



HAL
open science

Agrégation de données décentralisées pour la localisation multi-véhicules

Nadir Karam

► **To cite this version:**

Nadir Karam. Agrégation de données décentralisées pour la localisation multi-véhicules. Automatique / Robotique. Université Blaise Pascal - Clermont-Ferrand II, 2009. Français. NNT : 62T1071932HJQ . tel-00724489

HAL Id: tel-00724489

<https://theses.hal.science/tel-00724489>

Submitted on 21 Aug 2012

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

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

UNIVERSITÉ BLAISE PASCAL - CLERMONT II

ÉCOLE DOCTORALE
SCIENCES POUR L'INGÉNIEUR DE CLERMONT-FERRAND

Mémoire de thèse

présenté par

Nadir Karam

pour obtenir le grade de

DOCTEUR D'UNIVERSITÉ
Spécialité : VISION POUR LA ROBOTIQUE

Agrégation de données décentralisées pour la localisation
multi-véhicules

Soutenu publiquement le 14/12/2009 devant le jury

M. Didier Aubert, Directeur de recherches au LIVIC	Président et Rapporteur
M. Philippe Bonnifait, Professeur à l'IUT de Compiègne	Rapporteur
M. François Charpillet, Directeur de recherches à l'INRIA	Examineur
M. Frédéric Chausse, Maître de conférences HDR à l'IUT de Clermont-Ferrand	Examineur
M. Romuald Aufrère, Maître de conférences à l'université Blaise Pascal	Examineur
M. Roland Chapuis, Professeur à l'université Blaise Pascal	Directeur de thèse
M. Javier Ibanez-Guzman, Technocentre RENAULT	Invité

Remerciements

Les travaux présentés dans cette thèse ont été réalisés au sein du Groupe Automatique, Vision et Robotique (GRAVIR) du Laboratoire des Sciences et Matériaux pour l'Electronique, et d'Automatique (LASMEA) Unité Mixte de Recherche (UMR 6602) de l'université Blaise Pascal de Clermont-Ferrand et du Centre National de Recherche Scientifique (CNRS).

Je tiens tout d'abord à remercier Monsieur Michel Dhome, Directeur de recherche au CNRS et directeur du LASMEA pour m'avoir accueilli au sein du laboratoire et pour l'aide précieuse qu'il m'a apporté pendant la rédaction de ce manuscrit.

Je remercie Monsieur Didier Aubert, Directeur de recherche au LIVIC - Versailles pour l'intérêt qu'il a porté à mon travail en acceptant d'être rapporteur et pour avoir présidé mon jury de soutenance.

Mes remerciements s'adressent également à Monsieur Philippe Bonnifait, Professeur à l'IUT de Compiègne pour l'honneur qu'il m'a fait en acceptant de juger mon travail.

Je tiens aussi à remercier Monsieur François Charpillet, Directeur de recherche à l'INRIA de Nancy-Grand Est pour avoir accepté de participer à l'évaluation de mon travail.

Je souhaite également remercier Monsieur Javier Ibanez-Guzman pour l'intérêt qu'il a porté à mon travail en acceptant d'assister à la soutenance de ma thèse.

Je voudrais témoigner toute ma gratitude à Monsieur Roland Chapuis, Professeur de l'Université Blaise Pascal et mon directeur de thèse pour la confiance qu'il m'a accordé durant toutes ses années, pour sa disponibilité, ces précieux conseils, sa patience et pour son humanité et bien d'autres qualités tant au travail qu'en dehors. Ce fut un réel plaisir de travailler avec lui.

Mes remerciements vont aussi à Frédéric Chausse, Maître de conférences HDR à l'IUT de Clermont Ferrand et co-dérecteur de ma thèse pour l'aide qu'il m'a apporté et sa disponibilité,

sa gentillesse et la grande sympathie qu'il m'a témoigné durant ces années.

Je tiens aussi à exprimer mes sincères remerciements à Romuald Aufrere, Maitre de conférences à l'Université Blaise Pascal co-directeur de ma thèse pour sa gentillesse et pour l'aide précieuse qu'il m'a apporté durant toute ma thèse et spécialement durant la rédaction de ce manuscrit.

Mes remerciement vont aussi à Monsieur Jean-Pierre Dérutin, Professeur à l'Université Blaise Pascal et responsable de l'équipe Système de perception (PERSYST) du groupe GRA-VIR pour la confiance qu'il m'a accordé durant notre collaboration pour le projet CRISTAL.

Bien sure je remercie l'ensemble des collègues et amis que j'ai pu côtoyer au LASMEA notamment les membres de l'équipe techniques Francois, Serge et Laurent pour l'aide apportée durant la mise en œuvre de mes expérimentation, mes collègues de bureau et amis, Noël, David, Sébastien, Hicham, Clément, Datta, Thomas, Damien pour la bonne humeur qui régnait toujours dans ce dernier.

Enfin mes dernières pensées vont à mes parents, mes sœurs et l'ensemble de ma famille et amis qui m'ont toujours soutenus et encouragé et auxquels je dédie ce travail.

Résumé

Les avancées technologiques notables dans le domaine de la communication sans fil ont ouvert de nouvelles perspectives pour améliorer les systèmes d'aide à la conduite existants. En effet il a été prouvé que l'échange de données de positionnement absolu et relatif entre des véhicules d'un même groupe peut améliorer la précision de la localisation de tous ces membres. Ce ci est appelé localisation coopérative. Le travail présenté dans ce manuscrit s'inscrit dans ce cadre et propose une approche totalement distribuée permettant de combiner les informations provenant des capteurs de tous les véhicules d'un groupe pour obtenir l'estimation la plus précise possible de la pose de ces derniers tout en minimisant les informations transmises sur le réseau de communication. Le principe de l'approche proposée est que chaque véhicule utilise ses propres capteurs pour mettre à jour une estimation de l'état du groupe puis la transmet à ses congénères. La fusion des états reçus et l'état construit dans chaque véhicule donne l'estimation de l'état global du groupe qui intègre les informations provenant de tous ses membres. L'approche proposée a été validée expérimentalement en simulation et dans un environnement réel sur des véhicules urbains.

Mots clés

Localisation multivehicules, fusion de données, filtrage de KALMAN, communication véhicule à véhicule, système embarqué, temps réel.

Abstract

The technological advances in wireless communication opened up new perspectives to enhance the existing driver assistance systems. It has been shown that the exchange of absolute and relative positioning information between the vehicles of the same group improves the localization precision of the whole group. This is called cooperative localization. The presented work is part of this framework and introduces a distributed approach which combines the data provided by the sensors of all the group vehicles to obtain the most accurate estimation of their pose while minimizing the quantity of exchanged data over the communication network. The principle of the proposed approach is that each vehicle uses its own sensors to maintain an estimation of the group state and sends it to the other vehicles. The fusion of the received states with the built one gives the estimation of the global state of the group which summarizes the sensor data coming from all its members. The proposed approach has been tested and validated by simulations and real experiments with a group of road vehicles.

Key words

Multivehicle localization, data fusion, KALMAN filtering, vehicle to vehicle communication, embedded system, real time.

Table des matières

Introduction	1
Cadre général	1
Objectifs de la thèse	2
Organisation du mémoire	3
I Outils et approches de localisation	5
1 Outils pour la localisation d'un robot mobile	7
1.1 Les méthodes de localisation	8
1.1.1 Le positionnement relatif	8
1.1.1.1 Positionnement par odométrie	9
1.1.1.2 Positionnement inertiel	13
1.1.1.3 Positionnement relatif "hybride"	13
1.1.2 Le positionnement absolu	14
1.1.2.1 Positionnement par satellites	15
1.1.2.2 Positionnement local	20
1.1.2.3 Bilan	20

1.1.3	Les systèmes de positionnement hybride	21
1.2	Filtrage	22
1.2.1	Le filtrage optimal	22
1.2.2	Le filtre de Kalman	24
1.2.3	Le filtre de Kalman étendu EKF	26
1.3	Conclusion	29
2	Localisation coopérative d'un groupe de robots	31
2.1	Principe de la localisation coopérative	32
2.2	Localisation coopérative centralisée	35
2.3	Localisation coopérative distribuée	37
2.3.1	Les approches basées sur l'échange des informations capteurs	38
2.3.2	Les approches basées sur l'échange d'état	39
2.3.2.1	Consanguinité des données	39
2.3.2.2	Les solutions au problème de consanguinité des données dans la littérature	45
2.4	Approche privilégiée dans ce mémoire	48
2.5	Conclusion	49
II	Localisation collective par fusion d'états	51
3	Localisation collective : cas idéal	55
3.1	Estimation de l'état du groupe	56
3.1.1	Evolution de l'état global du groupe	58

Table des matières

3.1.2	Mise à jour de l'état du groupe	59
3.2	Localisation collective	61
3.2.1	Principe général	61
3.2.2	Notion de sous-états	62
3.2.3	Fusion décentralisée	63
3.3	Conclusion	67
4	Localisation collective : cas réaliste	69
4.1	Ordonnancement des observations	70
4.1.1	Désynchronisation	70
4.1.2	Latences	71
4.2	Localisation collective	72
4.2.1	Estimation de l'état du groupe	72
4.2.1.1	Evolution de l'état	73
4.2.1.2	Mise à jour de l'état du groupe	74
4.2.1.3	Principe de la mise à jour	75
4.2.1.4	“Nettoyage” de l'état du groupe	76
4.2.2	Fusion des états de l'environnement	78
4.3	Conclusion	81
III	Mise en œuvre et résultats	83
5	Simulation de l'approche de localisation collective	87
5.1	Description des simulations	87

5.2	Génération des données de simulation	88
5.3	Scénario de l’expérimentation	90
5.3.1	Scénario 1 : Cas simple avec deux robots	90
5.3.2	Scénario 2 : Cas d’un groupe de trois robots	97
5.4	Conclusion	105
6	Expérimentation réelle	107
6.1	Plate-forme expérimentale	108
6.1.1	Le cycab	108
6.1.2	Capteurs utilisés	108
6.2	Scénario de l’expérimentation	110
6.3	Construction de l’état du groupe	111
6.3.1	Localisation	111
6.3.2	Détection du véhicule de tête	112
6.4	Résultats et discussions	114
6.4.1	Résultats de la localisation coopérative	115
6.4.1.1	Trajectoire 1	115
6.4.1.2	Trajectoire 2	120
6.5	Problèmes rencontrés	124
6.6	Conclusion	124
	Conclusion et perspectives	129
	Principales contributions	129

Table des matières

Perspectives 130

Bibliographie **132**

Liste des figures

1.1	Modèle cinématique de type char	10
1.2	Modèle cinématique de type tricycle	11
1.3	Positionnement absolu par satellites	17
1.4	Positionnement relatif par satellites	18
2.1	Principe général : le robot 1 détecte le robot 2 et partage cette information . . .	34
2.2	Principe général : localisation des robots 1 et 2 après la fusion de données . . .	34
2.3	Principe général : le robot 3 détecte le robot 2 et partage cette information . . .	34
2.4	Principe général : localisation des robots 1, 2 et 3 après la fusion de données . .	34
2.5	Architecture centralisée pour la localisation coopérative	36
2.6	Architecture distribuée pour la localisation coopérative	37
2.7	Cause de la sur-convergence schéma 1	40
2.8	Problème de sur-convergence : Etats initiaux des robots R_1 et R_2	41
2.9	Problème de sur-convergence : Détection et localisation des robots	42
2.10	Problème de sur-convergence : Echange des données de localisation et fusion .	42
2.11	Problème de sur-convergence : Evolution de l'état des 2 robots	42
2.12	Problème de sur-convergence : Détection, localisation et échange de données. .	42

2.13 Cheminement des données pour l'exemple à deux robots	43
2.14 Cause de la sur-convergence schéma 2	44
2.15 Problème de sur-convergence : Etats initiaux des robots R_1 et R_2	44
2.16 Problème de sur-convergence : Echange des données de localisation et fusion	45
2.17 Dépendance des données et sur-convergence	46
2.18 Exemple d'un arbre de dépendances dans la cas d'un groupe de 4 robots.	47
2.19 Pose relative entre deux robots mobiles.	47
3.1 Etat du robot R_s	56
3.2 Pose relative entre les robots du groupe	60
3.3 Echange de données : Avant la communication	64
3.4 Echange de données : après la communication	65
3.5 Utilisation d'un robot intermédiaire pour communiquer les états	66
4.1 Ordonnancement des observations : Désynchronisation	71
4.2 Ordonnancement des observations : arrivée d'une observation	71
4.3 Ordonnancement des observations : insertion de l'observation	72
5.1 Simulation des latences des données.	89
5.2 Trajectoires prises par les deux robots. d1 : distance maximum de détection. d2 : distance maximum de communication.	90
5.3 Etat du groupe estimé dans les deux robots : les deux robots n'échangent aucune information. (a1) : état du groupe estimé dans le robot R_1 . (a2) : état du groupe estimé dans le robot R_2 . (b1) : état global du groupe fusionné dans le robot R_1 . (b2) : état global du groupe fusionné dans le robot R_2	92

Liste des figures

- 5.4 Etat du groupe estimé dans les deux robots : les deux robots peuvent échanger leurs données de localisation mais ne peuvent pas se détecter ni se localiser mutuellement. (a1) : état du groupe estimé dans le robot R_1 . (a2) : état du groupe estimé dans le robot R_2 . (b1) : état global du groupe fusionné dans le robot R_1 . (b2) : état global du groupe fusionné dans le robot R_2 93
- 5.5 Etat du groupe estimé dans les deux robots : les deux robots se détectent, se localisent l'un l'autre et échangent les données. (a1) : état du groupe estimé dans le robot R_1 . (a2) : état du groupe estimé dans le robot R_2 . (b1) : état global du groupe fusionné dans le robot R_1 . (b2) : état global du groupe fusionné dans le robot R_2 95
- 5.6 Ecart type des erreurs des localisations des robots R_1 et R_2 selon les axes de la longitude et de la latitude obtenues dans les états fusionnés des deux robots. . . 96
- 5.7 Trajectoires prises par les quatre robots. d1 : distance maximum de détection (60m). d2 : distance maximum de communication (100m). 97
- 5.8 Etat du groupe estimé dans les trois robots équipés : Le robot R_1 détecte le robot R_4 mais aucun échange de données n'a encore été effectué. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 98
- 5.9 Etat du groupe estimé dans les trois robots équipés : Les robots R_1 et R_3 communiquent. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 99
- 5.10 Etat du groupe estimé dans les trois robots équipés : Les robots R_1 et R_3 peuvent mesurer leurs poses relatives et échanger ces données. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 100

5.11	Etat du groupe estimé dans les trois robots équipés : Les trois robots peuvent mesurer leurs poses relatives et communiquer ces données. Le robot R_3 peut mesurer la pose relative du robot R_4 . (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3	102
5.12	Etat du groupe estimé dans les trois robots équipés : Les trois robots communiquent entre eux, le robot R_2 mesure les poses relatives des deux autres robots et les robots R_1 et R_3 ne peuvent mesurer que la pose relative du robot R_2 . (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3	103
5.13	Etat du groupe estimé dans les trois robots équipés : les robots R_2 et R_3 peuvent encore mesurer leurs poses relatives et échanger des informations. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3	104
5.14	Ecart type des erreurs des localisations des robots R_1 , R_2 et R_3 selon les axes de la longitude et de la latitude obtenues dans les états fusionnés des trois robots.	105
6.1	Le Cycab	109
6.2	Configuration de l'expérience réelle	110
6.3	modèle tricycle utilisé pour l'évolution de l'état des robots	111
6.4	Télémètre laser utilisé pour la détection du véhicule de tête.	112
6.5	Image fournie par le télémètre laser	113
6.6	Trajectoire 1	114
6.7	Trajectoire 2	115
6.8	Résultats de la localisation : Trajectoire 1.	116

Liste des figures

6.9	Distance euclidienne entre la valeur moyenne de la position estimée après la fusion des états et la valeur de la position de référence donnée par le GPS centimétrique	117
6.10	Distance de Mahalanobis entre la pose estimée après la fusion et la pose fournie par le GPS centimétrique	118
6.11	Zoom sur les trajectoires résultantes	119
6.12	Résultats de la localisation : Trajectoire 2.	120
6.13	Distance euclidienne entre la valeur moyenne de la position estimée après la fusion de états et la valeur la position de référence donnée par le GPS centimétrique. 121	
6.14	Distance de Mahalanobis entre la pose estimée après la fusion et la pose fournie par le GPS centimétrique. La zone I_1	122
6.15	Effet de la perte des informations sur poses relatives	123

Introduction

Cadre général

Les avancées technologiques notables dans le domaine de la communication sans fil ont ouvert de nouvelles perspectives pour améliorer les systèmes d'aide à la conduite existants. En effet la communication véhicule à véhicule (V2V pour *vehicle to vehicle*) ou véhicule à infrastructure (V2I pour *vehicle to infrastructure*) promet de grandes avancées de l'efficacité des systèmes d'aide à la conduite et de la sécurité des usagers.

Ces dernières années, de grands constructeurs automobiles tels que General motors aux Etats Unis, Daimler, BMW et Volkswagen en Europe ou encore Honda au Japon, se sont lancés dans le développement de cette technologie en collaboration avec des laboratoires de recherche.

Ainsi, en Europe, par exemple, plusieurs projets tels que Car2Car, NOW (pour *Network on Wheels*), CVIS (pour *Cooperative Vehicle-Infrastructure Systems*) ou encore WiSafeCar ont vu le jour. Le but principal de ces projets étant d'apporter des améliorations dans les domaines de la sécurité routière et de la gestion du trafic.

En échangeant en temps réel des données sur leur position, leur vitesse et leur direction respectives, ces systèmes peuvent détecter un risque potentiel représenté par un autre véhicule et alerter le conducteur à temps pour éviter le danger. Ceci peut être notamment appliqué pour assister le conducteur dans les intersections à visibilité réduite.

La communication dans ce cas est faite généralement de véhicule à véhicule et, bien que la zone couverte soit limitée, elle convient à l'application.

Dans les projets CVIS et WiSafeCar par exemple, les véhicules envoient les informations

sur leurs position, vitesse et direction à un système centralisé qui les traite pour tenir à jour avec précision l'état du trafic. Les informations collectées sont ensuite redistribuées sur l'ensemble de la flotte. Une bonne visibilité sur l'état du trafic permet de le gérer d'une manière plus efficace. Ceci peut passer par une meilleure planification des trajets des usagers afin de diminuer les embouteillages par exemple, ou encore par le calcul d'une vitesse optimale à conseiller aux usagers pour permettre une meilleure fluidité du trafic. Toutes ces actions contribueront à diminuer les consommations de carburant, et par là même, les émissions de CO_2 ainsi que le stress des conducteurs.

Dans le projet PReVENT, l'échange de données ne se limite pas uniquement aux informations propres aux véhicules mais aussi sur l'environnement. Les véhicules utilisent des capteurs tels que des caméras ou des télémètres laser pour détecter d'éventuels vulnérables tels que des piétons et également pour permettre de prévoir leur trajectoire. Cette approche fournit aux autres véhicules une vue plus détaillée de l'environnement et permet d'alerter les conducteurs sur d'éventuels dangers autres que les véhicules.

Dans tous les projets cités précédemment, les données échangées entre véhicules, que ce soit de véhicule à véhicule ou par l'intermédiaire d'une infrastructure, sont utilisées uniquement pour propager l'information avec les autres usagers. Tous les véhicules communiquant entre eux, ils peuvent à tout moment se localiser et connaître les positions des autres congénères ou les éventuels vulnérables.

La communication inter-véhicules peut apporter une autre plus-value aux systèmes multi-véhicules.

En effet, il a été démontré que la mesure des informations relatives entre les véhicules (position, orientation des uns par rapport aux autres) et l'échange de ces informations améliorent le système de localisation en diminuant les erreurs de positionnement.

Par exemple, dans le cas de deux véhicules, si le positionnement relatif entre ces derniers est connu, un gain en précision sur le positionnement des deux véhicules est obtenu. Ceci est appelé dans la littérature **localisation coopérative** ou **localisation collective**.

Objectifs de la thèse

Ce travail de thèse s'inscrit dans ce cadre de la localisation coopérative et se propose d'étudier les apports de la communication véhicule à véhicule pour la localisation. Une approche de

combinaison des données échangées entre les véhicules est proposée. Le but de cette approche est d'obtenir dans chaque véhicule, la meilleure vision possible de l'état de l'environnement incluant les positions des autres véhicules communicants et des éventuels obstacles ou vulnérables détectés.

Dans la suite, ce travail est positionné dans le contexte de la robotique mobile afin de faciliter sa comparaison avec les approches proposées dans la littérature. Il faut cependant garder en mémoire que l'approche doit être applicable dans un environnement ouvert tel qu'un cadre urbain ou routier. En effet, les recherches qui traitent de la localisation coopérative se positionnent souvent dans un cadre expérimental où beaucoup de problèmes liés aux environnements complexes ne sont pas traités. Toutes les approches rencontrées dans la littérature traitent, par exemple, d'un groupe de robots évoluant dans un environnement fermé avec un nombre de robots connu et fixe. Cette hypothèse n'est pas réaliste et ne peut s'appliquer dans un environnement urbain, par exemple.

L'hypothèse est toujours faite que les robots peuvent détecter et identifier leurs congénères. Or l'identification du véhicule peut uniquement être envisagée par la détection de la plaque minéralogique si le capteur utilisé est une caméra vidéo mais reste impossible avec d'autres capteurs tels qu'un télémètre laser.

L'approche proposée dans ce mémoire est une approche de localisation coopérative décentralisée d'un groupe de robots mobiles. Cette méthode permet de combiner de façon optimale toutes les informations dont dispose le robot (ses propres informations et les informations que les autres robots ont pu lui communiquer) afin d'obtenir l'estimation la plus complète et la plus précise possible de l'état du groupe.

Cette approche a été expérimentée sur des véhicules expérimentaux mais les conditions imposées font qu'elle est tout à fait applicable dans le cas de véhicules urbains dans un environnement ouvert.

Organisation du mémoire

Ce mémoire se divise en trois parties.

La première partie présente les différentes méthodes de localisation d'un robot mobile. Le premier chapitre (chapitre 1) de cette partie décrit quelques méthodes et capteurs utilisés pour effectuer la localisation mono-robot. Il précise aussi des outils de fusion de données utilisés pour combiner les différentes informations issues des capteurs du robot mobile et ainsi obtenir une estimation de sa pose dans un repère absolu. Ce chapitre ne constitue pas un état de l'art

détaillé des méthodes de localisation et de fusion de données, il a pour but principal de montrer la manière dont un robot mobile peut estimer sa position dans un repère donné.

Le deuxième chapitre de cette partie (chapitre 2) présente l'état de l'art des méthodes de localisation multi-robots. Il met en évidence les différents problèmes rencontrés et les solutions proposées dans la littérature.

La deuxième partie de ce mémoire présente notre contribution et détaille l'approche de localisation coopérative proposée. Cette partie est divisée en deux chapitres.

Dans le chapitre 3, le principe général de l'approche proposée est détaillé pour un cas idéal. Certains problèmes liés aux applications réelles se déroulant en milieu ouvert ont ici été ignorés comme par exemple les latences des données, l'identification des robots détectés, le nombre des robots du groupe, etc. Le chapitre 6 montre ensuite la prise en compte par l'approche proposée des contraintes ignorées dans le chapitre 3.

La troisième partie de ce mémoire présente les résultats obtenus après l'implémentation de l'approche proposée. Cette partie se divise en deux chapitres : le premier (chapitre 5), présente les résultats obtenus en simulation dans le cas de deux robots puis dans le cas d'un groupe de quatre robots. Les résultats réels sont présentés dans le second (chapitre 6). Afin de valider l'approche de localisation coopérative proposée, celle-ci a été implémentée sur des véhicules expérimentaux et testée dans une configuration de convoi de deux véhicules. Dans ce chapitre, les résultats sur le gain en précision et sur les effets des pertes de données sont détaillés.

Enfin, nous terminons ce mémoire par quelques conclusions et perspectives.

Première partie

Outils et approches de localisation

Chapitre 1

Outils pour la localisation d'un robot mobile

Sommaire

1.1	Les méthodes de localisation	8
1.1.1	Le positionnement relatif	8
1.1.2	Le positionnement absolu	14
1.1.3	Les systèmes de positionnement hybride	21
1.2	Filtrage	22
1.2.1	Le filtrage optimal	22
1.2.2	Le filtre de Kalman	24
1.2.3	Le filtre de Kalman étendu EKF	26
1.3	Conclusion	29

1.1 Les méthodes de localisation

Quel que soit le domaine d'application pour lequel est destiné un robot mobile, une localisation précise de ce dernier reste une tâche primordiale. Cette tâche est généralement accomplie en combinant les informations issues des différents capteurs qui décrivent le mouvement ou la pose du robot. En robotique mobile, il existe trois différentes méthodes de positionnement :

- **La localisation relative** où la pose (la position et l'orientation) du robot est déterminée par intégrations successives d'informations décrivant la dynamique du robot, c'est-à-dire les composantes de vitesse et d'accélération à partir d'une position d'origine connue du robot. Cette méthode de localisation présente le problème de l'accumulation des erreurs liées aux informations des capteurs et au modèle utilisé pour décrire le mouvement du robot. Ceci mène à une dérive à long terme de la position du robot.
- **La localisation absolue** où la pose du robot est déterminée par le traitement des données fournies par un capteur qui donne des informations sur la position et la direction dans un référentiel propre à ce capteur (ex : référentiel géodésique pour le positionnement par GPS, le référentiel de la carte des amers visuels pour le positionnement par vision etc...).
- **Le positionnement hybride** où la pose du robot est déterminée en combinant les méthodes de positionnement relatif et de positionnement absolu. Ce type de positionnement permet d'obtenir de meilleurs résultats de localisation que les deux premières méthodes en compensant les différents problèmes propres à celles-ci (problème de dérive à long terme dû à l'accumulation des erreurs liées aux capteurs et au modèle utilisé pour décrire le déplacement du robot dans le cas de la localisation relative et les problèmes d'occultations du capteur de positionnement dans le cas de la localisation absolue). La combinaison des informations est généralement faite avec un processus de filtrage bayésien.

Dans les sections suivantes nous allons faire une brève étude de ces différentes méthodes de localisation.

1.1.1 Le positionnement relatif

Le positionnement relatif peut être 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 Les méthodes de localisation

1.1.1.1 Positionnement par odométrie

La localisation par odométrie est très répandue en robotique mobile de par 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 robot en mesurant la rotation de ses roues. Ces mesures sont généralement réalisées par des codeurs incrémentaux couplés à l'axe des roues, la précision de la mesure dépend directement de la résolution du codeur. Le changement d'orientation peut être déduit de la différence de distance parcourue par les roues droites et gauches du robot.

Le calcul de la position relative et du cap du robot $(x, y, \varphi)^T$ est réalisé par intégrations successives de ses déplacements élémentaires. La localisation par odométrie nécessite la connaissance du diamètre des roues, de la largeur de l'essieu et du modèle cinématique du robot.

Il existe différentes modélisations des véhicules. Souvent, seul un modèle cinématique est utilisé. Deux modèles sont présentés ici.

Le modèle cinématique de type char

Ce modèle considère un robot composé de deux roues (voir figure 1.1).

La cadence d'acquisition des mesures odométriques peut être considérée comme assez élevée pour réaliser des petits déplacements élémentaires et considérons également que le robot se déplace sans glissement.

En premier lieu, calculons la vitesse longitudinale, la vitesse latérale du robot selon l'axe (x_r, y_r) et la vitesse de rotation du robot en fonction des vitesses angulaires des roues w_d pour la roue droite et w_g pour la roue gauche.

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

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 robot soit considérée nulle.

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

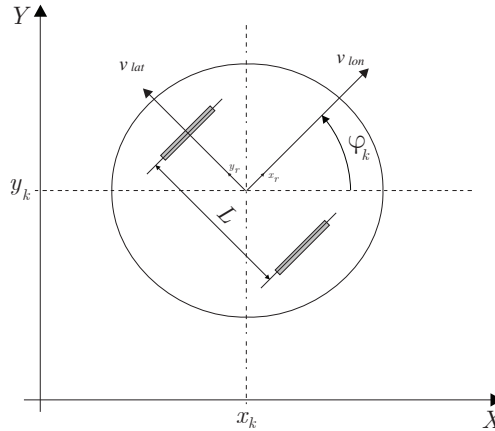


FIGURE 1.1 – Modèle cinématique de type char

relations :

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

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

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

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

Modèle cinématique de type tricycle

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

1.1 Les méthodes de localisation

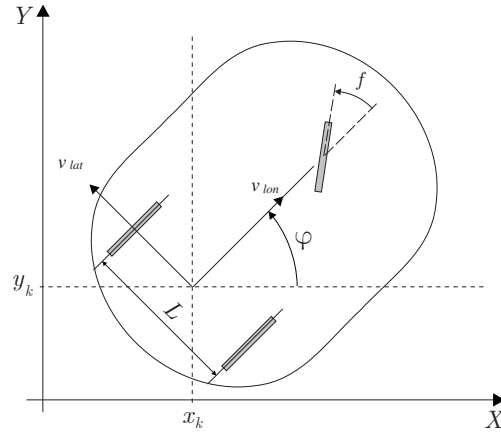


FIGURE 1.2 – Modèle cinématique de type tricyle

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \\ \dot{f} \end{pmatrix} = \begin{pmatrix} \cos(\varphi) \\ \sin(\varphi) \\ \tan \frac{f}{L} \\ 0 \end{pmatrix} v_{lon} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v_r \quad (1.4)$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \\ \dot{f} \end{pmatrix} = \begin{pmatrix} \cos(f) \cos(\varphi) \\ \sin(f) \sin(\varphi) \\ \sin \frac{f}{L} \\ 0 \end{pmatrix} v_{lon} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v_r \quad (1.5)$$

où \dot{f} est la vitesse de braquage de la roue directrice et v_{lon} la vitesse longitudinale des roues arrières ou de la roue avant du robot 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{f} devient inutile) et en intégrant dans le temps \dot{x} , \dot{y} , $\dot{\varphi}$, le déplacement du robot entre les instant k et $k - 1$ est :

$$\begin{cases} x_k = x_{k-1} + v_{lon} \Delta_T \cos(\varphi_k) \\ y_k = y_{k-1} + v_{lon} \Delta_T \sin(\varphi_k) \\ \varphi_k = \varphi_{k-1} + \tan \frac{f}{L} \end{cases} \quad (1.6)$$

Ce modèle est réaliste sous la contrainte qu'à tout instant $|\varphi| \leq \varphi_{max} \leq \frac{\pi}{2}$, 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 instantané maximal R_{max} et la vitesse longitudinale.

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

Modélisation des erreurs odométriques

Le principal défaut de cette méthode est une dérive de la position proportionnelle à la distance parcourue par le robot. Cette dérive est due à l'accumulation des erreurs lors des intégrations successives. Borenstein et *al.* dans [BEF96] classent ces erreurs en deux catégories :

1. **les erreurs systématiques** : ces erreurs s'ajoutent à chaque itération de la localisation du robot et sont de deux natures différentes :
 - a) erreurs sur les paramètres mécaniques du robot (les roues de diamètres différents, non alignement des roues, etc ...) et b) les erreurs de mesures (résolution des codeurs, fréquence d'échantillonnage inexacte, etc ...). Les erreurs systématiques du premier 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 robot.
2. **les erreurs non-systématiques** : ces erreurs sont directement dues aux contacts robot / terrain, elles présentent un caractère aléatoire et dépendent en général des caractéristiques du sol : surface de roulement irrégulière (trous et bosses), présence d'objet sur la route ou encore glissement du robot (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 robot, l'une ou l'autre de ces sources d'erreurs devient prépondérante par rapport à l'autre. Par exemple dans le cadre d'applications en environnement intérieur les erreurs non-systématiques étant quasi-inexistantes, ce sont donc 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.

1.1 Les méthodes de localisation

1.1.1.2 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 Isaac Newton, la localisation du robot est déduite de la double intégration des composantes accélérométriques a du robot :

$$\text{vitesse} \quad v = \int a \, dt \quad (1.8)$$

$$\text{position} \quad p = \int v \, dt \quad (1.9)$$

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.

$$\text{orientation} \quad \theta = \int w \, dt \quad (1.10)$$

Les différents systèmes inertiels peuvent être classés en deux catégories : les systèmes "à plates-formes stabilisées" (capteurs solidaires d'une plate-forme isolée des mouvements mécaniques) et les systèmes "à composant liés" (les capteurs sont directement solidaires du corps du véhicule). 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 (accéléromètres, gyromètres). De plus amples informations sur les deux catégories de systèmes et les différents capteurs inertiels classiquement utilisés pourront être trouvées dans [SSN⁺].

1.1.1.3 Positionnement relatif "hybride"

En raison du prix des capteurs inertiels, une localisation précise par une approche purement inertielle semble incompatible avec les applications robotiques. En effet les capteurs accessibles en terme de coût présentent l'inconvénient majeur de dériver rapidement avec le temps. De nombreux systèmes [Vag93] [TaDS01] proposent de combiner simplement l'odométrie pour mesurer la vitesse linéaire et un gyromètre pour mesurer la vitesse de rotation du véhicule et ainsi remédier à ce problème de dérive. Ce type d'approche présente l'avantage d'être facile à mettre en œuvre et peu onéreux mais la localisation est dans ce cas moins précise et nécessite d'être recalée par des informations extéroceptives.

Une autre méthode proposée dans [BF96] 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 l'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 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.

1.1.2 Le positionnement absolu

Les méthodes de positionnement absolu reposent généralement sur la perception d'éléments remarquables disposés dans l'environnement 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 stratégie peuvent être rencontrés :

- 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 robot.

Quel que soit le cas de figure, le processus de positionnement nécessite toujours une connaissance des positions de ces éléments dans l'environnement.

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 contrepartie la localisation dépend directement de la visibilité des éléments remarquables par le système de perception et nécessitent souvent un aménagement adéquat 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.

1.1 Les méthodes de localisation

1.1.2.1 Positionnement par satellites

Description des systèmes existants

Trois systèmes [Thu02] de positionnement par satellites existent, 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 et de 5 stations américaines au sol dont la principale se trouve aux Etats-Unis. Le système GPS, est accessible gratuitement aux civils depuis 1995 et permet une localisation avec une précision de 10 mètres à 2σ , ce qui le rend le plus utilisé au monde.

GLONASS : GLONASS (GLObal Navigation Satellite System) [Glo02] 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. La particularité de GLONASS est que chaque satellite possède sa propre fréquence d'émission ce qui augmente la résistance au brouillage. Malheureusement, à l'heure actuelle seuls 16 satellites sont actifs ce qui ne permet de couvrir que le territoire russe.

GALILEO : GALILEO [BL01] [Gal02] 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 système GALILEO sera composé de 30 satellites. 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. Le système GALILEO devrait être opérationnel en 2013.

Principe du positionnement Le positionnement par satellites peut être réalisé selon deux méthodes différentes. La première est une simple triangulation en fonction des informations émises 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 triangulation, 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. Dans le cas du système GPS par exemple, 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 triangulation, 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 1.3).

