N^o d'ordre: D.U : 2082
EDSPIC : 503

THÈSE

présentée

DEVANT L'UNIVERSITÉ BLAISE PASCAL - CLERMONT II

pour obtenir

le grade de : *DOCTEUR DE L'UNIVERSITÉ BLAISE PASCAL*
Mention: électronique et Systêmes

par

Pierre AVANZINI

équipe d'accueil : GRAVIR, groupe Rosace (LASMEA, UMR 6602 CNRS)
école Doctorale : Sciences Pour l'Ingénieur

Titre de la thèse :

Modélisation et Commande d'un Convoi de Véhicules Urbains par Vision

Soutenue publiquement le 6 décembre 2010

COMPOSITION DU JURY

M.	Jean-Pierre DERUTIN	Président
M.	Dominique MEIZEL	Rapporteur
M.	Thierry FRAICHARD	Rapporteur
M.	Javier IBANEZ-GUZMAN	Examinateur
M.	Benoît THUILOT	Examinateur
M.	Philippe MARTINET	Directeur de thèse

Table des matières

Table des matières **3**

Remerciements **9**

Introduction **11**

1 Bibliographie et travaux précurseurs **21**
1.1 Nouvelles pratiques et systèmes de transport 21
1.1.1 Véhicules urbains privés 22
1.1.2 Le co-voiturage 25
1.1.3 De nouveaux concepts et services 26
1.1.4 Applications de services de navigation en milieu urbain 29
1.1.5 Applications de services de navigation dans un milieu non-urbain 39
1.2 Commande de robots mobiles en formation 41
1.2.1 Navigation en convoi et fonctionnalités associées 42
1.2.2 Notion de stabilité du convoi 43
1.2.3 Stratégies de commande locales 48
1.2.4 Stratégies de commande globales 56
1.2.5 Bilan 62
1.3 Expérience capitalisée avant la thèse 64
1.3.1 Navigation de véhicules en convoi 64
1.3.2 Bilan : 72

2 Génération de la trajectoire de référence **79**
2.1 Introduction 79
2.2 Étude bibliographique 79
2.2.1 Génération de trajectoire hors-ligne 80
2.2.2 Génération et modification en ligne de trajectoire 87
2.3 Génération de la trajectoire 90
2.3.1 Représentation de la trajectoire 90
2.3.2 Optimisation de la trajectoire dans le cas hors ligne 93
2.3.3 Génération en ligne de la trajectoire de référence 95
2.3.4 Détails sur l'implémentation de l'algorithme 98
2.4 Résultats de simulation pour la génération en ligne de la trajectoire . . . 104

2.4.1 Création en ligne de la trajectoire . . . 104
2.4.2 Critères de qualité de la trajectoire . . . 104
2.4.3 Analyse des paramètres . . . 106
2.4.4 Recherche d'un indicateur sur la qualité de la trajectoire . . . 110
2.5 Conclusion . . . 114

3 Localisation par vision monoculaire pour la conduite en convoi 117
3.1 Introduction . . . 117
3.2 État de l'art : localisation d'un véhicule par vision, estimation du facteur d'échelle . . . 118
3.2.1 Localisation partielle . . . 118
3.2.2 Localisation absolue . . . 120
3.3 Analyse des distorsions . . . 131
3.3.1 Analyse empirique . . . 131
3.3.2 Influence des distorsions sur les performances des lois de commande 135
3.3.3 Évaluation hors-ligne des facteurs d'échelle . . . 140
3.4 Stratégie de correction en-ligne : approche télémétrique . . . 145
3.4.1 Suivi du véhicule de tête . . . 145
3.4.2 Evaluation des facteurs d'échelle . . . 147
3.5 Stratégie de correction en-ligne : approche odométrique . . . 151
3.5.1 Modèles d'évolution : monde réel / virtuel . . . 151
3.5.2 Principe de la méthode . . . 152
3.5.3 Estimation de la vitesse curviligne dans le monde vision . . . 154
3.6 Simulations . . . 157
3.6.1 Optimisation locale avec les données télémétriques . . . 158
3.6.2 Estimation à partir de données odométriques . . . 161
3.7 Conclusion . . . 164

4 Expérimentations 165
4.1 Contexte expérimental . . . 165
4.1.1 Plateforme expérimentale : PAVIN . . . 166
4.1.2 Véhicules expérimentaux : CyCabs et RobuCabs . . . 166
4.1.3 Capteurs utilisés . . . 167
4.1.4 Communication sans fil . . . 169
4.1.5 Architecture informatique du convoi . . . 170
4.2 Cohérence temporelle et synchronisation des informations du convoi . . . 171
4.2.1 Acquisition asynchrone des données de commande . . . 171
4.2.2 Mise à jour de l'état du convoi . . . 172
4.3 Résultats expérimentaux . . . 175
4.3.1 Guidage manuel du convoi : localisation par récepteur RTK-GPS 175
4.3.2 Guidage du convoi en mode tout automatique : localisation par vision monoculaire . . . 181

5 Conclusion 191

Annexes **195**

A Eléments de géométrie projective et épipolaire pour la reconstruction 3D à partir d'une séquence d'images **197**
A.1 Modèle sténopé d'une caméra 198
A.2 Géométrie épipolaire 199
A.2.1 Matrice essentielle 199
A.2.2 Quelques algorithmes 200

B Localisation par vision monoculaire pour la navigation autonome **205**
B.1 Reconstruction 3D de la séquence de référence 205
B.1.1 Mise en place de l'algorithme et prérequis 206
B.1.2 Etapes de la reconstruction 206
B.2 Localisation en temps réel 209
B.3 Résultats 210
B.3.1 Précision sur la position 211
B.3.2 Précision sur l'orientation 215
B.3.3 Temps de calcul et espace mémoire 215

C Suivi des véhicules expérimentaux grâce à des données télémétriques **217**
C.1 Méthodologie 217
C.1.1 Définition de la zone d'intérêt 218
C.1.2 Localisation d'un RobuCab 218
C.1.3 Localisation d'un CyCab 219
C.2 Performances et limitations 222
C.2.1 Protocole de simulation 223
C.2.2 Résultats obtenus 225

Résumé

Cette thèse concerne la commande d'un convoi de véhicules avec l'objectif sociétal de réduire les problèmes de pollution et d'engorgement dans les milieux urbains. La recherche se concentre ici sur la navigation coopérative d'une flotte de véhicules communicants en s'appuyant sur une approche de commande globale : chaque véhicule est contrôlé à partir d'informations partagées par l'ensemble de la flotte, en s'appuyant sur des techniques de linéarisation exacte. Le développement de nouvelles fonctionnalités de navigation fait l'objet des deux contributions théoriques développées dans ce manuscrit et leur mise en œuvre constitue la contribution pratique.

Dans un premier temps, on s'intéresse à la mise en place d'un mode manuel de navigation dans lequel le premier véhicule, guidé par un opérateur, définit et retransmet la trajectoire à suivre aux membres du convoi. Il convient que la trajectoire, dont la représentation évolue au fur et à mesure de l'avancée du véhicule de tête, soit numériquement stable afin que les véhicules suiveurs qui sont asservis dessus puissent être contrôlés avec précision et sans subir de perturbations. A ces fins, la trajectoire a été modélisée par des courbes B-Spline et un algorithme itératif a été développé pour étendre cette dernière selon un critère d'optimisation évalué au regard des positions successives occupées par le véhicule de tête. Une analyse paramétrique a finalement permis d'aboutir à une synthèse optimale de la trajectoire en terme de fidélité et de stabilité de la représentation.

Dans un second temps, on considère l'intégration d'une stratégie de localisation par vision monoculaire pour la navigation en convoi. L'approche repose sur une cartographie 3D de l'environnement préalablement construite à partir d'une séquence vidéo. Cependant, un tel monde virtuel comporte des distorsions locales par rapport au monde réel, ce qui affecte les performances des lois de commande en convoi. Une analyse des distorsions a permis de démontrer qu'il était possible de recouvrer des performances de navigation satisfaisantes à partir d'un jeu de facteurs d'échelle estimés localement le long de la trajectoire de référence. Plusieurs stratégies ont alors été élaborées pour estimer en-ligne ces facteurs d'échelle, soit à partir de données odométriques alimentant un observateur, soit à partir de données télémétriques intégrées dans un processus d'optimisation. Comme précédemment, l'influence des paramètres a été évaluée afin de mettre en évidence les meilleures configurations à utiliser en vue d'applications expérimentales.

Pour finir, les algorithmes développés précédemment ont été mis en œuvre lors d'expérimentations et ont permis d'obtenir des démonstrateurs en vraie grandeur comprenant jusqu'à quatre véhicules de type CyCab et RobuCab. Une attention particulière a été accordée à la cohérence temporelle des données. Celles-ci sont collectées de manière asynchrone au sein du convoi. L'utilisation du protocole NTP a permis de synchroniser l'horloge des véhicules et le middleware AROCCAM d'estampiller les données et de gérer le cadencement de la commande. Ainsi, le modèle d'évolution des véhicules a pu être intégré afin de disposer d'une estimation précise de l'état du convoi à l'instant où la commande est évaluée.

Abstract

Vehicle platooning is addressed in this thesis with the aim of contributing in reducing congestion and pollution in urban areas. The researches here focus on cooperative navigation of a fleet of communicating vehicles and relies on a global control approach : each vehicle is controlled from information shared by the entire fleet, using exact linearization techniques. This manuscript presents two theoretical contributions, introducing two new navigation functionalities, and the third contribution consists in their practical implementation.

As a first part, the introduction of a manual navigation mode has been investigated, in which the first vehicle, guided by an operator, defines and broadcasts the path to be followed by platoon members. In that case, the numerical representation of the trajectory must be extended without disturbing the portion previously generated, to ensure that the entire platoon can precisely and smoothly follow the leader track. To meet these requirements, the trajectory is modeled using B-Spline curves and an iterative path creating algorithm has been developed from successive positions collected by the lead vehicle during its motion. A parametric analysis has finally resulted in the design of an optimal trajectory with respect to the fidelity of the path representation and to the robustness regarding the disturbances that could arise during the creation procedure.

In a second part, a localization strategy relying on monocular vision has been integrated in platoon control algorithms. The localization approach is based on a learning phase during which a video sequence is used to perform a 3D mapping of the environment. However, such a virtual vision world is slightly distorted with respect to the actual metric one. This affects the performances of the platoon control laws. An analysis of distortion has demonstrated that platooning performances can be recovered, provided if a set of local scale factors attached to the reference trajectory is available. Several strategies have then been designed to estimate online such scale factors, either from an observer relying on odometric data, or from an optimization process based on telemetric data. As before, intensive simulations has been run to evaluate parameter influence and highlight the best configurations to use during experiments.

Finally, above mentionned algorithms have been implemented in order to present full-scale demonstrators with up to four vehicles (either CyCab or RobuCab). Particular attention is data temporal consistency : since data are collected asynchronously within the platoon, a NTP has been used to synchronize vehicle clock, so that middleware AROCCAM can stamp vehicle data and manage the control scheduling. Thus, vehicle models can be integrated in order to obtain an accurate estimate of the platoon state at the time the control laws are evaluated.

Remerciements

Ces travaux ont été réalisés au LASMEA (LAboratoire des Sciences et Matériaux pour l'Électronique et d'Automatique). J'adresse donc naturellement mes remerciements aux acteurs de cet établissement qui m'a offert un cadre de recherche confortable et des moyens techniques de grande qualité.

Je tiens à exprimer mes remerciements à Dominique Meizel, Thierry Fraichard et Javier Ibanez-Guzman qui ont eu l'amabilité d'être respectivement rapporteurs et examinateur de cette thèse.

Je voudrais exprimer ma gratitude à Philippe Martinet pour m'avoir proposé cette thèse et ouvert des perspectives de travail intéressantes. Il a su notamment partager sa vision exigente de la Recherche et m'a toujours incité à maintenir mes investigations au meilleur niveau international.

Je tiens à remercier chaleureusement Benoît Thuilot qui m'a encadré au quotidien. Grâce à ses conseils avisés, son extrême minutie et son perfectionnisme, il a su guider mes pas aux moments opportuns et j'ai pu en sa compagnie développer mes capacités de rigueur et d'analyse.

Grand merci à Jean-Pierre Dérutin pour avoir présidé ma thèse et surtout pour ses encouragements, son soutien et sa confiance.

Je voudrais également remercier l'ensemble des membres de l'équipe ROSACE (ROboticS and Autonomous Complex systEms) pour leur accueil, les échanges scientifiques et leur amabilité.

Je tiens également à exprimer ma reconnaissance à l'équipe technique, notamment François Marmoiton, Serge Alizon et Laurent Malaterre, qui ont été d'un grand soutien pour la mise en place des expérimentations et pour leur contribution à l'ambiance chaleureuse qui règne au LASMEA.

J'adresse également mes meilleurs sentiments aux doctorants du LASMEA avec qui j'ai passé d'agréables moments. J'ai une pensée particulière pour Lolo, Ben, Yoyo, Fabio, Johnny, Tej, François et Laëti, Réda et Hala, Ahmed, Jerem, Nico, Franky, Richie, le vieux Seb et Redwan qui ont rendu cette aventure très réjouissante.

Enfin un merci tout particulier à ma famille et à Nejwa, mon grand bonheur.

Introduction

Quand nous parlons déplacements, nous parlons forcément voiture, dès lors que plus de 80% du kilométrage parcouru en France l'est en automobile. Cet engouement se témoigne aussi par la constante augmentation du nombre de véhicules en circulation. Le parc automobile en France a ainsi évolué, passant de 18 millions en 1980 pour atteindre près de 31 millions en 2009[1]. Malgré la récente crise du secteur de l'automobile, celle-ci est toujours considérée comme un élément incontournable dans la vie de tous les jours. Et d'ailleurs si cela nous touche tant, "c'est aussi parce que nous entretenons avec la voiture une relation singulière", rappelle Courrier International[2]. Cette relation est irriguée d'imaginaires collectifs, véhiculés par un siècle de pubs et de road-movies célébrant la voiture-liberté. Dans l'esprit d'une majorité, la voiture c'est avant tout un formidable objet technologique qui évolue en permanence, indispensable pour travailler et se déplacer et qui fournit directement ou indirectement un emploi à des millions de personnes grâce à l'industrie automobile.

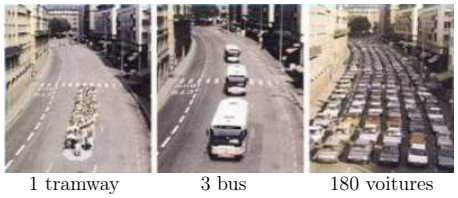
Cependant, ces dernières années la voiture est sujette à controverses et son image se ternit. Pour les détracteurs, les infrastructures routières engloutissent tous les ans d'importants investissements, défigurent les paysages, dénaturalisent le territoire et menacent les écosystèmes. L'impact environnemental le plus connu est la pollution atmosphérique due aux gaz d'échappements, qui cause des maladies respiratoires et contribue au réchauffement de la planète. Selon l'agence française de sécurité sanitaire (AFSSET), la pollution atmosphérique, liée pour près d'un tiers aux rejets polluants des voitures, serait responsable chaque année du décès de 6500 à 9500 personnes en France. Pour certains, l'automobile est également source de pollution sociale, les plus pauvres et les plus fragiles étant les plus directement touchés par les nuisances liées à l'automobile et aux infrastructures routières (bruits, accidents, détours imposés aux piétons et aux cyclistes, relégation urbaine, etc.).

Ces inconvénients sont d'autant plus notoires en milieu urbain. En effet, plusieurs caractéristiques propres au contexte urbain ne sont pas propices à l'usage de la voiture. De par sa forte concentration dans ces milieux, l'automobile est devenue le principal responsable, avec les appareils de chauffage domestique, des smogs urbains (brume épaisse),

[1] estimation du Comité des Constructeurs Français d'Automobiles (CCFA)
[2] dossier automobile de l'hebdomadaire n° 975 du 9 juillet 2009

situation chronique dans les capitales asiatiques. D'autre part, alors que l'espace est un bien particulièrement rare et coûteux en ville, les transports consomment un espace important et tout particulièrement l'automobile, comme le montre l'illustration ci-dessous. L'automobile est partout, 80% de l'espace public lui est consacré. Ainsi, quelques 160

"Comment transporter 240 personnes ?"
tramway, bus où véhicules personnels ?


1 tramway 3 bus 180 voitures

hectares sont "artificialisés" chaque jour en France. Autre chiffre tout aussi impressionnant : on a mesuré entre 1992 et 2004, 20% d'extension des surfaces en France pour 6% de population en plus. Cet espace est d'autant plus gaspillé au vu des nombreuses places de parkings ou ouvrages toujours vides (un bon tiers des places de la Défense, par exemple), les voiries surdimensionnées (files de circulation inutiles, maxis giratoires...) ou les voiries secondaires quasi-désertes. Dans la plupart des grandes agglomérations la réalité est très éloignée des plaquettes de présentation des projets urbains qui laissent espérer une cohabitation harmonieuse entre chacun et où l'automobile est étrangement absente.

Il faut aussi mentionner que la voiture est très préjudiciable lorsqu'elle engendre des embouteillages. Ces derniers sont avant tout synonymes de pertes de temps et de productivité. Ces engorgements entraînent néanmoins aussi des coûts supplémentaires, tels que ceux liés à l'exploitation et aux atteintes à l'environnement.

Voici quelques brèves illustrant les précédents propos :

- Au rythme actuel, le trafic d'heure de pointe à Sydney passera de 83000 autos (en 2008) à 250000 en 2013 [3].

- Enfin, le dernier chic à Sao Paulo, paralysé par les embouteillages, est de prendre l'hélicoptère. En moyenne, 500 d'entre eux sillonnent le ciel en permanence [4].

Enfin, ces problèmes de mobilité dans les mégalopoles sont tellement préoccupants que même de grands journaux exploitent ce sujet sans vérifier leurs sources et colportent d'énormes canulars :

- Un embouteillage géant de plus de 100km a bloqué une autoroute du nord de la Chine pendant 10 jours [5].

Les alternatives existent. Sur de courts trajets et en terrain urbain, le vélo constitue un moyen de locomotion aussi efficace que la voiture, tout en présentant des atouts

[3] journal *The Sydney Herald* paru le 21 juin 2008
[4] journal en ligne *The Daily Planet* paru le 22 mars 2008
[5] journal *Libération* paru le 25 août 2010

supplémentaires. Mais son utilisation reste réservée à des parcours bien choisis. Notre culture vélo s'arrête souvent à des promenades sur pistes cyclables et petites routes de campagne. Quand il s'agit d'aller au travail, faire les courses ou accompagner les enfants à l'école, le temps mis pour le trajet et la fatigue endurée décourage les citadins. A cela s'ajoute les dangers de la circulation en ville et une utilisation qui n'est pas toujours pratique dans les situations où il y a des intempéries ou des biens à transporter. A ces difficultés, les services publics offrent une solution par l'usage des transports en commun, adaptés à la ville, moins stressants, désengorgeant le trafic et réduisant les pollutions sonores et atmosphériques. Pourtant, lorsque l'on peut choisir entre les transports en commun et la voiture individuelle, cette dernière est presque toujours privilégiée. La liberté de déplacement (en horaire et en itinéraire) qu'octroie la voiture personnelle la rend en effet plus souple que les transports collectifs. A l'exception de l'hyper-centre où les contraintes de stationnement et de circulation sont très élevées, l'automobile est plus performante. Au demeurant, les transports publics ne peuvent répondre à tous les besoins, notamment pour tout transport de charges ne serait-ce que modérément lourdes ou volumineuses ou le déplacement pour les personnes âgées.

