dégénérescence du filtre particulaire avec interaction . . . . .	181
6.4	Atténuation du phénomène de dégénérescence avec le GPF . . . . .	184
6.5	Jeu de particules après correction par une information GPS . . . . .	185
6.6	Nouvel opérateur de brassage génétique "métropolisé" . . . . .	186
6.7	Erreurs latérales et longitudinales . . . . .	189
6.8	Evolution du véhicule dans le rond point . . . . .	190
6.9	Erreurs latérales et longitudinales : "vision" indisponible . . . . .	190
6.10	Evolution du véhicule dans le rond point : vision indisponible . . . . .	191
6.11	Erreurs latérales et longitudinales : pertes d'informations GPS . . . . .	192
6.12	Erreurs latérales et longitudinales : pertes d'informations "vision" . . . . .	193
6.13	Erreurs latérales et longitudinales : sans biais GPS . . . . .	195
6.14	Multimodalité des informations "vision" et changement de voie . . . . .	196
A.1	Repères utilisés vue de coté . . . . .	207
A.2	Repères utilisés vue de dessus . . . . .	208
B.1	Géoïdes et ellipsoïdes de la Terre . . . . .	212
B.2	Système de coordonnées géographiques et géocentriques . . . . .	213

---

B.3	Système de coordonnées géocentriques et locales . . . . .	214
B.4	La projection UTM . . . . .	216
B.5	Les différents types de projections cartographiques . . . . .	217
B.6	Les projections Lambert . . . . .	218
C.1	Le véhicule expérimental VELAC . . . . .	221
C.2	L'alternateur . . . . .	222
C.3	L'onduleur et les batteries . . . . .	222
C.4	Le gyromètre . . . . .	223
C.5	Les récepteurs GPS . . . . .	224
C.6	La caméra numérique . . . . .	225
C.7	Le télémètre laser RIEGL LMSZ210-60 . . . . .	225
C.8	Images délivrés par le télémètre et la caméra . . . . .	226
C.9	Radar et Lidar placés dans le pare-chocs . . . . .	227
C.10	Ordinateur embarqué . . . . .	227
C.11	Synoptique de l'architecture informatique . . . . .	228

# Liste des tableaux

1.1	Erreurs des systèmes GPS et DGPS . . . . .	36
1.2	Précisions obtenues par le système GNSS . . . . .	37
1.3	Précisions obtenues avec les générations GSM/UMTS . . . . .	40
1.4	Avantages et inconvénients des méthodes de positionnement relatif et absolu . . . . .	53
3.1	Fonctions d'autocorrélation et temps de corrélation des processus de Gauss-Markov d'ordre 1 à n . . . . .	97
4.1	Erreur de prédiction en statique selon les modélisations stochastiques . . . . .	125
4.2	Erreur de prédiction en dynamique selon les modélisations stochastiques . . . . .	126
4.3	Latences estimées pour différentes vitesses d'évolution du véhicule . . . . .	130
5.1	Moyennes et écart-type des erreurs de localisation . . . . .	161
6.1	Moyenne et écart-type des erreurs de localisation . . . . .	189
6.2	Précision du système de localisation en fonction du nombre de particules et du filtre particulaire utilisé . . . . .	197
6.3	Temps d'exécution des différentes étapes des filtres SIR et du GPF . . . . .	199
B.1	Propriétés de l'ellipsoïde IAG-GRS80 . . . . .	215
B.2	Propriétés de l'ellipsoïde Clarke 1880 IGN . . . . .	216
B.3	Les constantes de projections Lambert . . . . .	219
C.1	Caractéristiques de la caméra numérique . . . . .	225
C.2	Caractéristiques de l'ordinateur embarqué dans VELAC . . . . .	227

# Listes des Algorithmes

2.1	Filtre de Monté Carlo à procédure SIS . . . . .	68
2.2	Filtre de Monté Carlo pondéré . . . . .	69
2.3	Filtre de Monté Carlo à mémoire limitée . . . . .	70
2.4	Filtre de Monté Carlo avec procédure SIS-R . . . . .	72
2.5	Filtre particulaire avec interaction SIR . . . . .	73
2.6	Filtre particulaire régularisé RPF . . . . .	78
2.7	L'évolution génétique . . . . .	81
2.8	Filtre particulaire génétique GPF . . . . .	84
2.9	L'opérateur de brassage génétique "métropolisé" . . . . .	85
2.10	L'opérateur de mutation génétique "métropolisé" . . . . .	85
4.1	Structure algorithmique sous-jacente au filtrage . . . . .	142
6.1	Filtre particulaire avec interaction lié à notre application . . . . .	180
6.2	Filtre particulaire génétique lié à notre application . . . . .	183