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 (récepteur). 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

1.1 Les méthodes de localisation

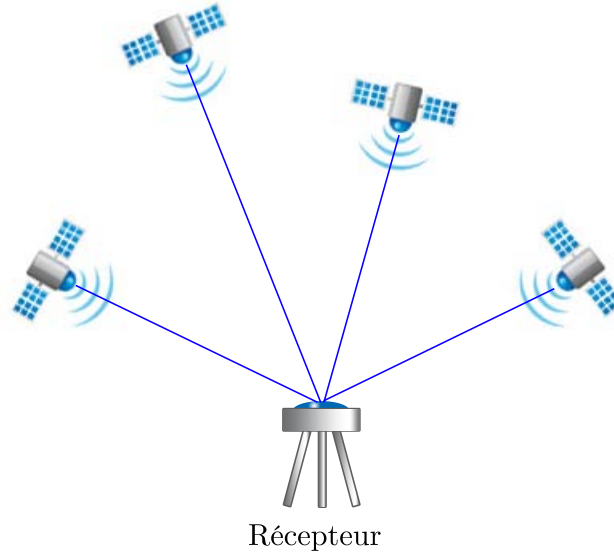


FIGURE 1.3 – Positionnement absolu par satellites

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.

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. Le récepteur intégrera les corrections dans le calcul de sa position et améliorera 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. Le récepteur mobile peut donc appliquer les corrections à ses propres mesures de pseudo-distances.

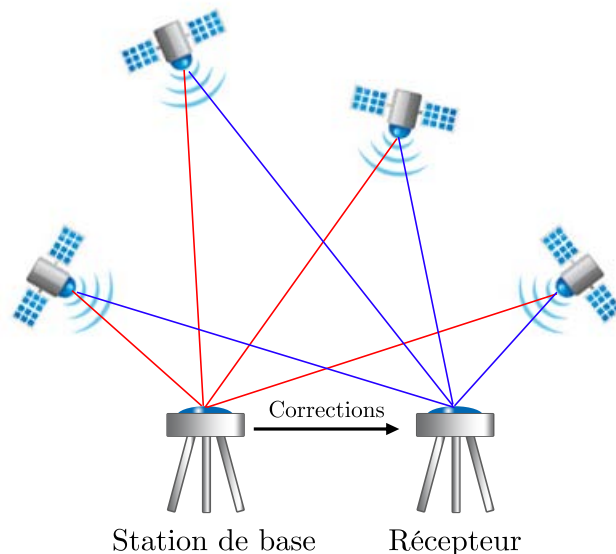


FIGURE 1.4 – Positionnement relatif par satellites

La technologie DGPS augmente considérablement la précision la faisant passer à moins d'un mètre. L'inconvénient d'un tel système est sa portée, qui n'excède pas quelques 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.

La mesure de phase : Les récepteurs GPS 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 généré 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

1.1 Les méthodes de localisation

parle alors de RTK (*Real Time Kinematic*). 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 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.

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 avec au moins 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.

Fonctionnement 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 affecte la qualité des positions calculées. Cette qualité peut être caractérisée par le calcul des facteurs de dégradation géométrique appelés DOP (*Dilution Of Precision*) donnant ainsi à l'utilisateur une idée de la confiance à accorder 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 de 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 évaluer son influence sur les mesures de pseudo-distances qui en découlent.

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 augmentent souvent la distance du trajet satellite-récepteur et donc aussi la durée de propagation du signal. Le récepteur calcule alors une distance satellite-récepteur fautive ce qui entraîne une erreur sur la position.

1.1.2.2 Positionnement local

Depuis les vingt dernières années, de nombreux travaux sur le thème de la localisation locale mono-véhicule ont vu le jour. Ces approches permettent de donner une estimation de la position du véhicule par rapport à son environnement. Si des informations de cet environnement ont été cartographiées au préalable, on obtient ainsi un positionnement absolu. Ces approches sont divisées en deux catégories. La première [Auf01] [DK02] [BBC⁺02] vise à interpréter des images afin d'en déduire la position locale du véhicule. La seconde catégorie regroupe des approches basées sur l'extraction d'indices visuels simples, comme par exemple des points de Harris [HS96], aisément extractibles dans les données capteurs. Ces approches permettent soit une estimation des déplacements [CMR07], soit une localisation par rapport à des données déjà enregistrées [RLDL07].

1.1.2.3 Bilan

Le positionnement par satellites est la principale méthode de localisation de robots mobiles en environnement extérieur. La précision de ce système de positionnement est étroitement liée au coût du récepteur. De plus, les occultations et les effets de multi-trajets dégradent fortement la précision du système et plus particulièrement en environnement urbain. Le positionnement

1.1 Les méthodes de localisation

par vision est l'approche se rapprochant le plus du système de perception humain. Compte tenu du nombre incroyable d'informations contenues dans une image, il est possible de déterminer l'attitude d'un robot dans son environnement. Ces approches de localisation locale fournissent uniquement une attitude du robot dans son référentiel. Pour combiner ces informations avec d'autres données référencées dans un repère absolu, il est nécessaire d'utiliser une cartographie 3D de l'environnement. Ainsi des approches de positionnement hybride combinant les informations des systèmes de positionnement relatif et absolu ont été développées plus récemment.

1.1.3 Les systèmes de positionnement hybride

Les différentes méthodes de positionnement relatif ou absolu 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'informations, la visibilité des satellites pour le positionnement par GPS 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 partielles et/ou entachées d'erreurs.

Pour mettre en œuvre des systèmes de positionnement hybrides, de nombreuses techniques de fusion de données sont disponibles. Une très bonne étude bibliographique est présentée dans [GLM04]. Ces techniques peuvent être séparées en quatre grandes classes : 1) les approches analytiques telles que le filtre de Kalman et ses extensions [Lan06a], 2) les approches numériques telles que le filtre particulaire [Lan06a] [MSW02], 3) les approches ensemblistes que ce soit les approches à erreur bornée [Hal97] ou les approches par analyse d'intervalles [Kie99b] [SKL05] et enfin 4) les approches basées sur la théorie des croyances [Roy02] [NB02] ou la théorie des possibilités [Ous98].

Dans cette thèse, nous allons nous intéresser plus particulièrement au filtrage basé sur le filtre de Kalman étendu. Ce filtrage permet de fusionner les informations des capteurs de façon optimale. Sa simplicité d'implémentation et sa rapidité d'exécution font de cet outil, l'un des

plus utilisés pour de la localisation en robotique mobile.

1.2 Filtrage

1.2.1 Le filtrage optimal

Le problème du filtrage consiste à estimer l'état de variables d'un système dynamique (variables d'états) 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,

$$\underline{X}_{t \geq 0} = \underline{X}_0 + \int_0^t f(\underline{X}_t, \underline{W}_t, t) dt \quad (1.11)$$

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.

$$\underline{Y}_k = h_k(\underline{X}_k) + \underline{V}_k \quad (1.12)$$

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 $n < k$.

1.2 Filtrage

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 :

$$E[(\hat{\underline{X}}_k - \underline{X}_k)(\hat{\underline{X}}_k - \underline{X}_k)^T] \quad (1.13)$$

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 :

$$\hat{\underline{X}}_k = E[\underline{X}_k | \underline{Y}_1, \dots, \underline{Y}_k] \quad (1.14)$$

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

$$\hat{\underline{X}}_k = \text{Arg Max}_{\underline{X}_k} [p(\underline{X}_k | \underline{Y}_1, \dots, \underline{Y}_k)] \quad (1.15)$$

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

$$p_{k|k} = p(\underline{X}_k | \underline{Y}_{1:k}) \quad (1.16)$$

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 itérative 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 $\underline{Y}_{1:k-1} = (\underline{Y}_1, \dots, \underline{Y}_{k-1})$

$$p_{k|k-1} = p(\underline{X}_k | \underline{Y}_{1:k-1}) \quad (1.17)$$

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 modèle d'évolution du système), en réalisant la transition de $p_{k-1|k-1}$ à $p_{k|k-1}$ de la manière suivante,

$$p_{k|k-1} = \int_{\mathcal{R}_d} p(\underline{X}_k, \underline{X}_{k-1}) p_{k-1|k-1} d\underline{X}_{k-1} \quad (1.18)$$

où \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.

$$p_{k|k} = \frac{p(\underline{Y}_k | \underline{X}_k) p(\underline{X}_k | \underline{Y}_{1:k-1})}{p(\underline{Y}_k | \underline{Y}_{1:k-1})} \quad (1.19)$$

soit

$$p_{k|k} = \frac{p(\underline{Y}_k | \underline{X}_k) p_{k|k-1}}{\int_{\mathcal{R}_d} p(\underline{Y}_k | \underline{X}_k) p_{k|k-1} d\underline{X}_k} \quad (1.20)$$

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).

1.2.2 Le filtre de Kalman

Ce filtre a été développé par Kalman [Kal60] pour le cas discret puis repris par Kalman et Bucy [KB61] pour le cas continu. Le modèle d'état est supposé linéaire, avec un bruit additif Gaussien (équation (1.21)). L'observation est une fonction linéaire de l'état entachée d'un bruit additif Gaussien (équation (1.22)).

$$\underline{X}_k = \mathbf{F}_k \underline{X}_{k-1} + \mathbf{G}_k \underline{U}_k + \Phi \underline{W}_k \quad (1.21)$$

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 $k-1$,

1.2 Filtrage

- $\underline{U}_k \in \mathcal{R}^{d_2}$ 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 Φ 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.

$$\underline{Y}_k = \mathbf{H}_k \underline{X}_k + \Psi \underline{V}_k \quad (1.22)$$

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,
- Ψ 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 :

$$\begin{aligned} p(\underline{W}_k) &\sim \mathcal{N}(0, \mathbf{Q}_k) \\ p(\underline{V}_k) &\sim \mathcal{N}(0, \mathbf{R}_k) \end{aligned} \quad (1.23)$$

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 :

$$\begin{aligned} \bar{\underline{X}}_0 &= E[\underline{X}_0] \\ \mathbf{P}_0 &= E[(\underline{X}_0 - \bar{\underline{X}}_0)(\underline{X}_0 - \bar{\underline{X}}_0)^T] \end{aligned} \quad (1.24)$$

De par le fait de la linéarité des équations d'état et de mesure et 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 (1.18) et de correction (1.20), le filtre de Kalman peut se résumer par les trois étapes suivantes.

$$\begin{cases} p(\underline{X}_{k-1} | \underline{Y}_{1:k-1}) = \mathcal{N}(\underline{X}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) \\ p(\underline{X}_k | \underline{Y}_{1:k-1}) = \mathcal{N}(\underline{X}_{k|k-1}, \mathbf{P}_{k|k-1}) \\ p(\underline{X}_k | \underline{Y}_{1:k}) = \mathcal{N}(\underline{X}_{k|k}, \mathbf{P}_{k|k}) \end{cases} \quad (1.25)$$

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

$$\underline{X}_{k|k-1} = \mathbf{F} \underline{X}_{k-1|k-1} + \mathbf{G} \underline{U}_k \quad (1.26)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^T + \Phi\mathbf{Q}_k\Phi^T \quad (1.27)$$

et

$$\underline{X}_{k|k} = \underline{X}_{k|k-1} + \mathbf{K}_k(\underline{Y}_k - \mathbf{H}\underline{X}_{k|k-1}) \quad (1.28)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_{k|k-1} \quad (1.29)$$

avec

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^T(\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^T + \Psi\mathbf{R}_k\Psi)^{-1} \quad (1.30)$$

En général, on appelle l'innovation l'écart entre la mesure et sa prédiction, elle est donnée par $\tilde{Y}_{k|k} = \underline{Y}_{k|k} - \underline{Y}_{k|k-1}$ en sachant que $\underline{Y}_{k|k-1} = \mathbf{H}_k\underline{X}_{k|k-1}$. 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.

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.

1.2.3 Le filtre de Kalman étendu EKF

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 [May79] 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

1.2 Filtrage

modifier les équations (1.21) et (1.22) de la manière suivante :

$$\begin{cases} \underline{X}_k = f_k(\underline{X}_{k-1}, \underline{U}_k, \underline{W}_k) \\ \underline{Y}_k = h_k(\underline{X}_k, \underline{V}_k) \end{cases} \quad (1.31)$$

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 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 (1.25) deviennent :

$$\begin{cases} p(\underline{X}_{k-1} | \underline{Y}_{1:k-1}) \simeq \mathcal{N}(\underline{X}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) \\ p(\underline{X}_k | \underline{Y}_{1:k-1}) \simeq \mathcal{N}(\underline{X}_{k|k-1}, \mathbf{P}_{k|k-1}) \\ p(\underline{X}_k | \underline{Y}_{1:k}) \simeq \mathcal{N}(\underline{X}_{k|k}, \mathbf{P}_{k|k}) \end{cases} \quad (1.32)$$

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

$$\underline{X}_{k|k} = f_k(\underline{X}_{k-1|k-1}, \underline{U}_k, 0) + \mathbf{F}_k [\underline{X}_{k-1} - \underline{X}_{k-1|k-1}] + \mathbf{\Phi}_k \underline{W}_k \quad (1.33)$$

où

$$\mathbf{F}_k = \left. \frac{\partial f_k(X, \underline{U}_k, 0)}{\partial X} \right|_{X=\underline{X}_{k-1|k-1}} \quad (1.34)$$

$$\mathbf{G}_k = \left. \frac{\partial f_k(\underline{X}_{k-1|k-1}, U, 0)}{\partial U} \right|_{U=\underline{U}_k} \quad (1.35)$$

$$\mathbf{\Phi}_k = \left. \frac{\partial f_k(\underline{X}_{k-1|k-1}, \underline{U}_k, W)}{\partial W} \right|_{W=\underline{W}_k} \quad (1.36)$$

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

$$\underline{Y}_k = h_k(\underline{X}_{k|k-1}, 0) + \mathbf{G}_k [\underline{X}_k - \underline{X}_{k|k-1}] + \mathbf{\Psi}_k \underline{V}_k \quad (1.37)$$

où

$$\mathbf{H}_k = \left. \frac{\partial h_k(X, 0)}{\partial X} \right|_{X=\underline{X}_{k|k-1}} \quad (1.38)$$

$$\mathbf{\Psi}_k = \left. \frac{\partial h_k(\underline{X}_{k|k-1}, V)}{\partial V} \right|_{V=\underline{V}_k} \quad (1.39)$$

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

$$\underline{X}_{k|k-1} = f_k(\underline{X}_{k-1|k-1}, \underline{U}_k, 0) \quad (1.40)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{G}_k^T \mathbf{S}_k \mathbf{G}_k^T + \Phi_k \mathbf{Q}_k \Phi_k^T \quad (1.41)$$

et

$$\underline{X}_{k|k} = \underline{X}_{k|k-1} + \mathbf{K}_k (\underline{Y}_k - h_k(\underline{X}_{k|k-1}, 0)) \quad (1.42)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \quad (1.43)$$

où

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \Psi_k \mathbf{R}_k \Psi_k)^{-1} \quad (1.44)$$

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 les estimations 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 :

$$\begin{aligned} & E[(X_{k|k-1} - \underline{X}_{k|k-1})(X_{k|k-1} - \underline{X}_{k|k-1})^T] \\ & E[(X_{k-1|k-1} - \underline{X}_{k-1|k-1})(X_{k-1|k-1} - \underline{X}_{k-1|k-1})^T] \end{aligned} \quad (1.45)$$

ainsi que du degré de non-linéarité des équations d'état et d'observation f et h . Plus ces quantités sont petites plus l'approximation est meilleure. 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 plusieurs approches ont été développées comme l'UKF [JU02], les filtres DD1, DD2 [Mag65] [MASD98] et les techniques IMM [NPR98]. Des approches reposant sur des techniques de Monté Carlo appelées filtres particulières ont aussi été développées.

1.3 Conclusion

Dans ce mémoire, nous ne décrivons cependant pas le principe de fonctionnement de ces techniques numériques car l'approche développée dans le cadre de cette thèse sur la localisation coopérative multi-véhicules repose essentiellement sur une méthode analytique type filtre de Kalman étendu. En effet, les approximations nécessaires (linéarité autour d'un point de fonctionnement) pour utiliser ce type de filtre optimal sont suffisantes dans notre cas.

1.3 Conclusion

Ce chapitre a présenté les outils utilisés classiquement pour la localisation dans le domaine de la robotique mobile et dont certains seront repris dans la suite de ce mémoire.

Tout d'abord, nous avons présenté les différentes méthodes de positionnement envisageables pour un robot mobile. Le positionnement relatif permet de mesurer les déplacements en intégrant des mesures issues de capteurs proprioceptifs tels que les capteurs odométriques ou encore une centrale inertielle. Ce type d'approche permet d'obtenir de bons résultats de positionnement relatif à court terme, mais dans le long terme, cette méthode de localisation a tendance à dériver, du fait de l'accumulation des erreurs. Les systèmes de positionnement absolu ne souffrent du problème de dérive mais restent très dépendants de l'environnement et des points de repère utilisés pour la localisation. Ceci les rend peu robustes aux occultations.

Les méthodes hybrides permettent de combiner les avantages des méthodes relatives et absolues. Les informations de positionnement absolu et relatif peuvent être combinées à l'aide de différentes techniques de fusion. Le filtre de Kalman étendu est l'une des méthodes de fusion de données les plus courantes dans le domaine de la localisation en robotique mobile du fait d'une part de ses bonnes performances en regard de sa complexité, et d'autre part du fait que son formalisme se prête particulièrement bien à l'approche d'agrégation proposée dans le cadre de ce travail de thèse.

Le chapitre suivant présente les aspects fondamentaux de la localisation coopérative de robots mobiles.

Chapitre 2

Localisation coopérative d'un groupe de robots

Sommaire

2.1	Principe de la localisation coopérative	32
2.2	Localisation coopérative centralisée	35
2.3	Localisation coopérative distribuée	37
2.3.1	Les approches basées sur l'échange des informations capteurs	38
2.3.2	Les approches basées sur l'échange d'état	39
2.4	Approche privilégiée dans ce mémoire	48
2.5	Conclusion	49

La localisation précise est l'un des plus importants problèmes pour un robot mobile autonome. Le robot doit avoir une estimation précise de sa pose (position et orientation) afin de pouvoir accomplir la tâche qui lui est confiée. Pour les applications à un seul robot comme nous l'avons vu dans le chapitre précédent, la localisation est obtenue en combinant les informations issues des capteurs proprioceptifs qui décrivent le mouvement du robot et des informations provenant de capteurs extéroceptifs tels que le GPS par exemple qui décrivent la pose du robot dans un repère absolu.

Dans le cas d'un système multi-robots, il est possible de combiner non seulement les informations propres à chaque robot, mais aussi les informations provenant des autres robots du groupe. Ce thème a fait l'objet de plusieurs recherches durant les dernières années.

Ce chapitre permet d'introduire dans un premier temps le principe général de la localisation coopérative. Dans la suite de ce chapitre, les deux principales techniques de localisation collective (centralisée et distribuée) sont décrites. Une attention particulière est donnée aux méthodes se basant sur l'échange des états des différents robots composant le groupe car l'approche présentée dans ce mémoire repose sur ce principe. Dans un dernier paragraphe, un bilan présentant les principaux avantages et inconvénients des approches décrites dans la littérature est donné ainsi que le cahier des charges que nous souhaitons respecter dans le développement de notre approche.

2.1 Principe de la localisation coopérative

Toutes les approches de localisation multi-véhicules, décrites dans la littérature, considèrent un groupe de N robots mobiles avec pour objectif majeur d'améliorer la localisation individuelle de chacun de ces robots. Chaque robot membre du groupe est équipé de capteurs lui permettant de se localiser mais également de détecter, d'identifier et de mesurer la pose relative de ses congénères.

Kurazume et *al.* ont introduit l'idée de la localisation multi-véhicules dans [KNH94]. Face au problème de l'accumulation des erreurs de localisation dans un environnement inconnu, les auteurs proposent une solution appelée *amers portables* ou *portable landmarks* où la navigation des robots se fait en groupe. Le groupe est divisé en deux équipes. Quand l'une des deux équipes

2.1 Principe de la localisation coopérative

se déplace, l'autre équipe reste fixe et sert d'amer pour la première. Dans une deuxième phase l'équipe qui était en mouvement s'arrête et sert à son tour d'amer pour la deuxième équipe qui se met à son tour en mouvement. Ce scénario est répété jusqu'à ce que le groupe atteigne son objectif. L'approche proposée permet de réduire les erreurs de localisation par rapport à une navigation basée sur des capteurs proprioceptifs. Une des particularités de cette approche est qu'aucune fusion des informations fournies par les différents robots du groupe n'est réalisée. Dans ce cas, il s'agit uniquement d'une localisation multi-véhicules et non d'une localisation coopérative multi-véhicules.

Si chaque robot est également équipé d'un système de communication afin de partager des informations avec les autres membres du groupe, la notion de localisation coopérative peut être adoptée.

La figure 2.1 présente un exemple qui montre l'effet de l'échange de données relatives et celui de la prise en compte des interdépendances sur la localisation globale du groupe.

Supposons en premier lieu que le robot 1 estime sa pose dans l'ellipse d'incertitude représentée en vert sous le robot 1. De la même manière le robot 2 estime sa pose dans l'ellipse représentée en jaune.

Quand le robot 1 détecte le robot 2, il peut estimer la pose de ce robot 2. Cette estimation est représentée dans l'ellipse verte sous le robot 2 (figure 2.1). Une fois ces informations échangées entre les deux robots, le robot 2 dispose de deux estimations de sa propre pose (celle qu'il a estimée et celle qu'il a reçue du robot 1). Il peut alors fusionner les deux estimations pour obtenir une estimation plus précise de sa pose. Cette nouvelle estimation est représentée en jaune dans la figure 2.2. A présent, les estimations des poses des robots 1 et 2 sont liées l'une à l'autre. On dit alors qu'elles sont interdépendantes.

Maintenant, si nous supposons qu'un 3ème robot arrive et détecte le robot 2. Le robot 3 est alors capable de localiser le robot 2 dans l'ellipse d'incertitude dessinée en rouge sous le robot 2 comme représentée dans la figure 2.3. En échangeant cette information avec les autres robots, le robot 2 obtient encore une fois deux estimations de sa pose, en fusionnant ces deux informations il affine encore la précision de sa localisation. Comme la pose du robot 1 était liée à la pose du robot 2, l'amélioration de la précision de l'estimation de la pose du robot 2 engendre une amélioration automatique de celle du robot 1. Cet effet est dû à la prise en compte des interdépendances entre les deux estimations.

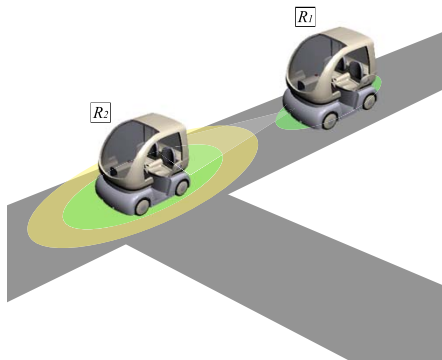


FIGURE 2.1 – Principe général : le robot 1 détecte le robot 2 et partage cette information

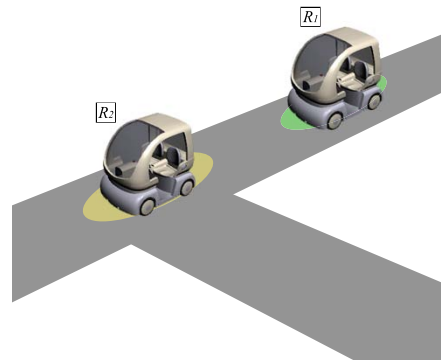


FIGURE 2.2 – Principe général : localisation des robots 1 et 2 après la fusion de données

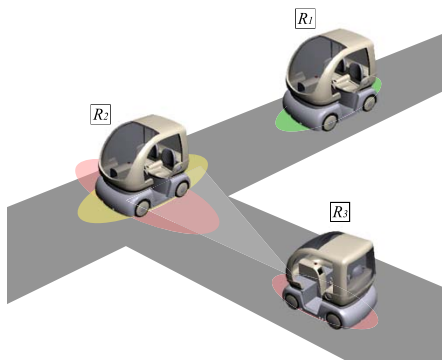


FIGURE 2.3 – Principe général : le robot 3 détecte le robot 2 et partage cette information

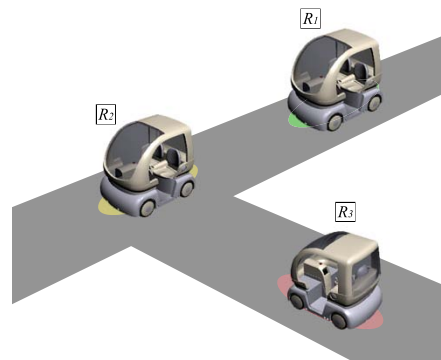


FIGURE 2.4 – Principe général : localisation des robots 1, 2 et 3 après la fusion de données

2.2 Localisation coopérative centralisée

Nous remarquons que le gain en précision d'un tel système par rapport à un système de localisation classique faisant intervenir un seul robot est dû au fait de l'utilisation des autres robots du groupe comme capteurs extéroceptifs. Les données provenant des autres membres fournissent des informations supplémentaires sur la pose du robot dans un repère absolu. L'information reçue peut aussi être considérée comme une localisation relative par rapport à un point de référence de position connue qui est dans ce cas le robot qui a envoyé cette information.

Les approches de localisation coopérative proposées dans la littérature, peuvent être regroupées selon leur topologie en deux grandes catégories :

- Les systèmes basés sur une architecture centralisée.
- Les systèmes basés sur une architecture distribuée.

Dans les sections suivantes, nous allons décrire ces deux types d'architecture ainsi que leurs avantages et inconvénients.

2.2 Localisation coopérative centralisée

L'implémentation la plus simple d'un système de localisation coopérative est fondée sur l'utilisation d'un système centralisé qui communique avec tous les robots du groupe comme représenté dans la figure 2.5.

Le système centralisé contient l'estimation de l'état du groupe et le met à jour en traitant les informations provenant des capteurs de tous ses membres. Ces informations peuvent décrire les déplacements des robots, la pose dans un repère absolu ou encore la pose relative entre deux robots.

Dans [RB00], Roumeliotis et al présentent un système de localisation coopérative d'un groupe de robots mobiles par une approche centralisée. Dans cette approche un Filtre de Kalman Etendu est utilisé pour mettre à jour l'état du groupe.

Plusieurs études se sont basées sur cette approche de Roumeliotis pour étudier par exemple la propagation des incertitudes dans le cas de N robots [RR03] ou encore les effets des informations relatives entre les robots dans un cas général [MPS05]. Des approches se proposent d'utiliser d'autres outils de fusion de données tels qu'une estimation par maximum de vraisemblance [HMS03b] et un estimateur maximum a posteriori [NRM09]. Des approches utilisant différents types de capteurs ont aussi été proposées. Dans [MG SVM05] par exemple, l'auteur présente une approche de localisation coopérative basée sur la mesure de l'orientation relative à l'aide d'une caméra vidéo. Dans [LVH⁺05] l'auteur mesure la pose relative entre les robots

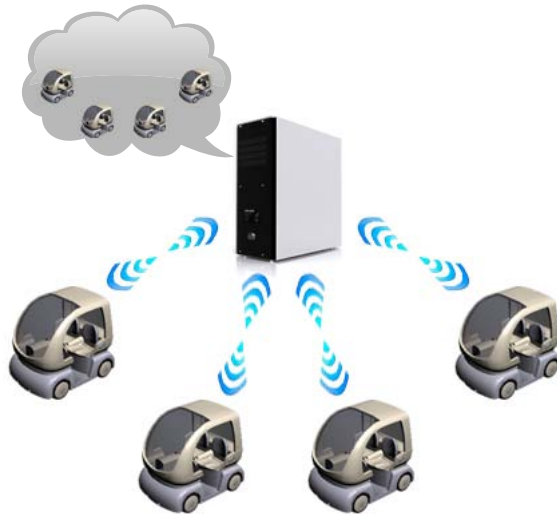


FIGURE 2.5 – Architecture centralisée pour la localisation coopérative

en utilisant la capture des ondes acoustiques audibles.

Les approches basées sur un système centralisé présentent les avantages suivants :

- La simplicité du traitement : les informations provenant des robots du groupe sont traitées de façon centralisée et le résultat est mis à la disposition de tous les membres du groupe.
- La manière naturelle avec laquelle sont gérées les interdépendances : chaque information traitée a un impact non seulement sur le robot qu'elle concerne directement, mais aussi sur tous les robots dont les poses sont liées à la pose de ce dernier.

La majorité des approches citées précédemment a seulement été implémentée et testée en simulation et fait abstraction des différents problèmes liés aux applications réelles. Dans un cas réaliste, une architecture centralisée présenterait des inconvénients de taille tels que :

- La zone d'évolution restreinte des robots : en effet on peut voir que les robots du groupe doivent être en contact permanent avec l'infrastructure centrale. La zone d'évolution des robots est de ce fait directement dépendante de la portée du système de communication choisi.
- La vulnérabilité du système : un système centralisé nécessite une infrastructure sur laquelle repose toute l'application. Une défaillance de cette infrastructure compromet tout le système.

Compte tenu de ces inconvénients majeurs, il n'y a quasiment plus, de nos jours, de travaux de recherche reposant sur ce principe de localisation collective. La solution à ces problèmes

2.3 Localisation coopérative distribuée

est de s'affranchir du système central et de distribuer l'estimation de la pose du groupe sur ses membres. Ce type d'architecture est appelé Architecture distribuée (ou décentralisée).

2.3 Localisation coopérative distribuée

La particularité d'une architecture distribuée est qu'elle ne possède pas de serveur central communiquant avec la flotte. Les traitements sont disséminés au sein même de chaque véhicule qui coopère. Nous pouvons parler dans ce cas de localisation coopérative distribuée. La figure 2.6 représente une architecture distribuée dans le cas de quatre robots mobiles.



FIGURE 2.6 – Architecture distribuée pour la localisation coopérative

Le principe de base de la localisation coopérative dans le cas d'une architecture distribuée reste le même que celui des approches basées sur une architecture centralisée. Le groupe de robots est vu comme un seul système qui est mis à jour avec les informations provenant de tous les membres. La différence est que chaque robot doit avoir une version de l'état du groupe qu'il met à jour avec ses propres informations et avec les données collectées et transmises par ses congénères.

Ce type d'architecture a été introduit dans [RB00] et [RB02] par Roumeliotis et *al.*. L'auteur propose un système de localisation collective basé sur un filtre de Kalman étendu. Cette approche est basée sur la modification des équations du filtre de Kalman pour les distribuer sur

les différents membres du groupe. D'autres travaux s'inspirent de cette approche et proposent des solutions basées sur différents capteurs et outils mathématiques pour la fusion des informations. Par exemple, une approche *égocentrique* est présentée dans [HMS03a] où chaque robot estime les positions des autres membres par rapport à la sienne à l'aide d'une caméra vidéo et d'un télémètre laser. Les informations collectées sont partagées parmi les membres du groupe et fusionnées à l'aide d'un filtre à particules.

Les approches distribuées peuvent tout de même être regroupées en deux catégories selon le type de données échangées entre les robots :

- Les approches basées sur l'échange des informations capteurs.
- Les approches basées sur l'échange d'état fusionné.

2.3.1 Les approches basées sur l'échange des informations capteurs

Dans une approche basée sur l'échange des informations capteurs, lorsqu'un robot mesure une information susceptible de mettre à jour sa version de l'état du groupe, il l'envoie aussitôt à tous ses congénères pour qu'ils puissent mettre à jour de la même manière leur propre version. Dans [MFP02] par exemple, Madhavan et *al.* présentent un système distribué de localisation coopérative d'un groupe hétérogène de robots mobiles. L'approche proposée a été implémentée et testée dans un cas réel en extérieur avec une flotte de deux robots mobiles. Pour obtenir la localisation coopérative, les robots échangent les informations sur les poses relatives (distance entre deux robots et orientation relative) ainsi que les informations qui décrivent leur positionnement (information GPS et données proprioceptives).

Ce type d'approche permet à tous les robots d'estimer l'état global du groupe de la même manière. Une fois les communications effectuées, tous les robots peuvent mettre à jour leur version de l'état du groupe et obtiennent le même résultat.

L'inconvénient majeur est la nécessité d'une disponibilité permanente de la communication entre les véhicules. Cette solution ne peut être envisagée que pour une flotte réduite de robots. En effet, plus il y a de robots dans le groupe, plus grande est la quantité de données à transmettre. Cette quantité devient encore plus importante dans le cas de l'utilisation de capteurs qui délivrent des informations à une fréquence élevée. Le cas d'un groupe de robots hétérogènes ferait encore augmenter la quantité de données transmises car en plus des données capteurs, il faudrait transmettre aussi des informations sur le type de capteurs et sur la manière dont il faut

2.3 Localisation coopérative distribuée

les traiter.

Les inconvénients de ce type d'approches les restreignent à des applications très spécifiques.

2.3.2 Les approches basées sur l'échange d'état

Pour ces approches, les robots utilisent les informations issues de leurs capteurs pour mettre à jour leur version de l'état du groupe et échangent le résultat de cette mise à jour avec leurs congénères. L'état résultant intègre non seulement les dernières informations utilisées pour la mise à jour mais aussi toutes celles qui l'ont précédées. Le robot qui reçoit cet état peut alors le fusionner avec sa propre version pour obtenir une localisation plus précise du groupe.

Ce type d'approches permet de limiter la quantité de données transmises à travers le réseau de communication. En effet, quel que soit le type de capteurs utilisé, seul l'état du groupe est échangé entre les différents congénères. La quantité de données transmises est donc limitée à la taille de l'état estimé.

De plus, n'ayant pas de problème lié au débit du réseau et à l'hétérogénéité des capteurs, le nombre de robots constituant le groupe peut être simplement étendu.

Les approches basées sur l'échange d'état font cependant face au problème de la consanguinité des données également nommé rumeur ou *Data Incest* en anglais. Ce problème survient lorsque une information est prise en compte plus d'une fois dans un système de fusion de données alors que ce même système suppose l'indépendance de ces dernières. L'effet engendré dans ce cas est une sur-convergence vers une valeur fausse.

2.3.2.1 Consanguinité des données

Les figures 2.7 et 2.14 présentent des situations typiques responsables de la consanguinité des données.

La figure 2.7 illustre le cas où la donnée est fusionnée dans le nœud 1 et le résultat de cette fusion est transmis au nœud 2 où il est fusionné avec d'autres données. Le résultat de la fusion dans le nœud 2 est retransmis au nœud 1 pour une dernière fusion. Dans ce cas, la donnée est fusionnée 2 fois dans le nœud 1. La première fois de façon directe et la deuxième de façon

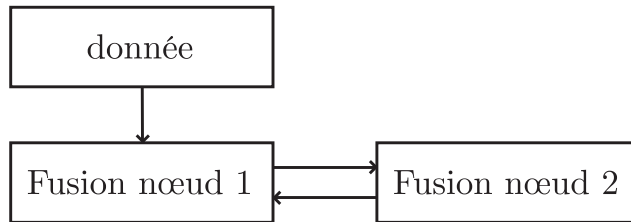


FIGURE 2.7 – Cause de la sur-convergence schéma 1

indirecte en passant par le nœud 2.