Pour finir cette introduction générale, j'aimerais citer Bruno Marzloff, sociologue à la direction de la plateforme d'échange du groupe Chronos[6], qui réunit des acteurs des transports, des médias et de la Ville. Lors d'une conférence au journal Le Monde (19 novembre 2009) où étaient présents les acteurs historiques et les nouveaux opérateurs de mobilité, il a dévoilé, selon lui, les quatre clés qui façonnent le nouveau paradigme des mobilités :

> *- le numérique, en atteste l'iPhone présent dans quasiment toutes les interventions. Le numérique, c'est aussi l'open source et l'interopérabilité des données.*
>
> *- les régulations. Le partage de la donnée publique amorce la boucle vertueuse. Les contributions des usagers font partie de la "donnée publique", complétées des traces (des passes Navigo, des véhicules, des mobiles) qu'il faut libérer.*
>
> *- le partage. Malgré le vandalisme, Vélib a montré que l'on était arrivé à maturité sur la question des partages. Il a eu le rôle de révélateur. Autolib*[7] *pourrait avoir le même impact.*
>
> *- les modèles. Les études de TNS Sofres*[8] *soulignent l'inertie des comportements. Comment en sortir ? Encore une fois, Vélib a agi comme catalyseur de la pratique du vélo urbain parce que l'offre a révélé la demande.*

[6] Chronos est un cabinet d'études sociologiques et de conseil en innovation qui observe, interroge et analyse l'évolution et les enjeux des mobilités.

[7] projet de voitures électriques en libre service qui concernent Lyon et Paris. À Paris, Autolib est pour l'instant un projet du maire Bertrand Delanoë, avec une mise en place prévue pour septembre 2011

[8] leader français des études marketing et d'opinion

Les deux premiers points qui viennent d'être cités présagent l'irruption du numérique et des technologies de l'information dans les nouveaux moyens de transport : les réponses à la mobilité urbaine de demain sont basées sur des systèmes de transports intelligents qui s'adaptent, interagissent et communiquent entre eux et proposent une interface avec les utilisateurs. Par ailleurs, un concept intéressant pour adresser les problèmes des transports urbains repose sur les services de partage, dont le succès commercial dépend de leur adaptation aux besoins émergents de la société.

Ce qu'il faut retenir ici, c'est que la voiture fut le choix de la société du siècle dernier ; aujourd'hui elle cherche sa place dans un environnement exigeant et il est normal à l'heure actuelle que l'on s'interroge sur nos systèmes de transport et leur place dans la maîtrise de la mobilité. Puisque la technologie ne nous donne pas encore les clés à un système de téléportation qui dispenserait de voyager, de nouvelles alternatives de transport sont nécessaires, tirant le maximum d'avantages et réduisant les désagréments des moyens de transport traditionnels.

Les véhicules serviciels

Le nouveau type de transport se situerait donc, au croisement des transports en commun public (bus, tramways, métros), des véhicules personnels et des déplacements pédestres, sans venir toutefois remplacer ces différentes entités, mais plutôt en apportant une complémentarité aux services actuels.

L'objectif principal de ce dernier peut donc se résumer à concevoir un moyen de transport pouvant contenir biens et objets encombrants, disponible 24h/24h, non polluant, s'intégrant pleinement dans le réseau de transport urbain actuel, modulable en fonction du nombre de passagers et offrant la même flexibilité que la voiture.

Une solution envisagée consiste à mettre à disposition des usagers des transports en commun (ou des utilisateurs de voiture personnelle) des véhicules urbains électriques en libre accès en des stations situées près des arrêts des transports publics (ou dans un parking à l'entrée de la ville, à disposition des utilisateurs de voitures personnelles désirant se rendre en ville). Ces véhicules électriques (donc non polluants), plus petits que les voitures traditionnelles, seraient donc une ressource utilisée par un individu pour se rendre d'une station à une autre station, le temps du trajet. Ensuite, arrivé à la station de destination, le passager libère son véhicule, qui pourrait alors être utilisé par un autre éventuel utilisateur. Grâce à ce système, l'usager dispose d'un véhicule, est alors en mesure de circuler en toute autonomie et ne perd pas de temps pour se garer (le véhicule étant remis immédiatement à disposition).

Dans plusieurs villes pionnières à travers le monde, de tels systèmes ont été mis en place afin d'évaluer leurs bénéfices. En France, l'opération Lisélec lancée à La Rochelle en septembre 1999, comprend 50 véhicules électriques, répartis par des agents d'exploitation sur 7 stations situées en des points stratégiques de l'agglomération rochelaise (centre ville, gare, campus,...), accessibles 24h/24, 7j/7, pour les personnes qui

ont souscrit un abonnement. La croissance constante des abonnements sur le site de La Rochelle atteste d'un réel engouement du public pour le système, notamment grâce à la flexibilité renforcée du système.

Ensuite, une fonctionnalité des plus intéressantes et pratiques serait de permettre aux véhicules de circuler de manière totalement autonome sans la moindre intervention humaine. Ainsi, tout utilisateur du véhicule n'aurait plus besoin de conduire, mais se laisserait guider, sans intervention et en toute sécurité, en un point de destination spécifié. En plus du transport de passager sans conducteur, cette fonctionnalité permettrait également au véhicule automatisé de se rendre d'un point à un autre sans passager. Dès lors, il serait envisageable d'imaginer le véhicule automatique, libre de passager, se rendant automatiquement à un point de rendez-vous fixé par un usager (pouvant être donc autre qu'une station). Arrivé au point de rendez-vous, l'utilisateur monterait dans le véhicule pour se rendre à destination voulue (manuellement ou automatiquement au gré du passager). Une fois que l'usager a quitté le véhicule à destination, le véhicule libre d'utilisateur rejoindrait une station pour maintenance ou rechargement des batteries, ou bien se rendrait directement en un nouveau point de rendez-vous fixé par un nouvel utilisateur.

Par la suite, afin d'augmenter le flux de circulation, plutôt que d'utiliser plusieurs véhicules fonctionnant indépendamment, il serait envisageable d'optimiser le fonctionnement des véhicules par leur marche en convoi (aussi bien pour le transport de passagers que pour le rapatriement des voitures urbaines vers une station). En effet, en imposant un faible écart entre les membres du convoi, les capacités de circulation sont accrues et cela permet de fluidifier le trafic. De plus, en contrôlant une distance minimale de sécurité entre les véhicules, les risques de collision sont considérablement amoindris. La sécurité se trouve donc renforcée. Ensuite, ce mode de fonctionnement jouit d'une grande modularité. Le nombre de véhicules formant le convoi est directement lié au nombre de passagers et seules les ressources nécessaires sont réquisitionnées, contrairement aux transports en commun qui ne peuvent pas s'adapter au nombre de passagers effectifs. Enfin, pour obtenir plus de flexibilité dans le choix de l'itinéraire, on peut envisager un mode de conduite manuel : les membres du convoi suivraient le véhicule de tête, alors conduit manuellement.

Ainsi, le fonctionnement automatique d'un convoi de véhicules permettrait, en complément des services de transport existants, l'élimination des problèmes de parking, la réduction des problèmes environnementaux et un désengorgement conséquent du trafic urbain.

Le sujet traité dans cette thèse est ainsi en rapport avec la conduite en convoi autonome de véhicules urbains. Elle consiste à développer et étendre les modes de fonctionnement de tels convois, validés ensuite par des expérimentations sur véhicules réels.

Enfin, une telle innovation laisserait entrevoir de multiples applications possibles. Ainsi, l'intérêt de cet investissement dépasse largement les domaines d'application évoqués. En effet, la marche en convoi par arrimage immatériel peut avoir des retombées précieuses dans des milieux divers et variés, tels que l'industrie (déplacement du personnel sur le site industriel ou gestion d'une flottille de chariots élévateurs), les parcs

d'attraction (mise à disposition d'un mode de déplacement flexible), trains de véhicules modulaires pour le tourisme en centre-ville, etc.

Programmes de recherche directement liés au sujet de la thèse

Sur le plan national, le PREDIT (Programme national de REcherche et D'Innovation dans les Transports) est un programme de recherche, d'expérimentation et d'innovation dans les transports terrestres, initié et conduit par les ministères chargés de la recherche, des transports, de l'environnement et de l'industrie, l'ADEME[9] et OSEO[10]. L'ANR[11] fait désormais partie également de ses financeurs. Stimulant la coopération entre secteurs public et privé, ce programme vise à favoriser l'émergence de systèmes de transport économiquement et socialement plus efficaces, plus sûrs, plus économes en énergie, et finalement plus respectueux de l'homme et de l'environnement. Le PREDIT 3 (2002-2006) est encadré par trois objectifs généraux :

- Assurer une mobilité durable des personnes et des biens
- Accroître la sécurité des systèmes de transport
- Réduire les impacts environnementaux et contribuer à la lutte contre l'effet de serre

Ce vaste programme fédérateur s'organise autour de 11 groupes opérationnels parmi lesquels le groupe 9 : intégration des systèmes d'information et de communication. L'objectif de ce groupe opérationnel est l'étude, la proposition et le test des nouveaux systèmes ou services basés sur les technologies de l'information et de la communication dans les transports.

Le début de mon travail de recherche concorde avec la date clôture du projet ***Mobi-VIP*** (Véhicules Individuels Publics pour la Mobilité en centre ville), directement associé à ce groupe opérationnel. Ce projet, engagé sur trois ans (2003-2006), s'est intéressé aux recherches et aux expérimentations de briques technologiques visant à la mise en place de services de mobilité en milieu urbain basés sur un système de transport : les Véhicules Individuels Publics et sur un système d'information s'intégrant dans la politique de gestion globale des déplacements en centre ville. Les travaux, faisant intervenir pas moins de cinq laboratoires et de nombreux industriels, ont ainsi été menés selon deux axes réunissant des études prospectives, basées sur des études d'impact et des enquêtes sur les nouveaux services et TIC[12], et le développement de briques technologiques. Les recherches portent sur plusieurs thèmes transversaux dont la modélisation, la navigation, la conduite assistée, le conduite autonome et en convoi, les systèmes d'information et de communication. La conclusion du projet fut l'occasion, pour les différents acteurs,

[9] Agence De l'Environnement et de la Maîtrise de l'Energie
[10] Établissement public de l'Etat français, chargé de soutenir l'innovation et la croissance des PME
[11] Agence Nationale de la Recherche
[12] Technologies de l'Information et de la Communication

de présenter des démonstrations attestant la viabilité et les avancées scientifiques obtenues dans le cadre du projet. On retiendra que le projet fut un précurseur dans le sens où beaucoup de briques nécessaires à la mise en pratique étaient présentes, aboutissant à des possibilités de transfert technologique.

Ainsi le LASMEA, et plus particulièrement le groupe GRAVIR, a pu faire valoir ses nombreuses contributions dans les domaines de la perception et de la localisation d'un véhicule dans son environnement et dans le guidage autonome de véhicules.

Dans la continuité de MobiVIP, le projet ***CityVIP*** qui est porté par le LASMEA, s'insère également au programme PREDIT 3. Il s'étale sur une période de trois ans (2008-2011) avec un investissement de $3{,}2M€$. Le projet concerne le déplacement sûr de Véhicules Individuels Publics dans un contexte urbain et plus particulièrement les centres urbains à accès réglementé qui pourraient être le lieu privilégié de déploiements de tels véhicules. La faisabilité d'un système permettant d'offrir aux personnes, en particulier celles à mobilité réduite, la possibilité d'utiliser de petits véhicules électriques en mode partagé, est étudiee. Ces VIP pourraient être déclinés en version "conduite manuelle" ou en "flotte totalement automatisée". À la différence du projet MobiVIP où la problématique était considérée dans sa globalité, seules certaines briques nécessaires à la mise en pratique sont présentes dans ce projet. Les partenaires associent en effet leurs compétences autour d'un programme scientifique en quatre parties : localisation, conduite autonome, sécurité des déplacements, bases de données géo-référencées. Il s'agit, pour chaque véhicule, d'être doté d'un système permettant d'une part de le localiser précisément en temps réel et d'autre part d'assurer sa navigation autonome. L'attention est aussi focalisée sur l'optimisation de la répartition de la flotte en fonction de la demande, même en cas de conduite manuelle. Dans ce cas un pilote mènerait à destination une flotte de véhicules autonomes pour réapprovisionner les stations, d'où la nécessité de la modalité convoi. Les partenaires regroupent essentiellement des laboratoires de recherche, le LASMEA, l'équipe AROBAS de l'INRIA, l'équipe Lagadic-IRISA de l'INRIA, HEUDIASYC, XLIM, le LCPC, MATIS de l'IGN et la société BeNomad.

Le projet ***CRISTAL*** (Cellule de Recherche Industrielle en Systèmes de Transports Automatisés Légers) est un projet de mobilité urbaine innovant soutenu par le FUI [13] qui s'inscrit dans un contexte existant d'expérimentations de VIP (Véhicules Individuels Publics). Il ne revendique pas de grandes avancées technologiques mais base son succès commercial sur des innovations techniques et son réalisme économique. Le projet est porté par LOHR Industrie, industriel spécialisé dans la conception et la réalisation de systèmes de transport de biens et de personnes. Le consortium initial réunit aussi le LORIA, VU LOG et TRANSITEC SA et intègre de nombreux partenaires sous-traitant. L'INIRIA, l'UTBM, GEA SA, le LASMEA, PIMENTIC, GEOLIA et TECNOMADE participent ainsi au niveau de la recherche et du développement, alors que J2B se charge de mener les études territoriales.

[13] Fonds Unique Interministériel

Les acteurs du projet ont investi 8 millions d'euros pour les différentes recherches et études associées à la conception du véhicule Cristal, un système de transport public, individuel ou semi-collectif, adaptable à l'évolution de la demande de mobilité dans le temps et dans l'espace. Il s'agit d'un véhicule compact de 3,40m de long et d'une capacité de 6 personnes. Il expose des caractéristiques attrayantes comme le fait d'être non-polluant (moteur électrique), silencieux et de pouvoir évoluer indifféremment selon deux modes : en conduite manuelle (mode libre-service) ou en convoi d'abord matériel, puis dans une deuxième étape immatériel et supervisé à distance par l'opérateur local. Il est destiné à adapter en continu l'offre de transport individuel ou semi-collectif à l'évolution de la demande dans un centre urbain.



• Au travers du mode *"libre-service"*, le système Cristal fonctionne en flotte de véhicules disponibles en tout temps et en tout lieu dans un périmètre défini. Il répond à une demande ponctuelle, instantanée (sans réservation), dispersée dans le temps et dans l'espace. C'est un système de transport complémentaire aux autres systèmes existants. L'implantation de stations Cristal est prévue, ces dernières seront définies par des études territoriales spécifiques. L'usager, pré-identifié auprès de l'exploitant, pourra vérifier la disponibilité de la flotte et réserver un véhicule à la station de son choix. A la montée dans le véhicule, celui-ci sera configuré en mode "libre-service", avec accès au tableau de bord, et application du code de la route (par exemple, port de la ceinture obligatoire). En fin de course, le véhicule est rendu à la station de destination par l'usager.

• Un mode *"convoi"* a aussi été pensé incluant un conducteur professionnel rattaché à l'exploitant. Grâce à ce dernier, le système Cristal permet de répondre à une demande de déplacement planifiée (évènement, type de service spécifique), ponctuellement concentrée dans le temps et dans l'espace. Il offre la possibilité d'adapter la capacité par attelage d'abord matériel puis immatériel dans une seconde étape. Le conducteur choisit le nombre de mobiles composant son convoi, puis quitte la station. Pour ce mode, les tableaux de bord autres que celui du conducteur sont escamotés et inaccessibles aux

passagers. Le véhicule est considéré comme un mode de transport public, avec des places assises et debout.

Le véhicule Cristal a été labellisé par le pôle de compétitivité "Véhicule du Futur" d'Alsace Franche-Comté. Deux études territoriales sont en cours à Strasbourg et Montbéliard : elles sont destinées à développer ce transport innovant dans des conditions d'intégration qui auront été préalablement optimisées.

Objectifs de la thèse

La thèse exposée ici s'inscrit pleinement dans le cadre des grands projets cités précédemment. Elle s'appuie d'une part sur les travaux de thèse de J.Bom [Bom 06] qui a développé une stratégie de commande de véhicules en convoi, qui a été validée par des expérimentations avec des véhicules réels. Elle intègre d'autre part un système de cartographie 3D et de localisation pour robot mobile fondé sur la vision monoculaire, mis en place et validé par E.Royer [Royer 06] pendant sa thèse. L'objectif général de cette thèse est de contribuer au développement de nouvelles fonctionnalités et modes de navigation pour le convoi de véhicules.

Dans un premier temps, on souhaite apporter une plus grande flexibilité au système de navigation en intégrant des outils permettant de générer facilement une trajectoire à partir de données brutes de localisation absolue dans un système métrique. La finalité est ensuite de pouvoir modifier la trajectoire des véhicules de façon dynamique à travers un "mode manuel" de conduite. Le déplacement des véhicules doit se faire en fonction de la trajectoire empruntée en temps réel par un véhicule désigné comme leader. L'objectif est que le convoi soit monotrace sur la trajectoire du leader. La première contribution de cette thèse réside donc sur la représentation de la trajectoire, basée sur des courbes B-Splines uniformes, et sur les stratégies employées pour la rendre extensible et ajustable en temps réel.

Le deuxième objectif de cette thèse est d'intégrer le capteur vision pour calculer les entrées des lois de commande pour la navigation en convoi. On observera par la suite que le transfert de la stratégie de navigation par vision vers une application en convoi n'est pas directe. La reconstruction 3D de l'environnement n'est pas parfaite et de légères distorsions avec le monde métrique sont perceptibles. La deuxième contribution de cette thèse est donc l'adaptation des lois de commande à une nouvelle représentation de l'environnement. Deux approches ont été développées pour déterminer et prendre en compte des corrections dans le monde vision. La première fait intervenir deux véhicules et permet d'estimer les corrections par une mesure télémétrique directe de distance entre les deux véhicules. La deuxième solution, plus facile à mettre en œuvre et plus robuste, calcule les corrections grâce à l'utilisation de mesures odométriques injectées dans le modèle d'évolution du véhicule leader.

Les deux contributions ont toutes deux été validées après simulation et par des expérimentations sur des véhicules réels.

Organisation du manuscrit

Les travaux réalisés dans le cadre de cette thèse s'articulent autour des quatre chapitres suivants.