Ce cas peut également être transposé au problème rencontré dans les approches de localisation coopérative. Les nœuds 1 et 2 de la figure 2.7 représentent les robots 1 et 2 et la donnée peut représenter une mesure de la pose relative entre les deux robots par exemple. Le robot 1 met à jour son estimation du positionnement du groupe (détection du robot 2 dans notre cas) et transmet le résultat au robot 2. Ce dernier le fusionne avec sa propre estimation du positionnement du groupe. Lorsque le robot 2 reçoit une autre information il peut alors la fusionner avec son estimation et transmettre le résultat au robot 1 qui va le fusionner avec sa propre version. Après la fusion le robot aura fusionné la donnée sur la pose relative deux fois.

La figure 2.8 représente un exemple concret d'une situation de consanguinité des données dans le cas d'un groupe de deux robots R_1 et R_2 .

Le robot R_1 se localise à l'aide de ses capteurs proprioceptifs et extéroceptifs dans l'ellipse d'incertitude représentée en vert. De la même manière le robot R_2 se localise dans l'ellipse représentée en rouge.

Supposons que le robot R_1 détecte le robot R_2 et le localise dans l'ellipse verte représentée sous le robot R_2 dans la figure 2.9. Notons que cette estimation est calculée à partir de la pose du robot R_1 et de la mesure de la pose relative entre les deux robots. L'estimation de la pose du robot R_2 dépend donc fortement de celle du robot R_1 . De la même manière le robot R_2 détecte et localise le robot R_1 dans l'ellipse d'incertitude représentée en rouge sous le robot R_1 dans la figure 2.9. Cette estimation dépend elle aussi de celle du robot R_2 .

A cet instant, chacun des deux robots dispose d'une estimation de sa propre pose et une estimation de la pose de l'autre robot. Une fois la communication établie, les deux robots échangent ces informations.

2.3 Localisation coopérative distribuée

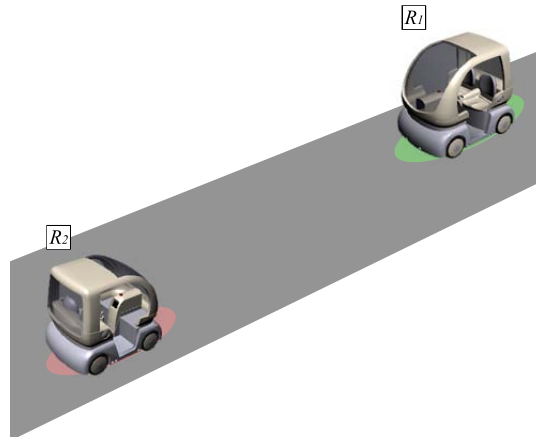


FIGURE 2.8 – Problème de sur-convergence : Etats initiaux des robots R_1 et R_2

Après la communication chacun des deux robots dispose de deux estimations de sa pose, celle qu'il a estimée avec ses capteurs et celle qu'il a reçue de l'autre robot. La fusion de ces deux estimations dans les deux robots leur donne une meilleure précision de localisation. L'erreur de localisation des deux robots après la fusion est représentée dans la figure 2.10.

Après la fusion, les estimations des poses du robot R_1 et R_2 sont dépendantes l'une de l'autre. En effet la pose du robot R_1 après la fusion dépend de son estimation propre et celle que lui a fournie le robot R_2 . De la même manière la nouvelle pose du robot R_2 dépend de sa propre pose et de celle du robot R_1 .

Supposons maintenant que les deux robots se déplacent et se localisent comme représenté dans la figure 2.11. La pose estimée des deux robots résulte de l'évolution des poses fusionnées à l'étape précédente et reste de ce fait toujours dépendantes l'une de l'autre.

Lorsque les deux robots se détectent une deuxième fois, chaque robot peut estimer la pose de l'autre. Le robot R_1 estime la pose du robot R_2 en utilisant l'estimation de sa propre pose et de la mesure de la pose relative du robot R_2 . La pose estimée du robot R_2 dépend donc de la pose du robot R_1 qui elle même depuis la fusion précédente dépend de la pose du robot R_2 . Donc la pose estimée du robot R_2 dépend de l'ancienne pose du robot R_2 .

De la même manière le robot R_2 estime la pose du robot R_1 et de la même manière cette pose dépendra de l'ancienne pose du robot R_1 .

Après la communication les deux robots disposent encore une fois de deux estimations de leurs

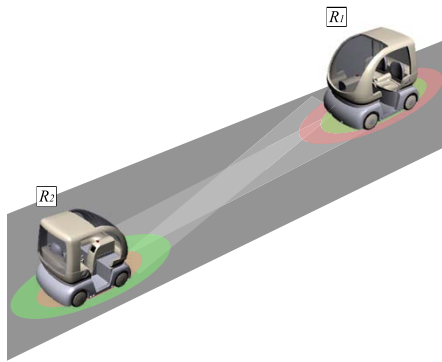


FIGURE 2.9 – Problème de sur-convergence : Détection et localisation des robots

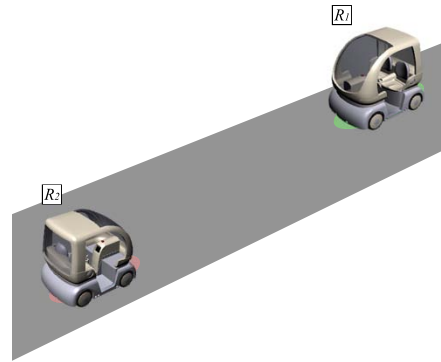


FIGURE 2.10 – Problème de sur-convergence : Echange des données de localisation et fusion

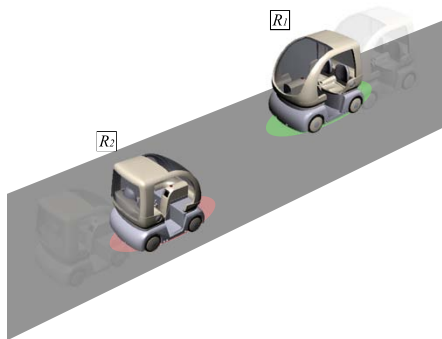


FIGURE 2.11 – Problème de sur-convergence : Evolution de l'état des 2 robots

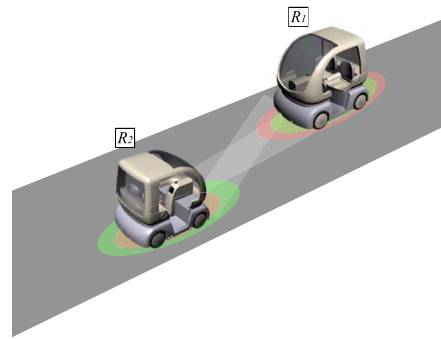


FIGURE 2.12 – Problème de sur-convergence : Détection, localisation et échange de données.

2.3 Localisation coopérative distribuée

poses, celle estimée avec leurs propres capteurs et celle reçue de l'autre robot. Comme nous l'avons vu, la pose du robot R_1 estimée avec ses propres capteurs est fortement liée à la pose qu'il a reçue du robot R_2 . La fusion de ses deux poses pourrait engendrer une sur-convergence vers une position fautive. Elle devient alors prohibée. De la même manière le robot R_2 ne peut plus utiliser les informations provenant du robot R_1 comme leurs deux poses sont dépendantes l'une de l'autre.

La figure 2.13 montre le cheminement que prennent les données dans cet exemple. Les flèches représentées en rouge montrent que la même donnée a pris deux chemins différents pour être fusionnée deux fois à l'étape 5.

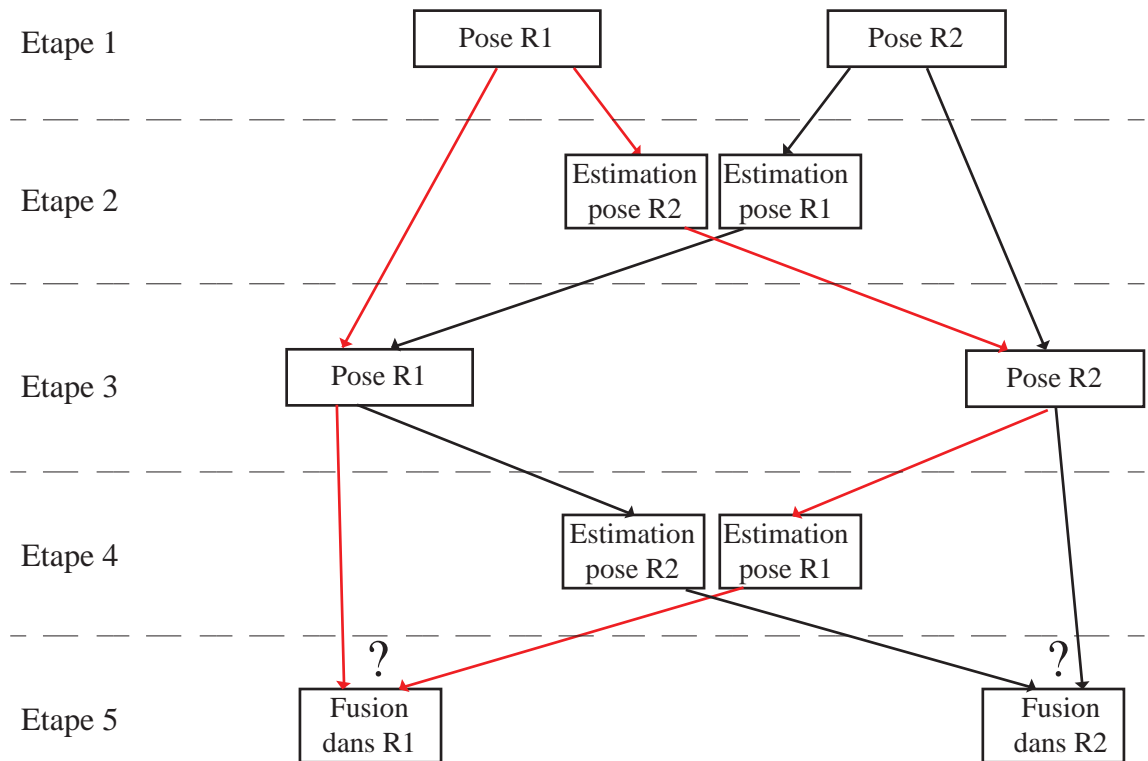


FIGURE 2.13 – Cheminement des données pour l'exemple à deux robots

La figure 2.14 présente un autre cas d'un problème de consanguinité des données. Dans ce cas la même donnée est fusionnée dans deux nœuds différents et les résultats de ces fusions sont fusionnés dans un troisième nœud. Dans ce cas la donnée a été prise en compte deux fois dans le nœud 3.

Ce cas peut être transposé au problème rencontré dans les approches de localisation coopérative.

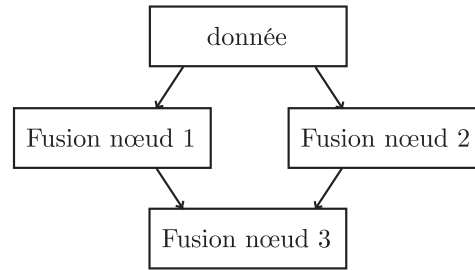


FIGURE 2.14 – Cause de la sur-convergence schéma 2

Les figures 2.15, 2.16 et 2.17 représentent un exemple concret d'une situation de consanguinité des données dans le cas d'un groupe de trois robots R_1 , R_2 et R_3 .

Sur la figure 2.15, le robot R_1 se localise à l'aide de ses capteurs proprioceptifs et extéroceptifs dans l'ellipse d'incertitude représentée en jaune. De la même manière le robot R_2 se localise dans l'ellipse représentée en vert. De plus ces 2 robots sont capables de se détecter mutuellement et de se communiquer l'estimation de leur pose ainsi que celle du robot détecté. La donnée dans ce cas représente la mesure de la distance relative entre les 2 robots.

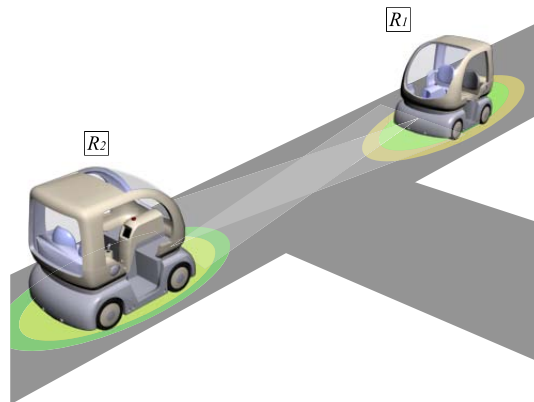


FIGURE 2.15 – Problème de sur-convergence : Etats initiaux des robots R_1 et R_2

Après la communication, chacun des deux robots dispose de deux estimations de sa pose, celle qu'il a estimée avec ses capteurs (en jaune pour le robot R_1 et en vert pour le robot R_2)

2.3 Localisation coopérative distribuée

et celle qu'il a reçue de l'autre robot (en vert pour le robot R_1 et en jaune pour le robot R_2). La fusion de ces deux estimations dans les deux robots leur donne une meilleure précision de localisation (voir figure 2.16) mais elles sont totalement dépendantes l'une de l'autre.

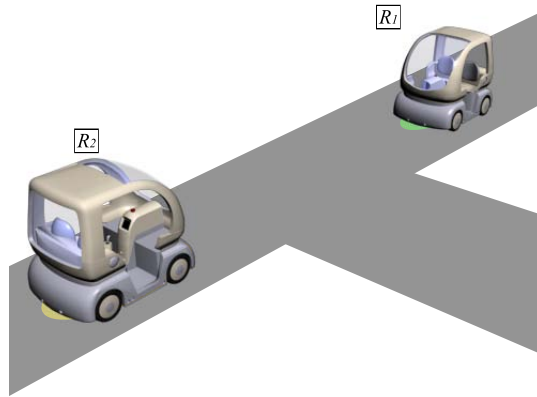


FIGURE 2.16 – Problème de sur-convergence : Echange des données de localisation et fusion

Sur la figure 2.17, le robot R_3 , qui arrive dans la même zone, est détecté par les 2 robots R_1 et R_2 . Après une phase de communication, le robot R_3 dispose ainsi de 3 différentes estimations de sa pose : sa propre estimation et les estimations fournies par les robots R_1 et R_2 qui sont dépendantes de leur propre pose et de la mesure des distances relatives entre les différents robots. Les 2 estimations des robots R_1 et R_2 ne sont pas totalement indépendantes car elles ont déjà pris en compte la mesure de leur distance relative. Ainsi, la fusion dans le robot R_3 des différentes estimations de sa pose est prohibée car la mesure de la distance relative entre les robots R_1 et R_2 serait indirectement prise 2 fois en considération.

2.3.2.2 Les solutions au problème de consanguinité des données dans la littérature

Le problème de consanguinité des données dans le cadre de la localisation collective a été traité dans plusieurs travaux ces dernières années.

Dans [HMS03a], Howard et *al.* présentent une approche dite *égocentrique* dans laquelle chaque robot du groupe estime les poses des autres robots par rapport à la sienne. Il envoie ensuite ses estimations aux autres afin d'être fusionnées pour obtenir une localisation coopérative. Les don-

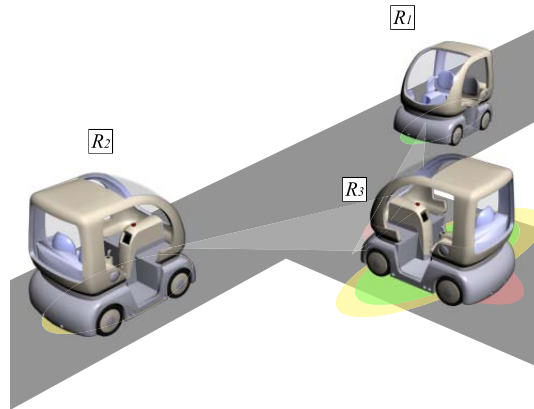


FIGURE 2.17 – Dépendance des données et sur-convergence

nées échangées dans cette approche sont les distributions des poses mesurées. Après l'échange des données, chaque robot met à jour la distribution de sa pose avec celle qu'il a reçue du robot qui l'a détecté. Pour contourner le problème de consanguinité des données et éviter les mises à jour circulaires, l'auteur tient à jour un arbre de dépendances dans lequel chacune des distributions des poses des robots du groupe constitue un nœud qui ne peut avoir qu'un seul parent et aucun ou plusieurs enfants tel qu'illustré dans la figure 2.18. Quand une distribution est utilisée pour mettre à jour une autre elle devient automatiquement un parent et les distributions ne peuvent pas mettre à jour un ascendant mais peuvent mettre à jour leurs descendants. Et pour éviter des situations de blocage l'auteur suppose qu'une distribution ne dépend que du parent direct ce qui autorise une distribution à mettre à jour le parent d'un parent. Cette approche ne résout pas réellement le problème de la consanguinité et présente encore de nombreux cas où une sur-convergence peut survenir. Le cas où un parent peut mettre à jour tous ses descendants en incluant les descendants des descendants est un cas typique de consanguinité et la supposition que la distribution ne dépend que du parent direct est une approximation assez grossière.

Roumeliotis et *al.* proposent dans [RR03] une solution plus rigoureuse au problème de consanguinité. Afin de distribuer le calcul de l'état du groupe sur les membres de ce dernier, l'auteur fait remarquer que dans le cas d'une mesure de pose relative simple $(\Delta x, \Delta y, \Delta \varphi)$ représenté dans la figure 2.19, l'équation de l'innovation et celle de sa matrice de covariance ne dépendent que des paramètres liés au deux robots concernés par la mesure relative. Les auteurs précisent que dans ce cas les deux robots peuvent échanger une partie de l'état du groupe (celle

2.3 Localisation coopérative distribuée

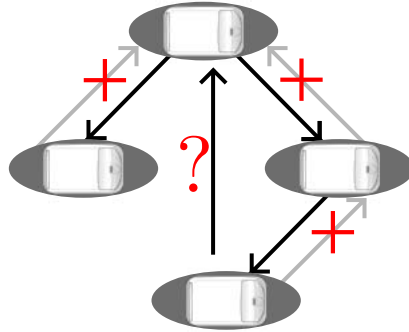


FIGURE 2.18 – Exemple d'un arbre de dépendances dans la cas d'un groupe de 4 robots.

qui les concerne) et la mesure de la pose relative ($\Delta x, \Delta y, \Delta \varphi$) afin que les deux robots puissent mettre à jour leur version de l'état du groupe.

Cette approche est une solution au problème de la consanguinité des données dans le cas

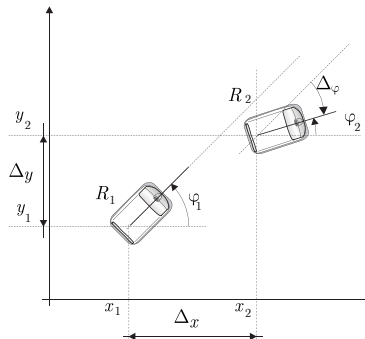


FIGURE 2.19 – Pose relative entre deux robots mobiles.

d'une approche de localisation coopérative distribuée par échange d'état. Elle combine les informations relatives mesurées entre les robots en tenant compte des interdépendances entre les estimations des poses de ses derniers, mais elle n'est pas optimale au sens où elle ne tient pas compte des informations provenant de tous les robots du groupe. L'exemple suivant permet de montrer un cas qui illustre cet inconvénient.

Prenons le cas d'un groupe de 3 robots mobiles (R_1, R_2 et R_3). Supposons que le robot R_1 dispose dans sa version de l'état du groupe une estimation des poses des 3 robots dépendantes les unes des autres. Le robot R_2 dispose aussi d'une estimation des poses des 3 robots mais dans son cas, il n'y a que les poses des robots R_1 et R_2 qui sont dépendantes. Celle du robot R_3 est indépendante des deux autres (ce cas peut arriver quand aucune pose relative n'a été mesurée).

Supposons que les robots R_1 et R_2 se croisent et mesurent leur pose relative. Les deux robots échangent alors les parties de l'état du groupe qui les concernent avec les informations sur les interdépendances qui les lient et la mesure de la pose relative pour que chaque robot puisse mettre à jour sa version de l'état du groupe.

Dans le robot R_1 , comme les poses des 3 robots sont dépendantes les unes des autres, la mise à jour des poses des robots R_1 et R_2 affecte aussi celle du robot R_3 . La pose du robot R_3 devient alors plus précise. Alors que dans le robot R_2 , comme la pose du robot R_3 est indépendante des deux autres, elle reste inchangée après la mise à jour.

Dans ce cas le robot R_2 ne profite pas de toutes les informations dont dispose le robot R_1 . La mise à jour dans le robot R_2 est donc juste mais pas optimale.

Par ailleurs, une solution approchée au problème de rumeur pourrait être les approches pessimistes de fusion telles que les techniques exploitant l'analyse d'intervalles [Kie99a] [SL05]. Dans ces cas, le problème sera, du fait du caractère pessimiste des approches, qu'aucun apport réel de précision ne sera donné.

Le problème de consanguinité des données est donc resté un problème ouvert pour lequel aucune solution satisfaisante n'a été apportée. L'approche présentée ici apporte une solution à ce problème.

2.4 Approche privilégiée dans ce mémoire

Dans ce mémoire nous présentons ainsi une approche de localisation coopérative d'un groupe de robots mobiles. Cette approche doit répondre au cahier des charges suivant :

- Le système de fusion de données doit intégrer toutes les informations disponibles de façon optimale en évitant le problème de consanguinité.
- L'approche proposée doit pouvoir être applicable dans un cas réel. Elle doit de ce fait être robuste à tous les problèmes liés à ce type d'applications tels que les limites fonctionnelles des capteurs, les pertes et les retards des communications.
- L'approche proposée doit pouvoir fonctionner en temps réel avec un grand nombre de robots.
- L'approche proposée doit pouvoir également être applicable dans un environnement ouvert dans lequel peuvent évoluer d'autres véhicules que les robots du groupe.

2.5 Conclusion

Dans ce chapitre nous avons présenté les différentes approches de localisation coopérative d'un groupe de robots mobiles. Nous avons identifié du point de vue de la topologie utilisée, deux grandes catégories d'approches : les approches basées sur une architecture centralisée et les approches basées sur une architecture distribuée.

L'architecture centralisée présente une solution théoriquement simple et rigoureuse aux problèmes liés à la localisation coopérative. Mais les inconvénients dont cette architecture souffre la rendent inadaptée aux applications réelles faisant intervenir un grand nombre de robots. En effet plus le nombre de robots augmente, plus la bande passante utilisée du réseau de communication sera élevée. Et plus la zone d'évolution des robots est grande, plus l'infrastructure nécessaire est onéreuse et difficile à mettre en place. De plus une telle architecture repose entièrement sur un seul point, ce qui la rend peu tolérante aux pannes. Aussi nous nous sommes orientés vers une architecture distribuée.

Dans les approches distribuées nous avons identifié deux catégories selon le type de données échangées : les approches basées sur l'échange des données capteurs et les approches basées sur l'échange de l'état mis à jour

- L'échange des données capteurs nécessite la transmission d'une grande quantité de données à travers le réseau de communication. Cette quantité dépend du nombre de robots qui composent le groupe et des types de capteurs utilisés.
- Les approches basées sur l'échange d'état permettent de limiter la quantité des données à transmettre à la taille de l'état du groupe. Cette quantité est indépendante du type des capteurs utilisés et de leur fréquence de rafraîchissement. Le défaut majeur de ces approches reste la consanguinité des données.

Dans la partie qui suit, nous présentons une approche de localisation collective distribuée originale basée sur des échanges d'états. La fusion de données est faite à l'aide d'un filtre de Kalman étendu. Dans cette approche nous proposons une solution qui permet d'obtenir une localisation coopérative du groupe de robots de façon optimale tout en tenant compte de toutes les interdépendances éventuelles entre les poses des robots du groupe évitant ainsi le problème de rumeur.

Nous verrons aussi que cette approche satisfait aussi toutes les contraintes imposées par le cahier des charges décrit dans cette section.

Deuxième partie

Localisation collective par fusion d'états

Dans cette partie nous détaillons notre approche de localisation collective. Cette approche par échange d'états a pour principal objectif de localiser un groupe de robots mobiles avec le maximum de précision possible tout en garantissant notamment une intégrité des résultats (gérer les problèmes de consanguinité par exemple). Cette approche est également capable de gérer tous les problèmes liés à un fonctionnement en situation réelle (pertes et retards de communication, limite des capteurs, etc). Le principe général de l'approche est d'effectuer la fusion des états en deux étapes. La première de façon individuelle avec les informations propres à chacun des robots et la seconde plus collective pour tenir compte de tous les états indépendants des robots du groupe.

Afin de s'affranchir des problèmes liés aux applications réelles et de se focaliser sur l'approche en elle-même, nous allons dans un premier temps nous mettre dans un cas idéal. Dans ce cas les problèmes tels que les retards qui affectent les données capteurs et les communications, les coupures de communications, etc, sont ignorés. Nous allons aussi supposer que le groupe de robots évolue dans un environnement fermé où seuls les membres du groupe sont présents. Nous supposons aussi que le nombre de robots du groupe est connu a priori et que les robots sont équipés de capteurs qui leur permettent de localiser et d'identifier tous les autres membres du groupe. Ce cas fera l'objet du chapitre 3.

Dans le chapitre 4 nous nous remettons dans un cas réaliste dans lequel les contraintes des applications réelles sont prises en compte. La première section de ce chapitre présente la stratégie utilisée pour régler le problème de désynchronisation des données et celui des retards liés aux algorithmes de traitement ou aux latences de communication et de transfert des données.

Dans la seconde section, le groupe est supposé évoluer dans un environnement ouvert où les robots peuvent croiser aussi bien des membres de leur groupe que d'autres véhicules non équipés. Le nombre de robots n'est pas connu à l'avance, l'approche doit gérer l'insertion et le retrait de robots dans le groupe. La difficulté dans ce cas réside dans le fait que les robots peuvent localiser les véhicules présents dans le champs de mesure de leurs capteurs (qu'ils appartiennent au groupe ou pas) sans pouvoir les identifier.

Chapitre 3

Localisation collective : cas idéal

Sommaire

3.1 Estimation de l'état du groupe	56
3.1.1 Evolution de l'état global du groupe	58
3.1.2 Mise à jour de l'état du groupe	59
3.2 Localisation collective	61
3.2.1 Principe général	61
3.2.2 Notion de sous-états	62
3.2.3 Fusion décentralisée	63
3.3 Conclusion	67

Ce chapitre a pour but de détailler l'approche de localisation collective par fusion d'état dans un cas idéal. En effet, les approches basées sur un échange des états de chaque robot du groupe, comme la notre, posent classiquement quelques problèmes. Nous supposons donc dans un premier temps le cas idéal où nous allons ignorer les problèmes applicatifs afin de nous focaliser sur la méthodologie et le principe de résolution de la fusion collective en évitant le phénomène de sur-convergence présenté dans le chapitre 2 de la partie I.

Le problème est modélisé comme suit : soit un groupe hétérogène de N robots mobiles R_i avec $i=1\dots N$ sachant que dans la suite de ce mémoire nous considérons que le robot du groupe où s'exécute la procédure que nous présentons sera nommé R_s (s pour self (soi)).

Les suppositions liées au cas idéal sont les suivantes :

- Chaque robot est capable de se localiser dans un référentiel absolu et commun à tous les

robots du groupe.

- Chaque robot est capable de localiser et d'identifier ses congénères.
- Tous les robots du groupe sont équipés d'un système de communication permettant l'échange de données entre eux.

Le but de l'approche proposée est d'obtenir une estimation la plus précise possible de l'état du groupe en tenant compte des interdépendances éventuelles entre les estimations des localisations de ses membres.

Cette approche peut être décomposée en deux parties. La première partie décrit le traitement qui s'effectue au sein de chaque robot afin de se localiser, de détecter et de localiser les autres membres du groupe. Ce traitement permet aux robots de construire une estimation de l'état du groupe. L'état global du groupe est obtenu en fusionnant les estimations de l'état venant de chacun de ses membres. L'étape de fusion est décrite dans la seconde section de ce chapitre.

3.1 Estimation de l'état du groupe

Nous considérons ici que l'état du robot R_s est représenté par un vecteur d'état $\underline{x}_s = [x_s, y_s, \varphi_s, v_s]^T$, où x_s et y_s sont les coordonnées du robot considéré dans un repère absolu, φ_s son orientation et v_s le module de sa vitesse tel que présenté dans la figure 3.1.

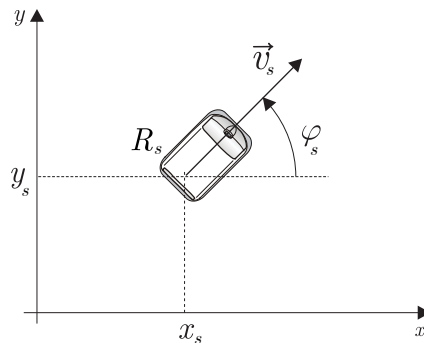


FIGURE 3.1 – Etat du robot R_s

Ce vecteur d'état est associé à une matrice de covariance P_s (équation (3.1)) qui décrit les

3.1 Estimation de l'état du groupe

erreurs liées à ses paramètres.

$$P_s = \begin{bmatrix} \sigma_{x_s}^2 & \sigma_{x_s y_s} & \sigma_{x_s \varphi_s} & \sigma_{x_s v_s} \\ \sigma_{y_s x_s} & \sigma_{y_s}^2 & \sigma_{y_s \varphi_s} & \sigma_{y_s v_s} \\ \sigma_{\varphi_s x_s} & \sigma_{\varphi_s y_s} & \sigma_{\varphi_s}^2 & \sigma_{\varphi_s v_s} \\ \sigma_{v_s x_s} & \sigma_{v_s y_s} & \sigma_{v_s \varphi_s} & \sigma_{v_s}^2 \end{bmatrix} \quad (3.1)$$

Nous supposons ainsi un modèle d'erreur gaussien avec :

$$\underline{x}_s \sim \mathcal{N}(\bar{x}_s, P_s) \quad (3.2)$$

où \bar{x}_s est l'espérance mathématique $E[\underline{x}_s]$.

Pour faire de la localisation collective, le groupe est représenté par un seul système $W = (\underline{X}, P)$, où \underline{X} est le vecteur d'état global qui regroupe les états de tous les robots du groupe et P sa matrice de covariance suivant les équations (3.3) et (3.4).

$$\underline{X} = [\underline{x}_1^T, \underline{x}_2^T, \dots, \underline{x}_s^T, \dots, \underline{x}_N^T]^T \quad (3.3)$$

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & \dots & P_{1,N} \\ P_{2,1} & P_{2,2} & \dots & P_{2,N} \\ \dots & \dots & P_{s,s} & \dots \\ P_{N,1} & P_{N,2} & \dots & P_{N,N} \end{bmatrix} \quad (3.4)$$

La matrice de covariance $P_{s,s}$ est équivalente à la matrice de covariance P_s du vecteur d'état.

Les sections suivantes décrivent le traitement effectué dans le robot R_s afin de maintenir à jour une estimation W^s de l'état du groupe W . Ce traitement consiste à fusionner les données proprioceptives et extéroceptives à l'aide d'un filtre de Kalman.

Le filtre de Kalman opère en deux phases : *Prédiction* et *Mise à jour*. La phase de prédiction utilise l'état estimé de l'instant précédent pour produire une prédiction de l'état courant. Dans l'étape de mise à jour, les observations de l'instant courant sont utilisées pour corriger l'état prédit dans le but d'obtenir une estimation plus précise.

La phase de prédiction est décrite dans la section 3.1.1 où un modèle d'évolution utilisant les données proprioceptives est utilisé pour l'estimation. La phase de mise à jour utilisant les données extéroceptives est détaillée dans la section 3.1.2.

3.1.1 Evolution de l'état global du groupe

En considérant que le traitement est réalisé au sein du robot R_s , celui-ci fait évoluer sa pose (le vecteur d'état \underline{x}_s et sa matrice de covariance P_s) selon un modèle qui tient compte des informations délivrées par ses capteurs proprioceptifs afin de décrire sa trajectoire de façon fidèle. Il fait aussi évoluer les poses des autres robots R_i (avec $i=1\dots N$ et $i \neq s$) selon un modèle qui ne dépend que des informations dont il dispose et qui sont estimées dans l'état du groupe. Ce modèle peut être un modèle cinématique à vitesse constante et cap constant ou encore un modèle à accélération constante.

La position du robot R_s évolue selon l'équation (3.5), celle des autres robots R_i par la relation (3.6)

$$\underline{x}_{s_{k+1}}^- = f_s(\underline{x}_{s_k}, u_{s_k}) \quad (3.5)$$

$$\underline{x}_{i_{k+1}}^- = \tilde{f}_s(\underline{x}_{i_k}) \quad (3.6)$$

Où $\underline{x}_{s_{k+1}}^-$ et $\underline{x}_{i_{k+1}}^-$ sont respectivement les prédictions des états des robots R_s et R_i , k représente le temps et u_{s_k} les informations proprioceptives qui décrivent le mouvement du robot R_s . Nous supposons que les informations capteurs $u_{s_k}^m$ sont affectées par des erreurs, la mesure réelle u_{s_k} est alors modélisée par :

$$u_{s_k} \sim \mathcal{N}(u_{s_k}^m, Q_{s_k}) \quad (3.7)$$

Q_{s_k} étant la covariance du bruit qui affecte la mesure $u_{s_k}^m$.

L'équation d'évolution des états des robots et de leurs matrices de covariance devient :

$$\hat{\underline{x}}_{s_{k+1}}^- = f_s(\hat{\underline{x}}_{s_k}, u_{s_k}^m) \quad (3.8)$$

$$\hat{\underline{x}}_{i_{k+1}}^- = \tilde{f}_s(\hat{\underline{x}}_{i_k}) \quad (3.9)$$

$$P_{ss_{k+1}}^- = F_{sx_{s_k}} P_{ss_k} F_{sx_{s_k}}^T + F_{su_{s_k}} Q_{s_k} F_{su_{s_k}}^T \quad (3.10)$$

$$P_{si_{k+1}}^- = \tilde{F}_{sx_{i_k}} P_{si_k} \tilde{F}_{sx_{i_k}}^T \quad (3.11)$$

Où

- $\hat{\underline{x}}_{s_k}^-$ et $\hat{\underline{x}}_{i_k}^-$ sont respectivement les positions prédites de \underline{x}_{s_k} et \underline{x}_{i_k} à l'instant k ,
- $F_{sx_{s_k}}$ et $F_{su_{s_k}}$ sont respectivement les jacobiennes de la fonction f_s par rapport à \underline{x}_s et u_s ,
- $\tilde{F}_{sx_{i_k}}$ la jacobienne de la fonction \tilde{f}_s par rapport à \underline{x}_i ,
- $P_{si_{k+1}}^-$ caractérise la matrice de covariance associée à l'estimation de la pose d'un robot R_i par le robot R_s .

Chaque robot fait évoluer l'état du groupe selon la procédure que nous venons de décrire et ce jusqu'à l'arrivée d'informations extéroceptives qui permettent de mettre à jour l'état estimé. Ce processus de mise à jour fait l'objet de la prochaine section.

3.1 Estimation de l'état du groupe

3.1.2 Mise à jour de l'état du groupe

A l'arrivée d'une information extéroceptive z_k (mesure GPS par exemple), le robot R_s met à jour W^s l'état estimé du groupe (i.e. l'état du groupe vu par le robot R_s). z_k est une variable aléatoire modélisée par l'équation (3.12)

$$z_k \sim \mathcal{N}(z_k^m, B_k) \quad (3.12)$$

Où z_k^m est la valeur mesurée et B_k est la variance du bruit qui affecte cette mesure.

La mise à jour de l'état W^s est faite selon les équations (3.13), (3.14), (3.15), (3.16) et (3.17).

$$z_{k+1}^- = (z_{k+1}^m - H_{k+1} \hat{X}_{k+1}^-) \quad (3.13)$$

$$S_{k+1} = H_{k+1} P_{k+1}^- H_{k+1}^T + B_{k+1} \quad (3.14)$$

$$K_{k+1} = P_{k+1}^- H_{k+1}^T S_{k+1}^{-1} \quad (3.15)$$

$$\hat{X}_{k+1} = \hat{X}_{k+1}^- + K_{k+1} z_{k+1}^- \quad (3.16)$$

$$P_{k+1} = (I - K_{k+1} H_{k+1}) P_{k+1}^- \quad (3.17)$$

Où

- z_{k+1}^- représente l'innovation de l'observation,
- S_{k+1} sa matrice de covariance,
- K_{k+1} représente le gain de Kalman,
- $\hat{X}_{k+1}^- = [\hat{x}_{1k+1}^-, \hat{x}_{2k+1}^-, \dots, \hat{x}_{Nk+1}^-]^T$ la prédiction du vecteur d'état X_{k+1} ,
- $\hat{X}_{k+1} = [\hat{x}_{1k+1}, \hat{x}_{2k+1}, \dots, \hat{x}_{Nk+1}]^T$ le vecteur d'état mis à jour,
- P_{k+1} sa matrice de covariance,
- z_{k+1}^m l'observation mesurée,
- H_{k+1} la matrice de mise à jour.

La matrice de mise à jour H_{k+1} dépend du type d'observation. Si l'observation est une information de positionnement absolu par GPS par exemple où les coordonnées x et y sont mesurées et sachant que nous considérons que le robot R_s sur lequel s'exécute la procédure est le premier de la liste, la matrice de mise à jour H_{k+1} sera donnée par l'équation (3.18) :

$$H_{k+1} = \begin{bmatrix} I_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 4(N-1)} \end{bmatrix} \quad (3.18)$$

Où $I_{n \times n}$ est une matrice identité de $n \times n$ dimensions et $0_{n \times n}$ est une matrice de zéros de $n \times n$ dimensions.

Quand le robot R_s détecte un autre robot R_i , il peut alors mesurer sa pose relative (position et orientation relative) cette observation peut être décrite par $z_k=(\Delta x_{si}, \Delta y_{si}, \Delta\varphi_{si})$ (voir figure 3.2).

$$\Delta x_{si} = x_i - x_s \quad (3.19)$$

$$\Delta y_{si} = y_i - y_s \quad (3.20)$$

$$\Delta\varphi_{si} = \varphi_s - \varphi_i \quad (3.21)$$

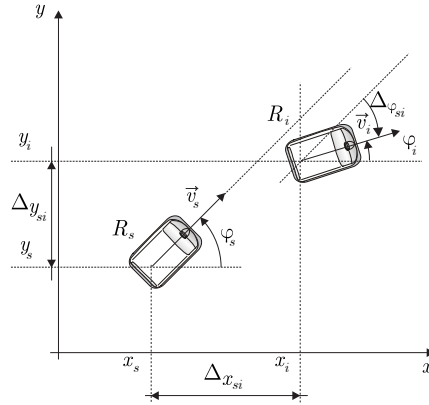


FIGURE 3.2 – Pose relative entre les robots du groupe

La matrice de mise à jour H_{k+1} dans ce cas sera une matrice de dimension $3 \times N$ avec $I_{3 \times N}$ de la colonne $4i - 3$ à $4i - 1$ et $-I_{3 \times N}$ de la colonne $4j - 3$ à $4j - 1$ et zéro ailleurs.

Exemple : si nous considérons que le robot R_s est le robot 2 dans un groupe de 3 robots et qu'il veuille mettre à jour l'état du robot 3, l'état du groupe serait donné par les équations (3.22) et (3.23) et la matrice de mise à jour H_{k+1} serait donnée par l'équation (3.24).

$$\underline{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (3.22)$$

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} \\ P_{2,1} & P_{2,2} & P_{2,3} \\ P_{3,1} & P_{3,2} & P_{3,3} \end{bmatrix} \quad (3.23)$$

3.2 Localisation collective

$$H_{k+1} = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \end{array} \right] \quad (3.24)$$

Ainsi pour le robot R_s , on obtiendra un état \underline{X}^s et la matrice associée P^s de l'ensemble W^s de la flotte.

3.2 Localisation collective

3.2.1 Principe général

Nous avons vu dans la section précédente comment une information z_k sera utilisée, dans le robot R_s pour mettre à jour sa carte de l'environnement W^s pour l'instant k notée $W_k^s = (\underline{X}_k^s, P_k^s)$.

Chaque robot est supposé capable de se localiser et de localiser les autres robots du groupe dans un référentiel commun à tout le groupe. Avant de communiquer et d'échanger les données entre les robots, les états globaux W^n mis à jour par chaque membre sont pour le moment indépendants, ayant été mis à jour à partir des données venant des capteurs propres à chaque robot. C'est grâce à cette indépendance des états que notre approche solutionne le problème de consanguinité car la localisation collective est obtenue en fusionnant les informations indépendantes venant de tous les robots du groupe.

En effet, il est important de conserver une indépendance des données (prise en compte unique de la même information) pour ne pas engendrer une sur-convergence vers une valeur fautive de l'estimation de l'état du groupe.

Lorsque le robot R_s communique avec un autre robot R_i appartenant au groupe, ils échangent leurs états $W^s = (\underline{X}^s, P^s)$ et $W^i = (\underline{X}^i, P^i)$ pour la première fois et puisque ces derniers sont indépendants, la fusion des deux est obtenue par exemple pour le robot R_s par simple mise à jour de l'état $W^s = (\underline{X}^s, P^s)$ par $W^i = (\underline{X}^i, P^i)$ selon les équations suivantes :

$$K = P_k^s - (P_k^s + P_k^i)^{-1} \quad (3.25)$$

$$\underline{X}_k^s = \underline{X}_k^s + K(\underline{X}_k^i - \underline{X}_k^s) \quad (3.26)$$

$$P_k^s = (I - K)P_k^s \quad (3.27)$$

avec \underline{X}_k^s : l'estimation de l'état à l'instant k dans le robot R_s et P_k^s la matrice de covariance associée.

Dans le cas de N robots, cette étape de mise à jour est faite pour chaque robot du groupe. L'estimation globale $W_k^s = (\underline{X}_k^s, P_k^s)$ qui en résulte pour le robot considéré R_s , résume toutes les informations collectées par les membres du groupe. Après la fusion de données, l'état W_k^s dans le robot R_s devient dépendant des états W_k^i de tous les autres robots.

A l'instant $k + 1$ le robot R_s maintient à jour son état W_k^s qui devient W_{k+1}^s selon le formalisme décrit dans la section 3.1. De la même manière les autres robots R_i maintiennent à jour leurs états W_k^i qui deviennent W_{k+1}^i . L'état W_{k+1}^i de chaque robot R_i est dépendant de l'état précédent W_k^i .

Lorsque les membres du groupe communiquent à nouveau, le robot R_s reçoit tous les états W_{k+1}^i mis à jour par les autres robots. Il est censé alors mettre à jour son état W_{k+1}^s par les états W_{k+1}^i . Or comme nous l'avons vu, l'état W_{k+1}^s dépend de l'état W_k^s qui dépend des $(N - 1)$ états W_k^i . Comme les $(N - 1)$ états W_{k+1}^i reçus dépendent aussi des $(N - 1)$ états W_k^i , il devient impossible de fusionner à nouveau W_{k+1}^s avec les W_{k+1}^i reçus sans risquer une sur-convergence des estimations des poses des robots du groupe.

3.2.2 Notion de sous-états

Pour résoudre ce problème il faut s'assurer que les états fusionnés et échangés entre les robots sont indépendants les uns des autres. La solution que nous proposons consiste à considérer l'état global considéré comme la fusion d'un ensemble de N sous-états indépendants les uns des autres selon l'équation (3.28).

$$W^s = \mathcal{G}_1 (W_1^s, W_2^s, \dots, W_s^s, \dots, W_N^s) \quad (3.28)$$

Avec W_s^s l'état mis à jour par le robot R_s avec ses propres capteurs, $W_{i \dots N}^s$ (avec $i \neq s$) sont les états groupe mis à jour par les $N - 1$ autres robots avec leurs propres capteurs et reçus par le robot R_s . \mathcal{G}_1 est la fonction qui fusionne les états pour obtenir W^s qui est l'état global fusionné du groupe de robots.

Ainsi, les seules informations communiquées entre les différents du groupe sont les sous-états

3.2 Localisation collective

W_i^s qui sont obligatoirement indépendants. La fusion décentralisée dans chaque robot R_s appartenant au groupe est réalisée sur ces sous-états indépendants. Le résultat de cette fusion qui correspond à l'état global du groupe n'est jamais communiqué entre les robots. L'objectif est d'obtenir le même état général dans chaque robot sans jamais se le communiquer. Ainsi le problème de consanguinité est résolu par le fait qu'uniquement des états indépendants sont fusionnés et communiqués.

Algorithme 1 Fonction (\mathcal{G}_1) Fusion des sous-états

Entrées: $W_i^s = (\underline{X}_i^s, P_i^s)$

Sorties: W^s

$$P^s \leftarrow P_1^s$$

$$\underline{X}^s \leftarrow \underline{X}_1^s$$

pour $j = 2$ à N **faire**

$$K \leftarrow P^s (P^s + P_j^s)^{-1}$$

$$\underline{X}^s \leftarrow \underline{X}^s + K (\underline{X}_j^s - \underline{X}^s)$$

$$P^s \leftarrow P^s - K P_j^s$$

fin pour

retourner $W^s = (\underline{X}^s, P^s)$.

3.2.3 Fusion décentralisée

L'échange de données est illustré dans les figures 3.3 et 3.4 dans lesquelles le robot R_1 est représenté en noir, le robot R_2 en gris et le robot R_N en blanc.

Chaque robot R_s a un ensemble de sous-états W_i^s avec $i = 1 \dots N$. Au début de l'application (figure 3.3) et avant la communication des données, le robot R_1 maintient à jour son sous-état W_1^1 avec les informations de ses propres capteurs. Les autres sous-états $W_2^1 \dots W_N^1$ sont initialisés avec des valeurs arbitraires associées à une très grande covariance (les valeurs arbitraires sont prises généralement autour de la position courante du robot R_1). De la même manière les robots $R_2 \dots R_N$ maintiennent respectivement à jour leurs sous-états $W_2^2 \dots W_N^2$. Les autres sous-états sont initialisés avec des valeurs arbitraires avec une très grande covariance. Quand les membres du groupe communiquent, chaque robot transmet son sous-état propre W_s^s et reçoit ceux des autres. Le robot R_1 par exemple, envoie son sous-état W_1^1 et reçoit de la même manière les sous-états W_2^2, \dots, W_N^N . La figure 3.4 représente les sous-états dont

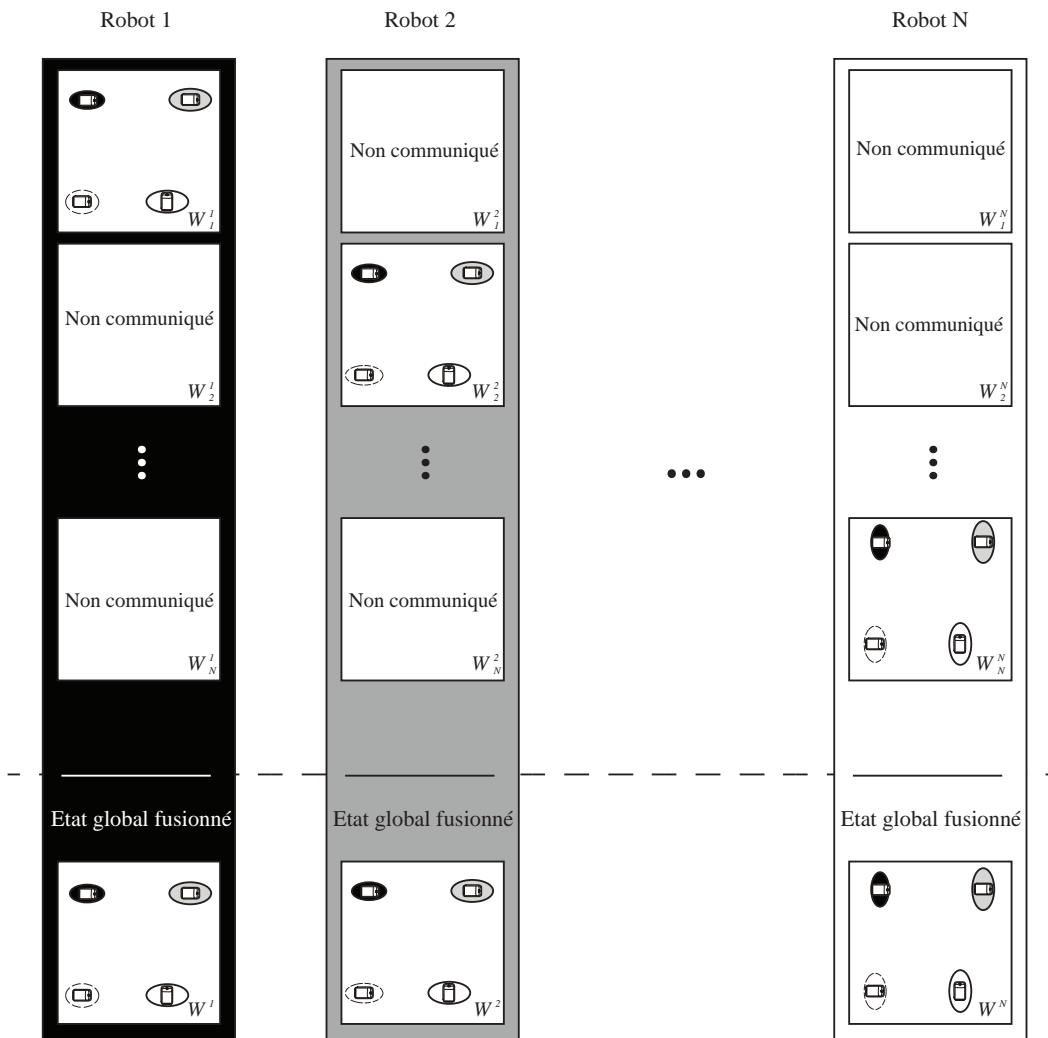


FIGURE 3.3 – Echange de données : Avant la communication

dispose chaque robot après l'échange de données.

Après la communication chaque robot R_s dispose des N états qui correspondent aux sous-états de chacun des membres du groupes. Pour obtenir l'état global du groupe W^s (l'état de la flotte vue par le robot R_s), chaque robot R_s doit fusionner les N sous-états W_1^s, \dots, W_N^s . Après la fusion l'état W^s combine de manière optimale les informations provenant de tous les robots du groupe en tenant compte des interdépendances entre les estimations des poses de chaque robot dans chaque sous-état.

3.2 Localisation collective

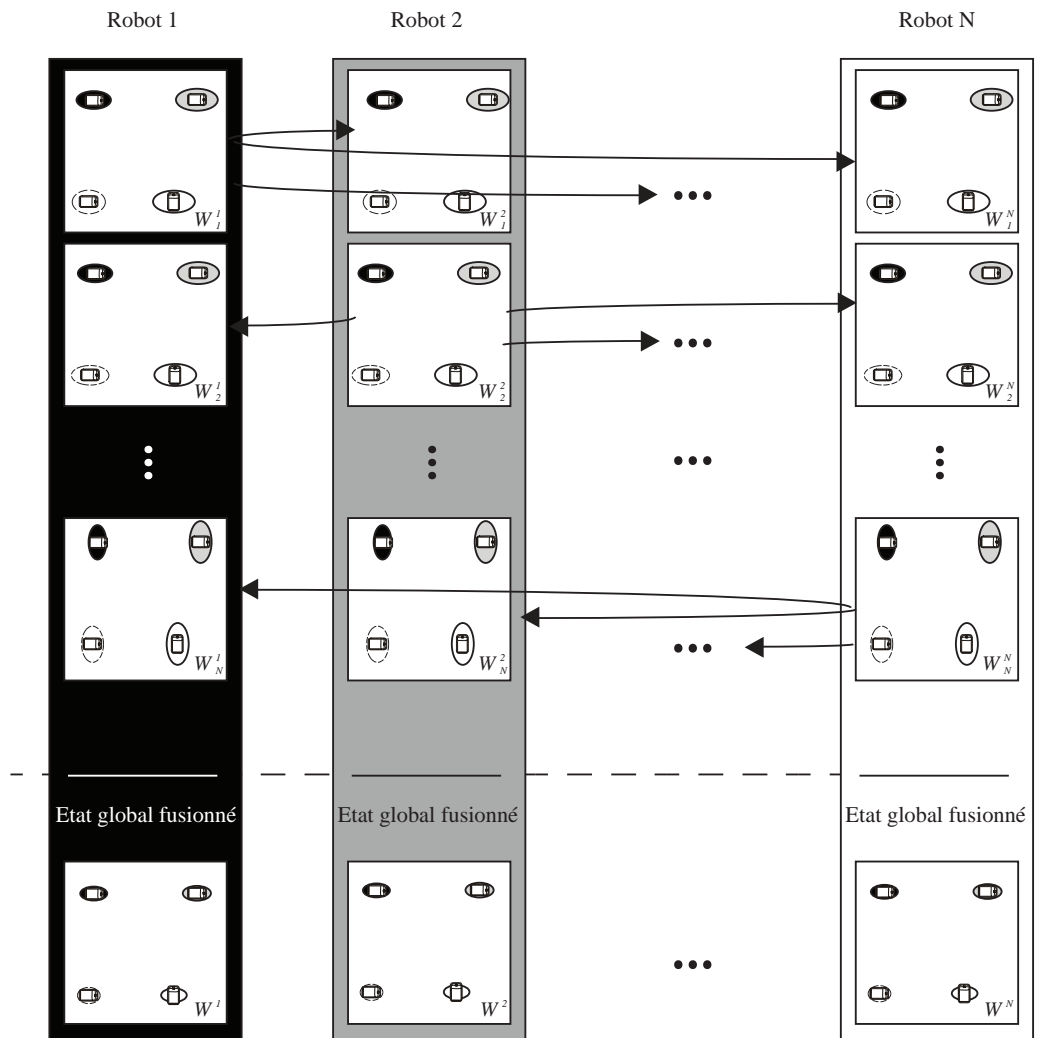


FIGURE 3.4 – Echange de données : après la communication

La fusion de tous les sous-états W_1^s, \dots, W_N^s peut être faite à tout instant par le robot R_s . Il est de ce fait capable d'obtenir la meilleure estimation possible de l'état global du groupe tenant compte des informations dont il dispose à tout instant. Il est important de noter que cet état fusionné n'est jamais communiqué entre les différents robots du groupe pour s'affranchir du problème de consanguinité.

Quand deux robots sont incapables de communiquer directement l'un avec l'autre, un troisième robot peut servir d'intermédiaire pour transmettre les sous-états de ces derniers. Par

exemple si le robot R_1 ne peut pas communiquer avec le robot R_3 , et s'il a déjà communiqué son sous-état W_1^1 au robot R_2 que celui-ci aura renommé W_1^2 , le robot R_2 peut alors transmettre au robot R_3 non seulement son sous-état W_2^2 mais aussi l'état W_1^2 qui correspond au sous-état du robot R_1 . Cet échange peut aussi être fait dans l'autre sens pour transmettre le sous-état du robot R_3 au robot R_1 . Ainsi, tous les robots disposent des mêmes informations sans pouvoir pourtant tous communiquer entre eux directement.

Un autre exemple est présenté sur la figure 3.5. Dans ce cas, nous considérons deux groupes de robots indépendants où chacun est constitué d'un certain nombre de robots. A l'intérieur de chaque groupe, les robots sont capables de se localiser et de communiquer entre eux et ainsi obtenir une localisation coopérative multi-véhicules optimale. Cependant, ces deux groupes étant trop éloignés l'un de l'autre, ils ne peuvent pas communiquer directement entre eux et donc partager leurs informations pour obtenir un état global de l'environnement.

Un robot positionné entre les deux groupes et à une distance autorisant une communication avec au moins un véhicule de chaque groupe peut servir d'intermédiaire. Ainsi, les états sont transmis entre les différents protagonistes par l'intermédiaire du robot central. Après la communication, la fusion des états indépendants dans chacun des robots permettra à l'ensemble de la flotte (les robots des deux groupes et le robot intermédiaire) de disposer d'une carte similaire de l'environnement.

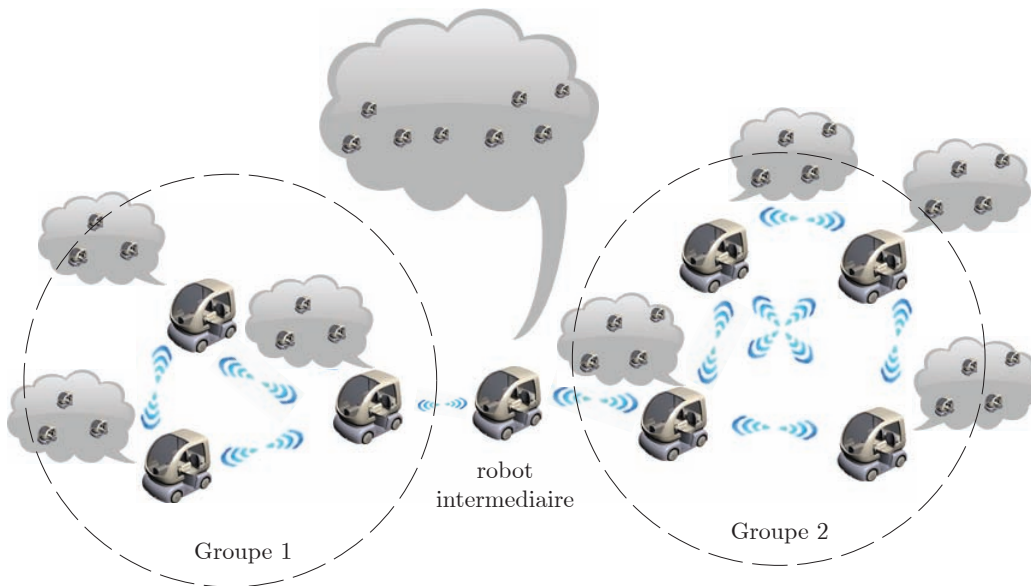


FIGURE 3.5 – Utilisation d'un robot intermédiaire pour communiquer les états

3.3 Conclusion

3.3 Conclusion

Dans ce chapitre, nous avons détaillé l'approche de localisation collective proposée dans un cas idéal. Les problèmes liés aux applications réelles sont ignorés afin de se focaliser sur l'approche de fusion de données.

Dans cette approche, chaque robot met à jour une version de l'état du groupe en utilisant uniquement les capteurs dont il dispose. Cet état est envoyé aux autres membres du groupe et reçoit de la même manière les états transmis par ces derniers. La fusion de l'état qu'il a mis à jour avec ceux reçus des autres robots permet d'obtenir un état global fusionné qui contient des informations de localisation plus précises et plus complètes.

L'idée clé de cette approche est que l'état global obtenu après la fusion des états reçus et l'état mis à jour en local n'est jamais échangé avec les autres membres. Chaque robot peut utiliser cet état pour accomplir la tâche qui lui est confiée mais ne le retransmet jamais aux autres robots. L'état échangé est celui mis à jour avec les capteurs propres du robot.

Cette stratégie permet de garantir l'indépendance des états échangés entre les robots et permet de les fusionner sans problème de sur-convergence.

Chapitre 4

Localisation collective : cas réaliste

Sommaire

4.1 Ordonnancement des observations	70
4.1.1 Désynchronisation	70
4.1.2 Latences	71
4.2 Localisation collective	72
4.2.1 Estimation de l'état du groupe	72
4.2.2 Fusion des états de l'environnement	78
4.3 Conclusion	81

Dans ce chapitre nous allons montrer les adaptations de l'approche de localisation coopérative présentée dans le chapitre 3 de la partie II pour faire face aux différents problèmes rencontrés dans le cadre d'une application réelle. En effet, pour le moment l'approche présentée résout uniquement le problème principal de la consanguinité grâce à l'échange de sous-états indépendants et qui sont ensuite fusionnés dans chacun des robots du groupe.

Ainsi, afin de généraliser l'approche présentée dans le chapitre 3 nous allons relâcher les contraintes les plus restrictives. Dans ce chapitre, nous considérons un groupe hétérogène de N robots avec les spécifications suivantes :

- Les robots du groupe sont capables de se positionner dans un référentiel absolu et commun à tous.
- Les robots sont potentiellement capables de localiser d'autres robots dans leur entourage par rapport à leur position.
- Les robots ne sont pas capables d'identifier les autres membres qu'ils détectent.
- Les robots du groupe sont équipés d'un système de communication sans fil.

- Le nombre total de robots du groupe n'est pas connu a priori. Des robots peuvent s'insérer dans le groupe ou en sortir à tout moment.
- L'environnement où évoluent les robots n'est pas fermé. Des véhicules qui n'appartiennent pas au groupe peuvent être présents.
- Tout robot communiquant peut être considéré comme membre du groupe.

La section 4.1 présente la stratégie adoptée pour solutionner le problème de désynchronisations et de latences des données reçues et la section 4.2 présente l'extension de l'approche de localisation proposée afin qu'elle soit applicable dans un cas réel.

4.1 Ordonnancement des observations

4.1.1 Désynchronisation

Dans une application réelle, chaque capteur délivre des informations à une fréquence différente. Il est facile de comprendre que les méthodes de localisation collective basées sur un échange de données (voir 2.3.1) sont très difficiles à adapter à ce problème car il serait nécessaire de communiquer en quasi-permanence des données entre les différents robots du groupe. Dans l'approche proposée ces informations sont traitées quand elles arrivent mais jamais communiquées entre les robots. Les informations provenant des capteurs proprioceptifs et extéroceptifs sont considérées comme des informations qui vont changer l'état courant. S'il s'agit d'une information d'un capteur proprioceptif elle sera utilisée pour l'évolution de l'état actuel. Par contre, s'il s'agit d'une information fournie par un capteur extéroceptif, l'état actuel est extrapolé jusqu'à la date correspondant à cette information extéroceptive. L'extrapolation est réalisée avec un modèle d'évolution qui n'est pas fonction des données proprioceptives. L'état est alors mis à jour avec la donnée extéroceptive.

Ce problème est traité en détail dans [Lan06b].

La figure 4.1 présente un exemple du traitement de deux informations odométriques et une observation GPS.

4.1 Ordonnement des observations

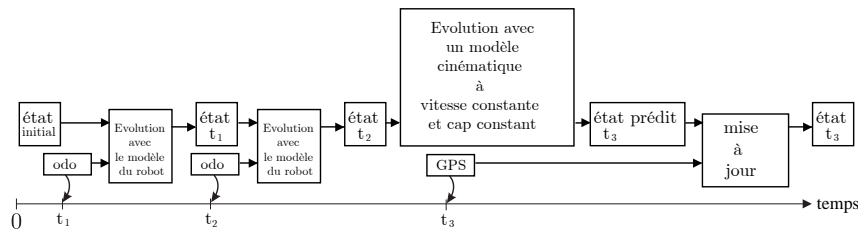


FIGURE 4.1 – Ordonnement des observations : Désynchronisation

4.1.2 Latences

Les observations peuvent être affectées de retard. Par exemple, les informations relatives données par le traitement d'une image caméra ou d'une nappe télémètre ne sont utilisables qu'après l'exécution de l'algorithme de traitement. L'information utile disponible à l'instant t correspond en réalité à l'instant $t - \Delta_t$, où Δ_t est le temps que prend l'algorithme de traitement pour transformer la donnée brute en information utile.

Ce problème sera résolu par la mise en place d'une procédure de datation des observations par rapport à un référentiel absolu. De cette façon les informations utiles sont affectées à la date réelle de leur arrivée et replacées correctement dans l'ordre chronologique.

Dans la figure 4.2 par exemple, l'observation obs_4 qui arrive, date de l'instant t_4 qui est anté-

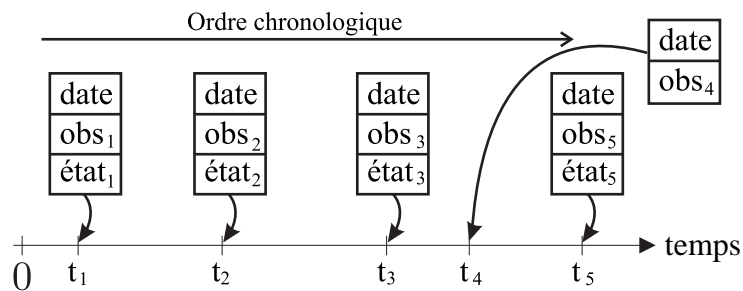


FIGURE 4.2 – Ordonnement des observations : arrivée d'une observation

rieur à t_5 , la date de l'état actuel. Avant l'insertion de l'observation, l'état $état_5$ était le résultat de la mise à jour de l'état $état_3$ à l'instant t_3 et l'observation obs_5 . A l'arrivée de la nouvelle observation obs_4 , elle est replacée dans l'ordre chronologique et le nouvel $état_5$ est recalculé. Dans ce cas, l'état actuel est recalculé depuis l'instant t_3 et sera le résultat de la mise à jour de l'état $état_3$ avec les observations obs_4 et obs_5 consécutivement tel que montré dans la figure 4.3. L'état $état_4$ sera associé à l'observation obs_4 et enregistré pour une éventuelle observation qui

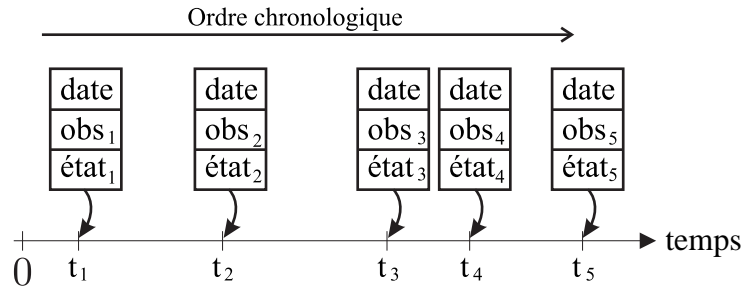


FIGURE 4.3 – Ordonnancement des observations : insertion de l’observation

viendrait s’insérer entre les observations obs_4 et obs_5 .
Ce problème est traité en détail dans [Tes07].

4.2 Localisation collective

Le but recherché reste le même que dans le cas idéal présenté dans le chapitre 3. Il s’agit d’obtenir l’estimation la plus précise et la plus complète possible de l’état des robots du groupe en combinant les informations collectées par chaque robot équipé d’un système de communication. L’approche proposée doit tenir compte de la désynchronisation des données, des délais et des pertes de communication.

Le principe utilisé reste également le même, chaque robot du groupe met à jour son état de l’environnement avec ses propres capteurs et lorsque la communication devient possible, il envoie ses données et reçoit celles collectées par les autres robots. L’état global de l’environnement est obtenu en fusionnant toutes ces informations.

4.2.1 Estimation de l’état du groupe

Chaque véhicule estime l’état de l’environnement qui l’entoure avec ses propres capteurs. Dans cette section nous allons décrire la procédure qui s’exécute dans l’un des robots du groupe que nous appellerons R_s (s pour self (soit)). La même procédure est exécutée dans les autres robots R_{o_i} du groupe (o pour others (autres)), avec $i = 1..M$ et où $M = N - 1$ est le nombre total des autres robots.

4.2 Localisation collective

L'état du robot R_s est représenté par son vecteur d'état $\underline{x}_s = [x_s, y_s, \varphi_s, v_s]^T$ et sa matrice de covariance P_s , où x_s et y_s sont les coordonnées du robot R_s dans un référentiel absolu et commun à tout le groupe, φ_s est son orientation et v_s est le module de sa vitesse.

L'état de chacun des autres robots est représenté par $\underline{x}_{o_i} = [x_{o_i}, y_{o_i}, \varphi_{o_i}, v_{o_i}]^T$ et sa matrice de covariance P_{o_i} , où $i=1\dots M$, x_{o_i} et y_{o_i} sont les coordonnées du robot R_{o_i} dans le même référentiel que R_s , φ_{o_i} son orientation et v_{o_i} le module de sa vitesse.

Le principe de la localisation collective reste le même que dans le cas idéal présenté dans le chapitre 3, le groupe est représenté par un seul système $W_s = (\underline{X}_s, P_s, \mathbf{id}_s)$, \mathbf{id}_s est l'identifiant du robot R_s , \underline{X}_s son vecteur d'état et P_s sa matrice de covariance comme dans les équations (4.1) et (4.2).

$$\underline{X}_s = [\underline{x}_s^T, \underline{x}_{o_1}^T, \underline{x}_{o_2}^T, \dots, \underline{x}_{o_{m_s}}^T]^T \quad (4.1)$$

$$P_s = \begin{bmatrix} P_{s,s} & P_{s,o_1} & \dots & P_{s,o_{m_s}} \\ P_{o_1,s} & P_{o_1,o_1} & \dots & P_{o_1,o_{m_s}} \\ \dots & \dots & \dots & \dots \\ P_{o_{m_s},s} & P_{o_{m_s},o_1} & \dots & P_{o_{m_s},o_{m_s}} \end{bmatrix} \quad (4.2)$$

Où m_s est le nombre de robots détectés par le robot R_s (avec $m_s < (M = N - 1)$). Initialement, le groupe est composé du seul robot R_s et sa taille augmente par la suite au fur et à mesure de l'arrivée des états fournis par les autres robots. Notons que l'état \underline{x}_s^T est toujours à la première position de l'état \underline{X}_s .

Comme dans le cas idéal présenté auparavant, le robot R_s tient à jour une estimation de l'environnement W_s en fusionnant les données de ses capteurs proprioceptifs et extéroceptifs à l'aide d'un Filtre de Kalman Etendu (EKF).