Chapitre 1 L'objectif de ce premier chapitre est tout d'abord de fournir une synthèse des travaux effectués pour la commande de robots mobiles en formation. Une classification des stratégies de commande employées est proposée et permet ensuite de situer les travaux de recherche effectués dans cette direction au sein du LASMEA parmi ceux existant dans la communauté. La commande longitudinale, développée par J.Bom [Bom 05] et les outils d'automatique utilisés pour sa conception, sont décrits plus en détail. Un bilan de la recherche sur les outils pour la navigation en convoi est dressé et les différents axes de recherche explorés au cours de cette thèse sont cités. L'intérêt des travaux réalisés en interne par E.Royer [Royer 05] sur la navigation est ensuite mis en évidence dans le cadre de notre application.

Chapitre 2 Dans ce chapitre, un état de l'art permet d'exposer les différents outils utilisés en robotique mobile pour décrire une trajectoire, qu'elle soit établie par avance ou générée en ligne. Notre stratégie de modélisation d'une trajectoire par des courbes B-Splines est ensuite décrite en fonction de deux modes de navigation du convoi : "mode tout automatique" qui se caractérise par une trajectoire fixe et pré-définie et "mode manuel" pour lequel la trajectoire se construit en temps réel en fonction du déplacement du véhicule de tête et qui nécessite d'être mise à jour à chaque instant pour les suiveurs. La qualité du processus de génération est finalement mesurée à partir de simulations puis directement corrélée avec les paramètres de l'algorithme.

Chapitre 3 Comme pour les chapitres qui précèdent, un état de l'art est effectué. Il propose une synthèse des travaux menés pour la navigation de robots mobiles par l'intermédiaire d'un dispositif de vision. La stratégie de localisation utilisée est justifiée par rapport à l'approche de commande pour la navigation en convoi. Des premiers tests sont présentés et révèlent les verrous scientifiques à soulever. Une fois ceux-ci identifiés, deux approches pour aborder la problématique sont explicitées et testées en simulation.

Chapitre 4 Enfin le dernier chapitre illustre et commente les différentes expérimentations réalisées sur véhicules réels pour mettre en œuvre les développements décrits dans les deux chapitres précédents.

Chapitre 1

Bibliographie et travaux précurseurs

La *robotique mobile*, à laquelle se rapporte cette thèse, représente une très vaste communauté et la quantité de travaux qui y sont associés rend ardu le positionnement d'apports personnels parmi les nombreuses branches de recherche et disciplines étudiées depuis des années. Dans ce chapitre, on va tout d'abord préciser le contexte de la recherche, à travers la description des nouveaux systèmes et services de transport, pour identifier de façon concrète le cadre qui entoure les notions de *véhicules urbains*, de *convoi* et plus généralement de *formation de véhicules*. Ensuite, un intérêt particulier est porté sur la fonctionnalité de navigation en convoi. Afin d'établir une synthèse des techniques employées pour aborder cette problématique, différentes stratégies de commande existantes sont détaillées, s'appuyant sur une classification en plusieurs approches selon l'environnement de travail et les objectifs à atteindre.

Enfin pour poser les bases de cette thèse, il convient de présenter les travaux réalisés au LASMEA qui traduisent la dynamique de recherche autour de la navigation de véhicules autonomes. La synthèse des lois de commande développées avant cette thèse afin de maintenir une formation de véhicules en convoi le long d'une trajectoire prédéfinie, est décrite ainsi que la stratégie de localisation par vision monoculaire pour la navigation autonome. Au vu de ces travaux, des perspectives de recherche sont mises en valeur et les axes de recherche, explorés au cours de cette thèse, sont définis.

1.1 Nouvelles pratiques et systèmes de transport

Le développement de nouveaux systèmes de transport urbain se traduit par des innovations perceptibles à plusieurs niveaux (technologie, infrastructure, logistique, etc.) et proposant des solutions à plus où moins long terme. Ainsi, une première réponse à la question “Quels véhicules pour la ville de demain ?”, s'appuyant sur les progrès de la technologie, est formulée par les généralistes de l'automobile. On appellera simplement cette première classe de véhicules “les VUP” comme Véhicules Urbains Privés.

1.1.1 Véhicules urbains privés

Une très grande activité industrielle et économique est en émergence depuis quelques années pour proposer des véhicules "plus propres". Les conditions sont aujourd'hui réunies pour que l'on puisse enfin sortir de la voiture à pétrole et la plupart des constructeurs suivent la tendance écologique. Les principales nouveautés concernent la source d'énergie alimentant le moteur. Mus par la force électromotrice de moteurs électriques alimentés par une batterie d'accumulateurs, certains véhicules électriques sont de plus munis de générateurs électriques internes : moteur thermique classique assurant selon la situation une partie de la traction ou une fonction de groupe électrogène (véhicule hybride), pile à combustible (véhicule à hydrogène) ou autre groupe électrogène, éventuellement des panneaux solaires intégrés à la carrosserie pour des véhicules spécialement économes.

• **les véhicules électriques** Quelques exemples, dérivés de la gamme tout-électrique des véhicules urbains, sont présentés à la Fig. 1.1. Dans cette catégorie, on peut tout d'abord citer deux concept-cars compacts s'adaptant parfaitement à l'urbanisation et à la ville. Le *Pivo 2* conçu par *Nissan* se distingue par son habitacle rotatif et l'intégration d'un compagnon cybernétique de voyage. La France n'est pas en reste avec un curieux biplace *Twizzy* conçu chez *Renault* et possédant un habitacle minimaliste comportant tout de même un airbag conducteur et deux airbags latéraux. Concernant les performances, ces véhicules possèdent un autonomie de $100km$ et développent une puissance de $15kW$ ($\simeq 20cv$). En comparaison les mêmes valeurs s'élèvent respectivement en moyenne à $150km$ et $75kW$ pour des voitures classiques du marché électrique. De son côté, un concept intéressant est proposé par *Eco&Mobilité*, une jeune entreprise implantée à Chauvigny en Poitou-Charentes. En effet, spécialisée dans la conception, la réalisation et la commercialisation de véhicules électriques pour le transport des personnes et des marchandises, elle propose une solution innovante aussi bien pour les particuliers que les professionnels, à l'image de son utilitaire le *Simply City*.



Pivo 2 : Nissan **Simply-city : Eco&Mobilité** **Twizzy : Renault**

FIG. 1.1 – Véhicules urbains "tout électriques" : quelques exemples

• **les véhicules hybrides** Ils comportent un moteur thermique (comme les actuelles voitures à essence, diesel), mais sont également dotés d'un moteur électrique et de batteries. Le moteur électrique entre en action soit pour donner plus de puissance à

la voiture, en appui du moteur thermique, soit seul, moteur thermique éteint. Concrètement, les récents modèles de voiture hybride permettent de rouler en tout-électrique à faible vitesse. Lors des phases de forte accélération, les deux moteurs fonctionnent en même temps afin de délivrer au véhicule la puissance maximale. L'intérêt de l'hybride est moindre lorsque l'on roule à vitesse constante relativement élevée, par exemple sur autoroute. Dans ce cas, la consommation est à peu près identique à celle d'un véhicule classique. Il s'agit donc d'une technologie de transition et, on le devine aisément, cette technologie fait l'objet d'une forte promotion des lobbies pétroliers.

Dans la gamme des véhicules hybrides on peut citer le *F-City*, illustré Fig. 1.2. Il s'agit d'une voiture sans permis développée par *FAM Automobiles*, soutenue par le pôle de compétitivité "Véhicule du Futur" à Belfort. Elle développe une puissance de $8kW$ et, après une période de 10h de charge, son autonomie varie de 80 à $100km$ selon la vitesse qui peut atteindre $65km.h^{-1}$.

• **les véhicules à air comprimé** La simplicité du stockage énergétique de la filière air comprimé, son bilan sanitaire ainsi que son coût potentiellement faible rendent cette technologie séduisante au premier abord. Cependant, les performances laissent sceptiques de nombreux experts. En effet pour compresser l'air, on consomme de l'électricité et le rendement énergétique global est 4 fois inférieur à celui de la voiture électrique à batterie. D'un point de vue purement énergétique, ce n'est donc pas une approche pertinente. Sans parler de l'encombrement de la réserve d'air comprimé, il s'agit aussi d'un véhicule bruyant.

Appartenant à cette catégorie, l'*AIRPod* est l'aboutissement des études de *MDI*, voir Fig. 1.2. Sorti des lignes de production en juin 2009, il répond à la demande d'Air France et KLM, ainsi qu'à l'appel d'offre de la ville de Paris, Autolib. Sa motorisation, souvent comparée à celle des véhicules électriques, affiche des performances équivalentes avec les autres modèles de voitures ultra compactes pré-cités, à savoir $6.8cv$ et une autonomie qui peut atteindre $200km$ selon les conditions d'utilisation. Cependant elle est beaucoup plus bruyante et le nombre élevé de pièces mobiles qui constituent son moteur laisse entrevoir un niveau de fiabilité inférieur à celui des véhicules électriques. D'un autre côté, son grand avantage repose sur ses capacités de recharge : des compresseurs installés en ville permettent de faire le plein d'air en seulement 3 minutes et un compresseur électrique interne permet de la recharger en quelques heures en la branchant à une prise de courant. D'un point de vue énergétique, bien que le stockage d'énergie par air comprimé apparaît moins performant que celui des batteries électriques, le poids excessif des batteries des voitures électriques vient entraver les performances.

• **les véhicules à hydrogène** Il existe deux types de véhicules à hydrogène. Le premier s'appuie sur un moteur à explosion basé sur la combustion interne de dihydrogène. Les véhicules du deuxième type sont en réalité des voitures électriques, mais à la place d'une batterie, elles possèdent un réservoir de dihydrogène comprimé. La fabrication de l'électricité se fait grâce à l'oxydation sur une électrode d'un combustible réducteur couplée à la réduction sur l'autre électrode d'un oxydant, tel que l'oxygène de l'air. L'inconvénient de cette technologie réside dans la difficulté de production du

dihydrogène, sa compression, son transport et son stockage. D'un point de vue des performances, des essais de moteurs à hydrogène (premier type) les ont révélé aussi performants que les moteurs diesel actuels. C'est pourquoi cette énergie plus propre que le moteur thermique classique a été commercialisée malgré son rendement énergétique plutôt faible ($\leq 50\%$). Concernant le deuxième type, les performances sont logiquement les mêmes que celles des véhicules électriques, cependant la solution dihydrogène de stockage de l'énergie est 3 fois moins efficace que le stockage électrochimique des batteries.

Ainsi le projet innovant de la marque *Honda* dénommé *Puyo*, illustré Fig. 1.2, brille plus de par ses attributs originaux que par sa motorisation qui s'appuie sur une pile à combustible (deuxième type). Elle possède en effet un système de rotation des roues qui aide aux manœuvres urbaines, une carrosserie aux formes arrondies, luminescente et qui change de couleur la nuit. Du côté des équipements, le tableau de bord prend la forme d'une simple plaque de verre et un joystick remplace le volant.

• **Autres** Certains véhicules, basés sur la technologie électrique font aussi preuve d'inventivité comme le *BLUECAR* du groupe *Bolloré-Pininfarina* et l'*Electric* produit par *Venturi* tous deux illustrés Fig. 1.2. L'originalité de la BLUECAR se situe dans ses batteries grâce à un concept unique reposant sur la technologie Lithium Métal Polymère qui, à poids équivalent, stocke cinq fois plus d'énergie qu'une batterie traditionnelle et se recharge en quelques heures. Des cellules photovoltaïques présentes sur le toit et la calandre contribuent aussi à l'alimentation de ses équipements électriques. De son côté l'Electric permet de produire et stocker les énergies renouvelables solaires et éoliennes, réduisant ainsi au maximum l'impact sur l'environnement. Un dernier véhicule, fonctionnant au gaz naturel, est plus axé sur la performance. Il s'agit du tricycle *Clever* (Fig. 1.2), conçu chez *BMW* et issu du projet éponyme financé par la Commission Européenne. Il se déplace à une vitesse de pointe d'environ $100 km.h^{-1}$ avec une consommation très faible : l'équivalent de $1.5L$ pour $100km$.

Pour conclure ce paragraphe on remarquera que la commercialisation actuelle de ces nouvelles technologies concerne essentiellement les moteurs hybrides et à combustion d'hydrogène de par leurs performances qui sont comparables à celles des moteurs thermiques actuels. Ces véhicules plus propres constituent ainsi une solution écologique à court terme. Cependant, à long terme il faudra s'orienter vers une motorisation électrique pour stopper les émissions de gaz. La réalisation d'un vecteur de stockage écologique et efficace de l'énergie reste encore une question ouverte.

Le cadre urbain, dans lequel se situe cette thèse, est peu exigeant en terme de performances sur la motorisation : les trajets effectués en villes sont souvent courts et la présence de piétons et d'autres systèmes de transport impose de rouler prudemment et à faible vitesse. Les perspectives d'une petite citadine écologique (Fig. 1.1 et 1.2) sont alors prometteuses et ces dernières constituent une part de marché grandissante, à l'image du projet Autolib.

L'inconvénient de l'utilisation de ces véhicules dans les métropoles, et non des moindres, est qu'ils ne peuvent pas à eux seuls résoudre les problèmes d'engorgement,



F-City : FAM Automobiles



AIRPod : MDI



Puyo : Honda



BLUECAR : Bolloré-Pininfarina



Electric : Venturi



Clever : BMW

FIG. 1.2 – Véhicules urbains : quelques exemples de nouvelles technologies

et ce quelle que soit la technologie retenue. Il va falloir les intégrer dans un service de partage pour traiter efficacement le problème. Sur ce point une solution en vogue, et qui présente de nombreux attraits, est constituée par le co-voiturage.

1.1.2 Le co-voiturage

Qu'il soit ponctuel sur une longue distance ou qu'il concerne un trajet régulier (domicile-travail par exemple), le co-voiturage est une pratique en pleine croissance. Par l'utilisation conjointe de véhicules, la collectivité y gagne par la diminution des embouteillages, de la pollution et des accidents de la route.

Comme le montre la pancarte ci-contre, affichée dans les années 40, la pratique du co-voiturage existe dans les pays développés depuis de nombreuses années. Cependant, elle ne connaît un réel engouement en France que depuis 2009 : les contraintes de pouvoir d'achat conjuguées à une réelle prise de conscience des limites écologiques de la voiture individuelle, concourent à une augmentation impressionnante de trafic sur les sites de co-voiturage français.



[1]

Dans certains pays des aménagements ont été prévus au niveau des infrastructures pour inciter et développer la pratique du co-voiturage :

• Au Canada et aux USA des voies réservées aux véhicules à occupation multiple (VOM) favorisent les véhicules transportant plusieurs passagers et réduisent les embouteillages. Des automobilistes prennent à des arrêts des passagers qu'ils ne connaissent

[1] Propagande du gouvernement des États-Unis durant la Seconde Guerre mondiale.

pas pour des trajets prédéfinis. Ce service est souvent gratuit pour le passager et la principale motivation pour le conducteur est de pouvoir utiliser les voies de co-voiturage VOM.

• Des aires de stationnement de co-voiturage ont également été délimitées. Ce sont des lieux sécurisés où les covoitureurs peuvent se poster pour attendre leur conducteur, ou pour pratiquer l'auto stop. Ces stationnements (parkings) permettent de se retrouver également et d'y laisser une voiture, afin de ne prendre qu'un véhicule pour plusieurs personnes. En France le département du Morbihan est un précurseur des aires de co-voiturage. Il a signé de nombreuses "conventions de signalisation d'une zone de co-voiturage" avec des grandes surfaces ayant des portions de parking peu exploitées. La Bretagne est la région de France où le mot "co-voiturage" est le plus recherché sur *Google*.

Cependant, la multiplicité et la diversité des acteurs et des sites sont un frein au développement du co-voiturage en France. En effet le principe repose sur la mise en relation de voyageurs adhérant au même site, afin de partager ou bénéficier d'un véhicule sur une partie ou l'intégralité d'un trajet. La France compte ainsi 78 sites de co-voiturage, gratuits pour la plupart, mais souffrant d'une faible fréquentation à cause du grand nombre de sites grand public existants. De plus, la plupart des sites de petites annonces gratuites proposent aussi une rubrique de co-voiturage. D'ailleurs, certains sites de petites annonces locales disposent de plus d'offres de co-voiturage qu'un site spécialisé, pour un trajet identique. Le regroupement d'acteurs (collectivité, entreprise, association, etc.) et la mise en commun des bases de données des sites pourraient répondre à ce problème. A cet effet, le Canada constitue un bon exemple pour les réseaux de co-voiturage. Le logiciel et la base de données du réseau de co-voiturage sont partagés par plusieurs sites web qui utilisent un habillage propre à leur portail respectif. Ce fonctionnement permet aux villes et aux institutions canadiennes de se joindre au réseau, et de bénéficier du logiciel à partir de leur propre site de manière transparente pour les utilisateurs. Lorsqu'un utilisateur est membre de l'un des sites Web affiliés au réseau, il l'est également pour tous les autres sites partenaires.

Malgré la prise de conscience écologique et le changement progressif des mentalités, le co-voiturage ne constitue pas une solution d'une ampleur suffisante notamment pour les grandes mégalopoles. L'optimisation des trajets par co-voiturage nécessite en effet un effort d'organisation conséquent supporté par des outils logiciels possédant une base de données commune, et des applications telles un calculateur de trajet versus les coûts de l'essence et un service de communication. Une révolution plus conséquente est amenée par l'apparition de nouveaux services de déplacement. Une perception nouvelle du transport en milieu urbain fait ainsi son apparition pour aboutir, à plus ou moins long terme, à une restructuration complète des systèmes de transport en ville.

1.1.3 De nouveaux concepts et services

La recherche d'une mobilité plus durable repose bien sur le progrès technologique et l'évolution des mentalités, mais aussi sur la planification urbaine, la tarification des

systèmes et les évolutions d'usages des modes. Seule une vue d'ensemble qui s'étend au-delà de la conception du véhicule et de l'entre-aide collective peut permettre de joindre simultanément des objectifs concernant qualité de vie, environnement, pression des transports sur l'espace et cohésion sociale.

On est aujourd'hui amené à restructurer les systèmes de transport en tenant compte de nouveaux **lieux**, proposant de nouvelles **fonctionnalités**. La cause principale de ceci repose sur la nécessité d'introduire le concept d'**intermodalité**, bien connu dans le milieu du transport de marchandises, aux transports de personnes au niveau urbain et péri-urbain.