4.2.1.1 Evolution de l'état

Le déplacement des robots peut être modélisé par la fonction f_s (équation (4.3)) pour le robot R_s et la fonction f_o (équation (4.4)) pour les autres robots R_{o_i} . Ces deux modèles d'évolution sont équivalents à ceux vus dans le paragraphe 3.1.1. Ici, l'ensemble des robots constituant le groupe sont mis à jour suivant ces modèles d'évolution.

$$\underline{x}_{s_{k+1}}^- = f_s(\underline{x}_{s_k}, \underline{u}_{s_k}^m) \quad (4.3)$$

$$\underline{x}_{o_{i_{k+1}}}^- = f_o(\underline{x}_{o_{i_k}}) \quad (4.4)$$

On suppose ici, que les données proprioceptives \underline{u}_s^m du robot R_s sont affectées par un bruit qui peut être modélisé par

$$\underline{u}_s^s \sim \mathcal{N}(\underline{u}_s, Q_s) \quad (4.5)$$

Où \underline{u}_s^m est la donnée mesurée et Q_s est la covariance du bruit qui l'affecte.

$$P_{s,s_{k+1}}^- = F_{s(\underline{x}_s)} P_{s,s_k} F_{s(\underline{x}_s)}^T + F_{s(\underline{u}_s^m)} Q_{s_k} F_{s(\underline{u}_s^m)}^T + B_s \quad (4.6)$$

$$P_{o_i, o_{i,k+1}}^- = F_o P_{o_i, o_{i,k}} F_o^T + B_o \quad (4.7)$$

k représente le temps, $\underline{x}_{s_{k+1}}^-$ et $\underline{x}_{o_{i,k+1}}^-$ sont respectivement les vecteurs d'état \underline{x}_{s_k} et $\underline{x}_{o_{i,k}}$ prédits à l'instant $k+1$, $F_{s(\underline{x}_s)}$ et $F_{s(\underline{u}_s^m)}$ sont respectivement les jacobiennes des fonctions f_s par rapport à l'état \underline{x}_s et \underline{u}_s , F_o est la jacobienne de la fonction f_o par rapport à l'état \underline{x}_{o_i} . B_s et B_o sont respectivement le bruit affectant le modèle d'évolution du robot R_s et celui qui affecte le modèle d'évolution des autres robots R_{o_i} .

L'état obtenu est la prédiction de l'état de l'environnement à l'instant $k+1$ utilisant l'état à l'instant k et les informations de capteurs proprioceptifs du robot R_s . Cet état va être mis à jour avec d'éventuelles informations extéroceptives. Ces informations peuvent concerner le robot R_s lui-même tel qu'une position GPS ou concerner son environnement tel que la détection d'un autre robot ou encore d'un obstacle extérieur au groupe.

4.2.1.2 Mise à jour de l'état du groupe

Au début de l'application, l'état W_s ne contient que l'estimation de la pose du robot R_s (\underline{x}_s et sa matrice de covariance $P_{s,s}$) comme dans l'équation (4.8). Aucun autre véhicule n'y figure ($m_s=0$).

$$\underline{X}_s = [\underline{x}_s], \quad P_s = [P_{s,s}] \quad (4.8)$$

Quand le robot R_s détecte un autre robot R_{o_1} , il mesure sa pose relative \underline{z}_r qui est supposée être bruitée par le modèle :

$$\underline{z}_r^m \sim \mathcal{N}(\underline{z}_r, B_z) \quad (4.9)$$

avec $\underline{z}_r^m = (\Delta x, \Delta y, \Delta \varphi)$ la pose relative mesurée et B_z la covariance du bruit qui l'affecte.

Comme l'état de l'environnement dans le robot R_s ne contient que sa propre pose, l'état du robot détecté peut être ajouté à l'état de l'environnement sans aucune ambiguïté. Pour garder

4.2 Localisation collective

les interdépendances entre les estimations des poses, l'état de l'environnement est mis à jour comme suit :

- L'état W_s est étendu avec une estimation de pose $(\underline{x}_0, P_{0,0})$ dont le vecteur d'état est initialisé avec une position arbitraire autour de la position du robot R_s associée à une grande covariance (4.10).

$$\underline{X}_{s_{k+1}}^- = \begin{bmatrix} \underline{X}_{s_{k+1}}^- \\ \underline{x}_0 \end{bmatrix}, \quad P_{s_{k+1}}^- = \begin{bmatrix} P_{s_{k+1}}^- & 0_{4 \times 4} \\ 0_{4 \times 4} & P_{0,0} \end{bmatrix} \quad (4.10)$$

où $0_{n \times n}$ est une matrice de zéros de dimension $n \times n$.

- L'état étendu est mis à jour avec la mesure $z_{r_{k+1}}^m$ selon les équations (4.11), (4.12), (4.13), (4.14) et (4.15).

$$z_{k+1}^- = z_{r_{k+1}}^m - H_{k+1} \underline{X}_{s_{k+1}}^- \quad (4.11)$$

$$S_{k+1} = H_{k+1} P_{s_{k+1}}^- H_{k+1}^T + B_z \quad (4.12)$$

$$K_{k+1} = P_{s_{k+1}}^- H_{k+1}^T S_{k+1}^{-1} \quad (4.13)$$

$$\underline{X}_{s_{k+1}} = \underline{X}_{s_{k+1}}^- + K_{k+1} z_{k+1}^- \quad (4.14)$$

$$P_{s_{k+1}} = (I - K_{k+1} H_{k+1}) P_{s_{k+1}}^- \quad (4.15)$$

La matrice de mise à jour H est celle de l'équation (4.16).

$$H = \begin{bmatrix} -I_{3 \times 4} & I_{3 \times 4} \end{bmatrix} \quad (4.16)$$

où $I_{3 \times 4} = \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 1} \end{bmatrix}$ et $I_{n \times n}$ est une matrice identité de $n \times n$ dimensions.

Après la mise à jour, l'état de l'environnement contient l'estimation de la pose du robot R_s et celle du robot détecté R_{o_1} , le nombre des autres robots devient alors $m_s=1$.

4.2.1.3 Principe de la mise à jour

Dorénavant quand le robot R_s détectera un autre robot R_o , comme il ne peut pas l'identifier en utilisant ses capteurs, il doit comparer sa pose avec les poses des m_s autres robots déjà présents dans son état de l'environnement W_s . La pose du robot détecté est estimée avec la pose actuelle du robot R_s et la pose relative $z_r^m = (\Delta x, \Delta y, \Delta \varphi)$ comme dans l'équation (4.17).

$$\begin{aligned} \tilde{\underline{x}}_o &= (x_s + \Delta x, y_s + \Delta y, \varphi_s + \Delta \varphi) \\ \tilde{P}_{o,o} &= \tilde{P}_{s,s} + B_z \end{aligned} \quad (4.17)$$

où \tilde{x}_o et $\tilde{P}_{o,o}$ sont la pose estimée du robot détecté et sa matrice de covariance, $\tilde{P}_{s,s}$ est la matrice de covariance associée à $\tilde{x}_s = [x_s, y_s, \varphi_s]^T$.

Comme le robot R_s est supposé incapable de mesurer la vitesse relative du robot R_o , la pose $(\tilde{x}_o, \tilde{P}_{o,o})$ est comparée avec $(\tilde{x}_{o_{i_{k+1}}}^-, \tilde{P}_{o_i, o_{i_{k+1}}}^-)$ dans l'état $(\underline{X}_{s_{k+1}}^-, P_{s_{k+1}}^-)$, où $\tilde{x}_{o_i}^-$ et \tilde{P}_{o_i, o_i}^- représentent respectivement $\underline{x}_{o_i}^-$ et P_{o_i, o_i}^- sans la composante de la vitesse v_{o_i} . Cette comparaison est effectuée en calculant la distance de Mahalanobis entre les deux estimations de poses comme dans l'équation (4.18)

$$d_i^2 = (\tilde{x}_o - \tilde{x}_{o_{i_{k+1}}}^-)^t (\tilde{P}_{o,o} \tilde{P}_{o_i, o_{i_{k+1}}}^-)^{-1} (\tilde{x}_o - \tilde{x}_{o_{i_{k+1}}}^-) \quad (4.18)$$

où $i=1 \dots m_s$. Le processus de comparaison génère un ensemble de m_s distances de Mahalanobis qui correspondent aux m_s autres robots présents dans l'état W_s .

- Si la distance minimale est inférieure au seuil défini avec la loi du χ^2 , cela signifie que le robot détecté existe déjà dans l'état W_s , la mesure z_r^m est alors utilisée pour mettre à jour la pose du robot correspondant selon les équations (4.11), (4.12), (4.13), (4.14) et (4.15) avec la matrice de mise à jour H définie dans l'équation (4.19).

$$H = \begin{bmatrix} -I_{3 \times 4} & 0_{3 \times (4(D-1))} & I_{3 \times 4} & 0_{3 \times (4(m_s-D))} \end{bmatrix} \quad (4.19)$$

Où $1 \leq D \leq m_s$ est la position du robot qui correspond à la distance de Mahalanobis minimale dans le vecteur de l'état de l'environnement W_s . Dans ce cas le nombre m_s des autres robots dans l'état W_s ne change pas.

- Si la distance minimale est supérieure au seuil, cela signifie que le robot détecté n'est pas dans l'état W_s , il est alors ajouté selon l'équation (4.20) et mis à jour selon les équations (4.11), (4.12), (4.13), (4.14) et (4.15), avec la matrice de mise à jour H définie dans l'équation (4.21).

$$\underline{X}_{s_{k+1}}^- = \begin{bmatrix} \underline{X}_{s_{k+1}}^- \\ \underline{x}_o \end{bmatrix}, P_{s_{k+1}}^- = \begin{bmatrix} P_{s_{k+1}}^- & 0_{4(m_s+1) \times 4} \\ 0_{4 \times 4(m_s+1)} & P_{0,0} \end{bmatrix} \quad (4.20)$$

$$H = \begin{bmatrix} -I_{3 \times 4} & 0_{3 \times (4m_s)} & I_{3 \times 4} \end{bmatrix} \quad (4.21)$$

Dans ce cas le nombre m_s des autres robots dans l'état est incrémenté ($m_s = m_s + 1$).

4.2.1.4 “Nettoyage” de l'état du groupe

Dans les sections précédentes nous avons montré le processus de détection, d'ajout et de mise à jour des autres robots dans l'estimation de l'état de l'environnement. Le robot R_s maintient ses robots dans son état W_s tant qu'il arrive à les détecter et à mettre à jour leurs poses.

4.2 Localisation collective

Dans le cas contraire l'erreur affectée à leur pose devient trop grande et peut alors mener à des erreurs d'association (la distance de Mahalanobis entre la pose d'un robot détecté et la pose d'un robot dont l'erreur est très grande est souvent inférieure au seuil d'association).

Pour résoudre ce problème une étape de suppression de robots de l'état W_s est effectuée. Après la mise à jour de l'état W_s , les robots dont la pose présente une erreur supérieure à une valeur maximale autorisée sont supprimés.

L'équation 4.22 présente un exemple où l'état \underline{x}_{o_n} du robot R_{o_n} est sélectionné pour la suppression car l'erreur sur sa pose décrite dans la matrice P_{o_n, o_n} est plus grande que l'erreur maximale autorisée. Dans ce cas tous les éléments correspondants à ce robot (encadrés dans l'équation 4.22) sont supprimés.

$$\underline{X}_{s_k} = \begin{bmatrix} \underline{x}_s \\ \underline{x}_{o_1} \\ \dots \\ \dots \\ \underline{x}_{o_{m_s}} \end{bmatrix}, \quad P_{s_k} = \begin{bmatrix} P_{s,s} & P_{s,o_1} & \dots & \dots & P_{s,o_{m_s}} \\ P_{o_1,s} & P_{o_1,o_1} & \dots & \dots & P_{o_1,o_{m_s}} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ P_{o_{m_s},s} & P_{o_{m_s},o_1} & \dots & \dots & P_{o_{m_s},o_{m_s}} \end{bmatrix} \quad (4.22)$$

La suppression d'un robot de l'état W_s arrive typiquement dans les cas suivants :

- Le robot R_{o_n} sort du champ d'action des capteurs utilisés pour la détection, sa pose n'est pas mise à jour et son erreur associée grandit jusqu'à ce qu'il soit supprimé de W_s .
- Après une mauvaise association, un robot peut être ajouté par erreur dans W_s alors qu'il était déjà présent. Dans ce cas, l'une des deux versions n'est plus mise à jour, son erreur augmente jusqu'à ce qu'elle soit supprimée.
- Après une fausse détection (fausse positive), un robot peut être ajouté par erreur alors qu'il n'existe pas. La pose ajoutée ne sera plus mise à jour et sera supprimée après quelques itérations.

Les processus *Evolution de l'état de l'environnement*, *Mise à jour de l'état de l'environnement* *Nettoyage de l'état de l'environnement* permettent au robot R_s de maintenir un état qui décrit sa vue de l'environnement. Ces mêmes processus sont exécutés dans les autres robots du groupe, et chaque membre peut aussi construire une estimation de l'état de cet environnement. En fusionnant toutes ces estimations, on obtient une version beaucoup plus complète et précise que celles estimées dans chaque robot.

4.2.2 Fusion des états de l'environnement

Quand la communication devient possible, le robot R_s envoie l'état de l'environnement qu'il a tenu à jour avec ses capteurs, et reçoit ceux des robots qui sont dans la zone de communication. Il peut alors procéder à la fusion de son état de l'environnement avec ceux qu'il a reçus.

Nous pouvons considérer que le robot R_s met à jour un état $W_s=(\underline{X}_s, P_s, \mathbf{id}_s)$ et reçoit L états $W_l=(\underline{X}_l, P_l, \mathbf{id}_l)$ avec $l=1\dots L$ des autres robots présents dans la zone de communication. L'état global de l'environnement est obtenu en fusionnant W_s avec les L états W_l reçus, selon l'équation (4.23)

$$W^s = \mathcal{G}_2 (W_s, W_1, W_2, \dots, W_l, \dots, W_L) \quad (4.23)$$

On note W^s l'état global fusionné de l'environnement dans le robot R_s . Cet état a la forme décrite dans l'équation (4.24), W_l est l'état de l'environnement reçu de l'autre robot $R_{\mathbf{id}_l}$.

$$W^s = (\underline{X}^s, P^s, \mathbf{id}^s) \quad (4.24)$$

où \underline{X}^s est le vecteur de l'état global de l'environnement, P^s sa matrice de covariance et \mathbf{id}^s est un vecteur de dimension $m_s + 1$ qui contient les identifiants des robots présents dans l'état global fusionné.

La fusion des états de l'environnement est obtenue en mettant à jour l'état W_s avec les états W_l reçus selon l'algorithme 2.

Algorithme 2 Fonction (\mathcal{G}_2) Fusion des états de l'environnement

Entrées: $(W_s, W_1, W_2, \dots, W_L)$

Sorties: W^s

W^s initialisation

pour $l = 1 \dots L$ **faire**

$W_l = \text{EVOL}(W_l, \Delta t)$

$W^s = \text{FUSE}(W^s, W_l)$

fin pour

return W^s .

où W^s est initialisé avec l'état W_s selon l'équation (4.25).

$$\underline{X}^s = \underline{X}_s, \quad W^s = W_s, \quad \mathbf{id}^s = [\mathbf{id}_s, 0, \dots, 0]^T \quad \text{et} \quad m^s = m_s \quad (4.25)$$

Dans le vecteur \mathbf{id}^s les éléments nuls veulent dire que le robot associé n'a pas encore été identifié. La fonction *EVOL* permet de faire évoluer l'état W_l à la date de W^s . L'état W_l reçu par

4.2 Localisation collective

la communication peut avoir une date antérieure à celle de W^s . La fonction *FUSE* permet de mettre à jour l'état $W^s=(\underline{X}^s, P^s, \mathbf{id}^s)$ avec un autre état $W_l=(\underline{X}_l, P_l, \mathbf{id}_l)$ reçu du robot $R_{\mathbf{id}_l}$. Cette fonction est détaillée ci-dessous.

Considérons que les états W^s et W_l contiennent respectivement $M^s=(m^s + 1)$ et $M_l=(m_l + 1)$ robots (une position propre au robot et respectivement m^s et m_l estimations de poses d'autres robots). Le nombre de robots et leurs positions dans le vecteur dans W^s et W_l sont différents. Pour fusionner les deux états correctement, une étape d'association de données est indispensable pour trouver les correspondances entre les différentes estimations. Une fois les correspondances trouvées, il devient possible de fusionner les deux états en une seule étape afin de garder les interdépendances entre les estimations des poses des différents robots.

Le processus de fusion est divisé en deux étapes :

- *Association de données* : Le robot R_s parcourt les deux états W^s et W_l à la recherche de correspondances entre les poses des robots présents. Ceci est fait en calculant la distance de Mahalanobis entre les estimations de poses dans les deux états à fusionner et en la comparant au seuil défini par la loi du χ^2 . Si cette distance est inférieure au seuil, les deux véhicules en question correspondent. La pose du robot présent dans W^s sera mise à jour avec celle du robot présent dans W_l . Bien sûr, le robot R_s ne compare pas sa pose et celle du robot $R_{\mathbf{id}_l}$.
- *Mise à jour de l'état W^s avec W_l* : Le processus d'association de données fournit une *table de correspondances* entre les robots présents dans les deux états. Dans cette table, deux cas sont possibles :
 - La pose $(\underline{x}_{o_b}, P_{o_b, o_b})$ dans l'état W_l correspond à la pose $(\underline{x}_{o_a}, P_{o_a, o_a})$ dans l'état W^s . Cela veut dire que le robot correspondant a été détecté par les deux robots R_s et $R_{\mathbf{id}_l}$, la pose $(\underline{x}_{o_a}, P_{o_a, o_a})$ sera alors mise à jour avec $(\underline{x}_{o_b}, P_{o_b, o_b})$.
 - La pose $(\underline{x}_{o_b}, P_{o_b, o_b})$ dans W_l ne correspond à aucune autre pose dans W^s . Cela veut dire que le robot détecté par $R_{\mathbf{id}_l}$ n'a pas été détecté par R_s ou a été supprimé de W^s . Dans ce cas pour chaque robot non associé, un nouveau robot sera ajouté à l'état W^s en étendant ce dernier avec une pose $(\underline{x}_0, P_{0,0})$ initialisé avec une des valeurs arbitraires et une très grande covariance selon l'équation (4.26). Le robot non associé sera affecté au robot ajouté dans W^s .

$$\underline{X}^s = \begin{bmatrix} \underline{X}^s \\ \underline{x}_0 \end{bmatrix}, \quad P^s = \begin{bmatrix} P^s & 0_{N_s \times 4} \\ 0_{4 \times N_s} & P_{0,0} \end{bmatrix} \quad (4.26)$$

$$M^s = M^s + 1$$

Après l'extension, l'état global de l'environnement contient M^s robots. La matrice de mise à jour H peut être construite en utilisant la table de correspondances et sera de dimension $4M_l \times 4M^s$ avec des $I_{4 \times 4}$ là où il le faut pour mettre à jour les robots dans W^s avec les robots correspondants dans W_l . L'état W^s est mis à jour selon les équations (4.27), (4.28) et (4.29).

$$K = P^s H^T (H P^s H^T + P_l)^{-1} \quad (4.27)$$

$$\underline{X}^s = \underline{X}^s + K(\underline{X}_l - H \underline{X}^s) \quad (4.28)$$

$$P^s = (I - K)P^s \quad (4.29)$$

Et enfin

$$\underline{\mathbf{id}}^s(D) = \mathbf{id}_l \quad (4.30)$$

où D est la position de la pose du robot $R_{\mathbf{id}_l}$ dans l'état global de l'environnement W^s . L'état global de l'environnement $W^s = (\underline{X}^s, P^s, \underline{\mathbf{id}}^s)$ retourné par la fonction de fusion a la forme décrite dans les équations (4.31) et (4.32).

$$\underline{X}^s = [\underline{x}^{sT}, \underline{x}^{o_1T} \dots \underline{x}^{o_{N_s}T}]^T, \quad \underline{\mathbf{id}}^s = [\mathbf{id}_s, \mathbf{id}_{o_1} \dots \mathbf{id}_{o_{N_s}}]^T \quad (4.31)$$

$$P^s = \begin{bmatrix} P_{s,s}^s & P_{s,o_1}^s & \dots & P_{s,o_{N_s}}^s \\ P_{o_1,s}^s & P_{o_1,o_1}^s & \dots & P_{o_1,o_{N_s}}^s \\ \dots & \dots & \dots & \dots \\ P_{o_{N_s},s}^s & P_{o_{N_s},o_1}^s & \dots & P_{o_{N_s},o_{N_s}}^s \end{bmatrix} \quad (4.32)$$

\underline{X}^s est le vecteur de l'état global de l'environnement et P^s sa matrice de covariance. Le vecteur $\underline{\mathbf{id}}^s$ contient les identifiants des robots dans l'état global W^s . Les robots non identifiés sont représentés par un identifiant nul. Un robot peut être détecté et localisé par plusieurs robots du groupe et rester non identifié dans deux cas :

- Le robot appartient au groupe mais n'a pas encore échangé son état de l'environnement avec aucun de ses congénères. Ce robot sera identifié dès qu'il communiquera avec les autres.
- Le robot non identifié n'appartient pas au groupe ou ne dispose pas de système de communication.

4.3 Conclusion

4.3 Conclusion

Dans cette section nous avons détaillé l'approche de localisation collective par échange d'état modifiée pour l'application dans un cas réaliste. Dans ce cas, les problèmes liés aux applications réelles ignorés dans le chapitre 3 ont été pris en compte. Le principe de base reste le même mais la prise en compte des contraintes imposées a nécessité l'élaboration de nouvelles stratégies pour la construction et la fusion des états du groupe.

Comme le nombre de robots constituant le groupe est inconnu, chaque robot commence l'application avec l'état d'un groupe dont il est le seul membre. La taille de son groupe augmente à mesure que ce robot détecte d'autres membres. L'identification des mobiles détectés étant impossible, à chaque détection d'un congénère, il entame une étape de mise en correspondance afin de voir si le robot détecté est déjà présent dans le groupe ou s'il s'agit d'un nouvel arrivant. Dans le premier cas l'état du groupe est mis à jour avec cette nouvelle pose, et dans le second l'état est étendu pour ajouter le nouveau robot.

Après l'échange de données avec les congénères, chaque robot fusionne l'état estimé en local et ceux reçus des autres membres. La taille des états reçus dépend du nombre de robots détectés par les congénères, elle n'est de ce fait, pas nécessairement la même que celle de l'état local. La fusion des états reçus avec l'état local nécessite, elle aussi une étape de mise en correspondance. Cette étape permet de trouver les correspondances entre les robots présents dans l'état local et les robots présents dans les états reçus des autres membres afin de fusionner les estimations correctement.

L'état obtenu après la fusion est le résultat de la localisation coopérative et contient la combinaison des données provenant de tous les membres du groupe. Cet état sera utilisé localement pour d'autres applications éventuelles, mais ne sera jamais communiqué à nouveau.

Comme le nombre de robots dans le groupe est dynamique, il est a été nécessaire d'élaborer des stratégies pour traiter l'insertion et le retrait d'un robot dans le groupe.

L'insertion de nouveaux robots est traitée par l'extension de la taille de l'état afin d'accueillir le nouvel arrivant. Et la sortie est traitée par l'étape dite de "nettoyage" du groupe. Cette étape permet de supprimer les robots dont la pose n'a pas été mise à jour depuis longtemps en recherchant les robots dont l'erreur de localisation est plus grande qu'un seuil défini.

L'approche proposée a été implémentée et testée en simulation et dans un cadre réel. Les résultats obtenus sont présentés dans la partie suivante.

Troisième partie

Mise en œuvre et résultats

Dans cette partie, nous présentons la mise en œuvre du système de localisation collective par fusion d'état.

Dans le premier chapitre de cette partie nous présentons les résultats de l'approche proposée dans le cadre d'une simulation réaliste. Pour cette étude expérimentale nous simulons un groupe de quatre robots. Différentes contraintes liées aux applications réelles ont été simulées telles que les erreurs et les limites opérationnelles des capteurs, la désynchronisation des données, les latences et les retards liés aux traitements des données et aux communications.

Dans le deuxième chapitre nous présentons les résultats dans le cadre d'une application réelle avec un groupe de deux véhicules expérimentaux. L'expérimentation est faite dans la configuration d'un convoi. Pour montrer la propagation de l'erreur de localisation entre les deux robots, nous avons équipé le véhicule de queue d'un GPS centimétrique et celui de tête d'un GPS bas coût.

Chapitre 5

Simulation de l'approche de localisation collective

Sommaire

5.1	Description des simulations	87
5.2	Génération des données de simulation	88
5.3	Scénario de l'expérimentation	90
5.3.1	Scénario 1 : Cas simple avec deux robots	90
5.3.2	Scénario 2 : Cas d'un groupe de trois robots	97
5.4	Conclusion	105

Ce chapitre présente les résultats de la simulation de l'approche de localisation collective par échange d'état. Pour que l'expérimentation soit réaliste, nous avons simulé les problèmes typiques qui peuvent être rencontrés dans une expérimentation réelle tels que les latences liées aux informations des capteurs et à la communication, etc.

5.1 Description des simulations

L'approche de localisation collective réaliste présentée dans le chapitre 4 de la partie II a été testée en simulation dans le cas d'un groupe de deux puis de quatre robots mobiles. Par la suite nous faisons les hypothèses suivantes :

- Les robots sont équipés de capteurs proprioceptifs, tels que des capteurs odométriques qui mesurent leurs déplacements, d'un capteur permettant de les positionner dans un repère absolu et commun à tout le groupe tel que le GPS par exemple et d'un capteur leur permettant de détecter les congénères à proximité et de mesurer leurs poses relatives.
- Aucun robot ne dispose de capteur lui permettant d'identifier les robots détectés.
- Tous les capteurs simulés délivrent des informations affectées d'erreurs supposées gaussiennes dont l'écart type correspond à ce que l'on peut obtenir avec des capteurs réels.
- Les capteurs simulés délivrent les informations d'une manière désynchronisée et à des fréquences différentes.
- Les robots disposent d'un système de communication qui leur permet d'échanger des données entre eux. La communication entre deux robots est établie lorsque ces derniers sont à une distance inférieure à la portée maximale du dispositif de communication.
- Les données transférées peuvent être affectées de retards et la communication entre deux robots peut être interrompue momentanément même si ces derniers sont assez proches pour communiquer.

5.2 Génération des données de simulation

Trajectoires des robots : Les déplacements réels des robots sont générés à partir d'un vecteur de vitesses, d'un vecteur d'angles au volant et un modèle cinématique d'évolution. Les positions des robots sont calculées en intégrant les valeurs de vitesses et d'angles au volant de proche en proche à l'aide d'un modèle cinématique de type tricycle en partant d'une pose connue. Ce modèle est détaillé dans la section 1.1.1.1 du chapitre 1.

Génération des données capteurs : Les trajectoires obtenues représentent la vérité terrain. Les valeurs contenues dans les vecteurs de vitesses et d'angles au volant représentent les vraies informations que devraient fournir les capteurs d'un robot qui emprunte ces trajectoires.

Les informations mesurées par les capteurs des robots sont considérées comme étant les valeurs vraies additionnées à un bruit gaussien d'écart type variable selon le type de capteur simulé :

- Les données provenant des capteurs odométriques et d'angle au volant sont générées à partir des vecteurs de vitesses et d'angles au volant utilisés pour la génération des trajectoires des robots. L'écart type du bruit ajouté dans ce cas est de 2° pour les angles au volant et de $0.03m/s$ pour la vitesse du robot.
- Les données de positionnement absolu (type GPS) sont générées à partir de trajectoires

5.2 Génération des données de simulation

vraies obtenues avec les données de vitesse et d'angles au volant et le modèle cinématique du robot. Les données de positionnement sont sous échantillonnées afin de simuler la lenteur de ce type de capteurs par rapport à l'odométrie par exemple. L'écart type du bruit ajouté dans ce cas, est de $4m$.

- Les mesures de poses relatives entre les robots sont générées en utilisant les différences entre les poses (positions et orientations) vraies des robots. Les écarts types des bruits ajoutés sont de $0.25m$ pour les poses relatives selon les axes x et y et de 10° pour l'orientation relative.

Simulation des latences et l'absence d'information : Les données capteurs générées sont datées et classées par ordre chronologique (figure 5.1). Lors du déroulement du scénario de simulation, ces données sont lues dans un ordre semi chronologique. A chaque instant le robot peut recevoir des données qui peuvent dater de l'instant présent mais aussi des données arrivées en retard qui datent d'instants passés.

Dans la figure 5.1 par exemple, à l'instant k le robot peut recevoir une des données mises en

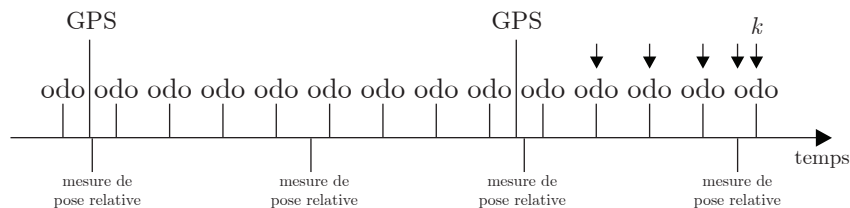


FIGURE 5.1 – Simulation des latences des données.

évidence par des flèches. Notons que le nombre de données susceptibles d'arriver à cet instant dépend du retard maximal que l'on veut simuler. Le choix de la donnée à traiter à cet instant est fait de façon aléatoire. Cette donnée est marquée comme traitée pour ne pas être réutilisée par la suite. Si la donnée choisie de façon aléatoire a déjà été traitée, aucune donnée n'est traitée et le système de simulation passe à l'instant suivant. Le retard simulé dans les scénarios présentés est d'une valeur maximale de 2 secondes.

5.3 Scénario de l'expérimentation

L'approche proposée a été testée dans le cas de deux scénarios de simulation. Le premier scénario représente un cas simple qui fait intervenir un groupe de deux robots qui se déplacent en ligne droite et qui se croisent pour mesurer leurs poses relatives et faire de la localisation coopérative.

Le deuxième scénario traite un cas plus complexe qui fait intervenir quatre robots mobiles. Trois de ces robots sont équipés des capteurs nécessaires pour faire de la localisation coopérative mais le quatrième n'est équipé d'aucun de ces capteurs. Nous présentons dans ce cas, d'une part les résultats de l'approche avec trois véhicules, et d'autre part la gestion du robot non équipé (ou obstacle) dans l'approche de localisation coopérative.

5.3.1 Scénario 1 : Cas simple avec deux robots

La figure 5.2 représente les trajectoires prises par les deux robots. Les robots R_1 et R_2 démarrent des positions respectives et suivent les trajectoires représentées en rouge pour le robot R_1 et en vert pour le robot R_2 . La distance maximum en dessous de laquelle les robots peuvent se détecter et mesurer leurs poses relatives est représentée par d_1 qui est égale dans ce cas à 60m. La distance d_2 , qui vaut 100m dans notre cas, représente quant à elle la distance maximum de communication. Au-delà de cette distance la communication entre les véhicules devient impossible.

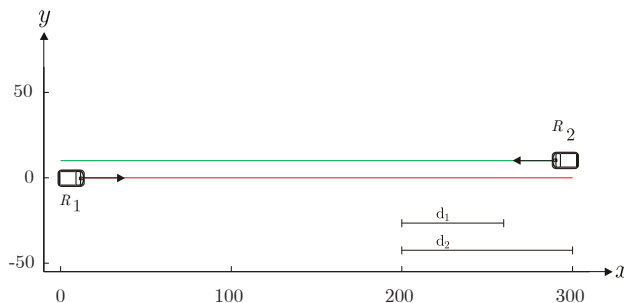


FIGURE 5.2 – Trajectoires prises par les deux robots. d_1 : distance maximum de détection. d_2 : distance maximum de communication.

5.3 Scénario de l'expérimentation

Dans les figures 5.3, 5.4 et 5.5, la pose réelle de chaque robot est représentée par un point. Les estimations des poses des robots sont représentées par une ellipse d'incertitude à trois écarts types centrée sur la pose estimée. L'absence d'ellipse sur les figures signifie que la pose n'est pas estimée. Les nombres représentés à côté des poses des robots représentent les identifiants des robots.

Les figures 5.3, 5.4 et 5.5 représentent les états du groupe estimés dans chacun des deux robots à différents instants durant le déroulement du scénario de la simulation. Les figures 5.3(a1), 5.4(a1) et 5.5(a1) représentent la version de l'état du groupe estimé dans le robot R_1 , cet état est appelé W_s dans le chapitre 4. Les figures 5.3(a2), 5.4(a2) et 5.5(a2) représentent la version de l'état du groupe estimée dans le robot R_2 .

Les figures 5.3(b1), 5.4(b1) et 5.5(b1) représentent l'état global fusionné dans le robot R_1 après l'échange de données. Cet état est appelé W^s dans le chapitre 4. Les figures 5.3(b2), 5.4(b2) et 5.5(b2) représentent l'état global fusionné dans le robot R_2 .

La figure 5.6 représente l'évolution de l'erreur de localisation des deux robots dans les états globaux fusionnés. Cette localisation est représentée dans les figures 5.3(b1), 5.4(b1) et 5.5(b1) pour le robot R_1 et dans les figures 5.3(b2), 5.4(b2) et 5.5(b2) pour le robot R_2 .

L'erreur de localisation est représentée par l'écart type de l'estimation de la position selon l'axe de la longitude et celui de la latitude (figure 5.6).

Les deux robots démarrent de leurs positions initiales et prennent leurs trajectoires respectives. Au début de l'application, la distance entre les deux robots est plus grande que les distances maximum de communication et de détection. Ce cas est représenté dans la figure 5.3. Chaque robot se localise en fusionnant les informations issues de ses capteurs mais aucune information n'est échangée entre les deux robots. L'état global fusionné des deux robots (figure 5.3(b1) et figure 5.3(b2)) est alors identique à l'état mis à jour par chaque robot (figure 5.3(a1) et figure 5.3(a2)). Ce cas peut être assimilé à de la localisation mono-véhicule. L'erreur de localisation dans ce cas est représentée dans la figure 5.6 entre les instants t_0 et t_1 . L'écart type moyen de l'estimation de la position dans ce cas est de $3.2m$ selon l'axe de la longitude et de $2.5m$ selon l'axe de la latitude.

Lorsque la distance entre les deux robots devient inférieure à la distance maximum de communication, les robots commencent à échanger leurs versions de l'état du groupe. Une fois l'échange effectué chaque robot peut fusionner les deux versions de l'état du groupe à sa disposition (sa propre version et celle qu'il a reçue de son congénère). Ce cas est représenté dans la figure 5.4. Nous pouvons voir dans les figures 5.4(b1) et 5.4(b2) que chacun des deux robots

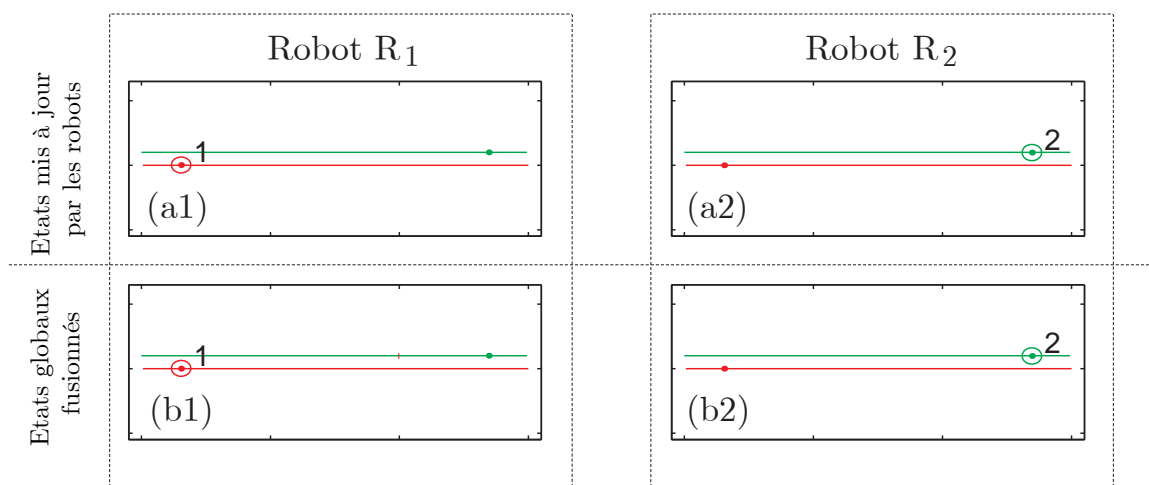


FIGURE 5.3 – Etat du groupe estimé dans les deux robots : les deux robots n'échangent aucune information. (a1) : état du groupe estimé dans le robot R_1 . (a2) : état du groupe estimé dans le robot R_2 . (b1) : état global du groupe fusionné dans le robot R_1 . (b2) : état global du groupe fusionné dans le robot R_2 .