Definition : *L'intermodalité représente un système soutenu par une logistique complexe faisant appel à plusieurs modes de transport. Ces derniers sont étudiés pour permettre un transfert efficace des passagers (et/ou marchandises) durant le parcours origine-destination.*

Une intégration efficace des différents systèmes modaux permettrait en effet un transfert des modes de transport individuels vers le transport public, ce qui est généralement considéré comme souhaitable. On s'attache ainsi à redonner une certaine attractivité aux systèmes publics qui apparaissent comme segmentés, lents et inadaptés à la société actuelle. La caractéristique centrale à l'intermodalité est la fourniture d'un service via un seul ticket pour les passagers. Plutôt que de chercher à exploiter individuellement les avantages propres à un mode de transport, que ce soit en matière de coût, de service, de fiabilité et sécurité, on capitalise les richesses de chacun pour tirer profit de la multiplicité des transports. Une illustration de ce principe se trouve Fig. 1.3. On considère l'exemple typique du trajet maison (dans une ville quelconque A)- travail (en centre-ville B) où *"une personne"* rime en règle générale avec *"une voiture"* aux heures de pointe. Voici les clefs qui découlent de l'intermodalité, mises en valeur au travers de cet exemple :

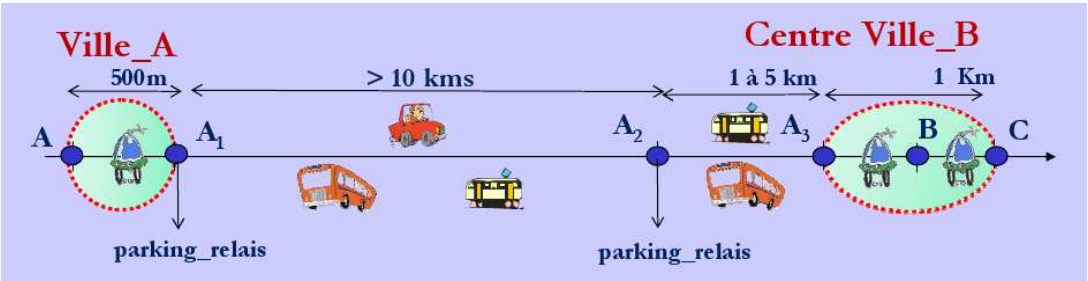


FIG. 1.3 – Exemple tiré du projet CityVIP

L'utilisation de **pôles d'échange** est primordiale puisqu'elle encourage l'intermodalité entre la voiture particulière et les transports publics. Dans notre exemple Fig. 1.3, le voyageur rejoindra le parking relais le plus proche en périphérie de sa ville (au point

A_1) pour accéder au réseau de transport lui permettant de joindre son lieu de travail. Au niveau d'une grande ville (B) un autre parking relais (point A_2) peut se situer en zone péri-urbaine permettant ainsi la transition vers le centre-ville. Un autre type de station, plus adapté en centre-ville, peut également faire figure de pôle d'échange (point A_3). A ce niveau, l'intermodalité se conçoit par la mise à disposition d'informations données au passager concernant l'intégralité de son voyage et l'analyse des fréquences des différents modes de transport. Il s'agit également de permettre aux voyageurs de transformer leur temps d'attente en un temps utile, par la création dans les gares de services divers.

Le développement de **VIP** (Véhicules Individuels Publics) constitue une solution de mobilité alternative à un véhicule privé. Cette offre en libre-service, complémentaire à celle des opérateurs de transport public, est destinée à la desserte d'une zone géographique restreinte : hyper-centre ou lieu d'attraction particulier (aéroports, hôpitaux, etc.). Les VIP interviennent donc sur un territoire souvent réservé aux véhicules privés, qui leur confère un de leurs plus grands avantages puisque cela reste un mode de transport individuel. Ils présentent la caractéristique d'avoir faible empreinte écologique, possèdent en général une capacité allant de 2 à 5 voyageurs, et sont disponibles moyennant une tarification cohérente avec celle des transports publics. Ces derniers seraient utilisés, dans l'exemple Fig. 1.3, pour tout type de trajet en ville (A) et centre-ville (B).

Une particularité, présentée à part, de ces VIP est leur intégration dans un système de **gestion dynamique** de la flotte de véhicules. Pour être efficace, une répartition proportionnée des véhicules est requise permettant ainsi de répondre avec justesse à la demande. Un élément capital réside dans la capacité de redéploiement des véhicules et d'atteindre rapidement certaines zones stratégiques (comme les pôles d'échange, lieux où s'effectue la maintenance, etc.). La complexité de la gestion de la flotte dépend de l'importance de la couverture géographique du service. La flexibilité du service est aussi un facteur primordial et l'on différencie plusieurs types d'offre. D'un côté le simple autopartage nécessite, en plus d'une réservation préalable du véhicule, la remise du véhicule au point de départ. En comparaison, un service en "trace directe" laisse plus de liberté à l'utilisateur puisqu'il n'est pas nécessaire de ramener le véhicule à la station d'origine. La location peut donc s'interrompre lors d'une visite ou une pause déjeuner, auquel cas un autre véhicule sera sollicité pour un éventuel autre déplacement. Dans le dernier cas, une dernière distinction se porte sur le lieu d'accès au véhicule qui peut être soit en station soit "au fil de la rue" après un appel téléphonique ou une réservation par Internet.

Quels que soient le lieu (centre-ville, banlieue ou périphérie) et le mode de transport, une gamme étendue de **services** et **fonctionnalités** sont à envisager pour offrir une liberté et un confort d'utilisation accrus à l'utilisateur. Des capacités de conduite en mode automatique ou dual (conduite manuelle et automatique) représenteraient par exemple un atout important autant dans la gestion de la flotte, autonome dans ses déplacements, que pour l'utilisateur dont l'attention ne serait pas portée sur la conduite

du véhicule. La marche en convoi permettrait aussi d'optimiser le flux de circulation et d'amoindrir les risques de collision.

Au vu de ce qui vient d'être présenté, la conception d'une telle approche de la mobilité en milieu urbain n'aurait pas pu advenir sans une série de préalables innnovants touchant à l'organisation et au contrôle de l'information. Au cœur de l'intermodalité actuelle se trouvent des systèmes de traitements et d'acheminement de données essentiels à tout transport de passagers empruntant plusieurs modes de façon sécuritaire, fiable et rentable. L'EDE [2] est une technologie en plein essort qui aide les compagnies à faire face à l'incessante complexification d'un système de transport global. Les applications de services de navigation qui sont en activité à l'heure actuelle et qui adoptent une vision multi-modale répondent à trois grandes problématiques. La première concerne la gestion des services de mobilité fournis par les différentes structures afin d'améliorer le système de transport. Le domaine de l'urbanisme est aussi ciblé à travers les infrastructures et stations de desserte impliquées dans la mise en place de l'application. Le dernier point clef repose finalement sur la conception et la technologie du véhicule ciblé ainsi que son guidage.

Pour cette bibliographie, il est important de séparer les véhicules urbains des autres à cause des hypothèses de commande. Les véhicules urbains roulent sans glissement, à faible vitesse et on n'a pas besoin de modéliser le comportement du moteur qui est électrique.

1.1.4 Applications de services de navigation en milieu urbain

Le concept de VIP urbains repose sur des voitures de faibles dimensions, tout particulièrement adaptées à la conduite en milieu urbain. Ces engins sont généralement équipés d'un moteur électrique, produisant ainsi un faible bruit et ne polluant pas l'environnement. Des projets très prometteurs, introduisant de tels véhicules, ont ainsi vu le jour à travers le monde et peuvent être classifiés en deux approches :

Les véhicules pilotés par l'usager Au Japon, Toyota a lancé en 1999 une étude de consommation sur l'utilisation coopérative de petits véhicules électriques et autres types de véhicules peu polluants. Ce système de transport, nommé Crayon, a été majoritairement mis en place à Toyota City, une grande ville de l'industrie automobile dans la préfecture d'Aichi. Les expérimentations font partie de "START 21" (Smooth Traffic with Advanced Roads in Toyota City for the 21st Century), un plan de soutien destiné à revitaliser la ville en utilisant des systèmes de transport intelligents. Trois types d'études comportementales ont été menées, correspondant à des besoins sur de courts trajets. La première porte sur le co-voiturage d'employés gouvernementaux résidents dans la ville et partageant les voitures pour des déplacements quotidiens en semaine. D'autres essais s'intéressent à l'utilisation coopérative de véhicules. Ils ciblent les organisations situées dans la zone centrale de la ville et concernent les déplacements en semaine liés à une activité économique. La dernière étude se rapporte à des trajets de courte distance pour

[2] Échange de Données Électroniques

rejoindre par exemple la gare la plus proche qui mènera l'utilisateur à sa destination du week-end. Les participants peuvent ainsi réserver, par téléphone ou Internet, des véhicules électriques nommés "e-com". Le principe est assez simple. Un enregistrement s'effectue au niveau d'un terminal situé dans la station (Fig. 1.4 partie 1), grâce à une carte à puce (partie 2). Après utilisation du véhicule (partie 3), ce dernier doit être ramené à la station d'origine puis chargé (partie 4). Cinq stations, pourvues des 17 véhicules électriques, sont réparties dans Toyota City. Depuis son démarrage, l'expérience

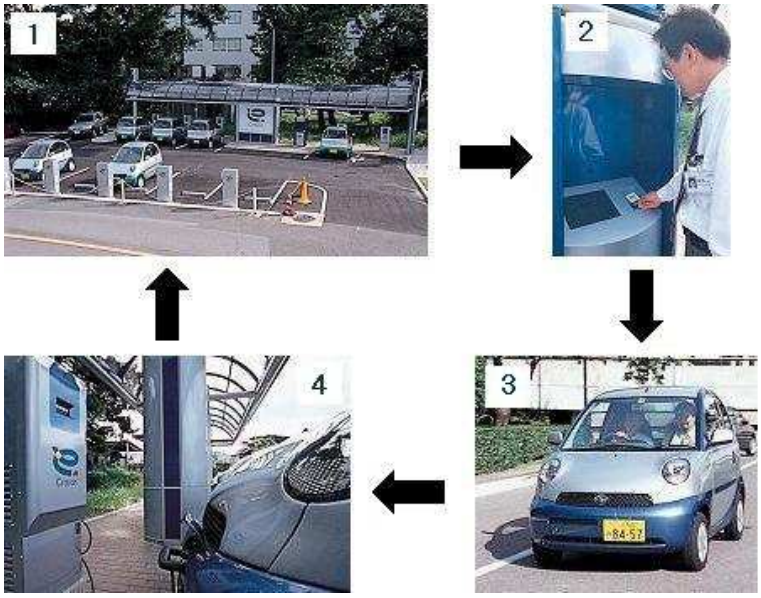


FIG. 1.4 – Principe d'utilisation des véhicules électriques à Toyota City

s'est élargie d'année en année. Selon l'enquête de l'exercice 2001, les effets comprenaient une augmentation du nombre de passagers ferroviaires, préférant l'utilisation combinée du train puis de véhicules électriques et délaissant ainsi leur voiture. Une certaine revitalisation des quartiers a également été constatée en raison de l'augmentation de la fréquence des visites au centre-ville. De plus, les émissions de dioxyde de carbone et d'oxydes d'azote ont été réduites de 132 kg et 693 grammes par mois, respectivement. À partir de Décembre 2004 et jusque 2006, date de fin du test, 302 personnes du grand public et 983 personnes employées par les 35 entreprises membres ont été enregistrées.

En France plusieurs villes comme Paris, Lyon, Marseille, Grenoble et Strasbourg ont lancé des initiatives d'auto-partage, impliquant une réservation avant de pouvoir disposer d'une voiture et un rapatriement à son point d'origine après emploi.

La ville de La Rochelle, pionnière en matière d'écologie et de déplacements urbains depuis plus de 30 ans se démarque par la mise en place d'un projet très novateur. Un service de véhicules électriques en libre-service a effectivement été initié dès 1999 dans l'agglomération rochelaise. Il s'agit du projet Lisélec, déjà mentionné dans l'introduc-

tion, dont le schéma de principe est décrit sur la figure 1.5. Le service est constitué d'une flotte de véhicules électriques, accessibles 24 h sur 24 et 7 jours sur 7 répartis autour de 7 stations situées à travers la ville. La personne abonnée accède à Liselec grâce à un "pass" qui lui a été remis et qui permet de prendre une des 50 voitures puis de la restituer dans l'une des stations. Les stations Lisélec, identifiées par une signalétique spécifique, sont



FIG. 1.5 – Schéma illustrant le projet Lisélec

implantées sur les points stratégiques de la ville : gares, équipements administratifs, quartiers commerçants, zones d'activités ou de loisirs, zones résidentielles isolées. Ces stations sont équipées de bornes de charge électrique et d'une unité de gestion électronique qui assure l'interface entre les voitures garées et le poste central, chargé de superviser le système. Les mouvements de véhicules de la flotte Lisélec, départs comme retours, sont détectés et le poste central est informé, en temps réel, de l'effectif restant. Sur le terrain, un personnel spécialisé, les "jockeys", déplace les voitures d'une station à une autre, sur instruction du poste central, pour maintenir une répartition optimale de la flotte. Ces équipes veillent également au bon état des voitures et à leur mise en charge sur les bornes de station. Un système de vidéo-surveillance connecté au poste central est intégré aux stations.

Une enquête réalisée en février 2000 auprès de 130 abonnés, montre que les usagers sont satisfaits du service. Les avantages économiques et pratiques provoquent l'engoue-

ment du grand public et l'on compte ainsi 2500 courses par mois, la distance moyenne d'un trajet étant de 6 km pour une durée moyenne de 30 minutes. La majorité des usagers, constituée à 33% d'étudiants, effectue des trajets directs de station à station. Fin avril 2002, on dénombrait 485 abonnés avec une progression de 6 nouvelles inscriptions par semaine en moyenne. Depuis fin 2006, le service est exploité par la société Proxiway, filiale de Veolia Transport.

Avec Autolib', la ville de Paris veut faire évoluer encore plus les mentalités, en présentant le 10 décembre 2009 un appel d'offres portant sur la mise en place d'un système de 3000 voitures électriques en libre service, disponibles dans 1 000 stations (dont 300 en banlieue) équipées de bornes. L'idée serait de disposer de voitures s'adaptant parfaitement à l'urbanisation et à la ville, à l'image du concept car élaboré au MIT (voir Fig. 1.6), doté d'un châssis pliable lui permettant de s'empiler avec les autres voitures, comme des chariots de supermarché. Le projet devrait couvrir Paris et 29 villes d'Ile-de-France. Ainsi Autolib' réplique le système Vélib' mais sans commettre l'erreur de l'arrêter aux frontières de la capitale, permettant à chacun d'emprunter un véhicule à tout moment (service 24 h sur 24 et 7 jours sur 7) sans réservation préalable et sans contrainte d'avoir à le redéposer où on l'a pris. Autolib' est une première mondiale, puisqu'aucun système de ce type et de cette ampleur n'existe aujourd'hui. Les collectivités précédemment citées (ville de La Rochelle et Toyota City), pour ne citer qu'elles, s'y sont essayées, avec de petites flottes ; mais aucune métropole n'a tenté, à ce jour, la mise en œuvre du concept à grande échelle. Paris fait figure de laboratoire, tant tout est à inventer : l'ajustement entre offre et demande de véhicules, l'équilibre économique du système, la gestion d'un parc automobile entièrement électrique. Et cela tout en gérant des facteurs moins prévisibles, comme l'adhésion de la population à un système où l'on emprunte un véhicule dont la place vient d'être libérée par un autre conducteur. Ce

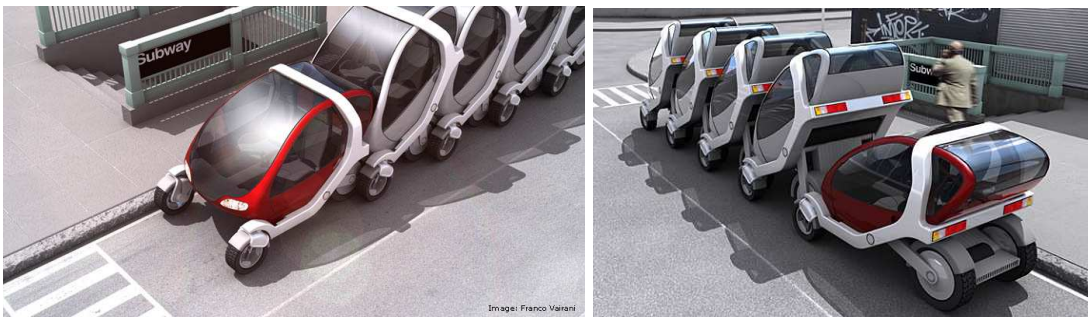


Fig. 1.6 – The City Car, projet du MIT pour des services de type Autolib', présenté en 2006 à la biennale d'architecture de Venise

service de transport s'inscrit dans le cadre d'un bouquet de modes de transport offert aux franciliens, pour permettre aux usagers d'avoir un choix plus large de solutions de mobilité. L'accès aux différents types de transport, à savoir : transports en commun (métro, bus, tramway), vélo (avec notamment Vélib'), taxi, l'offre de location de voi-

ture classique et les services d'auto'partage et de co-voiturage, va ainsi permettre aux usagers d'adapter leurs demandes selon l'heure et le lieu de leur déplacement.

La tarification du service Autolib' ne devra pas être trop bon marché pour ne pas concurrencer inutilement d'autres services de transport pertinents. Elle ne devra pas non plus être trop élevée afin d'inciter les automobilistes à renoncer à leurs véhicules particuliers. Les opérateurs qui répondront à l'appel d'offres devront alors équilibrer leurs comptes pour arriver à un coût d'abonnement de 15 euros par mois, augmenté d'une tarification à la demi-heure d'utilisation allant de 4 à 6 euros.

Plusieurs outils sont envisagés afin d'évaluer la disponibilité des véhicules. L'exploitant suivra par exemple en temps réel le taux de remplissage des stations et pourra, le cas échéant, identifier les stations nécessitant un réapprovisionnement. Par ailleurs, les usagers disposeront également de supports d'information en temps réel afin de connaître l'état du parc et pourront être orientés sur les véhicules disponibles immédiatement à proximité.

Toujours en France, la société VU Log propose des solutions logicielles pour la gestion de systèmes de transport urbain, adaptées pour les opérateurs de mobilité urbaine, les collectivités locales, les entreprises et autres complexes logistiques. Pionnier des systèmes d'information au service de la mobilité urbaine, VU Log propose une solution qui gère la mise à disposition et le suivi de véhicules urbains de proximité en libre service ou en auto-partage, avec et sans réservation pour des véhicules électriques ou thermiques à 2 ou 4 roues. Issue des laboratoires de l'INRIA, la société VU Log valide toutes ses solutions depuis plus de 2 ans, dans son laboratoire, et en réel dans la ville d'Antibes avec plus d'une dizaine de véhicules électriques et thermiques en circulation 24h/24, 365 jours par an. De par son historique et son expérience en réel à Antibes, VU Log a également fortement investi dans les logiciels de back office (sites Web, systèmes informatiques), ce qui leur permet, au travers de leur centrale de mobilité Cité VU, de gérer au mieux la flotte de véhicules.

Très vite, des fonctionnalités de conduite entièrement automatisée ont vu le jour sur ces véhicules, en particulier pour la marche en convoi autonome. Ceci s'intègre dans le cadre des développements pour les transports publics urbains (désignés par l'acronyme ITS : Intelligent Transportation Systems). Afin de classifier chaque dispositif de transport selon la souplesse dont il fait preuve, on distingue par la suite les solutions soumises à une infrastructure pour la navigation de celles qui en sont libres.

Les véhicules automatiques : ***avec infrastructure dédiée ou systèmes intégrés à la route*** Le principal mode de transport individuel se déplaçant sur un chemin entièrement dévolu qui a été étudié est le Personal Rapid Transit (PRT), que l'on pourrait traduire en français par "transport personnel automatisé". Il s'agit d'un moyen de transport collectif léger permettant de se déplacer à la demande et sans arrêt intermédiaire dans de petits véhicules indépendants. Le principe de commande considère des logements virtuels qui caractérisent la position et l'évolution désirées des véhicules le

long de la piste, cette dynamique étant définie d'une manière cohérente entre les différentes boucles du réseau de navigation. Chaque véhicule réel se voit alors attribuer un de ces emplacements virtuels et l'erreur de positionnement est régulée par une boucle de commande.

Afin d'obtenir une meilleure représentation du concept de PRT, une description générale plus détaillée est donnée dans les paragraphes suivants à travers un exemple.

Le dispositif le plus innovant est proposé par la société anglaise *Advanced Transport Systems Ltd*, dont le dispositif est composé de véhicules électriques pilotés par ordinateur et se déplaçant le long de "rampes" de guidage. Les véhicules, nommés *ULTra* comme *Urban Light Transport*, sont disponibles sur demande 24 h sur 24. Le chemin de guidage utilisé est entièrement passif, sans éléments mécaniques ou systèmes d'alimentation ; seul des rebords sont visibles pour permettre un guidage optique le long des voies et des pneumatiques en caoutchouc assurent alors le contact avec la surface de navigation.

Les modules, illustrés Fig. 1.7-b), peuvent accueillir 4 passagers munis de bagages et sont aussi entièrement accessibles aux poussettes et fauteuils roulants. Leur poids léger (820kg) et leur faible rayon de braquage (5m) leur permettent de parcourir des itinéraires complexes en s'appuyant sur une infrastructure adaptable (Fig. 1.7-a). Le dispositif peut ainsi opérer à plusieurs niveaux (tunnels, en élévation) et peut s'intégrer à une architecture de bâtiments existante. Chaque station comprend un unique quai

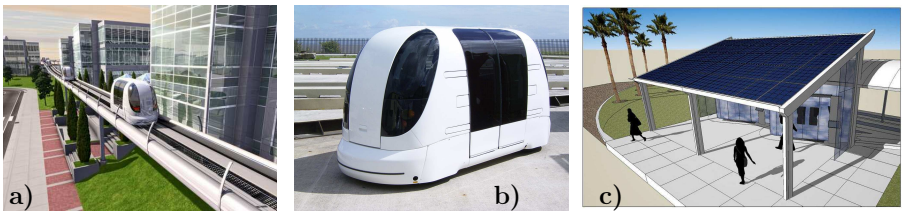


FIG. 1.7 – Dispositif de transport "ULTra"

composé de plusieurs espaces dédiés à la montée à bord permettant de gérer jusqu'à 400 passagers par heure. L'intégration de panneaux photovoltaïques (Fig. 1.7-c) est envisagée dans certaines stations, où le climat est propice, pour alimenter le dispositif en énergie solaire.

Malgré une navigation dans un environnement entièrement contrôlé sur une voie de guidage exclusive, les modules entièrement automatiques sont surveillés en tout temps par le personnel de la salle de contrôle. Ce dispositif de navigation rapide, qui est en développement depuis le milieu des années 90, est actuellement exploité sur deux sites : l'installation d'essai de Cardiff ainsi que le premier système commercial de transport personnel au monde ouvert au public en 2010 à l'aéroport de Londres Heathrow.

D'autres solutions, plus souples et moins coûteuses, s'appuyent sur des systèmes de perception intégrés à la route. Parmi les systèmes qui offrent de très bonnes perfor-

mances, plusieurs solutions technologiques peuvent être distinguées : par exemple les systèmes filoguidés, ceux à base de balises magnétiques ou de transpondeurs.

Dans les applications basées sur le filoguidage, les véhicules sont en mesure de se localiser par rapport au chemin de référence via un câble placé suivant cette référence. Des capteurs mesurent le champ créé par le câble, souvent intégré dans la chaussée et parcouru par un courant alternatif. Un exemple, très simple sur le plan de la commande, est proposé par le projet BURST (Bright URban System for Transportation). Il devrait permettre le convoyage autonome de véhicules urbains nommés “Serpentine”, illustrés par la figure 1.8 [Gillet 99]. Ceux-ci sont alimentés par induction à partir d'une voie active enfouie dans la chaussée, sans contact direct. De plus, celle-ci permet également de réaliser les mesures nécessaires pour la commande par filoguidage. Un site pilote de démonstration sur les quais d'Ouchy de Lausanne (Suisse) est en place depuis 2001. A terme, ce projet devrait permettre d'équiper l'ensemble de l'agglomération de Lausanne avec ce type d'engins.



FIG. 1.8 – Exemple d'une flotte de véhicules “Serpentine”

Des balises magnétiques, placées à la surface de la voie peuvent aussi être utilisées pour décrire le chemin de référence. Le projet le plus ambitieux relève de la thématique de l'autoroute automatisée (AHS : Automated Highway System) ciblée dans le cadre du projet PATH (California Partners for Advanced Transit and Highways). Ce dernier ne concerne pas le milieu urbain ; il sera donc détaillé ultérieurement en section 1.1.5. Par contre dans certaines villes, des systèmes de guidage magnétique permettent le transport en commun en site propre (TCSP). Le tracé de repères magnétiques est uniquement implanté sur les voies ou espaces qui leur sont réservés, en parallèle aux voies de circulation classiques, permettant ainsi d'alléger des axes lourdement utilisés pendant les heures de pointe. La ligne de transport peut être entièrement équipée mais peut aussi emprunter des voies communes aux transports personnels où un conducteur reprend alors le contrôle du véhicule. Ce type de guidage offre de par sa précision la possibilité de s'arrêter à proximité du quai. Il en résulte un alignement de la marche sur le quai permettant une amélioration du flux de voyageurs entrants et sortants et une meilleure accessibilité aux personnes à mobilité réduite. A Eindhoven, Pays-Bas, un bus guidé nommé Phileas a été développé suivant cette technologie par Samenwerkingsverband Regio Eindhoven (SRE). Le conducteur dirige le bus uniquement en dehors des voies guidées ou sous certaines circonstances comme des conditions de tempête ou une chaussée à surface trop glissante. Le système permet de se déplacer avec une vi-

tesse moyenne de 30 km/h (incluant les arrêts), ce qui est mieux que le bus classique (évoluant aux alentours de 20 à 25 km/h) et similaire au métro.

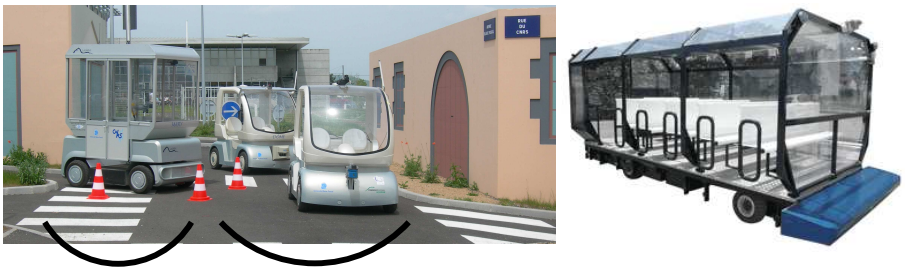
Une autre méthode, à coût moins élevé, consiste à utiliser des dispositifs optiques. Ce système de guidage repose sur le principe du traitement de l'image. Une caméra lit devant le véhicule le marquage peint sur la chaussée et matérialisant la trajectoire optimale. En France, la société Irisbus a déjà équipé les villes de Clermont-Ferrand et Rouen avec un bus piloté par guidage optique, appelé Civis. La société commercialise ce modèle sous le nom de tramway Civis bien qu'il soit réellement un bus guidé et non un tramway ou un tramway sur pneumatiques. Actuellement les bus à guidage optique (guidage Siemens Optiguide) sont des bus Agora et Citelis. Cette technologie de guidage équipe également des trolleybus Cristalis à Castellon (Espagne).

Tous ces capteurs offrent de bonnes performances de localisation. Cependant, il est nécessaire préalablement de baliser ou d'équiper l'infrastructure où évoluent les véhicules. Ceci engendre un coût élevé. Ainsi, à San Diego, l'Interstate 15 a été équipée d'aimants pour un coût proche de 10 000 dollars par mile, ce coût étant tout de même appelé à diminuer significativement lorsque le processus d'installation sera automatisé. Enfin, la portabilité des applications développées est quasi-inexistante si l'on souhaite s'affranchir de tels équipements et/ou infrastructures.

Les véhicules automatiques : *ne nécessitant aucune infrastructure ou systèmes intégrés à la route* La navigation dans un tel contexte est rendue ardue puisque l'environnement extérieur n'émet aucune information spécifique pour le guidage et ne permet pas non plus de maintenir le système au sein de la zone navigable. Une connaissance a priori de l'environnement est très recommandée et se présente souvent sous la forme d'une base cartographique métrique, plane et fixe. Un processus de planification exploite alors ce modèle de l'environnement afin de spécifier le déplacement des véhicules.

Ces considérations imposent aux véhicules d'évoluer sur voie propre sur un site protégé où ne circulent que des VIPs ou des piétons. D'ailleurs, d'un point de vue juridique, aucune réglementation ou procédure d'homologation n'existe à ce jour pour l'évolution de véhicules en mode automatique dans un flux urbain. Une certification peut néanmoins être obtenue pour la mise en place de tels dispositifs en site propre (grands parkings, hôpitaux, aéroports, parc d'attraction...).

En France, la société ROBOSOFT qui est issue de l'INRIA, a mis sur le marché une gamme complète de 6 véhicules électriques intelligents et modulaires. Le plus connu est le CyCab, illustré Fig. 1.9-a. On trouve des exemplaires dans le monde entier et 3 d'entre eux ainsi que 2 RobuCabs (Fig. 1.9-b) constituent d'ailleurs la base de développement et de recherche du LASMEA sur le guidage automatique. Les produits de cette gamme sont modulables et permettent des adaptations multiples en terme de design mécanique, d'équipement logiciel et d'accessoires robotiques, visant à proposer des solutions efficaces aux problématiques de transports automatiques rencontrées.



a) RobuCab **b) CyCab** **c) RobuRide**

FIG. 1.9 – Quelques véhicules de la gamme ROBOSOFT : a) CyCab, b) RobuCab et c) RobuRide

ROBOSOFT a notamment livré 3 RobuRides à Vulcania (Fig. 1.9-c), le Parc de l'aventure de la Terre situé au cœur du parc naturel régional des volcans d'Auvergne. Ces derniers, rebaptisés VolcanBuls, permettent aux visiteurs de faire un tour d'environ 1 kilomètre à travers le parc, pour observer la chaîne des Puys et se familiariser avec leur histoire. Aujourd'hui Vulcania est le premier site au monde dans lequel un tel système de transport automatique a été mis en place et certifié. L'absence d'infrastructure, mais aussi de conducteur, et son exploitation dans une zone partagée par des promeneurs en font un exemple particulièrement intéressant pour tous les sites accueillant du public, comme les centres-villes, les aéroports, les hôpitaux, les campus, les parcs d'attraction... Une vidéo de l'utilisation sur le site est disponible en suivant le lien *http ://www.youtube.com/watch ?v=E-1Qht1fGCY*

La localisation est assurée par un GPS différentiel, hybridé avec une centrale inertielle et de l'odométrie : en cas de perte momentanée du signal GPS, le relais est pris par la centrale inertielle. Cette solution conduit à une précision de suivi de trajectoire centimétrique, et ce jusqu'à 12 km/h, la vitesse maximum du VolcanBul.

Un exemple très représentatif est le programme Praxitèle, mené en France par l'INRIA (Institut National de Recherche en Informatique et en Automatique) dans le cadre du projet LA Route Automatisée (LARA), débuté en 1993. Les recherches, décrites de façon plus approfondie par la suite dans ce manuscrit, adressent des applications en libre accès aux véhicules et l'automatisation de voitures en convoi, dans le domaine de la route intelligente.

De plus, le programme Praxitèle a également donné naissance à des véhicules urbains électriques, équipés de moyens électroniques de gestion, de contrôle et de communication, nommés PRAXIS (ancêtres du Cycab, illustrés par la Fig. 1.10). L'expérience de Saint-Quentin-en-Yvelines, où ces engins ont été mis à la disposition des usagers, encourage de telles innovations à l'avenir.

Un autre exemple, le projet VIPA, est labellisé par le pôle de compétitivité ViaMéca et réunit le consortium formé par le LASMEA (Unité Mixte de Recherche CNRS /



FIG. 1.10 – Exemple d'un convoi de deux véhicules dans le cadre du projet Praxitèle

Université Blaise Pascal), le constructeur automobile LIGIER et le bureau d'études APOJEE. Le projet porte sur la mise en circulation du VIPA (Véhicule Individuel Public Autonome), un véhicule électrique de transport de passagers autonome en terme de conduite grâce à un système de localisation et de guidage basé sur le traitement d'images vidéo et d'algorithmes de localisation par vision en temps réel. Le cahier des charges des VIPAs comprend la navigation individuelle et en flotte sur des distances relativement courtes (entre 0,3 et 1,5km) et en présence de piétons et éventuellement de véhicules à faible vitesse. Le VIPA, dont la vitesse est limitée à 20km/h, peut accueillir jusqu'à 4 personnes et évolue dans des zones telles que des sites industriels, parcs, aéroports... Il ne nécessite aucune infrastructure de guidage ; équipé d'un ordinateur de bord et d'une caméra fixée à l'avant du véhicule, le VIPA se dirige grâce à sa mémoire visuelle, programmée en deux étapes :

• Hors exploitation : apprentissage de la trajectoire à suivre par simple mémorisation visuelle (enregistrement vidéo du parcours), puis extraction des "amers" visuels (objets particuliers facilement identifiables) les plus significatifs (portes, fenêtres, trottoirs...) et calcul de leurs positions 3D.

• En ligne : guidage automatique fondé sur le calcul, pour chaque nouvelle acquisition vidéo, de la position précise du véhicule à partir de la connaissance préalable des "amers" 3D et de leur observation dans l'image courante (triangulation).

Le principe a déjà été testé grandeur nature sur le site industriel d'Airbus à Toulouse pendant la période s'étalant du 6 au 17 octobre 2008. 4 véhicules de type CyCab, évoluant sur une voirie réservée, assuraient alors la liaison aller-retour sur l'avenue du Languedoc aux heures d'affluence (4 arrêts sur 1,5km). 1500 personnes ont pu être déplacées en toute satisfaction au cours de ces 10 jours ouvrables.

La mise en service public de 10 VIPAs est projetée pour la fin de l'année 2010 au niveau du CHU de Clermont-Ferrand. Les véhicules, au service des personnes à mobilité réduite, établieront le lien entre le parking et l'entrée de l'établissement hospitalier et reviendront tous seuls à leur point de départ. L'homologation de ce projet sera une première mondiale.

Si certaines contraintes liées à la conduite en milieu urbain sont à prendre en compte, de lourdes considérations peuvent également être éliminées. En effet, en étudiant les



FIG. 1.11 – Le VIPA présenté dans le cadre de la semaine de la mobilité à Clermont-Ferrand

spécificités propres à la conduite en milieu urbain, quelques hypothèses simplificatrices peuvent être introduites. Par exemple, l'environnement de travail peut être assimilé à un terrain non accidenté. De plus, compte tenu de leur faible vitesse de fonctionnement et du revêtement des surfaces de fonctionnement (généralement de l'asphalte), une hypothèse de roulement sans glissement est entièrement justifiée. Enfin, les véhicules urbains étant électriques, la modélisation du système de propulsion se trouve simplifiée comparée aux engins à combustion.

1.1.5 Applications de services de navigation dans un milieu non-urbain

Bien que le cadre d'application ainsi que certaines hypothèses de commande soient différents du milieu urbain, la mise en place de ces stratégies vise des objectifs similaires et leur étude reste très intéressante dans notre cas.

Constatant que le volume des biens transportés sur les routes européennes avait doublé au cours des années 80, des études sur le convoyage de marchandises ont été réalisées dans le cadre du projet CHAUFFEUR, afin d'alléger la charge de travail portée par les conducteurs ainsi que les effets négatifs de cette augmentation sur la sécurité routière et l'environnement. Sous la coordination de DaimlerChrysler, ce projet s'est penché sur de futures applications concernant les transports routiers automatiques, et notamment la navigation d'une flotte de véhicules et leur maintien sur la voie de circulation par vision. Les composants et systèmes nécessaires ont été développés, comme des capteurs vidéo, des contrôleurs pour le train directeur, le freinage et le braquage des roues. A l'issue du projet, des démonstrations impliquant des camions de DaimlerChrysler, IVECO et Renault ont été mises en œuvre dans des conditions de trafic réelles. Des convois de trois véhicules, illustrés Fig. 1.12 ont ainsi été guidés par un conducteur humain sur l'autoroute de Brenner. Ce travail a été accompagné d'un examen des aspects légaux et administratifs liés à l'application d'une "accroche virtuelle" entre les véhicules. Un premier rapport détaillé a été écrit et souligne les étapes nécessaires à la certification

et les problèmes potentiels. Un second décrit les pré-requis opérationnels et techniques pour la mise en œuvre d'un convoi automatisé.



FIG. 1.12 – Convoi de 3 camions réalisé dans le cadre du projet CHAUFFEUR

Cependant, le programme PATH (Partners for Advanced Transit and Highways) reste de loin le projet de plus grande envergure à ce jour. PATH est un programme multi-disciplinaire créé en 1986 et ayant pour mission principale d'appliquer les technologies de pointe pour augmenter à la fois la capacité et la sécurité des autoroutes, tout en réduisant les problèmes de congestion du trafic, de pollution atmosphérique et de consommation énergétique. Les recherches effectuées dans le domaine des Systèmes de Transports Intelligents (ITS) traitent un large champ d'applications comprenant aussi bien les systèmes avancés d'information et de gestion des transports (ATMIS : Advanced Transportation Management and Information Systems), les outils d'aide à la décision et de mesure des performances des équipements routiers (AVCSS : Systèmes Avancés de Commande et de Sécurité des Véhicules) et la commande totale des véhicules. Les applications traitant de la conduite automatisée de véhicules routiers, présentent des caractéristiques très particulières et s'éloignent assez des conditions rencontrées en centre-ville. Les véhicules, généralement à combustion, nécessitent une modélisation approfondie de la dynamique du moteur. Ensuite, le fonctionnement à vitesse élevée (typiquement autour de 100 km/h) et les voies de circulation de faibles courbures imposent de nouvelles considérations. Les objectifs de commande peuvent se retrouver modifiés. Ainsi, pour tenir compte des variations de vitesse, tout en assurant la sécurité du convoi, l'objectif de la commande longitudinale est de maintenir, non pas une distance, mais un écart temporel entre véhicules. Enfin, les paramètres de modélisation de ces véhicules ne sont pas toujours constants en cours de fonctionnement : par exemple, les variations de vitesse ou d'accélération, ainsi que les intempéries susceptibles d'être rencontrées, sont quelques uns des facteurs engendrant la variation de ces paramètres. Les travaux expo-



FIG. 1.13 – Convoi de 8 voitures réalisé dans le cadre du projet PATH

sés dans [Huppé 03] et [Huppé 04] traitent ainsi de l'influence des conditions routières et climatiques canadiennes. Un exemple de convoi formé de 8 voitures est présenté sur la Fig. 1.13. Les usagers des autoroutes équipées de balises magnétiques en Californie peuvent déjà se doter de systèmes d'aide à la conduite leur permettant de connaître en temps-réel l'état de l'infrastructure et du trafic.