5.3 Scénario de l'expérimentation

dispose à présent dans son état global fusionné, de l'estimation de sa propre pose (calculée avec ses propres capteurs) et l'estimation de la pose de son congénère reçue de ce dernier. Comme aucune mesure de pose relative n'est encore possible, chacun des deux robots ne dispose que d'une seule version de l'estimation de sa pose (sa propre version). L'échange de données dans ce cas n'a alors aucune incidence sur la précision de la localisation. Il ne fait que propager les informations de localisation. L'erreur de localisation dans ce cas est représentée dans la figure 5.6 entre les instants t_1 et t_2 . L'écart type moyen dans cet intervalle ne change pas, car aucune pose relative n'a encore été mesurée.

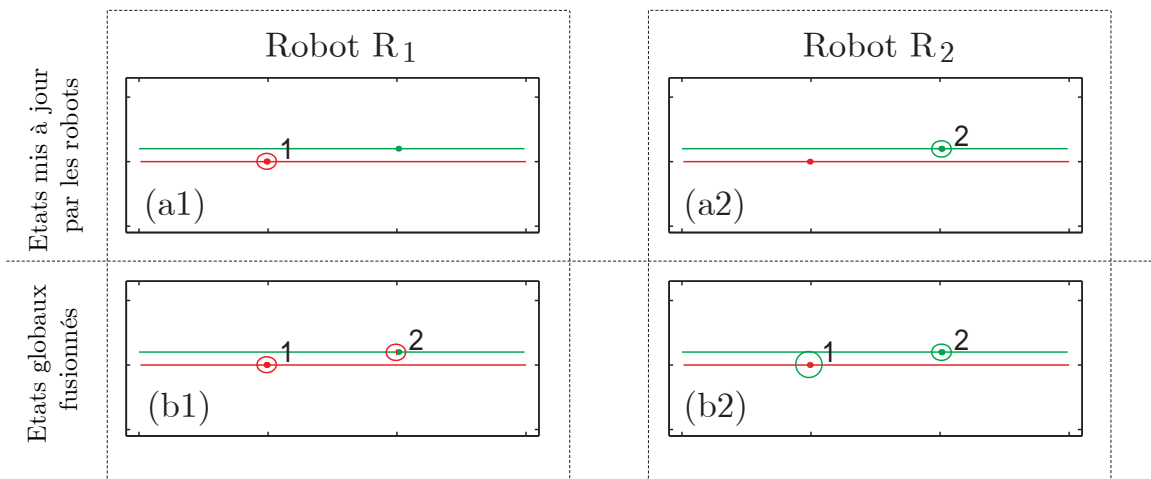


FIGURE 5.4 – Etat du groupe estimé dans les deux robots : les deux robots peuvent échanger leurs données de localisation mais ne peuvent pas se détecter ni se localiser mutuellement. (a1) : état du groupe estimé dans le robot R_1 . (a2) : état du groupe estimé dans le robot R_2 . (b1) : état global du groupe fusionné dans le robot R_1 . (b2) : état global du groupe fusionné dans le robot R_2 .

Lorsque la distance entre les deux robots devient inférieure au seuil de détection les deux robots peuvent mesurer leurs poses relatives et mettent à jour en conséquence leur version de l'état du groupe pour le communiquer par la suite. Les figures 5.5(a1) et 5.5(a2) montrent que chaque robot dispose dans sa version de l'état du groupe d'une estimation de sa propre pose et d'une estimation de la pose de son congénère calculée à partir des informations relatives mesurées.

Simulation de l'approche de localisation collective

En recevant l'état du robot R_2 , le robot R_1 se retrouve avec quatre estimations de poses de robots : deux poses dans son propre état (sa propre pose et celle du robot qu'il a détecté) et deux poses dans l'état qu'il a reçu du robot R_2 (la pose du robot R_2 et une pose d'un autre robot que R_2 a détecté). Afin de fusionner les deux états, une étape de mise en correspondance est effectuée. Dans cette étape le robot R_1 parcourt l'état reçu du robot R_2 pour trouver les correspondances entre les robots présents dans les deux versions de l'état du groupe. Le critère de correspondance est décrit dans la section 4.2.1.3 du chapitre 4. La mise en correspondance permet de déterminer que le robot détecté par le robot R_1 est bien le robot R_2 et celui détecté par le robot R_2 n'est autre que le robot R_1 . Chaque robot a ainsi deux estimations de sa pose (une estimée en local et l'autre estimée par le congénère). La fusion de ces informations permet aux deux robots d'obtenir les états globaux représentés dans les figures 5.5(b1) et 5.5(b2). Nous pouvons remarquer que les ellipses d'incertitude qui représentent les erreurs de localisation des robots R_1 et R_2 dans les états globaux fusionnés (figures 5.5(b1) et 5.5(b2)) sont plus petites que celles présentes dans les versions estimées par les deux robots (figures 5.5(a1) et 5.5(a2)). Ce cas est représenté dans la figure 5.6 entre les instants t_2 et t_3 . La diminution de l'erreur de localisation des deux robots est due à la fusion des données locales et celles reçues du congénère. L'écart type moyen de l'erreur de position dans cet intervalle passe de $3.2m$ à $2.5m$ selon la longitude et de $2.5m$ à $1.9m$ selon la latitude, ce qui fait un gain en précision de 25%. Lorsque les deux robots s'éloignent et redeviennent dans l'impossibilité de mesurer leurs poses relatives, l'erreur de localisation dans les états globaux fusionnés augmente à nouveau pour atteindre l'erreur obtenue dans le cas d'une localisation mono-robot. Ce cas est représenté dans la figure 5.6 à partir de l'instant t_3 .

5.3 Scénario de l'expérimentation

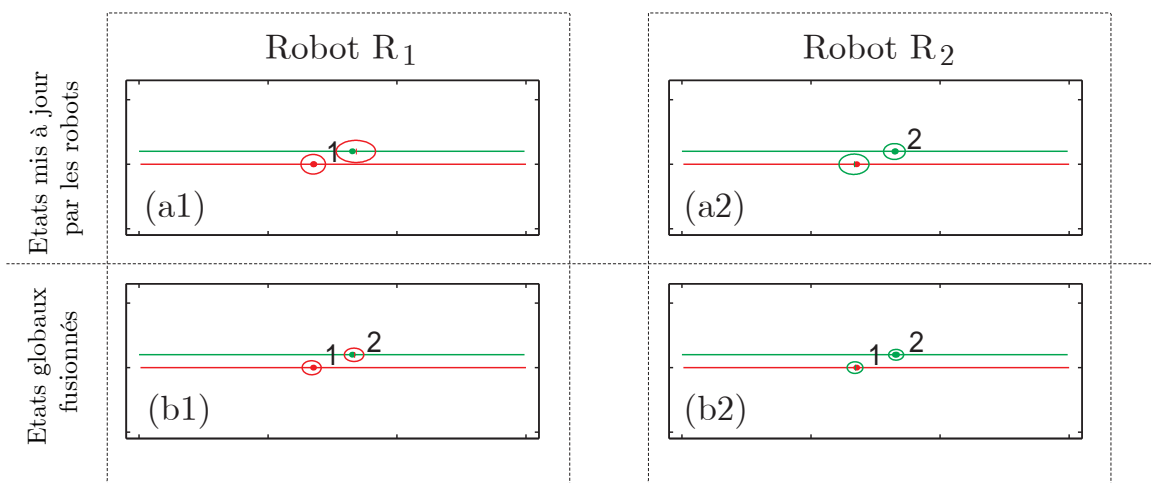


FIGURE 5.5 – Etat du groupe estimé dans les deux robots : les deux robots se détectent, se localisent l'un l'autre et échangent les données. (a1) : état du groupe estimé dans le robot R_1 . (a2) : état du groupe estimé dans le robot R_2 . (b1) : état global du groupe fusionné dans le robot R_1 . (b2) : état global du groupe fusionné dans le robot R_2 .

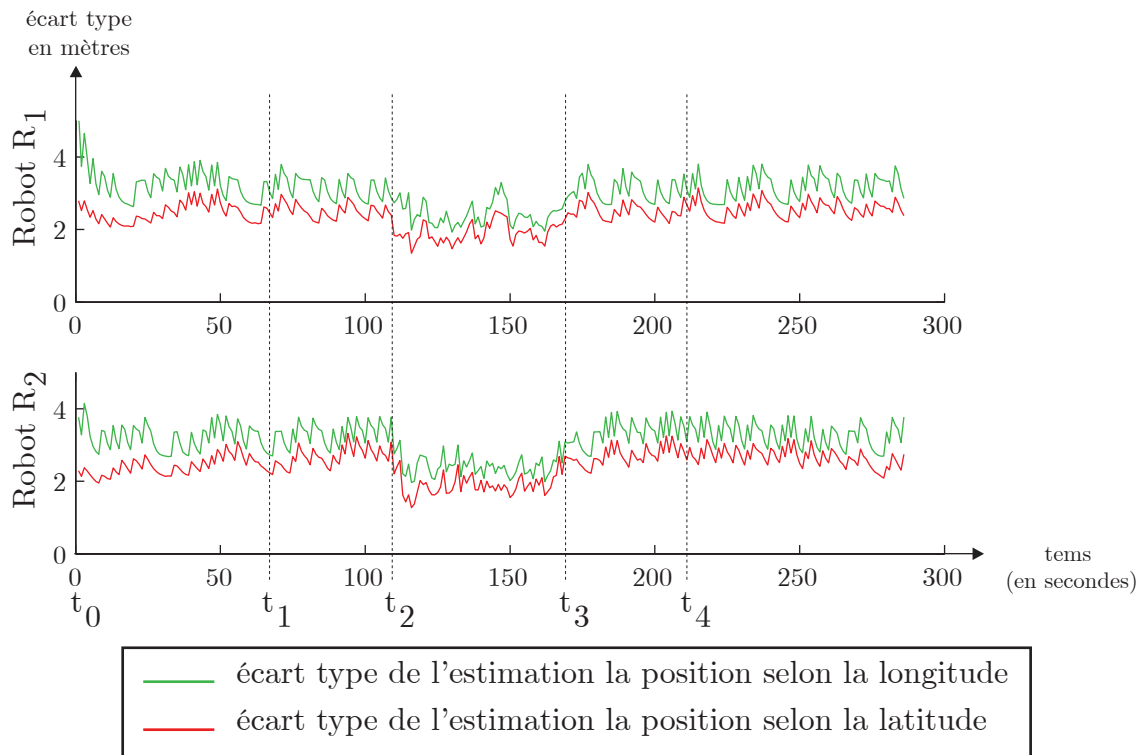


FIGURE 5.6 – Ecart type des erreurs des localisations des robots R_1 et R_2 selon les axes de la longitude et de la latitude obtenues dans les états fusionnés des deux robots.

5.3 Scénario de l'expérimentation

5.3.2 Scénario 2 : Cas d'un groupe de trois robots

Ce scénario fait intervenir un groupe de quatre robots mobiles. Trois d'entre eux sont équipés des capteurs de localisation et de détection et du système de communication. Le quatrième robot ne dispose d'aucun de ces équipements. Il joue le rôle d'un obstacle mobile.

La figure 5.7 représente les trajectoires prises par les quatre robots. d_1 et d_2 représentent respectivement les distances maximum de détection et de communication $d_1 = 60m$ et $d_2 = 100m$.

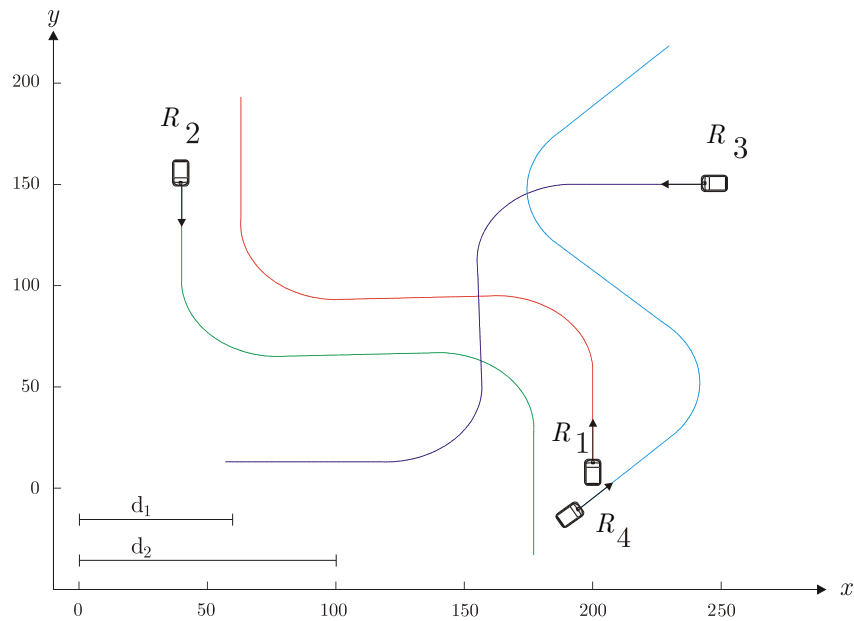


FIGURE 5.7 – Trajectoires prises par les quatre robots. d_1 : distance maximum de détection (60m). d_2 : distance maximum de communication (100m).

Dans les figures 5.8, 5.9, 5.10, 5.11, 5.12 et 5.13, les vignettes (a1), (a2) et (a3) représentent respectivement les versions de l'état du groupe estimées dans les robots R_1 , R_2 et R_3 . Les états globaux fusionnés dans les trois robots équipés sont représentés dans les vignettes (b1), (b2) et (b3).

Les quatre robots démarrent de leur position initiale et prennent leurs trajectoires respectives. La figure 5.14 représente l'évolution de l'erreur de localisation des robots R_1 , R_2 et R_3 calculés dans leur état global respectif pendant le déroulement du scénario.

Au début du scénario, les distances entre les robots R_1 , R_2 et R_3 sont supérieures à la distance maximum de communication. Dans ce cas chaque robot met à jour sa version de l'état du groupe avec ses propres capteurs en estimant sa pose et celle des robots qu'il a détectés. Ce cas est représenté dans la figure 5.8. Nous pouvons voir par exemple dans la vignette (a1) que le robot R_1 a détecté le robot R_4 . Le robot R_4 dans l'état global fusionné dans le robot R_1 (vignette (b1)) à un identifiant nul. En effet comme le robot R_4 n'échange aucune donnée avec les autres robots il ne pourra jamais s'identifier auprès des autres robots.

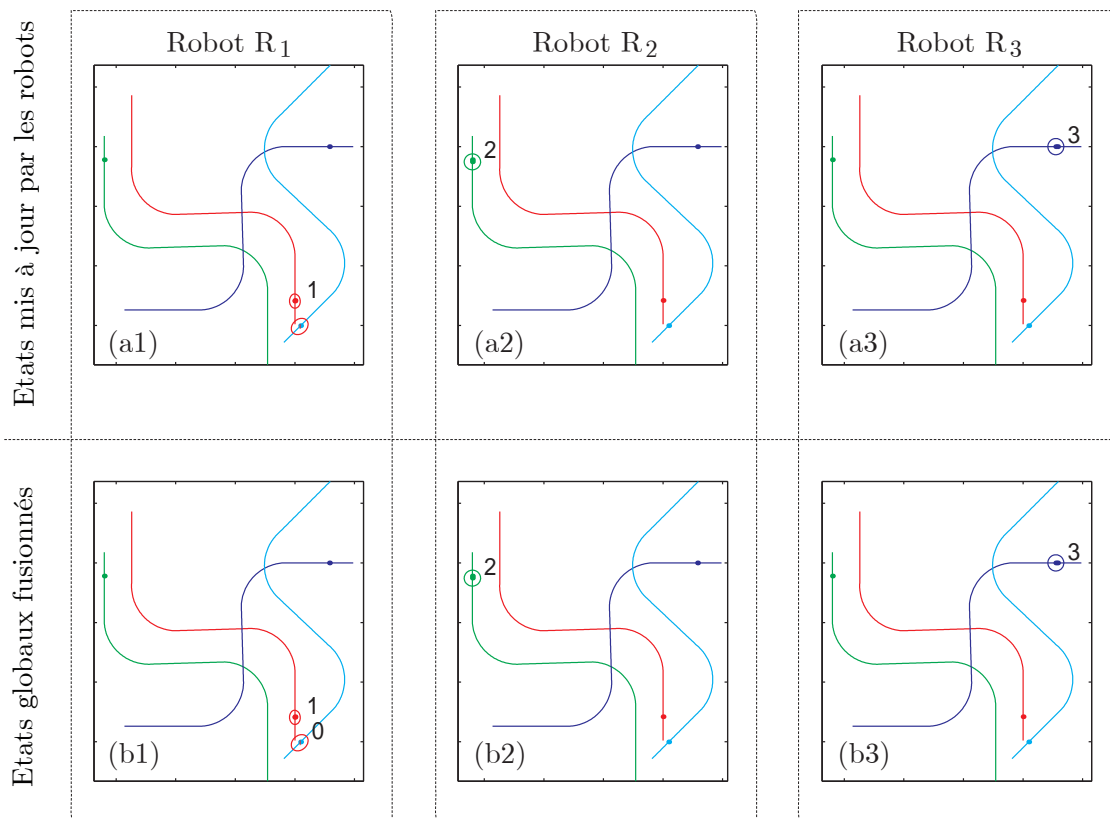


FIGURE 5.8 – Etat du groupe estimé dans les trois robots équipés : Le robot R_1 détecte le robot R_4 mais aucun échange de données n'a encore été effectué. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 .

5.3 Scénario de l'expérimentation

Lorsque les robots R_1 et R_3 sont assez proches pour communiquer, ils échangent leurs versions de l'état du groupe. Le résultat de la fusion de ces états est représenté dans la figure 5.9. Les vignettes (b3) montrent que le robot R_3 dispose dans son état global fusionné des estimations des poses des robots R_1 et R_4 . Comme aucune pose relative n'a encore été mesurée, l'échange de données entre les robots R_1 et R_3 et la fusion des états n'a aucune incidence sur l'erreur de localisation de ces deux robots. L'erreur de localisation dans ce cas est représentée dans la figure 5.14 entre les instants t_0 et t_1 . Les écarts types des estimations moyens dans cette intervalle sont compris entre $3.3m$ et $4m$ sur la longitude et la latitude.

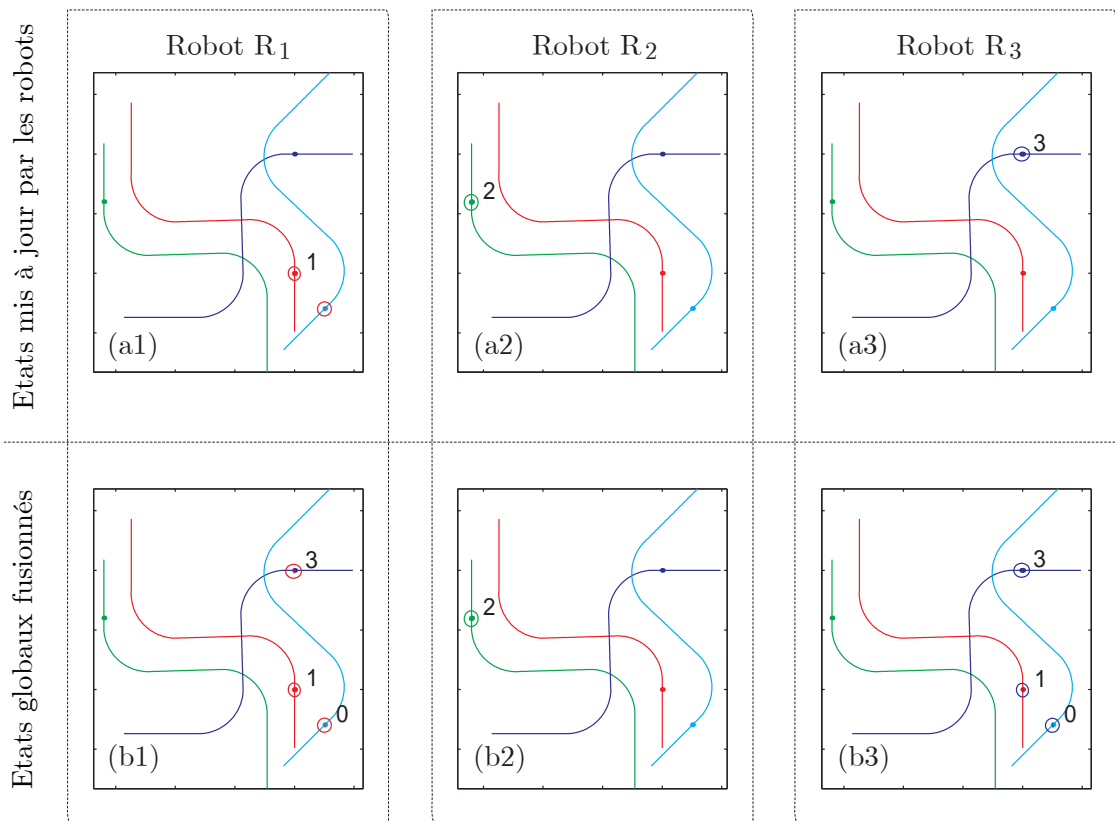


FIGURE 5.9 – Etat du groupe estimé dans les trois robots équipés : Les robots R_1 et R_3 communiquent. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 .

Lorsque les robots R_1 et R_3 se croisent, ils mesurent leurs poses relatives et échangent ces

Simulation de l'approche de localisation collective

données. La fusion permet d'obtenir une meilleure localisation. Ce cas est représenté figure 5.10 et l'erreur de localisation dans ce cas est représentée dans la figure 5.14 entre les instants t_1 et t_2 . Nous pouvons voir que les erreurs de localisation des robots R_1 et R_3 ont diminué par l'effet de la localisation coopérative. L'écart type moyen de l'estimation de la position dans cette intervalle passe de $3.2m$ à $2.5m$ et de $3.8m$ à $2.1m$ selon la longitude et de $3.3m$ à $2.2m$ et de $3.5m$ à $2.3m$ respectivement pour les deux R_1 et R_3 . En effet chaque robot dispose dans son état du groupe d'une estimation de sa propre pose et de celle du robot qu'il détecte.

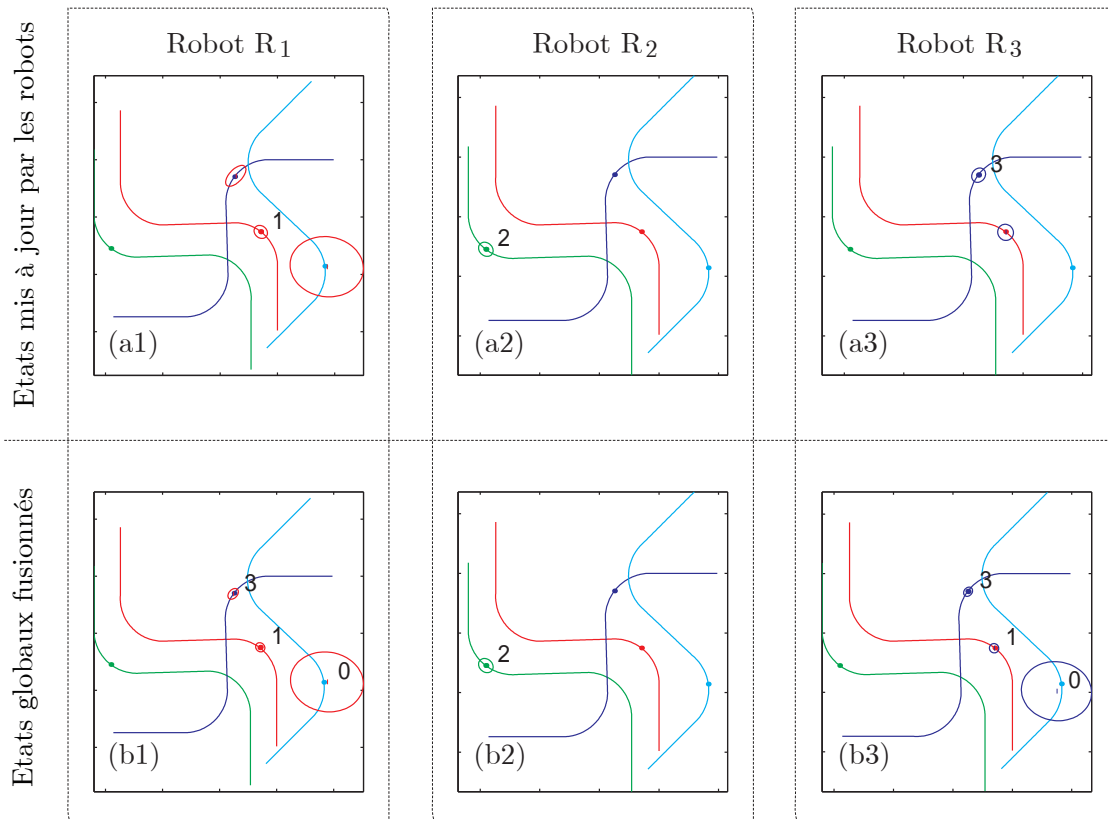


FIGURE 5.10 – Etat du groupe estimé dans les trois robots équipés : Les robots R_1 et R_3 peuvent mesurer leurs poses relatives et échanger ces données. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 .

L'instant où les trois robots peuvent se détecter représente le cas le plus favorable de ce

5.3 Scénario de l'expérimentation

scénario. En effet les trois robots disposent chacun de trois estimations de leur pose (la première est celle estimée en local par le robot lui-même et les deux autres sont celles estimées par ces deux congénères) la fusion des états reçus avec la version de l'état du groupe dans chacun des robots lui fournit une localisation plus précise de l'ensemble du groupe. Ce cas est représenté dans la figure 5.14 entre les instants t_2 et t_3 . L'écart type moyen de l'estimation de la position dans cet intervalle selon la longitude et la latitude est de $2.0m$ et $1.7m$ pour le robot R_1 , de $1.9m$ et $1.6m$ pour le robot R_2 et de $1.7m$ et $1.8m$ pour le robot R_3 . Ce qui fait un gain en précision de 40% à 60% par rapport à l'erreur de localisation dans l'intervalle entre t_0 et t_1 .

La figure 5.11 montre aussi que les robots R_1 et R_2 disposent d'une estimation de la pose du robot R_4 . Cette estimation leur a été transmise par le robot R_3 .

Dans la figure 5.12, nous pouvons voir que les robots R_1 et R_3 peuvent mesurer la pose relative du robot R_2 mais ne peuvent pas mesurer leurs poses relatives mutuelles. Le robot R_2 quant à lui peut mesurer les poses relatives de ses deux congénères R_1 et R_3 . Après l'échange de données, les robots R_1 et R_3 disposent de deux estimations de leur poses (leur propre estimation et celle reçue du robot R_2). Le robot R_2 quant à lui dispose de trois estimation de sa pose (sa propre estimation et les deux estimations transmises par les deux autres robots R_1 et R_3). Après la fusion des données échangées, la pose du robot R_2 est plus précise que celle des deux autres robots. Ceci est montré dans la figure 5.14 entre les instants t_3 et t_4 . Nous pouvons voir que l'erreur de localisation sur la pose du robot R_2 est meilleure que celle des robots R_1 et R_3 .

La figure 5.13 représente le cas où les robots R_2 , R_3 sont encore assez proches pour pouvoir mesurer leurs poses relatives. Le robot R_1 quant à lui ne peut ni détecter ses congénères ni communiquer avec ces derniers. L'erreur de localisation des trois robots est représentée dans ce cas sur la figure figure 5.14 entre les instants t_4 et t_5 . L'erreur des deux robots qui mesurent encore leurs poses relatives reste plus petite que celle du robot R_1 qui tend vers l'erreur de localisation mono-robot.

A partir de l'instant t_5 de la figure 5.14 la distance entre les robots devient trop grande pour que ces derniers puissent se détecter et mesurer leurs poses relatives. Dans ce cas, même si la communication reste encore possible entre certains de ces robots, l'erreur de localisation augmente pour atteindre l'erreur obtenue avec une localisation mono-robot. En effet chaque robot ne dispose alors que d'une seule estimation de sa pose (celle qu'il a estimée par ses propres capteurs). La fusion des éventuels états reçus avec celui construit n'améliore plus la précision de localisation.

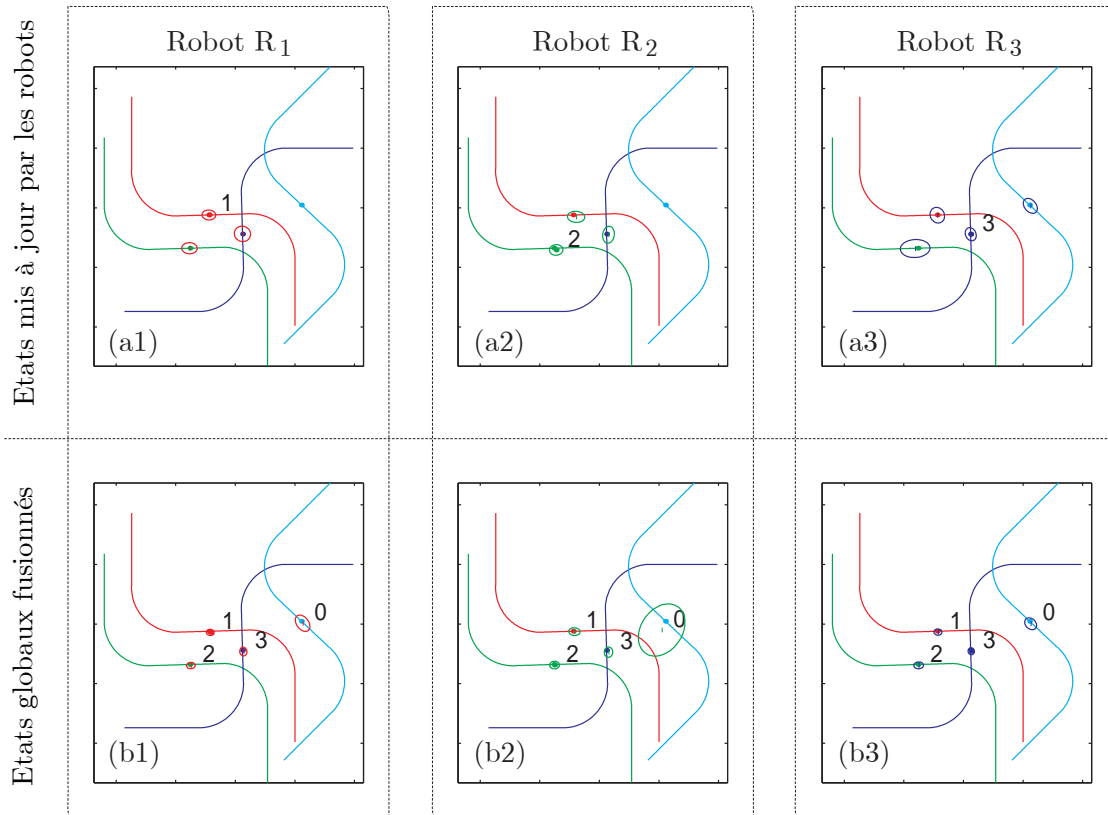


FIGURE 5.11 – Etat du groupe estimé dans les trois robots équipés : Les trois robots peuvent mesurer leurs poses relatives et communiquer ces données. Le robot R_3 peut mesurer la pose relative du robot R_4 . (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 .

5.3 Scénario de l'expérimentation

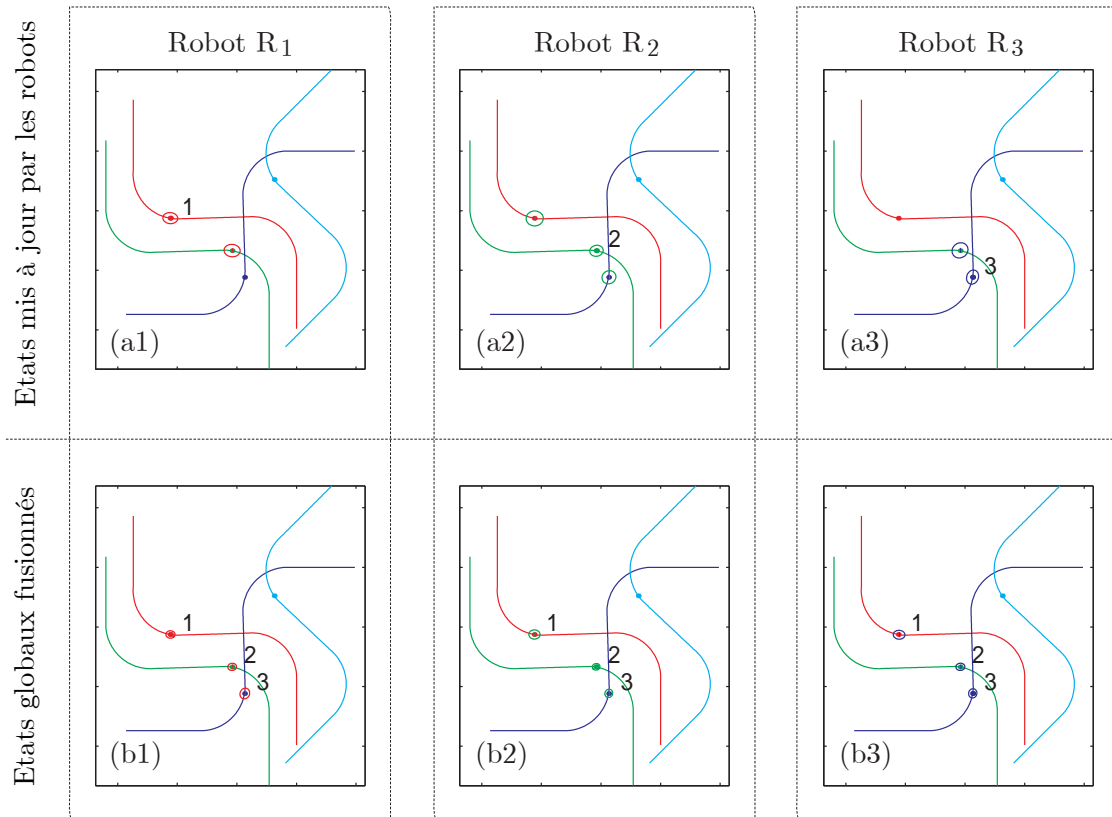


FIGURE 5.12 – Etat du groupe estimé dans les trois robots équipés : Les trois robots communiquent entre eux, le robot R_2 mesure les poses relatives des deux autres robots et les robots R_1 et R_3 ne peuvent mesurer que la pose relative du robot R_2 . (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 .