Pour conclure cette section, il faut retenir dans un premier temps que la mobilité urbaine se développe actuellement selon un nouveau cadre conceptuel. Afin de s'accorder à la transformation des modes de vie dans les espaces métropolitains, l'adaptation des pratiques de mobilité suit une évolution tant sur le plan écologique que sur les services proposés. Cela nécessite le développement d'outils concernant l'organisation et la gestion de flottes de véhicules, l'utilisation de nouvelles technologies (informatiques, capteurs, réseaux de communication, ...) et l'implication de laboratoires de recherche scientifique. On retiendra également que des expérimentations dans ce sens ont déjà commencé. Ainsi des travaux de recherche sont disponibles et concernent des applications urbaines ou sur voies de navigation rapides. L'objectif de la section suivante va être de présenter les fondements théoriques impliqués dans les expérimentations qui viennent d'être citées, mais aussi de couvrir un domaine de recherche plus étendu afin d'aborder d'une façon globale la commande d'une formation de robots mobiles.

1.2 Commande de robots mobiles en formation

Un grand nombre d'applications transverses se rapportent à la commande de robots mobiles en formation, ayant comme dénominateur commun la nécessité de se déplacer de façon organisée. Par exemple, un groupe de robots peut être utile pour de la mani-

pulation en équipe ou le transport de gros objets. D'autres applications intéressantes sont l'exploration de surfaces spatiale ou terrestre, la surveillance, le déminage, la recherche et le sauvetage ou encore la cartographie d'environnements. Dans ce manuscrit, l'état de l'art concerne uniquement le domaine des transports, et plus particulièrement la navigation de robots mobiles avec une formation qui matérialise un convoi.

Dans la suite, la formulation du problème de navigation, ainsi que les fonctionnalités et technologies impliquées, sont brièvement introduites. Une notion particulière de stabilité, incontournable lorsqu'un système est représenté par une chaîne d'agents interconnectés, est ensuite amenée afin d'illustrer la problématique et de cerner les éléments clefs de l'application considérée. Les différentes lois de commande pour mettre en œuvre un convoi sont finalement présentées et discutées.

1.2.1 Navigation en convoi et fonctionnalités associées

Dans cette sous-section, il est important de définir au préalable les tâches qui sont indissociables à la notion de formation en convoi. Cela permettra dans un second temps de cerner des fonctionnalités susceptibles d'intervenir dans leur réalisation.

Spécifications de la navigation en convoi Cette discussion adresse le problème de la navigation autonome d'une flotte de véhicules, disposés en file et se déplaçant vers une destination commune. Sur la base de cette définition très générale du convoi, plusieurs distinctions relatives à la liberté d'action accordée aux véhicules sont à prendre en compte. La tâche de navigation peut par exemple consister à amener un robot en un point d'arrivée, en essayant de minimiser la distance à parcourir tout en évitant une succession d'obstacles divers. Les travaux exposés ici se positionnent dans le cadre plus strict du suivi de trajectoire, c'est-à-dire qu'un chemin va servir de référence pour le guidage des véhicules. La trajectoire peut être prédéfinie, déduite en fonction de la position des véhicules qui précèdent ou encore simplement retransmise. Le respect de cette contrainte s'exprime en terme d'écart vis-à-vis de cette trajectoire et est alors assuré par un contrôle latéral du véhicule.

La deuxième tâche, qui traduit la notion de convoi dans notre situation, est caractérisée par le maintien d'une distance inter-véhicule désirée. Cette dernière traduit une contrainte longitudinale. Plusieurs règles concernant la consigne de distance sont généralement considérées. La consigne peut être choisie constante, ne tenant pas compte de la vitesse des véhicules, ou peut être fonction de la vitesse d'avancement du véhicule. Dans le dernier cas la consigne est généralement proportionnelle à la vitesse du véhicule contrôlé mais peut aussi être une fonction quadratique de la vitesse. Il est important de remarquer que la première règle est plus exigeante en terme de commande puisque chaque véhicule doit aligner sa position, vitesse et accélération avec le véhicule qui le précède. En conséquence, si l'on impose une inter-distance constante, l'information nécessaire est plus conséquente. Cela permet en contre-partie d'obtenir une capacité de circulation plus importante.

Fonctionnalités importantes pour la navigation en convoi Pour atteindre ces objectifs, la capacité du système à se localiser est une fonctionnalité essentielle. La localisation permet en effet de situer l'ensemble des robots d'une part par rapport au chemin de référence mais aussi éventuellement les uns par rapport aux autres au sein du convoi en fonction de l'information disponible. La localisation peut être de deux types selon le système de perception utilisé. Elle est dite absolue si la position de chaque véhicule est donnée dans un repère absolu, fixe et commun à tous les véhicules et dans lequel le chemin de référence est également exprimé. Par opposition la localisation dite relative va permettre de positionner un véhicule en fonction des autres éléments du convoi.

La capacité à se localiser est directement liée aux capteurs embarqués et à la fonction de perception des informations extérieures au système. Une localisation absolue nécessite l'utilisation d'un système de perception absolu. On distingue alors les dispositifs implantés dans une infrastructure spécifique (systèmes filoguidés, à base de balises magnétiques ou de transpondeurs) et ceux qui sont embarqués sur les robots mobiles (capteurs GPS ou caméra se repérant dans un environnement virtuel en comparant l'image courante aux images clefs stockées dans la mémoire visuelle). D'autres capteurs vont permettre de se localiser de manière relative dans le convoi. Le champ d'action de tels systèmes (caméras, télémètres laser, capteurs ultrasons) est relativement restreint. De ce fait, ce type de capteurs va permettre de repérer uniquement les véhicules les plus proches qui ne sont pas masqués. Il faut souligner qu'ils jouent un rôle important dans la perception de l'environnement des robots et le repérage d'éventuels obstacles.

Enfin une autre fonctionnalité des plus intéressantes lorsque l'on considère une formation se traduit par la mise en place d'une communication entre les agents. L'apport de la communication ouvre un ensemble d'applications très diverses et permet, notamment par la signalisation de dangers en amont, une plus grande sécurité. L'usage de communications entre les véhicules permettrait également à tous les membres du peloton de pouvoir se représenter l'état du convoi dans son ensemble, alors que les dispositifs de perception relatifs ne peuvent qu'obtenir des informations locales. L'utilisation d'une communication de proximité peut aussi s'avérer utile pour obtenir des informations précises sur l'état des autres véhicules. En effet les capteurs en place mesurant l'état interne d'un véhicule sont très souvent plus précis que ceux permettant de mesurer une information extérieure.

1.2.2 Notion de stabilité du convoi

Comme on le verra par la suite, de nombreuses stratégies ont été élaborées afin de réguler la position d'un véhicule au sein du convoi, et une difficulté dans l'établissement des lois de contrôle consiste à assurer le respect de la contrainte longitudinale. On peut alors définir la stabilité (longitudinale) d'un véhicule comme suit : c'est la capacité de tout véhicule du convoi à suivre le profil (borné) de vitesse et/ou d'accélération de son prédécesseur avec une erreur d'espacement et/ou de vitesse bornée.

Cependant, lorsque la taille du convoi devient importante, il devient intéressant d'observer le comportement du convoi dans son ensemble, comme l'interconnexion de

tous les véhicules et non plus chaque véhicule séparément. On considère alors avec attention le phénomène où de petites perturbations, introduisant des erreurs d'interdistance à un certain moment dans le convoi, s'amplifient sans limite et se propagent à travers la chaîne de véhicules. Cette notion de stabilité du convoi a été principalement abordée dans le cadre du développement de systèmes autonomes sur autoroute ("Automated Highway System" en anglais) dans lesquels on cherche à maintenir, en toute sécurité, un flot de circulation optimal en terme de débit.

Afin de traduire mathématiquement cette notion intuitive de stabilité ("string stability" en anglais) plusieurs notations et définitions ont été proposées [Chu 74] [Eyre 98] [Middleton 91] [Peppard 74] [Seiler 04] [Swaroop 96]. Celles qui servent le plus souvent de référence se trouvent dans la thèse de D. Swaroop [Swaroop 94] et font l'objet du paragraphe suivant.

Définitions Avant de définir les différentes formulations de la stabilité, voici les notations utilisées :

$\|f_i\|_\infty = \sup_{t\geq 0}|f_i(t)|$, et $\|f_i(0)\|_\infty = \sup_i|f_i(0)|$

$\|f_i\|_1 = \int_0^\infty |f_i(\tau)|d\tau$, et $\|f_i(0)\|_1 = \sum_{i=1}^\infty |f_i(0)|$

où f est une fonction vectorielle du temps dont le nombre d'éléments $f_i(t)$ est infini. Par la suite, l'erreur de positionnement du $j^{ème}$ véhicule ϵ_j est quantifiée par rapport au $j-1^{ème}$ véhicule comme la différence de distance avec la consigne longitudinale. On utilisera aussi la variable $S_i(t)$ qui représente l'évolution dans le temps de la somme des erreurs des i premiers véhicules :

$S_i(t) = \sum_{j=1}^i \epsilon_j(t)$

• ***Stabilité du convoi :*** Un convoi est dit stable si :

$$\forall\gamma > 0,\ \exists\delta > 0 \text{ tel que}$$
$$\max\left[\|\epsilon_i(0)\|_\infty,\ \|\dot\epsilon_i(0)\|_\infty,\ \|S_i(0)\|_\infty,\ \|\dot S_i(0)\|_\infty\right] < \delta \Rightarrow \sup_i\|\epsilon_i\|_\infty < \gamma \tag{1.1}$$

Dans cette formulation, on lie l'erreur maximale de positionnement des véhicules dans le temps avec les erreurs initiales de chaque véhicule ainsi qu'avec les erreurs entre chaque véhicule et le véhicule de tête.

• ***Stabilité faible du convoi :*** Un convoi est dit faiblement stable si :

$$\forall\gamma > 0,\ \exists\delta > 0 \text{ tel que :}\ \ \max\left[\|\epsilon_i(0)\|_1,\ \|\dot\epsilon_i(0)\|_1\right] < \delta \Rightarrow \sup_i\|\epsilon_i\|_\infty < \gamma \tag{1.2}$$

Cette formulation lie l'erreur maximale de positionnement des véhicules dans le temps avec les erreurs initiales cumulées des véhicules.

Remarque : Les définitions qui viennent d'être présentées ne tiennent pas compte du nombre de véhicules dans le convoi, qui peut très bien être infini. Un convoi de taille infinie n'a pas de signification, cela traduit simplement le souhait que l'ampleur des erreurs de positionnement dans le convoi ne dépende pas du nombre de véhicules mais

uniquement de la configuration initiale du convoi (On suppose en effet, pour des raisons de simplicité, que les perturbations se produisent à l'état initial des véhicules.).

Caractérisation des différentes stabilités Une façon plus intuitive de se représenter la stabilité du convoi est donnée par certains critères de caractérisation étudiés dans [Swaroop 94]. On considère la fonction de transfert H_i qui lie l'erreur du $i^{ème}$ véhicule à son prédécesseur :

$$H_i(s) = \frac{e_i(s)}{e_{i-1}(s)}, \tag{1.3}$$

où s représente ici la variable de Laplace. Il est alors démontré que la stabilité du convoi est obtenue si $\forall i,\ |H_i(j\omega)| < 1$ pour toute pulsation ω. La stabilité faible s'obtient elle si $|H_i(j\omega)| \leq 1$.

Performances et stabilité : les éléments clefs On pourrait évaluer l'efficacité d'une stratégie de contrôle en convoi par la capacité de circulation (flux), la diminution des erreurs absolues d'espacement inter-véhiculaires qu'elle garantit et la quantité d'information nécessaire pour implémenter cette stratégie en temps réel. Dans la littérature les performances de certaines stratégies de contrôle ont été évaluées à l'aune de ces critères. On choisit ici d'introduire cet état de l'art en présentant les résultats obtenus par Swaroop au cours de sa thèse [Swaroop 94]. Ce dernier a en effet proposé plusieurs lois de commande et répertorié les performances longitudinales obtenues en examinant les éléments clefs d'une stratégie de navigation en convoi :

1. le type d'information utilisé sur l'état des véhicules (position, vitesse, etc.)
2. la décision d'instaurer une communication dans le convoi et sa topologie
3. la consigne de régulation longitudinale

La grande variété des paramètres étudiés fait de ces résultats une base solide pour aborder le problème de la navigation en convoi.

• ***Modèle standard de déplacement des véhicules :*** Une grande proportion des stratégies de commande longitudinale qui vont suivre se focalise sur des modèles linéaires simples formés de deux intégrateurs. En effet beaucoup d'attention a été portée sur les systèmes autonomes évoluant sur autoroute, ainsi l'utilisation de modèles dynamiques se justifie facilement compte tenu des dimensions et de l'inertie de tels véhicules. Cependant les modèles réalistes sont souvent très complexes et prennent en compte de nombreux paramètres (motorisation, freinage, aérodynamisme, etc.). Dans la pratique la plupart des stratégies sont réalisées suivant deux niveaux : le plus haut niveau concerne le contrôle de l'accélération et le plus bas niveau le contrôle de la motorisation et du système de freinage de sorte que le comportement du système réponde à la consigne formulée par le niveau supérieur. Dans cette étude on ne souhaite pas aborder le contrôle bas niveau, c'est pourquoi les modèles d'évolution considérés par la suite sont des systèmes d'ordre deux (contrôle de l'accélération) choisis très simples car de nombreuses approches, comme celles présentées dans ce paragraphe [Swaroop 94], s'appuyent sur un modèle qui a été linéarisé.

Une autre caractéristique de ces stratégies est que l'évolution des véhicules s'opère dans un espace unidimensionnel, ce qui signifie que les courbures de la trajectoire sont négligées. Cette restriction, qui n'est pas réaliste dans un cadre urbain, se vérifie sur des voies de circulation rapide constituées essentiellement de lignes droites. Cependant, il est important de remarquer que les limitations mises en évidence dans cette situation seront toujours présentes si l'on étend le déplacement des véhicules à un mouvement dans le plan, puisque dans le cas où les véhicules évoluent selon une direction, la modélisation des véhicules est similaire.

Ainsi pour traiter le problème de la stabilité du convoi, le modèle dynamique du véhicule considéré dans [Swaroop 94] s'écrit simplement :

$$\ddot{x} = u, \tag{1.4}$$

où x est la distance parcourue par le véhicule et u la commande imposée. La représentation du convoi est illustrée sur la Fig 1.14 et les erreurs longitudinales sont notées $e_{i,j} = x_j - x_i - (i-j)d^\star$ et $e_i = e_{i,i-1}$, où $d^\star$ est la consigne de distante entre les véhicules. L'intérêt de Swaroop se porte naturellement en premier lieu vers les stratégies

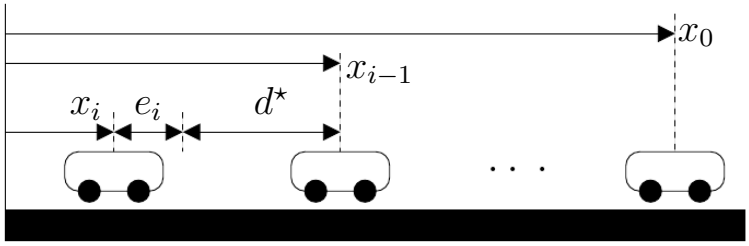


FIG. 1.14 – Représentation d'un convoi de véhicules

où la consigne $d^\star$ est constante, ayant ainsi une capacité de circulation maximale. Plusieurs solutions sont envisagées en fonction des informations utilisées sur les véhicules qui précèdent :

• ***commandes référencées sur le véhicule précédent :*** Afin que le système subisse le moins de perturbations possible par les délais de communication, des solutions minimales en terme d'information sont tout d'abord examinées. Deux commandes sont ainsi proposées ne nécessitant, pour chaque véhicule, que des informations concernant son prédécesseur : la première, donnée par l'équation (1.5) utilise uniquement des données pouvant s'obtenir avec les capteurs embarqués par le véhicule, à savoir la position et la vitesse relative du véhicule précédent, tandis que la deuxième, schématisée Fig. 1.14, intègre aussi l'accélération de ce dernier et par suite suppose une communication entre chaque paire de véhicules successifs (équation (1.6)).

$$u_i = -k_v \dot{e}_i - k_p e_i \tag{1.5}$$

$$u_i = k_a \ddot{x}_{i-1} - k_v \dot{e}_i - k_p e_i \tag{1.6}$$

Malheureusement la première solution est instable. On montre ci-dessous que la deuxième assure la stabilité au sens faible, mais qu'elle n'est pas robuste aux effets de bruit et de retard des actionneurs qui pourraient survenir.

Preuve : A partir de (1.4) et (1.6), on obtient facilement l'équation qui lie les erreurs de deux véhicules qui se succèdent :

$$\ddot{e}_i + k_v\dot{e}_i + k_p e_i = k_a\ddot{e}_{i-1} + k_v\dot{e}_{i-1} + k_p e_{i-1}$$

Ainsi, la fonction de transfert H_i s'exprime :

$$H_i(s) = \frac{k_a s^2 + k_v s + k_p}{s^2 + k_v s + k_p}$$

Pour la stabilité du convoi la valeur $k_a = 1$ s'impose alors : $k_a > 1$ implique que $|H_i(j\omega)| > 1$ à haute fréquence et $k_a < 1$ implique que $|H_i(j\omega)| > 1$ à faible fréquence. A l'opposé si $k_a = 1$ le critère de faible convergence est vérifié ($|H_i| \equiv 1$).

On modélise ensuite le retard en considérant que la commande calculée doit franchir un filtre passe-bas du premier ordre avant d'être appliquée au procédé, comme illustré Fig. 1.15. La commande calculée, notée w_i est alors différente de l'entrée du modèle (1.4) : $w_i = \tau\dot{u}_i + u_i$. La propagation des erreurs se traduit alors par :

$$H_i(s) = \frac{k_a s^2 + k_v s + k_p}{\tau s^3 + s^2 + k_v s + k_p}$$

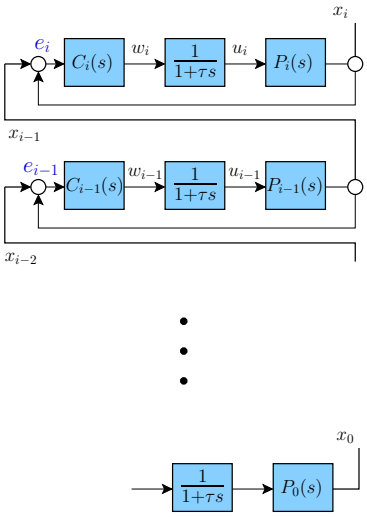


FIG. 1.15 – Modélisation du convoi "référencé véhicule précédent"

Ainsi, quelle que soit la valeur de τ on a alors $|H_i(j\omega)| > 1$ pour ω suffisamment petit. ■

• ***commandes référencées sur le leader :*** Les limitations précédentes peuvent se résoudre simplement par un asservissement référencé sur le véhicule leader, comme le décrit l'équation (1.7) :

$$u_i = \ddot{x}_0 - k_v\dot{e}_{i,0} - k_p e_{i,0} \tag{1.7}$$

De la même manière que précédemment, on montre que la commande est stable (au sens fort cette fois), cependant la mise en application est dangereuse car les informations du véhicule précédent ne sont pas prises en considération.