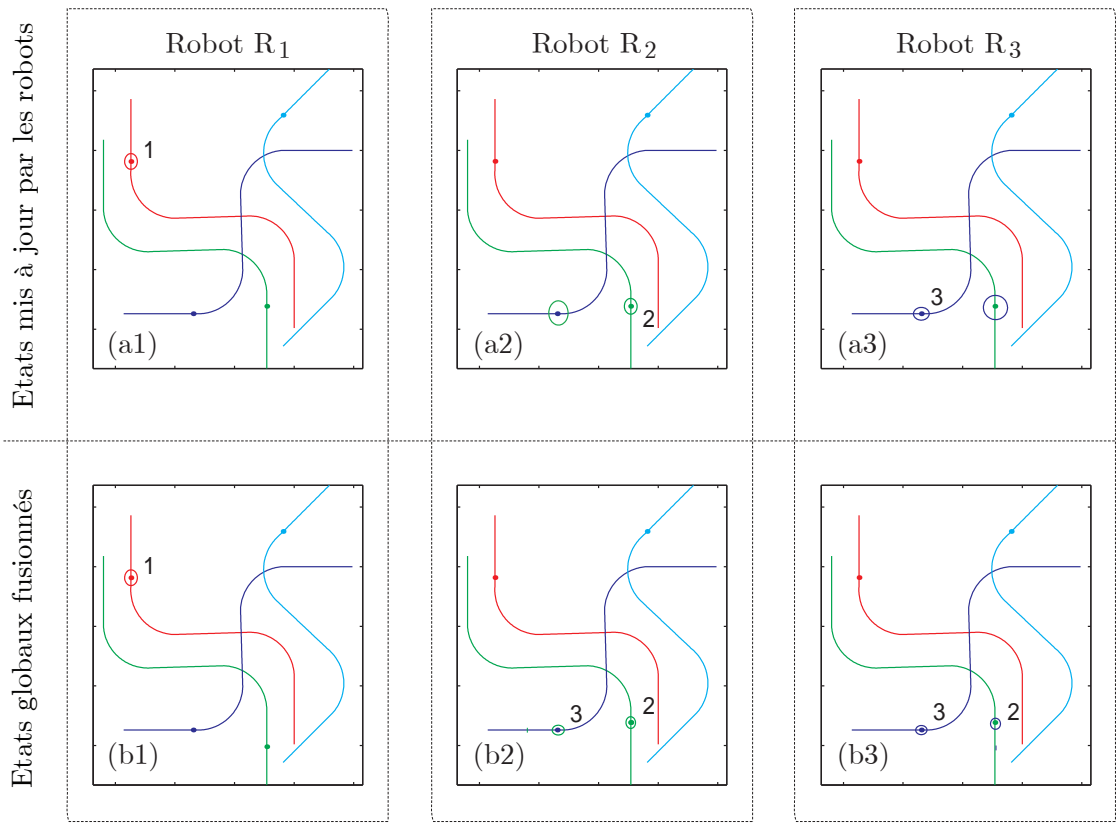


FIGURE 5.13 – Etat du groupe estimé dans les trois robots équipés : les robots R_2 et R_3 peuvent encore mesurer leurs poses relatives et échanger des informations. (a1), (a2) et (a3) : états du groupe estimés respectivement dans les robots R_1 , R_2 et R_3 . (b1), (b2) et (b3) : états globaux du groupe fusionnés respectivement dans les robots R_1 , R_2 et R_3 .

5.4 Conclusion

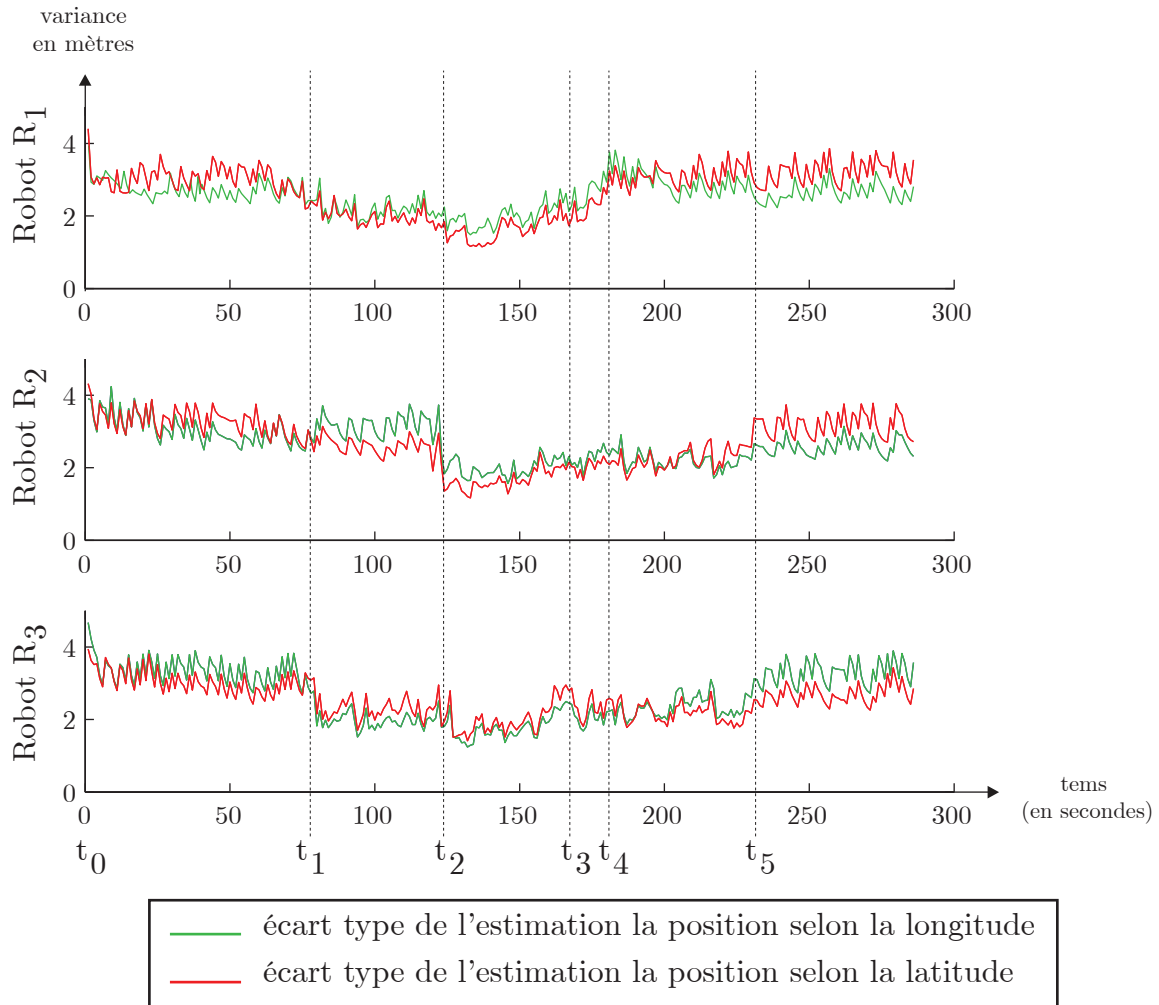


FIGURE 5.14 – Ecart type des erreurs des localisations des robots R_1 , R_2 et R_3 selon les axes de la longitude et de la latitude obtenues dans les états fusionnés des trois robots.

5.4 Conclusion

Dans ce chapitre nous avons présenté les résultats de la simulation de l'approche proposée. Nous avons testé, dans un premier temps, l'approche proposée dans le cas simple de deux robots qui suivent des trajectoires rectilignes pour se croiser, se détecter et mesurer leurs poses relatives. Le partage des données sur les positionnements relatifs relatives à travers l'état échangé permet à chaque robot d'obtenir deux estimations de sa propre pose. La fusion de ces deux états permet de diminuer les erreurs de localisations des deux robots du groupe. Nous avons vu que

Simulation de l'approche de localisation collective

dans ce cas, le gain en précision atteint 25%.

Dans un deuxième temps, l'approche a été testée dans un cas plus complexe qui met en œuvre un groupe de trois robots communicants et un robot non équipé qui joue le rôle d'obstacle mobile. Cette simulation permet de donner une idée sur le gain en précision obtenu avec une flotte de trois robots. Nous avons montré que ce gain peut atteindre 60% de la valeur de l'erreur initiale. Ce gain est, entre autre, fonction de la valeur de l'erreur initiale, de la quantité et de la précision des informations relatives échangées. Cette simulation permet de tester, entre autres, le processus de mise en correspondance effectué avant la fusion des états. Nous avons vu que les états échangés étaient fusionnés correctement et que l'identification des véhicules dans les *états globaux fusionnés* était faite correctement.

Les résultats obtenus permettent d'envisager l'implémentation de l'approche de localisation coopérative proposée dans un environnement réel sur la flotte de véhicules expérimentaux.

Chapitre 6

Expérimentation réelle

Sommaire

6.1	Plate-forme expérimentale	108
6.1.1	Le cycab	108
6.1.2	Capteurs utilisés	108
6.2	Scénario de l'expérimentation	110
6.3	Construction de l'état du groupe	111
6.3.1	Localisation	111
6.3.2	Détection du véhicule de tête	112
6.4	Résultats et discussions	114
6.4.1	Résultats de la localisation coopérative	115
6.5	Problèmes rencontrés	124
6.6	Conclusion	124

L'approche proposée a été implémentée et testée dans le cas d'un groupe de deux véhicules réels qui évoluent dans une formation de convoi. Nous présentons dans un premier temps la plate-forme expérimentale, les capteurs utilisés et le scénario des expérimentations. Ensuite nous présentons les résultats obtenus afin de valider l'apport de l'approche proposée dans ce mémoire.

6.1 Plate-forme expérimentale

6.1.1 Le cycab

Le groupe de robots sur lequel l'approche proposée a été testée est composé de deux véhicules de type cycab (figure (6.1)).

Le cycab est un véhicule expérimental électrique fabriqué par la société Robosoft. C'est un véhicule de petites dimensions (1,90 m de long et 1,20 m de large) qui est capable de transporter deux personnes à une vitesse maximale de 18 km/h.

Le cycab peut être conduit en mode automatique ou en mode manuel à l'aide d'un joystick. Pour les expérimentations, les deux cycabs utilisés sont conduits en mode manuel. L'architecture informatique implantée dans le cycab est composée de deux ordinateurs :

Le premier ordinateur est appelé *PC bas niveau* et gère les vitesses et les angles de braquages à appliquer aux roues. Il est connecté aussi aux capteurs odométriques et au télémètre laser. Il transfère les données issues de ces capteurs via une connexion Ethernet au deuxième ordinateur appelé *PC haut niveau*.

Le *PC haut niveau* est destiné à accueillir les algorithmes de traitement de données. Il exécute dans notre cas l'application de localisation coopérative proposée. Cette approche a été implémentée en C++ en utilisant AROCCAM (*Architecture d'Ordonnancement de Capteurs pour la Création d'Algorithmes Modulaires*) présenté dans [TCD⁺06].

6.1.2 Capteurs utilisés

L'expérimentation menée fait intervenir un groupe de deux cycabs qui se déplacent en mode convoi. La figure 6.2 présente cette configuration et les différents capteurs utilisés.

Le véhicule de tête est équipé des capteurs suivants :

- des capteurs odométriques qui décrivent le déplacement effectué par chacune des quatre roues,
- un GPS bas coût lui permettant de se localiser dans un repère absolu avec une précision qui varie de 3m à 9m selon la configuration des satellites et l'environnement du véhicule,
- un GPS centimétrique de type RTK (*Real Time Kinematic*) qui permet de localiser le véhicule dans le même repère que le GPS bas coût. Ce système permet d'obtenir des précisions de localisation de l'ordre du centimètre. Cette localisation sera utilisée comme

6.1 Plate-forme expérimentale



FIGURE 6.1 – Le Cycab

référence pour évaluer les performances de l'approche de localisation coopérative.

Le véhicule de queue possède les capteurs suivants :

- des capteurs odométriques qui décrivent le déplacement effectué par chacune des quatre roues,
- un GPS centimétrique de type RTK (*Real Time Kinematic*) qui permet de se localiser dans un repère absolu avec une précision de l'ordre du centimètre,
- un télémètre laser qui permet de détecter et d'estimer la pose relative (position relative et orientation relative) du véhicule de tête. Ce télémètre laser couvre un champ de 180° sur une distance de 80 mètres.

Les deux véhicules sont équipés d'un système de communication sans fil qui leur permet d'échanger leurs versions de l'état du groupe.

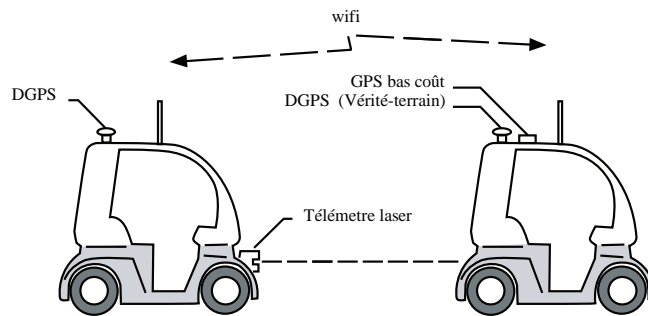


FIGURE 6.2 – Configuration de l'expérience réelle

6.2 Scénario de l'expérimentation

Les deux véhicules sont conduits manuellement en formation de type convoi.

Le véhicule de tête se localise grâce à la fusion des données issues des capteurs odométriques et des données fournies par le GPS bas coût. Le GPS centimétrique n'est présent que pour fournir une référence (vérité terrain) avec laquelle comparer les résultats de fusion et estimer les performances de l'approche proposée.

Le véhicule de queue se localise quant à lui en fusionnant les données issues de ses capteurs odométriques et son GPS centimétrique. Il peut aussi détecter et mesurer la pose relative du véhicule de tête à l'aide du télémètre laser dont il est équipé. Il peut alors construire et mettre à jour une version de l'état du groupe qui contient sa propre pose et celle du véhicule de tête.

Lors d'une expérimentation réelle, ce type de scénario pourrait permettre de gérer un convoi à partir uniquement du pilotage du véhicule de queue.

Les horloges des deux véhicules sont synchronisées régulièrement à travers le réseau de communication sans fil en utilisant le protocole NTP (Network Time Protocol). Ce protocole permet de synchroniser l'horloge locale de l'ordinateur sur une horloge de référence via un réseau informatique. Dans le cas présenté, le véhicule de queue se synchronise sur l'horloge du véhicule de tête. Cette synchronisation permet d'obtenir une précision inférieure à une milliseconde.

6.3 Construction de l'état du groupe

6.3.1 Localisation

Les deux véhicules se localisent en fusionnant les données proprioceptives issues des capteurs odométriques et des données fournies par le GPS centimétrique pour le véhicule de queue et GPS bas coût pour le véhicule de tête. La fusion de données est faite à l'aide d'un filtre de Kalman étendu qui utilise un modèle d'évolution de type tricycle (figure 6.3) présenté dans le chapitre 1. La différence ici, est que l'angle de braquage est calculé par la différence des vitesses des deux roues arrières. Ce modèle est décrit par les équations (6.1), (6.2) et (6.3).

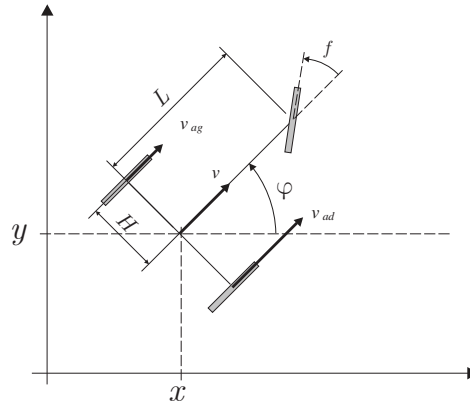


FIGURE 6.3 – modèle tricycle utilisé pour l'évolution de l'état des robots

$$v_k = \frac{v_{ad_k} + v_{ag_k}}{2} \quad (6.1)$$

$$f_k = \arctan\left(\frac{v_{ad_k} - v_{ag_k}}{v_{ad_k} + v_{ag_k}} \frac{L}{H}\right) \quad (6.2)$$

$$\begin{cases} x_{k+1} = x_k + v_k \Delta_T \cos(\varphi_k) \\ y_{k+1} = y_k + v_k \Delta_T \sin(\varphi_k) \\ \varphi_{k+1} = \varphi_k + v_k \Delta_T \tan\left(\frac{f_k}{L}\right) \end{cases} \quad (6.3)$$

Où v_{ad} et v_{ag} sont respectivement les vitesses des roues arrière droite et arrière gauche. v est la vitesse du véhicule rapportée au centre de l'essieu arrière. L et H sont respectivement la distance inter essieux et la moitié de la largeur de l'essieu arrière. f est l'angle de braquage calculé à partir des vitesses des roues arrières, Δ_T le temps écoulé entre deux mesures et x , y et

φ sont les composantes de l'état du véhicule (position et orientation) tel que représenté dans la figure 6.3.

6.3.2 Détection du véhicule de tête

La détection du véhicule de tête est faite à l'aide d'un télémètre laser de type *Sick* présenté dans la figure (6.4). Ce type de télémètre laser couvre un champ de 180° sur une distance de 80



FIGURE 6.4 – Télémètre laser utilisé pour la détection du véhicule de tête.

mètres. Dans la configuration convoi, ce télémètre fournit une image des points détectés (figure (6.5)).

La configuration en mode convoi permet de limiter la recherche du véhicule de tête dans la zone qui se situe devant le véhicule de queue. Cette zone est représentée en pointillés sur la figure 6.5. Cette limitation permet de diminuer la quantité de données à traiter et de simplifier la recherche du véhicule de tête.

Afin de détecter et de mesurer la pose relative du véhicule de tête nous avons élaboré un algorithme simple qui recherche une courbe continue qui constitue la réponse de l'arrière du cycab au faisceau laser (représenté en vert sur la figure 6.5). Cet algorithme parcourt les points présents dans la zone de recherche de proche en proche à la recherche d'une forme continue. Nous supposons que la forme détectée est le cycab de tête si la longueur de la courbe est entre

6.3 Construction de l'état du groupe

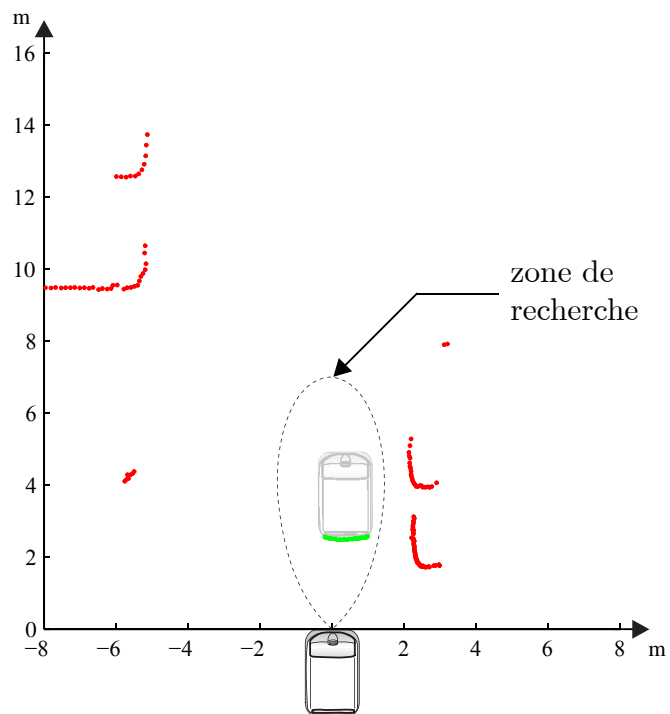


FIGURE 6.5 – Image fournie par le télémètre laser

0.8 m et 1.2 m. Cela permet de détecter le cycab de tête même quand une partie de l'arrière du véhicule ne se trouve pas dans la zone de recherche.

Cet algorithme fournit la position et l'orientation relative du véhicule de tête par rapport au véhicule de queue et les erreurs associées à ces mesures. L'état du groupe est alors mis à jour selon la méthodologie présentée dans la section 4.2.1.2 du chapitre 4.

6.4 Résultats et discussions

Dans cette section nous montrons les résultats de l'approche de localisation coopérative dans le cas d'un convoi de deux véhicules. Les expérimentations ont été effectuées sur deux trajectoires différentes. Les figures 6.6 et 6.7 représentent les trajectoires suivies par le convoi. La première trajectoire fait 193 mètres et la deuxième 113 mètres.

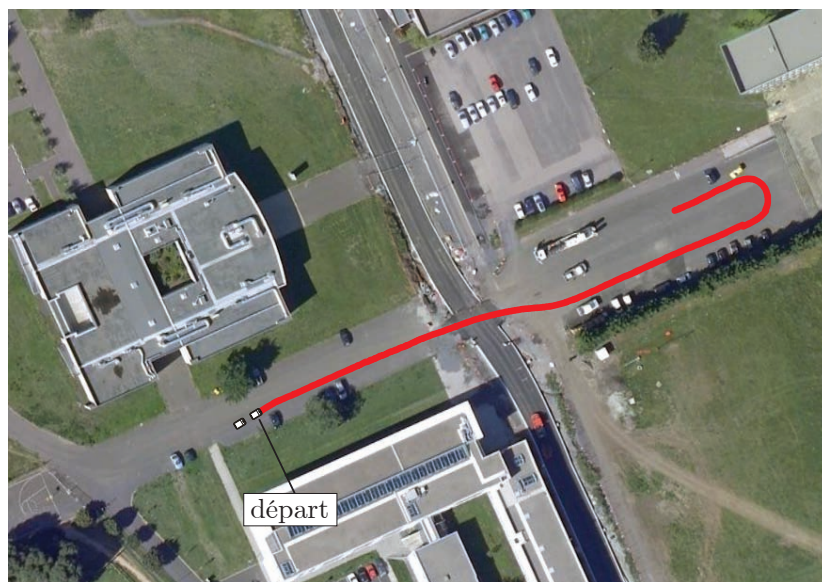


FIGURE 6.6 – Trajectoire 1

6.4 Résultats et discussions

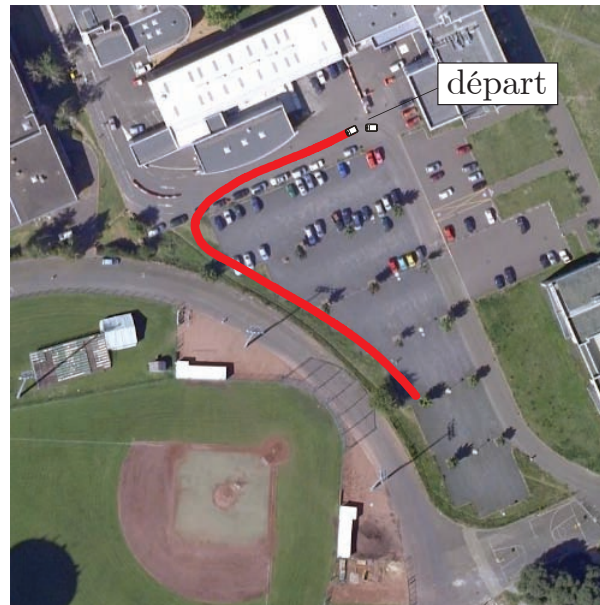


FIGURE 6.7 – Trajectoire 2

6.4.1 Résultats de la localisation coopérative

Le véhicule de queue se localise à l'aide de son GPS centimétrique et l'odométrie sur les roues arrières et détecte et localise le véhicule de tête avec le télémètre laser. Le véhicule de tête se localise quant à lui à l'aide de son GPS bas coût seulement. Les 2 véhicules construisent leurs versions de l'état du groupe en utilisant les capteurs à leur disposition et échangent ces informations à travers un réseau sans fil. Chaque véhicule fusionne l'état qu'il a obtenu avec celui reçu de l'autre véhicule. Dans ce cas, la carte du véhicule de queue est plus riche que celle du véhicule de tête puisque celle-ci contient une estimation de sa propre pose et une estimation de la pose du véhicule de tête.

6.4.1.1 Trajectoire 1

La figure 6.8 montre les résultats de l'algorithme de localisation dans le véhicule de tête. La courbe verte représente la trajectoire fournie par le GPS bas coût, la courbe rouge représente la trajectoire de référence fournie par le GPS centimétrique du véhicule de tête et la courbe bleue représente la trajectoire du véhicule de tête obtenue dans son état global fusionné. Nous pouvons voir que la trajectoire estimée après la fusion est quasiment superposée à la trajectoire de référence donnée par le GPS centimétrique.

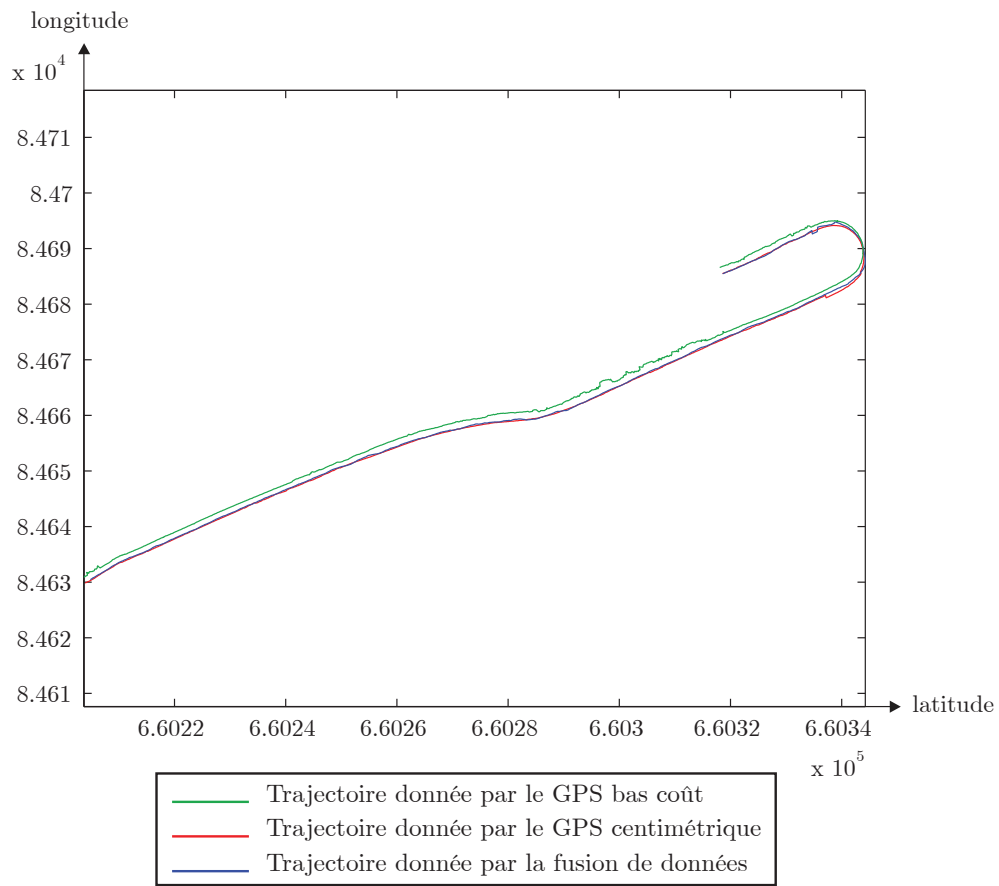


FIGURE 6.8 – Résultats de la localisation : Trajectoire 1.

6.4 Résultats et discussions

Pour évaluer les performances de l'algorithme de la localisation coopérative proposé nous allons caractériser l'erreur de localisation avec deux paramètres. Le premier paramètre est la distance euclidienne entre la position du véhicule de tête estimée après la fusion des états et la position de référence fournie par le GPS centimétrique. Le deuxième paramètre est la distance de Mahalanobis entre la distribution de la position du robot de tête estimée après la fusion des états et la distribution de la position fournie par le GPS centimétrique. Les figures 6.9 et 6.10 représentent respectivement l'évolution de la distance euclidienne et de la distance de Mahalanobis entre la pose estimée et la référence durant l'expérience.

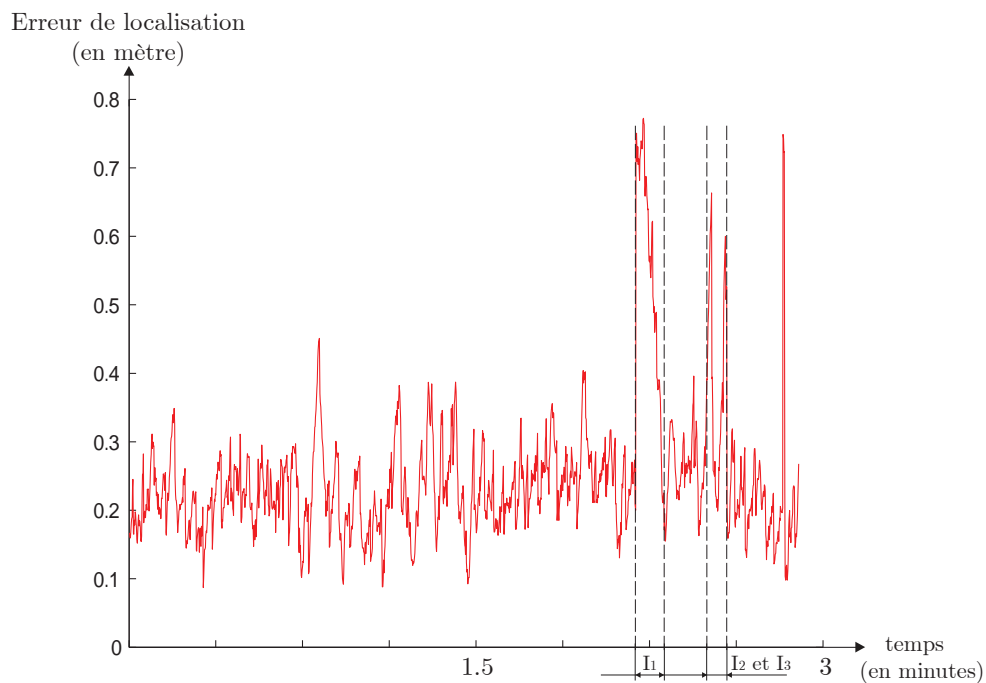


FIGURE 6.9 – Distance euclidienne entre la valeur moyenne de la position estimée après la fusion des états et la valeur de la position de référence donnée par le GPS centimétrique

Si nous considérons la loi du χ^2 comme référence, pour que le résultat de la localisation soit consistant (la position vraie est incluse dans l'ellipse d'incertitude de l'estimation de la pose), la distance de Mahalanobis doit être inférieure à un seuil qui est fonction de la dimension du vecteur à tester et de la probabilité de présence désirée. Dans ce cas nous allons mesurer la distance entre la positions donnée par le GPS centimétrique et l'estimation de la position du robot obtenue après la fusion des états à deux écarts types (95,4%). La table du χ^2 pour un vecteur à 2

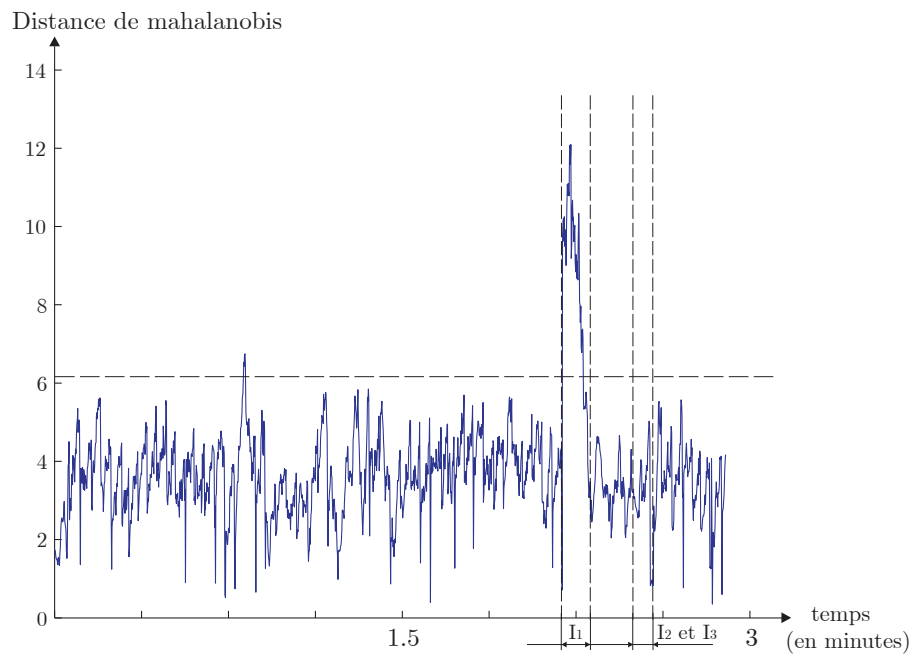


FIGURE 6.10 – Distance de Mahalanobis entre la pose estimée après la fusion et la pose fournie par le GPS centimétrique

6.4 Résultats et discussions

dimensions et une probabilité de présence à 95,4% donne un seuil de 6,15. La figure 6.10 montre que la distance de Mahalanobis reste autour de la valeur 4, ce qui est inférieur au seuil prédéfini.

Dans la figure 6.9 nous pouvons voir que la distance euclidienne entre la position estimée et la position de référence reste autour de 20 centimètres.

Le pic de l'erreur constatée dans l'intervalle de temps I_1 dans les figures 6.9 et 6.10 correspond à un saut du GPS centimétrique utilisé comme référence. Ce saut est mis en évidence dans la figure 6.11 dans la zone entourée et nommée I_1 . Les erreurs de localisation mises en évidence dans les zones I_2 et I_3 sont dues à l'incapacité au véhicule de queue de détecter le véhicule de tête ou à des mesures de distances relatives erronées. Ces cas sont visibles dans l'évolution de la distance euclidienne (figure 6.9 intervalle I_2 et I_3) mais comme l'erreur de localisation estimée grandit lorsque le véhicule n'est plus détecté, la distance de Mahalanobis dans cet intervalle reste sous les 6.15 donnés par la table du χ^2 (figure 6.10 intervalle I_2 et I_3).

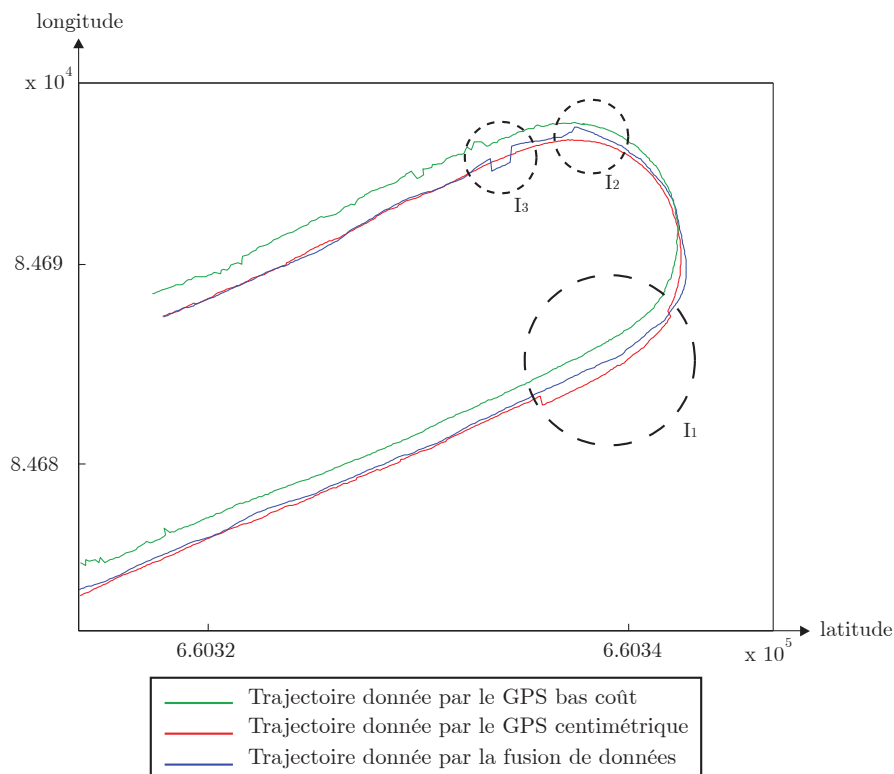


FIGURE 6.11 – Zoom sur les trajectoires résultantes

6.4.1.2 Trajectoire 2

La figure 6.12 représente les résultats de la localisation du véhicule de tête correspondant à la trajectoire 2. La courbe verte représente la trajectoire fournie par le GPS bas coût, la courbe rouge représente la trajectoire de référence fournie par le GPS centimétrique du véhicule de tête et la courbe bleue représente la trajectoire du véhicule de tête obtenue dans son état global fusionné. La figure 6.13 représente en rouge, la distance euclidienne entre la pose estimée après

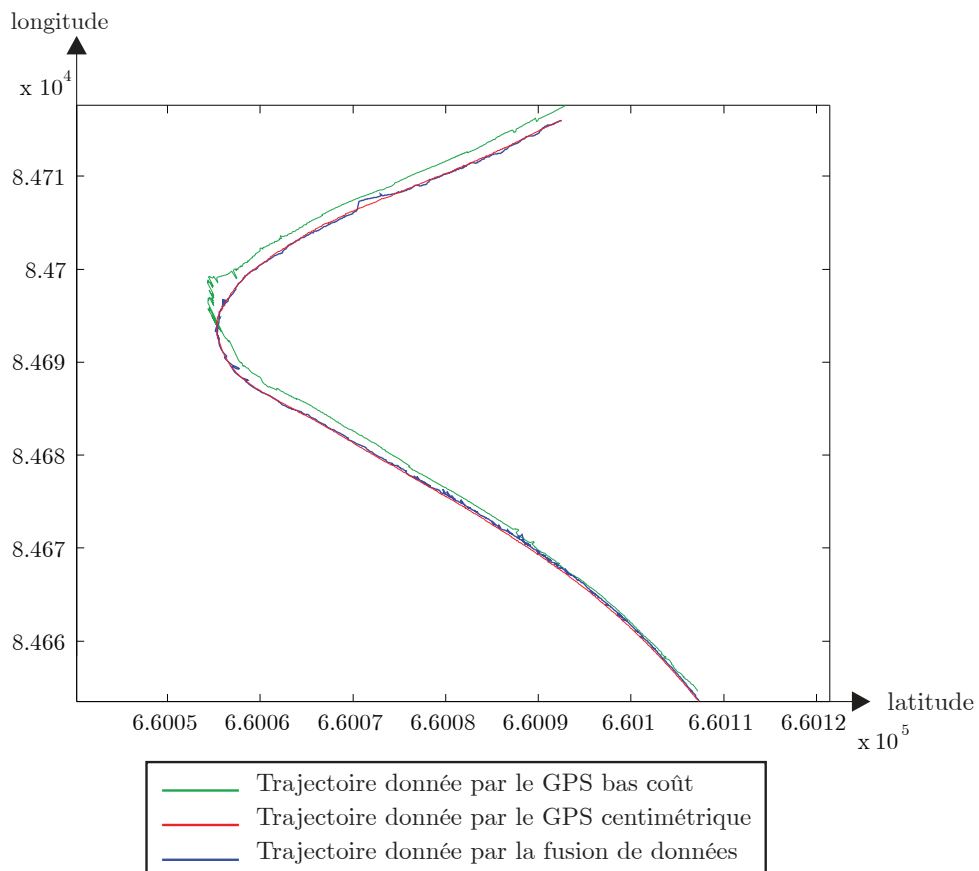


FIGURE 6.12 – Résultats de la localisation : Trajectoire 2.

la fusion et la trajectoire de référence donnée par le GPS centimétrique. Cette distance est comparée à titre indicatif à la distance euclidienne entre la position fournie par le GPS bas coût et la référence fournie par le GPS centimétrique (courbe verte).

Nous pouvons voir là aussi que l'erreur de localisation est comprise entre 20 et 50 centimètres. Le GPS bas coût avec lequel se localise le véhicule de tête fournit des positions à 140 centimètres en moyenne de la position de référence.

6.4 Résultats et discussions

La figure 6.14 présente l'évolution de la distance de Mahalanobis entre la pose estimée et la référence. Là aussi, nous pouvons voir que la distance reste en dessous des seuils définis par la table du χ^2 ce qui montre que l'algorithme de localisation coopérative reste consistant et la position vraie est toujours dans l'ellipse d'incertitude estimée.

La figure 6.15 présente un zoom de la figure 6.12 pour mettre en évidence l'effet de la non

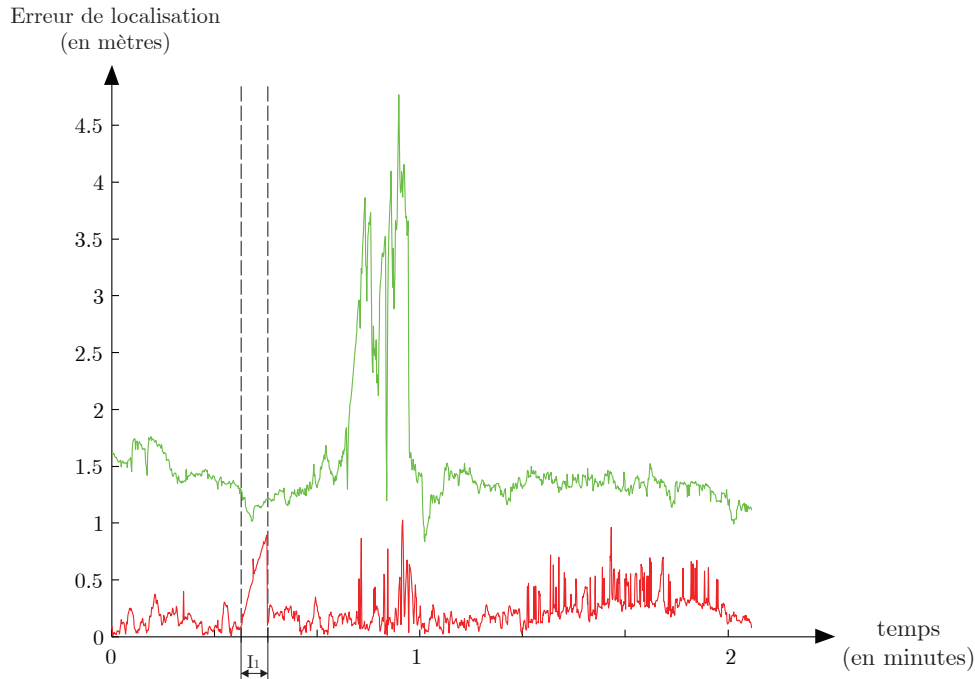


FIGURE 6.13 – Distance euclidienne entre la valeur moyenne de la position estimée après la fusion de états et la valeur la position de référence donnée par le GPS centimétrique.

détection du véhicule de tête par le véhicule de queue. Dans ce cas le véhicule de queue ne détecte plus le véhicule de tête et n'estime plus sa pose. Comme le véhicule de tête ne reçoit plus d'autre estimation de sa pose à part celle fournie par son GPS bas coût, sa position dans l'état global fusionné tend vers la position délivrée par le GPS. Ce cas est représenté dans la zone I_2 entourée dans la figure (6.15) et dans les intervalles I_2 des figures (6.13) et (6.14). Nous pouvons voir que lorsque la distance euclidienne augmente mais que la distance de Mahalanobis reste faible, la position vraie est maintenue dans l'ellipse d'incertitude car la variance associée à l'estimation de la pose augmente aussi.

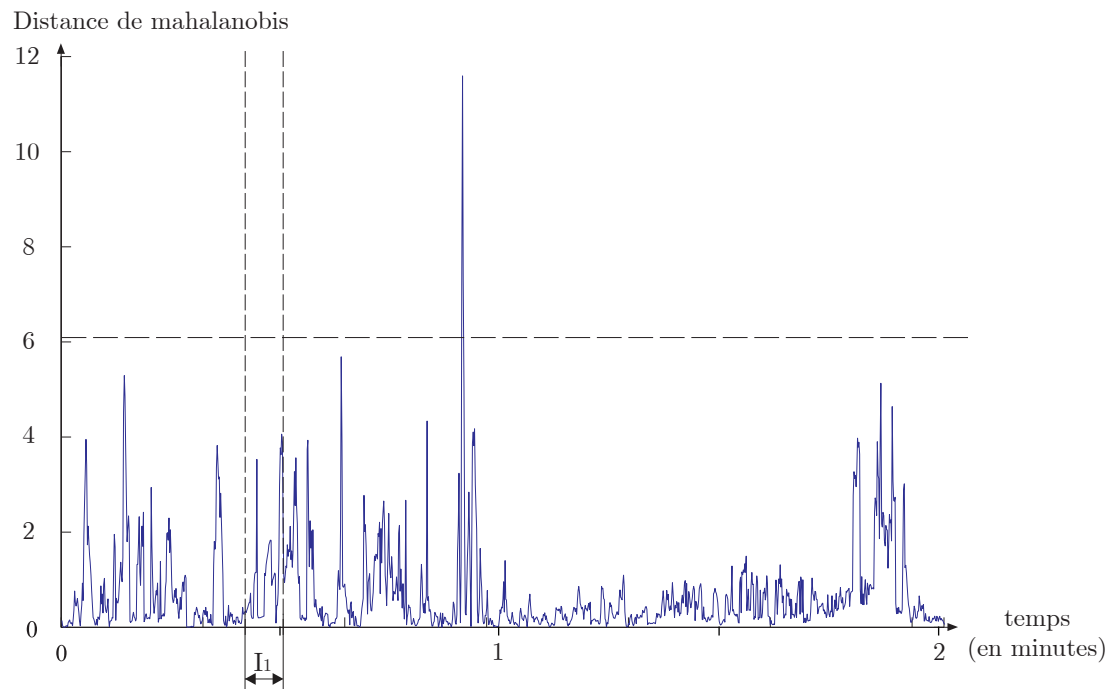


FIGURE 6.14 – Distance de Mahalanobis entre la pose estimée après la fusion et la pose fournie par le GPS centimétrique. La zone I_1

6.4 Résultats et discussions

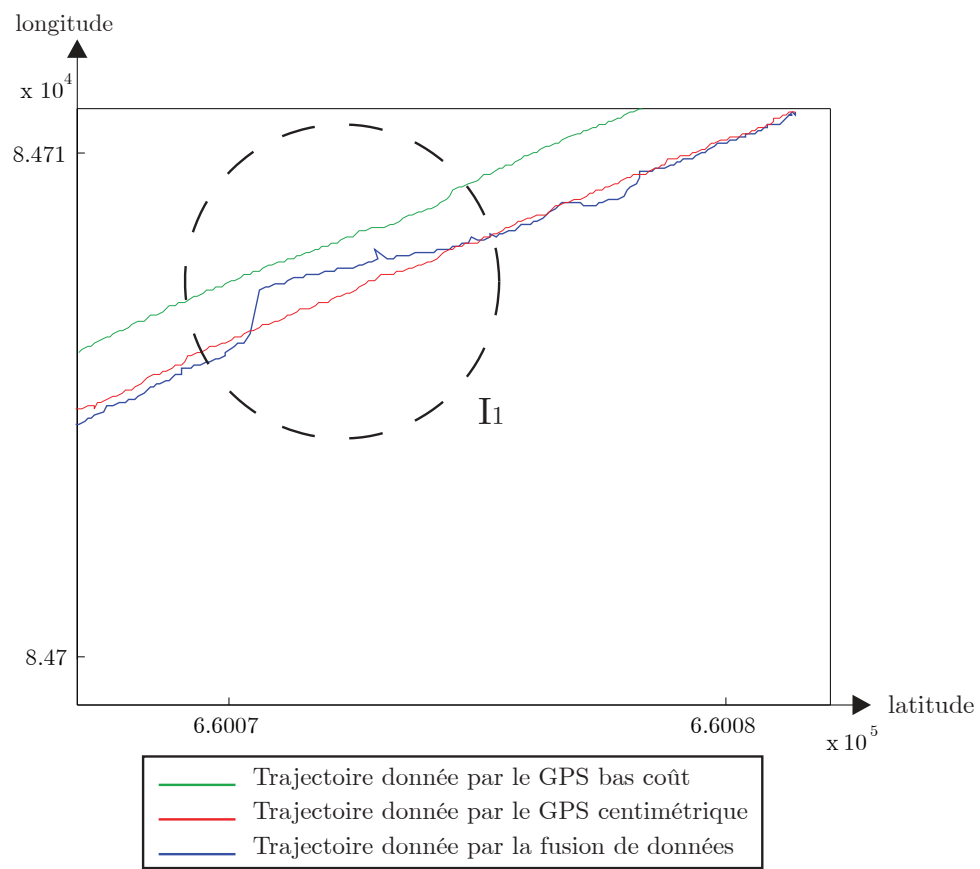


FIGURE 6.15 – Effet de la perte des informations sur poses relatives

6.5 Problèmes rencontrés

Les expérimentations réelles ont permis de mettre au jour des problèmes liés aux capteurs utilisés pour la localisation ou mesure des poses relatives.

L'approche de localisation coopérative proposée nécessite la détection et la mesure de la pose relative des autres robot. Cette tâche est difficile à réaliser dans un cas général. En effet, les robots n'ont pas toujours la même forme dans la réponse des capteurs ce qui rend la détection et le calcul de la pose relative difficiles. Pour éviter ce problème, nous avons opté pour une formation en convoi ce qui peut correspondre aussi à un cadre routier ou autoroutier. Dans ce type de formations, les véhicules sont toujours vus du sous le même angle. La forme de ces derniers dans la réponse d'un capteur de type télémètre laser est souvent la même. Ceci rend la détection et la mesure de la pose relative relativement simples à implémenter.

Le deuxième problème rencontré concerne le GPS bas coût et le biais donc souffrent les informations qu'il délivre. En effet les positions que ce capteur délivre sont non pas centrées sur la valeur vraie de la position du récepteur, mais sur une valeur biaisée. Comme l'évolution de ce biais est relativement lente, les mises à jour successives de la pose du robot avec le filtre de Kalman, engendre une convergence sur la pose biaisée avec une erreur estimée qui n'est pas intègre (i.e l'ellipse d'incertitude n'inclut pas la position vraie). Ce problème n'est pas traité de façon rigoureuse dans le travail présenté dans ce mémoire. Il est simplement contourné en interdisant la diminution de l'ellipse d'incertitude en dessous de l'ellipse fournie par la donnée brute du GPS. Cette stratégie permet de maintenir l'intégrité de l'estimation de la pose du véhicule de tête.

Nous présentons dans la conclusion de ce mémoire des solutions pouvant être envisagée pour résoudre le problème du biais GPS de manière plus rigoureuse.

6.6 Conclusion

Dans ce chapitre nous avons présenté les résultats de l'approche de localisation coopérative pas échange d'état dans le cas d'un groupe de deux véhicule qui se déplacent en formation de type convoi. Les deux véhicules sont équipés de capteurs proprioceptifs de type odomètres d'un système de positionnement absolu de type GPS et d'un système de communication pour échanger les états estimés. Afin de montrer l'apport de l'approche proposée, nous avons équipé le véhicule de queue d'un GPS centimétrique et le véhicule de tête d'un GPS bas coût qui donne une localisation avec une erreur de l'ordre de 9 mètres. Le véhicule de tête dispose aussi d'un

6.6 Conclusion

GPS centimétrique mais il n'est utilisé que pour avoir une vérité terrain. Le résultat de l'algorithme de localisation coopérative est montré principalement dans le robot de tête.

L'évolution de l'erreur de localisation du robot de tête après la fusion est présentée pour les deux scénarios. Cette erreur est exprimée par la distance euclidienne entre la position vraie délivrée par le GPS centimétrique et sa pose estimée après la fusion des états échangés. L'intégrité de la localisation obtenue est montrée par l'étude de l'évolution de la distance de Mahalanobis durant les expérimentations. Cette distance est calculée entre la position vraie donnée par le GPS centimétrique et l'estimation de la pose obtenue après la fusion des états.

L'étude de l'évolution de l'erreur de localisation montre que le robot de tête obtient une estimation de sa pose avec une précision de l'ordre de 20 centimètres. L'évolution de la distance de Mahalanobis montre que l'estimation de la pose du robot de tête reste intègre tout le long du déroulement des scénarios traités.

Conclusion

Le travail présenté dans ce mémoire est une contribution à l'étude et la mise en œuvre d'un système de localisation coopérative d'un groupe de robots mobiles. L'objectif d'un tel système est d'estimer, de manière la plus précise possible, les poses de tous les membres du groupe. Ceci est obtenu en combinant les informations issues des capteurs de chaque robot présent dans ce groupe.

Principales contributions

Nous avons développé une approche de localisation coopérative basée sur un échange d'état entre les robots (ou véhicules). Celle-ci permet de fusionner de manière optimale, les informations de localisation échangées parmi les membres du groupe. Le principe de cette approche est que chaque robot, estime l'état du groupe en fusionnant les données de ses propres capteurs. Dans cet état, la pose de ce robot, celle des congénères qu'il a détectés ainsi que celle d'éventuels obstacles sont représentées. Quand les robots du groupe communiquent, chaque membre envoie l'état du groupe qu'il a obtenu à ses congénères et reçoit celui construit par ces derniers. La fusion de l'état estimé en local et ceux reçus des autres membres permet d'obtenir un *état global fusionné*. Cet état combine toutes les informations collectées par tous les membres du groupe (tous les membres qui ont communiqué leurs états locaux).

L'idée clé de l'approche proposée est que l'*état global fusionné* est utilisé par les robots en local mais jamais communiqué à nouveau aux autres membres. En effet, après la fusion de données, l'*état global fusionné* contient les estimations optimales des poses de tous les robots. Cet état peut alors être utilisé en entrée pour des applications de plus haut niveau afin d'effectuer la tâche confiée au le robot. Contrairement aux approches classiques, l'état échangé à la communication suivante, est, non pas basé sur l'*état global fusionné*, mais sur l'état estimé en local échangé à la communication précédente. Il est ainsi, indépendant des états reçus des autres robots, et peut être transmis et fusionné sans problème de sur convergence.

Pour obtenir à tout moment, l'estimation optimale de l'état du groupe, les robots font évoluer en local leurs *états globaux fusionnés* en utilisant leurs capteurs proprioceptifs et les mettent à jour avec les informations extéroceptives de localisation ou de mesure de poses relatives. Cet état représente le résultat de la localisation coopérative et n'est destiné qu'à une utilisation interne à chaque robot. Compte tenu de l'application envisagée, les contraintes que nous nous sommes imposés sont compatibles avec une implémentation en milieu ouvert urbain ou routier. Dans ce type d'environnements le nombre de véhicules n'est pas connu à l'avance et change constamment. Par ailleurs l'identification des véhicules détectés est difficile, voire impossible à

obtenir selon les capteurs utilisés. Pour cela, l'approche proposée considère un groupe de taille variable et se propose de fusionner l'état estimé en local et les états reçus des congénères sans identification préalable des robots présents dans ces états. L'étape de fusion nécessite ainsi une étape de mise en correspondance qui permet de fusionner l'état des robots correctement. L'approche proposée permet aussi de traiter d'autres problèmes liés aux applications réelles tels que les latences qui affectent les données des capteurs ou des communications, la désynchronisation des données intégrées et les pertes éventuelles de communications.

L'approche de localisation coopérative par échange d'état a été implémentée et testée en simulation dans un premier temps, puis dans un cadre réel sur une flotte de véhicules expérimentaux.

Les résultats obtenus en simulation permettent de valider l'approche proposée dans un cas général. L'approche a été testée dans le cas de différents scénarios qui mettent en œuvre des groupes de différentes tailles. Le gain en précision de localisation dans l'*état global fusionné* dépend, non seulement du nombre de robots qui mesurent leurs poses relatives, mais aussi des précisions des capteurs et des algorithmes utilisés pour la mesure de ces poses relatives. Ce gain peut atteindre par exemple 60 % dans le cas de trois robots équipés de façon équivalente. La précision de la localisation dépend aussi du bon déroulement de l'étape de mise en correspondance des états à fusionner.

Les expérimentations réelles ont permis de montrer les résultats de la localisation coopérative dans le cas particulier de deux véhicules évoluant en formation de convoi. Les résultats obtenus permettent de démontrer l'apport d'une telle approche pour la localisation d'un robot évoluant dans un groupe.

Perspectives

Les résultats des expérimentations ont permis aussi de mettre en évidence des améliorations possibles à apporter à l'approche proposée. Certaines améliorations touchent l'approche en elle-même et d'autres concernent les algorithmes de prétraitement des données issues des capteurs utilisés.

Améliorations possibles pour l'approche de localisation coopérative : L'approche de localisation coopérative proposée, et particulièrement l'étape de mise en correspondance qui pré-

cède la fusion des états, suppose que tous les robots, expriment l'état qu'ils communiquent dans un repère commun à tous. Cette hypothèse est facilement réalisable avec l'utilisation d'un système de géolocalisation de type GPS, mais exclut du groupe les véhicules non équipés d'un tel système ce qui rend l'approche de localisation dépendante de ce type de capteurs.

En effet, dans le cas où un membre du groupe présente un problème de localisation absolue, les autres robots peuvent le détecter et mesurer sa pose. Quand ces robots échangent entre eux ces informations, ils peuvent les fusionner et obtenir un *état global fusionné* qui contient leurs poses mais aussi celle du robot qui a perdu son GPS. Cette pose a été mesurée par rapport aux autres véhicules. Quelle que soit la précision de cette pose, le robot concerné ne pourra jamais en bénéficier à cause du critère utilisé pour la mise en correspondance des poses dans les états à fusionner. Comme son état n'est pas exprimé dans le même repère que les autres, la fusion de son état local avec les états reçus lui est impossible dans l'état actuel de l'approche.

Une solution à ce problème serait d'exprimer l'état local de chaque robot du groupe dans le repère qui lui est propre. Dans ce repère, les erreurs sur les poses des autres robots ne dépendent que des erreurs de mesures relatives et de celles de l'évolution de ses poses. Après la communication des états calculés, une étape de mise en correspondance peut être envisagée en calculant la transformation (la translation et la rotation) entre les repères dans lesquels sont exprimées les poses des robots. Après cette étape, la transformation calculée peut être appliquée à l'état reçu afin de l'exprimer dans le même repère que l'état local et de pouvoir les fusionner.

Une telle approche n'est envisageable que si les états échangés sont riches en informations de poses et que le recouvrement entre les deux états est assez large pour pouvoir calculer la transformation entre les deux repères.

Améliorations possibles pour les algorithmes de prétraitement des données capteurs : La détection et la mesure précise des poses relatives des autres robots du groupe est une tâche primordiale pour le bon déroulement de l'approche proposée. Les expérimentations présentées dans ce mémoire sont faites dans le cas d'une formation de convoi. Ce choix a été motivé par nécessité de détecter les véhicules et de mesurer leurs poses relatives avec une précision satisfaisante pour la localisation coopérative. Dans le cas de la formation en convoi, la forme du véhicule de tête dans la réponse du capteur utilisé ne change pas ou peu durant l'application. L'algorithme de mesure de la pose relative est simple à implémenter mais n'est adapté qu'à ce type de formations.

Afin de tester l'approche proposée dans d'autres configurations, il est nécessaire de concevoir des briques de détection et de mesure de poses relatives plus performantes que celle implé-

mentée dans ce travail de thèse. Toutefois, un système qui détecte et mesure une pose relative d'un autre véhicule dans un cas général et sans connaissance a priori sur son gabarit reste un problème difficile à résoudre avec un seul type de capteurs. Une solution envisageable serait de combiner les informations de plusieurs capteurs afin de lever certaines ambiguïtés rencontrées avec un seul capteur. Par exemple, en détectant le coin d'un véhicule et ses deux cotés avec un télémètre laser, et en supposant que le véhicule est plus long que large, il est possible de mesurer son cap relatif à $\pm 180^\circ$. Cette ambiguïté peut être levée en utilisant une caméra vidéo par exemple.

Le deuxième problème rencontré est le celui du biais qui affecte les données de positionnement délivrées par le GPS bas coût. L'utilisation d'une donnée biaisée pour mettre à jour la position d'un robot avec un filtrage optimal engendre une convergence vers la valeur biaisée. L'estimation de la pose dans un tel cas n'est pas intègre (i.e. la position vraie n'est pas dans l'ellipse d'incertitude calculée). Dans le cas de l'approche proposée, ce biais peut engendrer des erreurs d'appariement qui peuvent perturber le bon fonctionnement de notre approche. Une solution à ce problème serait de calculer la position du récepteur GPS en utilisant les éphémérides des satellites et les pseudo-distances entre ces derniers et le récepteur. Si le calcul de la position se fait par rapport à un ensemble de satellites différent à chaque fois, il est fort probable que le biais change fortement d'un calcul de position à l'autre. Il reste à vérifier si les positions calculées sont centrées sur la vraie position du récepteur. Si cette hypothèse est vérifiée, la mise à jour de l'état d'un robot avec les poses calculées doit converger vers la vraie valeur de sa position.

Bibliographie

- [Auf01] 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.
- [BBC⁺02] 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.
- [BEF96] J. Borenstein, B. Everett, and L. Feng. Where am i ? Technical report, University of Michigan, Avril 1996.
- [BF96] J. Borenstein 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.
- [BL01] J. Benedicto and D. Ludwig. Galileo system and architecture. In *Digital Signal Processing Techniques for Space Communication Workshop*, Sessimbra, Portugal, 1-3 Octobre 2001.
- [CMR07] Andrew I. Comport, Ezio Malis, and Patrick Rives. Accurate quadrifocal tracking for robust 3D visual odometry. In *ICRA*, pages 40–45. IEEE, 2007.
- [DK02] G.N. Desouza and A.C. Kak. Vision for mobile robot navigation : a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2) :237–267, Février 2002.
- [Gal02] 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.

- [GLM04] D. Gruyer, A. Lambert, and B. Mourllion. Etat de l'art des stratégies de fusion de données pour la localisation. Livrable arcoss thème 3 t3.2, LIVIC(unité mixte INRET/LCPC), 2004.
- [Glo02] 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.
- [Hal97] 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.
- [HMS03a] A. Howard, M.J. Mataric, and G. Sukhatme. Putting the 'i' in 'team' : an ego-centric approach to cooperative localization. In *Proceedings, 2003 IEEE International Conference on Robotics and Automation*, 2003.
- [HMS03b] A. Howard, M.J. Mataric, and G.S. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation. In *Proceedings. 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems.*, pages 434–459, 2003.
- [HS96] C. Harris and M. Stephens. A combined corner and edge detector. pages 147–151, 1996.
- [JU02] S. J. Julier and J. K. Uhlmann. Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations. In *Proc. American Control Conf.*, volume 2, pages 887–892, 2002.
- [Kal60] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME, Journal of Basic Engineering*, 82 :34–45, 1960.
- [KB61] 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.
- [Kie99a] M. Kieffer. *Estimation ensembliste par analyse par intervalle : Application à la localisation de véhicules*. PhD thesis, Université de Paris Sud, 1999.
- [Kie99b] M. Kieffer. *Estimation ensembliste par analyse par intervalles : Application à la localisation de véhicules*. PhD thesis, Université de Paris sud, 1999.
- [KNH94] R. Kurazume, S. Nagata, and S. Hirose. Cooperative positioning with multiple robots. In *Proceedings, 1994 IEEE International Conference on Robotics and Automation*, volume 2, pages 1250–1257, 1994.

Bibliographie

- [Lan06a] J. Laneurit. *Perception multisensorielle pour la localisation d'un robot mobile en environnement extérieur, application aux véhicules routiers*. PhD thesis, Université Blaise Pascal, Clermont-Ferrand, France, Juillet 2006.
- [Lan06b] J. Laneurit. *Perception multisensorielle pour la localisation d'un robot mobile en environnement extérieur, application aux véhicules routiers*. PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, 2006.
- [LVH⁺05] Yuanqing Lin, Paul Vernaza, Jihun Ham, , and Daniel D. Lee. Cooperative relative robot localization with audible acoustic sensing. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages On DVD-ROM, 2005.
- [Mag65] D. T. Magill. Optimal adaptive estimation of sampled stochastic processes. *CONTROL*, AC-10 :434–439, Oct 1965.
- [MASD98] E. Mazor, A. Averbuch, Y. Bar Shalom, and J. Dayan. Interacting multiple model methods in target tracking : A survey. *IEEE Trans. Aerospace and Electronic Systems*, 34(1) :103–123, January 1998.
- [May79] P.S. Maybeck. *Stochastics models, estimation and control*. Academic Press, New York, USA, 1979.
- [MFP02] Raj Madhavan, Kingsley Fregene, and Lynne E. Parker. Distributed heterogeneous outdoor multi-robot localization. In *In Proceedings of the IEEE International Conference on Robotics and Automation*, pages 374–381, 2002.
- [MGSVM05] L. Montesano, J. Gaspar, J. Santos-Victor, and L. Montano. Cooperative localization by fusing vision-based bearing measurements and motion. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages On DVD-ROM, 2005.
- [MPS05] A. Martinelli, F. Pont, and R. Siegwart. Multi-robot localization using relative observations. In *Proceedings, 2005 IEEE International Conference on Robotics and Automation*, pages On DVD-ROM, 2005.
- [MSW02] Adam Milstein, Javier Nicolás Sánchez, and Evan Tang Williamson. Robust global localization using clustered particle filtering. In *In AAAI-02*, pages 581–586, 2002.
- [NB02] M.E El Najjar and P. Bonnifait. A road reduction method using multi-criteria fusion. In *IEEE Intelligent Vehicle Symposium*, Versailles, France, June 2002.
- [NPR98] M. Nørgaard, N. K. Poulsen, and O. Ravn. *Advances in derivative-free state estimation for nonlinear systems*, 1998. Revised edition, Oct 2004.

- [NRM09] Esha D. Nerurkar, Stergios I. Roumeliotis, and Agostino Martinelli. Distributed maximum a posteriori estimation for multi-robot cooperative localization. In *Proceedings, 2009 IEEE International Conference on Robotics and Automation*, pages On DVD–ROM, 2009.
- [Ous98] 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.
- [RB00] S.I. Roumeliotis and G.A. Bekey. Collective localization : A distributed kalman filter approach to localization of groups of mobile robots. In *Proceedings, 2000 IEEE International Conference on Robotics and Automation*, pages 2958–2965, 2000.
- [RB02] S.I. Roumeliotis and G.A. Bekey. Distributed Multi-Robot Localization. *IEEE Transactions on Robotics and Automation*, 18(5) :781–795, 2002.
- [RLDL07] Eric Royer, Maxime Lhuillier, Michel Dhome, and Jean-Marc Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, 74(3) :237–260, 2007.
- [Roy02] 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 compiègne, 2002.
- [RR03] S.I. Roumeliotis and I.M. Rekleitis. Analysis of multirobot localization uncertainty propagation. In *Proceedings. 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems.*, pages 1763– 1770 vol.2, 2003.
- [SKL05] E. Seigneze, 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.
- [SL05] E. Seigneze and A. Lambert. Experimental vehicle localization by bounded-error state estimation using interval analysis. In *Proceedings, 2005 IEEE International Conference on Robotics and Automation*, pages on DVD–ROM, 2005.
- [SSN⁺] K. Shaikh, R. Sharif, F. Nagi, H. Jamaluddin, and S. Mansor. Inertial navigation sensors, data processing and integration with gps for mobile mapping.
- [TaDS01] R. Thrapp and C. Westbrook ans D. Subramanian. Robust localization algorithms for an autonomous campus tour guide. In *IEEE International Conference on Robotics and Automation*, pages 2065–2071, Mai 2001.

Bibliographie

- [TCD⁺06] C. Tessier, C. Cariou, C. Debain, R. Chapuis, F. Chausse, and C. Rousset. A real-time, multi-sensor architecture for fusion of delayed observations : Application to vehicle localisation. In *Proceedings, 9th International IEEE Conference on Intelligent Transportation Systems*, pages On DVD-ROM, 2006.
- [Tes07] C. Tessier. *Système de localisation basé sur une stratégie de perception cognitive appliqué à la navigation autonome d'un robot mobile*. PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, 2007.
- [Thu02] J. Thurston. Galileo glonass and navstar : A report on gps for gis people. <http://www.gisvisionmag.com>, 2002.
- [Vag93] 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.

Résumé

Les avancées technologiques notables dans le domaine de la communication sans fil ont ouvert de nouvelles perspectives pour améliorer les systèmes d'aide à la conduite existants. En effet il a été prouvé que l'échange de données de positionnement absolu et relatif entre des véhicules d'un même groupe peut améliorer la précision de la localisation de tous ces membres. Ce ci est appelé localisation coopérative.

Le travail présenté dans ce manuscrit s'inscrit dans ce cadre et propose une approche totalement distribuée permettant de combiner les informations provenant des capteurs de tous les véhicules d'un groupe pour obtenir l'estimation la plus précise possible de la pose de ces derniers tout en minimisant les informations transmises sur le réseau de communication.

Le principe de l'approche proposée est que chaque véhicule utilise ses propres capteurs pour mettre à jour une estimation de l'état du groupe puis la transmet à ses congénères. La fusion des états reçus et l'état construit dans chaque véhicule donne l'estimation de l'état global du groupe qui intègre les informations provenant de tous ses membres.

L'approche proposée a été validée expérimentalement en simulation et dans un environnement réel sur des véhicules urbains.

Mots clés: Localisation multivehicules, fusion de données, filtrage de KALMAN, communication véhicule à véhicule, système embarqué, temps réel.

Abstract

The technological advances in wireless communication opened up new perspectives to enhance the existing driver assistance systems. It has been shown that the exchange of absolute and relative positioning information between the vehicles of the same group improves the localization precision of the whole group. This is called cooperative localization.

The presented work is part of this framework and introduces a distributed approach which combines the data provided by the sensors of all the group vehicles to obtain the most accurate estimation of their pose while minimizing the quantity of exchanged data over the communication network.

The principle of the proposed approach is that each vehicle uses its own sensors to maintain an estimation of the group state and sends it to the other vehicles. The fusion of the received states with the built one gives the estimation of the global state of the group which summarizes the sensor data coming from all its members.

The proposed approach has been tested and validated by simulations and real experiments with a group of road vehicles.

Key words: Multivehicle localization, data fusion, KALMAN filtering, vehicle to vehicle communication, embedded system, real time.