• ***commandes référencées sur le leader et le véhicule précédent :*** Afin de compenser les lacunes de chacune des deux précédentes méthodes, ce nouveau schéma est utilisé. La commande s'écrit :

$$u_i = f(\ddot{x}_{i-1}, \ddot{x}_0, \dot{e}_i, \dot{e}_{i,0}, e_i, e_{i,0}) \tag{1.8}$$

L'expression de f s'obtient en utilisant la théorie des systèmes plats. Comme on le désirait, cette dernière assure la stabilité (forte) du système et présente aussi des propriétés de robustesse (bruit et/ou retard des actionneurs).

• ***commandes référencées sur "r" véhicules précédents :*** La stabilité est aussi recherchée en utilisant les informations retransmises par plusieurs véhicules consécutifs :

$$u_i = \sum_{j=1}^{r} k_{aj}\ddot{x}_{i-j} - k_{vj}\dot{e}_{i,i-j} - k_{pj}e_{i,i-j}, \tag{1.9}$$

avec $x_{i-j} = x_0,\ \forall i \leq j$. Il est montré que la stabilité faible peut être obtenue en réglant convenablement les gains de commande, cependant la robustesse du système (bruit et/ou retard des actionneurs) n'est pas assurée.

Une commande avec une consigne d'interdistance proportionnelle à la vitesse du véhicule est finalement proposée en introduisant une nouvelle erreur longitudinale $\delta_i = e_i + h_w\dot{x}_i$, où h_w est la consigne de distance temporelle séparant les deux véhicules. La commande proposée est donnée par l'équation suivante :

$$u_i = -(\dot{e}_i + \lambda e_i)/h_w \tag{1.10}$$

Cette commande a l'avantage d'assurer la stabilité au sens faible par une méthode référencée sur le véhicule précédent uniquement. Sa robustesse est inversement proportionnelle à la distance temporelle imposée aux véhicules.

D'autres stratégies et combinaisons de stratégies que l'on ne mentionnera pas ici ont aussi été abordées par l'auteur. Un aperçu de la complexité de la problématique de commande (longitudinale) vient d'être exposé et cette première discussion sur un cas simple révèle l'intérêt de disposer d'informations en provenance d'une référence définie de façon appropriée. En effet, pour une consigne $d^\star$ constante, la commande basée sur les données du leader est la seule à pouvoir garantir la stabilité forte sur un plan théorique quelle que soit la longueur du convoi. De surcroît, il a été démontré récemment [Seiler 04] que la stabilité d'un convoi, dont le modèle de véhicule comporte un double intégrateur, ne peut pas être obtenue par une commande homogène (gains de commande identiques pour tous les véhicules) à partir des seules informations sur le véhicule précédent si l'on considère une consigne d'interdistance constante.

Après ces résultats théoriques on peut maintenant continuer sur l'état de l'art, qui traite en grande partie de la problématique longitudinale mais aussi de la régulation latérale. Les approches sont répertoriées selon la provenance des données utilisées lors du calcul des lois de commande. On distingue notamment les approches locales des approches globales.

1.2.3 Stratégies de commande locales

La plupart des approches citées dans la littérature s'appuyent sur des stratégies locales, c'est-à-dire que chaque véhicule est contrôlé à partir d'informations collectées dans son voisinage. Ces données peuvent être mesurées grâce aux capteurs disponibles à

bord ou retransmises, afin d'obtenir des informations plus précises et complètes sur l'état des véhicules à proximité. Les approches qui suivent une architecture de type *"leader-follower"* appartiennent à cette catégorie. Lorsqu'une formation générale de robots est ciblée, cela signifie qu'un groupe de robots suiveurs (follower) a pour tâche de suivre un autre robot qui joue le rôle de leader. Ce dernier peut se déplacer le long d'une trajectoire connue, être guidé manuellement par un opérateur ou jouer lui aussi le rôle de suiveur (on considère alors une hiérarchie de leaders). Dans le cadre d'un convoi de véhicules l'approche leader-follower, ou "de proche en proche", se traduit par une architecture de contrôle unidirectionnelle où chaque robot, mis à part le premier, s'asservit sur les informations de son prédécesseur et constitue la référence pour le suiveur. Dans la suite, les stratégies de contrôle unidirectionnelles sont tout d'abord présentées puis d'autres architectures plus complexes sont ensuite mises en avant.

Stratégies ayant une architecture de contrôle unidirectionnelle : Cette architecture se révèle très simple à mettre en œuvre et ne nécessite que peu ou pas de communication au sein du convoi, c'est pourquoi de nombreux auteurs ont choisi d'adopter cette solution. En revanche on a vu précédemment que la stabilité du convoi n'était pas garantie avec ce type d'architecture qui conduit à un comportement oscillatoire des véhicules à cause de l'accumulation des erreurs se propageant le long de la chaîne. Plusieurs solutions permettent cependant d'obtenir de bonnes performances longitudinales en pratique, si les convoi sont courts. Elles sont présentées ici selon trois catégories. La première considère un réglage des gains non homogène, c'est-à-dire que les véhicules sont contrôlés différemment selon leur position dans le convoi. La deuxième consiste simplement à relâcher la contrainte d'inter-distance entre les véhicules en passant de la consigne constante très exigeante à une consigne de distance temporelle plus souple. La troisième repose sur l'utilisation de modèles de véhicule plus simples, basés sur des équations cinématiques (pour ces derniers, la stabilité du convoi n'a que peu été explorée).

• ***Architecture de contrôle non homogène :*** Dans [Khatir 04] les auteurs se sont intéressés au problème de la stabilité du convoi avec une consigne d'interdistance constante en suivant une stratégie de type leader-follower. Ils considèrent le modèle dynamique suivant, qui généralise légèrement le modèle (1.4) :

$$m\ddot{x}_i + b\dot{x}_i = u_i,\ i = 1, 2, \ldots, N \tag{1.11}$$

où x_i représente de nouveau la distance parcourue par le $i^{ème}$ véhicule le long d'une droite, m sa masse et b est un coefficient d'amortissement. On a vu précédemment que ce problème n'avait pas de solution dans le cas homogène, c'est-à-dire lorsque chaque véhicule est contrôlé avec la même commande. Dans le cas présent les auteurs proposent de réguler l'erreur longitudinale $e_i = x_{i-1} - x_i - d^\star$ par un contrôleur de type PID où les coefficients correspondant à chacun des trois termes sont propres à chaque véhicule. Il est ainsi montré que la stabilité du convoi s'obtient au sens fort de façon itérative de la sorte : connaissant les gains KP_{i-1}, KI_{i-1} et KD_{i-1}, les gains du $i^{ème}$ véhicule

s'obtiennent en respectant les règles qui suivent :

$$KI_i \geq KI_{i-1} \tag{1.12}$$

$$KP_i = \frac{KI_i}{KI_{i-1}} KP_{i-1} + \frac{m}{KD_{i-1}} KI_{i-1} \tag{1.13}$$

$$KD_i = \frac{KI_i}{KI_{i-1}} KD_{i-1} + \frac{m}{KD_{i-1}} KP_{i-1} - b \tag{1.14}$$

Bien que ce schéma pointe des résultats de stabilité intéressants, les auteurs ne tirent aucune conclusion sur la robustesse de l'approche : les phénomènes perturbateurs fréquemment rencontrés en situation réelle (bruits, retard, etc.) ne sont pas introduits dans le modèle. Les équations (1.12) à (1.14) nous font remarquer que les gains croissent avec l'indice du véhicule dans le convoi et l'on peut se demander si dans la pratique des perturbations sur la commande ou des erreurs sur le modèle d'évolution n'exigeraient pas une différence plus importante dans le réglage des gains entre chaque paire de véhicules. Pour éviter d'avoir à subir des secousses de commande, dont l'intensité croît avec la valeur des gains, il faudrait alors contraindre le nombre de véhicules dans la chaîne. On constate par ailleurs qu'aucune expérimentation n'a à notre connaissance été menée suivant un tel schéma.

• ***Utilisation d'une consigne de distance temporelle :*** Un exemple intéressant est exposé dans le cadre du programme PRAXITÈLE [Massot 99]. La navigation d'une flotte de petits véhicules électriques est en effet réalisée avec succès. Le convoi, conduit par un véhicule de tête guidé manuellement, s'appuie sur un schéma de contrôle attrayant puisque chaque véhicule régule son comportement en exploitant uniquement les données mesurées par vision concernant son prédécesseur. Le dispositif vision comprend une caméra linéaire dont la fréquence d'acquisition s'élève à $1000Hz$. Elle est pourvue de filtres infrarouge et polarisé afin de minimiser l'influence de la lumière ambiante et de la réflexion. Cette caméra, montée à l'avant des véhicules (hormis le leader), permet de suivre une cible disposée à l'arrière des véhicules, composée de diodes agencées en lignes verticales. Ce dispositif permet de mesurer la position et l'orientation relative entre chaque paire de véhicules de façon très précise ($5mm$ d'erreur de position à $10m$). La vitesse relative est obtenue par filtrage des données de position reconstruites du véhicule précédent. Pour asservir la formation, chaque véhicule reçoit une commande latérale, commandant l'angle de braquage des roues directrices, et une commande longitudinale qui régule l'accélération du véhicule. Afin de simplifier le problème, les auteurs décident de séparer les deux commandes. Ce choix est discutable puisqu'il suppose que l'angle de braquage des roues n'a pas d'influence sur l'évolution longitudinale qui ne serait alors dictée que par l'accélération linéaire. C'est une approximation valide à faible courbure.

Plusieurs lois de contrôle, s'appuyant sur un modèle cinématique de type bicyclette, sont proposées dans [Daviet 96] pour la régulation latérale. Ce modèle, encore appelé modèle d'Ackermann, schématise le véhicule par une roue arrière motrice et une roue avant directrice. La commande la plus simple consiste à orienter la roue avant virtuelle du modèle en direction du véhicule précédent. Constatant que les véhicules coupaient la trajectoire pendant les courbes, deux nouvelles lois sont proposées. La première, à

nouveau en boucle ouverte, définit la commande par une formulation polynômiale plus complexe des variables d'état. Afin d'améliorer encore le recollement à la trajectoire de référence, la deuxième approche consiste à mémoriser pour chaque véhicule le chemin emprunté par son prédécesseur. L'écart latéral est alors défini comme la distance signée entre le point de contrôle du véhicule et sa projection sur la trajectoire. Une commande non-linéaire impliquant un modèle d'évolution d'ordre trois pour l'écart latéral est finalement proposée afin de réguler les véhicules sur la trajectoire. Malheureusement, aucun calcul n'est présenté sur l'obtention du modèle d'évolution latéral et par suite la commande est une heuristique donnée sans preuve de convergence.

Concernant la régulation longitudinale, la consigne de distance s'écrit classiquement selon l'équation (1.15) en considérant la vitesse du véhicule courant et la position des deux véhicules.

$$e_i = x_{i-1} - x_i - d_{min} - h\dot{x}_i \tag{1.15}$$

Un régulateur proportionnel-dérivé, qui n'est autre que la proposition de Swaroop (1.10) présentée à la section précédente, est utilisé afin de commander l'accélération du véhicule :

$$\ddot{x}_i = u_i \tag{1.16}$$

$$u_i = K_v \dot{e}_i + K_p e_i \tag{1.17}$$

Les gains K_p et K_v sont alors choisis de sorte que la commande appartienne à l'intervalle d'accélération réalisable pour le véhicule ($a_{min} \leq u_i \leq a_{max}$) et que chaque véhicule converge de façon asymptotique vers la distance désirée. La stabilité du convoi est alors assurée quel que soit le nombre de véhicules qui constituent la flotte. La dynamique longitudinale réelle des véhicules suit un modèle complexe qui a été simplifié en ne gardant que les termes prépondérant à basse vitesse ($\leq 50km.h^{-1}$). Elle est régulée par l'intermédiaire d'une commande bas-niveau qui agit sur la tension envoyée aux moteurs électriques et la pression des freins.

Une récente étude [Scheuer 09] a démontré que si l'on se place dans un cadre expérimental réaliste, c'est-à-dire que les accélérations et les vitesses des véhicules sont bornées, des collisions entre les véhicules peuvent survenir dans certaines configurations rendant du coup l'approche [Daviet 96] dangereuse. Le scénario, qui considère l'interaction la plus risquée entre deux véhicules, se décrit comme suit. Lorsqu'un problème se présente avec un véhicule, ce dernier va freiner et dans le pire des cas il suivra la valeur de décélération maximale. Une fois informé de la situation le véhicule suiveur, s'il est proche, fera de même pour éviter la collision. Dans ce cas de figure, il est facile de caractériser à chaque instant une distance de sécurité en fonction des interdistances, de la vitesse courante et de la décélération maximale des véhicules. Une solution est alors proposée afin d'éviter les situations risquées. Elle se traduit par le calcul d'une valeur seuil pour la commande en accélération assurant de ne pas empiéter dans la zone de sécurité définie précédemment. Des tests en simulation ont été réalisés en commandant chaque véhicule avec cette valeur seuil d'accélération : de bonnes performances longitudinales sont obtenues sans collision. La stabilité du convoi n'a pas encore été examinée avec cette commande.

Dans [Klinge 09] des résultats de synthèse intéressants sont fournis concernant cette topologie (communication de proche en proche ou absente) en suivant une consigne de distance temporelle. L'attention est focalisée sur les modèles d'évolution linéaires avec deux intégrateurs qui sont majoritairement représentés dans les applications de véhicules autonomes sur des voies rapides. Les auteurs ont mis en évidence et caractérisé un seuil h_0 de distance temporelle entre deux véhicules successifs en fonction du modèle, de la commande et des perturbations occasionnées sur la commande au delà duquel le convoi est stable. Ils ont aussi prouvé l'instabilité du convoi pour toute autre valeur h de consigne de distance temporelle inférieure au seuil h_0.

• ***Utilisation de modèles cinématiques :*** Dans certaines applications, et notamment celles qui s'inscrivent dans un cadre urbain, les véhicules considérés sont de dimensions modestes et circulent à des vitesses limitées sur des sols adhérents. Les phénomènes dynamiques ne sont pas prépondérants dans leur comportement, comme ils peuvent l'être à l'inverse dans le cas de poids-lourds circulant sur autoroute. Par conséquent la description du comportement par un modèle cinématique paraît la plus appropriée. Pour l'instant dans la littérature peu de travaux s'intéressent à la stabilité globale de ces approches. L'étude de cette caractéristique semble être focalisée sur les véhicules évoluant sur autoroute. Pourtant, l'avantage de ces approches est que les solutions proposées ne considèrent pas uniquement le contrôle longitudinal d'une flotte qui évolue en ligne droite ; l'évolution des systèmes est définie dans un espace bidimensionnel et le contrôle latéral s'ajoute à l'étude.

Dans [Xiang 10] les auteurs considèrent le traditionnel modèle unicycle de véhicule et le transforment pour exprimer les variations de la distance ρ et du cap relatif γ entre chaque paire de véhicules. Les lois de commande, très intuitives, sont définies proportionnelles aux deux variables précédentes :

$$v_i = k_\rho \rho_i,\ \omega_i = k_\gamma \gamma_i \tag{1.18}$$

où v_i et ω_i concernent respectivement la vitesse linéaire et angulaire du $i^{ème}$ véhicule. Afin de réduire la complexité de l'analyse d'un tel système (non linéaire), les auteurs ont choisi de réduire le comportement du véhicule de tête en deux modes élémentaires : l'évolution en ligne droite à vitesse linéaire constante et l'évolution circulaire avec les vitesses linéaire et angulaire constantes. Les positions d'équilibre pour ρ_i et γ_i des deux modes sont caractérisées en fonction des gains k_ρ et k_γ. La stabilité exponentielle des deux modes est ensuite prouvée localement par l'étude du système linéarisé tangent autour de ses points d'équilibre. Il est aussi montré que le réglage des gains peut être correctement ajusté pour permettre à l'ensemble des véhicules de suivre une trajectoire identique en mode de déplacement circulaire tout en conservant la liberté de manœuvre sur la consigne de distance entre les véhicules. Cela rend la transition entre les deux modes quasi immédiate puisque seule la consigne angulaire est modifiée. Cette propriété est aussi très intéressante car elle permet de procéder à un suivi de chemin alors que les véhicules ne possédent pas de formulation analytique de la trajectoire. Il est malheureusement regrettable que la stabilité du système ne considère que les déplacements à courbure constante du premier véhicule, d'autant plus que les contraintes de non ho-

lonomie interdisent ces changements de courbure sans passer par une vitesse linéaire v_1 nulle. La stabilité globale du convoi est examinée dans cette étude dans le premier mode : on considère encore une fois une flotte qui évolue en ligne droite et de plus à vitesse constante. La théorie de Lyapunov permet de prouver la convergence exponentielle de chaque véhicule, et ainsi du système global, à condition qu'une valeur seuil pour le gain k_γ soit dépassée.

Architecture de contrôle bidirectionnelle : Selon cette architecture, l'information utilisée ne concerne plus uniquement le véhicule qui précède mais on s'intéresse aussi au véhicule suiveur. Cette architecture découle naturellement du fait que la plupart des (bons) conducteurs humains utilisent les informations locales des véhicules juste devant ou derrière eux. Cette façon de faire couple chaque paire de véhicules adjacents pour former un système dynamique qui ressemble à une série de particules de masse / amortisseurs / ressorts, voir la Fig. 1.16. Modéliser les interactions des véhicules par ce phénomène physique est très intéressant car il est bien connu que les systèmes constitués d'une série de masses / amortisseurs / ressorts sont dissipatifs, c'est-à-dire que l'énergie du système tend à s'annuler et les distances entre les masses à se stabiliser sur la longueur du ressort. L'inconvénient potentiel d'une telle stratégie réside dans le fait que, si la stabilité n'est pas obtenue, les perturbations peuvent désormais se propager dans les deux sens du convoi.

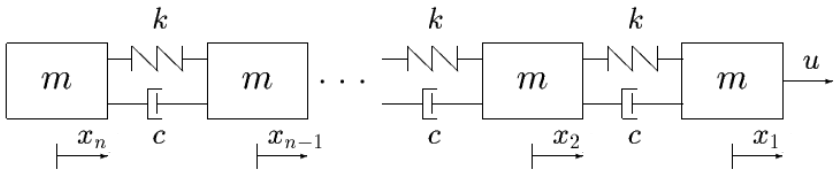


FIG. 1.16 – Modélisation Masses / Amortisseurs / Ressorts

Des résultats intéressants sur la stabilité sont donnés dans [Yanakiev 96]. L'évolution des véhicules est classiquement limitée à un déplacement en ligne droite et modélisée par un double intégrateur. On souhaite que le comportement des véhicules soit régi par une force de rappel de raideur k et une force d'amortissement d'élasticité c. Le bilan des forces appliquées à un convoi de N véhicules nous donne les équations dynamiques que l'on aimerait observer pour le système :

$$m\ddot{x}_1 = -k(x_1 - x_2 - d) - c(\dot{x}_1 - \dot{x}_2) + u \tag{1.19}$$

$$m\ddot{x}_i = k(x_{i-1} - 2x_i + x_{i+1}) + c(\dot{x}_{i-1} - 2\dot{x}_i + \dot{x}_{i+1}) \tag{1.20}$$

$$m\ddot{x}_N = -k(x_{N-1} - x_N - d) - c(\dot{x}_{N-1} - \dot{x}_N) \tag{1.21}$$

où m est la masse des véhicules, d la longueur du ressort (non comprimé) et u la force appliquée à la première masse qui représente le leader. En prenant le membre de droite des équations (1.19) à (1.21) pour commander l'accélération des véhicules, les auteurs

montrent alors que la stabilité du convoi s'obtient à la condition que le ratio c^2/km soit supérieur à une certaine constante qui augmente en fonction du nombre de véhicules dans le système.

Une nouvelle modélisation du système est alors étudiée afin d'avoir cette fois-ci une consigne de distance entre les véhicules qui dépend de la vitesse :

$$m\ddot{x}_1 = -k(x_1 - x_2 - d) - c(\dot{x}_1 - \dot{x}_2) - kh\dot{x}_1 + u \tag{1.22}$$

$$m\ddot{x}_i = k(x_{i-1} - 2x_i + x_{i+1}) + c(\dot{x}_{i-1} - 2\dot{x}_i + \dot{x}_{i+1}) - kh\dot{x}_i \tag{1.23}$$

$$m\ddot{x}_N = -k(x_{N-1} - x_N - d) - c(\dot{x}_{N-1} - \dot{x}_N) - kh\dot{x}_N \tag{1.24}$$

où h est la consigne de distance temporelle entre les véhicules. Il est montré dans ce cas que les conditions pour obtenir la stabilité du convoi sont moins restrictives que précédemment.

Dans [Yi 04] et [Contet 09], les auteurs considèrent une consigne d'interdistance constante et la principale amélioration avec la modélisation précédente (équations (1.19) à (1.21)) concerne l'espace d'évolution des véhicules. Ces derniers se déplacent dans le plan et l'état du $i^{ème}$ véhicule s'exprime alors selon deux composantes $X_i = (x_i, y_i)$. Les forces s'exerçant sur le système mécanique virtuel sont alors orientées et les équations dynamiques du système s'expriment comme suit :

$$m\ddot{X}_1 = -k(X_1 - X_2 - d_{1,2}) - c(\dot{X}_1 - \dot{X}_2) \tag{1.25}$$

$$m\ddot{X}_i = k(X_{i-1} - X_i - d_{i-1,i}) - k(X_i - X_{i+1} - d_{i,i+1}) + c(\dot{X}_{i-1} - 2\dot{X}_i + \dot{X}_{i+1}) \tag{1.26}$$

$$m\ddot{X}_N = -k(X_{N-1} - X_N - d_{N-1,N}) - c(\dot{X}_{N-1} - \dot{X}_N) \tag{1.27}$$

où $d_{i,j} = d^\star(X_i - X_j)/\|X_i - X_j\|$ et $d^\star \in \mathcal{R}^+$ est la distance euclidienne désirée entre les véhicules.

Dans [Yi 04] une modélisation cinématique des véhicules est finalement utilisée et permet de prendre en compte les contraintes de non-holonomie des véhicules. Les équations de la représentation d'état d'un robot unicycle sont dérivées pour obtenir un modèle dynamique dont on ne considère que les composantes liées à la position du véhicule (plus d'orientation). Ensuite, des commandes d'accélération linéaire et angulaire sont calculées par inversion du précédent modèle de sorte que les équations (1.25) à (1.27) soient vérifiées. La stabilité du convoi est prouvée en montrant que l'énergie du système est bien décroissante au cours du temps. Cette stratégie a l'intérêt d'être directement applicable puisqu'elle considère un modèle de véhicule qui n'est pas limité à un simple mouvement en ligne droite, cependant on constate que les véhicules ne suivent pas parfaitement la même trajectoire et ont tendance à couper les courbes. En effet, comme le montrent les équations (1.25) à (1.27), les forces qui caractérisent le modèle masse/amortisseur/ressort agissent sur le véhicule i selon un vecteur qui combine les directions $X_{i-1} - X_i$, $X_i - X_{i+1}$, $\dot{X}_{i-1} - \dot{X}_i$ et $\dot{X}_i - \dot{X}_{i+1}$ qui sont cohérentes par rapport à l'état courant du leader mais qui ne tiennent pas compte de la trajectoire empruntée par celui-ci.

Dans [Contet 09], une quatrième force F_t traduisant une contrainte latérale est introduite et permet d'obtenir de meilleurs résultats en courbe. Bien que se déplaçant

dans le plan, l'état des véhicules est alors exprimé dans l'espace (la composante selon l'axe vertical z est nulle) afin d'utiliser un calcul de produit vectoriel :

$$F_t = (X_{i+1} - X_i) \wedge k_m \theta^i_{i+1} z \tag{1.28}$$

où k_m est un paramètre réel et θ^i_{i+1} est l'orientation relative entre les $i^{ème}$ et $i+1^{ème}$ véhicules. F_t est orienté dans une direction orthogonale au vecteur $X_{i+1} - X_i$ ce qui peut intuitivement paraître bénéfique pour compenser le décalage à la trajectoire des véhicules en courbe. Pourtant le réglage du gain k_m n'est pas explicité et aucun argument théorique n'est formulé pour garantir le résultat. La stabilité d'un tel convoi, qui a été prouvée lorsque les véhicules évoluent en ligne droite ([Yanakiev 96]) n'est malheureusement pas abordée dans ([Contet 09]) dans le cas où des courbes sont présentes.

Ces travaux ([Yanakiev 96] [Yi 04] [Contet 09]) prouvent que l'architecture de contrôle bidirectionnelle est prometteuse. Elle apporte en effet des résultats théoriques intéressants sur la stabilité du convoi lorsque le nombre de véhicules est limité. Par ailleurs, la modélisation par une série de masses / amortisseurs / ressorts, répandue pour décrire les interactions dans un environnement incertain, présente ici des propriétés de robustesse aux incertitudes sur le modèle et au bruit des capteurs. La simplicité de cette stratégie comportementale affiche cependant certaines limitations pratiques assez conséquentes. En cas de défaillance d'un véhicule par exemple, le convoi entier va se trouver bloqué. Pour des applications dans un cadre urbain, ces solutions ne sont pas encore vraiment adaptées, puisque la flotte n'évolue pas en suivant la trace du leader : chaque véhicule coupe un peu plus la courbe empruntée par son successeur dans les virages et le passage du convoi sur des routes étroites n'est pas envisageable.

Architecture de contrôle cyclique : Une modélisation originale est proposée dans [Rogge 08] puisque les véhicules sont représentés par une topologie circulaire. Il ne s'agit pas d'une configuration géométrique des véhicules, ces derniers évoluant en ligne droite, mais de la transmission des informations : chaque véhicule est informé sur la distance le séparant de son prédécesseur et le véhicule de tête sur sa distance avec le dernier véhicule. Le modèle qui caractérise les véhicules est un point de masse en mouvement (similaire à celui utilisé dans [Khatir 04], voir eq. (1.11)) :

$$\ddot{x}_i + p\dot{x}_i = u_i,\ i = 1, 2, \ldots, N \tag{1.29}$$

où p est un paramètre de friction par unité de masse (ici la masse est fixée à 1). La commande u_i des véhicules s'écrit de la sorte :

$$u_i = K(x_{i-1} - x_i - L_i - h\dot{x}_i) \tag{1.30}$$

Ainsi chaque véhicule mesure sa distance avec le véhicule précédent et seule une communication est nécessaire entre le premier et le dernier véhicule. Selon cette topologie, il n'y a pas de leader proprement dit pour imposer un profil de déplacement. Ce sont

les paramètres L_i et h qui permettent de régler respectivement les inter-distances δ_i et la vitesse α du convoi qui sont liées par les relations :

$$\alpha = (-K \sum_{j=1}^{N} L_j)/N(p + hK) \tag{1.31}$$

$$\delta_i = 1/N \sum_{j=1}^{N} L_j - L_i \tag{1.32}$$

Dans le cas classique où on souhaite que les distances δ_i entre les véhicules soient identiques ($= \delta$), alors chaque véhicule calcule ses propres paramètres comme suit :

$$L_i = N\delta + L_1 \tag{1.33}$$

$$h = -(L_1 + (N-1)\delta)/\alpha - p/K \tag{1.34}$$

Cela suppose que les valeurs de N, L_1 (choisi arbitrairement) et δ soient connues par avance. Si l'on souhaite modifier la vitesse du convoi sans modifier la consigne δ, il suffit de régler la valeur de h. Si par contre la valeur de h est fixée et que l'on modifie L_1 alors la consigne δ sera ajustée en conséquence et la vitesse α suivra une courbe proportionnelle à δ. Les deux règles de suivi des véhicules (consigne constante ou proportionnelle à la vitesse) sont donc envisageables de cette façon. Une communication est tout de même nécessaire pour retransmettre de nouvelles valeurs pour les paramètres N, L_1 et δ (les deux dernières peuvent être définies par des fonctions continues) lorsque l'on souhaite modifier la vitesse et/ou la consigne longitudinale du convoi. Il a finalement été prouvé que le comportement induit par cette topologie respectait les contraintes de stabilité du convoi et était robuste aux phénomènes de perturbation.

1.2.4 Stratégies de commande globales

Contrairement aux approches abordées précédemment, dans le cadre des stratégies de commande globales chaque véhicule du convoi est contrôlé à partir de données partagées par l'ensemble de la flotte. Les capteurs extéroceptifs de proximité sont donc insuffisants et cette catégorie de stratégies se traduit par la nécessité d'adopter un moyen de communication approprié. Cette particularité peut apparaître comme une contrainte, toutefois le développement de stratégies globales vise à l'amélioration des performances du système en terme de stabilité face à des perturbations inattendues agissant sur les véhicules. Comme il a été illustré en introduction de cet état de l'art (section 1.2.2) avec les travaux de thèse de Swaroop, ces objectifs s'atteignent plus aisément lorsque des informations globales sont accessibles. C'est pourquoi l'introduction d'un outil de communication est souvent justifiée. Comme on va le découvrir par la suite, le choix de la topologie de communication va encore jouer un rôle clef dans la mise en pratique finale de l'application. On distingue à présent deux types d'architecture de contrôle. Si la totalité des informations concernant la formation doit être centralisée pour pouvoir générer les signaux de commande de chacun des agents, on parle alors d'architecture centralisée. Les données peuvent être soit collectées par un agent externe, par exemple

une station de contrôle, soit par un agent relais intégré à la formation. Lorsqu'une architecture décentralisée est adoptée, chaque agent exploite l'information qu'il obtient sur une partie restreinte des membres de la formation.

Remarque : Il existe plusieurs niveaux de "décentralisation". Dans la littérature on parle parfois de "high decentralized scheme/approach", cela fait référence aux stratégies qui ne nécessitent aucune communication et où le contrôle de chaque robot s'appuie uniquement sur des mesures issues de ses propres capteurs.

Architecture de contrôle centralisée : L'avantage de ces stratégies est de tenir compte de l'ensemble des interactions entre les différents sous-systèmes. On se place ainsi dans des conditions idéales pour l'élaboration de lois de commande et pour assurer la stabilité d'une formation. Paradoxalement certaines approches adoptent une architecture centralisée mais exploitent uniquement des informations locales pour asservir la flotte. C'est le cas dans [Das 02] où de simples contrôleurs sont réalisés pour effectuer des tâches de suivi de leader et d'évitement d'obstacles perceptibles dans le champ de vision. La centralisation est en fait utile pour la localisation des robots puisque la seule utilisation de caméras omnidirectionnelles ne leur permet pas de se localiser indépendamment dans la formation avec une grande précision. L'approche proposée tient compte d'informations angulaires partagées par les véhicules, pour retrouver la pose de la formation.

Parmi les approches où les informations globales sont intégrées au schéma de contrôle, deux catégories ont été envisagées dans la littérature. Une première façon d'aborder le problème de commande en formation est de le formuler comme un problème d'optimisation. A partir de l'erreur de formation d'un véhicule, calculée entre ce dernier et ses voisins, une fonction de coût est établie comme étant la somme des erreurs individuelles. Un exemple de contrôle optimal est donné dans [Levine 66] où une flotte de véhicules qui évoluent à grande vitesse est régulée avec une consigne de distance constante. Le contrôle de chaque véhicule est étudié afin de minimiser les erreurs de positionnement de chaque véhicule du convoi et nécessite ainsi la position et la vitesse de tous les véhicules de la chaîne. Comme on le verra dans le paragraphe suivant le problème d'optimisation peut aussi être formulé pour être traité de manière décentralisée (voir la référence [Gil-Pinto 07]).

Une autre méthode consiste en l'utilisation de coordonnées généralisées. Il s'agit habituellement de caractériser la position de la formation (P), son orientation (O) et sa forme (S) par rapport à un point de référence dans la formation. On suppose que la trajectoire, ainsi que la dynamique d'évolution du point de référence le long de la trajectoire ont été planifiées à l'avance. Sur cette base, les trajectoires des véhicules appartenant à la formation peuvent alors être spécifiées en temps réel en terme d'erreurs exprimées dans le nouveau système de coordonnées. Encore une fois, des réponses à ce type d'approche peuvent aussi bien se présenter sous une forme centralisée que décentralisée. Une stratégie centralisée se trouve par exemple dans [Caicedo 03]. Une analogie structurelle est établie avec la flotte de véhicules et l'évolution en ligne droite est contrôlée en s'appuyant sur des lois de la mécanique. Le système est ainsi régi par

les équations de Lagrange :

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}}\right) - \frac{\partial T}{\partial q} = Q \tag{1.35}$$

Dans cette équation , T représente l'énergie cinétique du système (il n'y a pas d'énergie potentielle) et Q les forces agissant sur ce dernier. Comme le montrent les équations (1.36) à (1.38), le vecteur d'état q est choisi pour faire ressortir deux éléments clefs : la localisation de la formation, assimilée à son centre de masse x_c, et sa forme géométrique définie par les distances $x_{i,j}$ entre les véhicules

$$q = [x_c,\ x_{2,1},\ \ldots,\ x_{N,N-1}]^T \tag{1.36}$$

$$x_c = \frac{1}{m}\sum_{i=1}^{N} m_i x_i,\ \text{avec } m = \sum_{i=1}^{N} m_i \tag{1.37}$$

$$x_{i,j} = x_j - x_i \tag{1.38}$$

où x_i représente la position du véhicule sur son axe d'évolution. On définit ensuite la matrice R qui fait le passage entre les coordonnées inertielles et généralisées. Cette dernière permet de calculer T et d'exprimer Q en fonction du vecteur de commande u constitué des forces appliquées à chaque particule. Une fois le calcul de T et Q injecté dans les équations de Lagrange, on obtient un système dynamique de la forme :

$$M\ddot{q} = Ru \tag{1.39}$$

On considère ensuite un profil d'évolution désiré pour la formation q_{des}, continu et différentiable au minimum deux fois afin d'exprimer la dynamique de l'erreur $e = q - q_{des}$ en fonction de u et q_{des} :

$$M\ddot{e} = Ru - M\ddot{q}_{des} \tag{1.40}$$

La loi de commande s'obtient finalement par inversion du système dynamique obtenu de sorte que l'évolution de l'erreur dépende de deux matrices de gains C et K :

$$u = R^{-1}(M\ddot{q}_{des} - C\dot{e} - Ke) \tag{1.41}$$

$$M\ddot{e} + C\dot{e} + Ke = 0 \tag{1.42}$$

La théorie de la stabilité de Lyapunov est finalement utilisée pour déterminer les contraintes sur les matrices C et K. Pour que les conditions de stabilité soient vérifiées, il résulte au final que la commande u_i du $i^{ème}$ véhicule requiert la connaissance du centre de masse x_c du système et des distances $x_{i,i-1}$ et $x_{i+1,i}$ avec ses plus proches voisins.

Cependant, bien que la notion de commande globale implique l'existence d'une topologie de communication sur les véhicules, procéder selon une architecture de contrôle centralisée devient très vite un problème de partage des données, surtout lorsqu'un grand nombre d'éléments constituent le convoi. L'importante quantité d'informations à échanger les rend peu attrayantes et ces stratégies sont généralement délaissées car la mise en pratique se révèle très limitée par la technologie de communication disponible.

Un autre inconvénient majeur de ces stratégies réside dans la faible tolérence aux défaillances et problèmes techniques sur les robots : un problème de communication sur l'un d'entre eux bloque le calcul de la commande pour tous les véhicules, et une perte de mobilité sur un robot va affecter le calcul de la commande des autres véhicules, "lestés" par ce dernier.

Architecture de contrôle décentralisée : La conception des commandes décentralisées ne prend pas en compte toutes les interactions entre les différents sous-systèmes. On cherche effectivement à restreindre au maximum le coût des communications entre les agents. Une structure de commande efficace et intuitive, précédemment introduite à la section 1.2.2 avec les travaux de Swaroop [Swaroop 94], est proposée dans [Shladover 79] et [Sheikholeslam 89a] - [Sheikholeslam 89b] - [Sheikholeslam 90] pour maintenir une formation de convoi avec un espacement intervéhiculaire constant sur une voie de circulation rapide. Le partage d'information concerne alors l'état du véhicule leader et chaque membre du convoi est censé pouvoir recouvrer l'état de son prédécesseur. Dans [Shladover 79], la retransmission de la vitesse et la position du véhicule leader permet d'élaborer une loi de commande longitudinale qui assure la stabilité du convoi. Dans [Sheikholeslam 89a], [Sheikholeslam 89b], [Sheikholeslam 90] le modèle dynamique utilisé introduit les effets du poids du véhicule et de la résistance de l'air (ces deux termes sont simplifiés en supposant la route horizontale et l'absence de vent). La synthèse de lois de commande longitudinales nécessite en complément de connaître l'accélération du leader. Plusieurs solutions sont testées en fonction de l'information disponible sur le véhicule qui précède (en plus de celles du leader) : uniquement sa position ou le triplet position - vitesse - accélération. Dans chacun des cas on observe un comportement stable et sans oscillation du convoi. On constate aussi que l'erreur d'espacement décroît plus rapidement dans la chaîne si la dynamique de l'erreur avec le véhicule qui précède est connue.

D'autres stratégies, qui perçoivent la formation des véhicules d'une façon plus générale, considèrent un ensemble de données partagées qui ne se rapportent pas uniquement à un seul agent qualifié de leader. Ces stratégies s'appuyent communément sur la théorie des graphes et/ou représentent la formation par analogie avec une structure virtuelle. Ces deux notions sont explicitées dans les paragraphes qui suivent.

• ***Application de la théorie des graphes au contrôle d'une formation de véhicules :*** La théorie des graphes est en effet un outil important pour l'analyse de la stabilité d'une formation. Elle permet de représenter les interconnexions entre les robots et en particulier les échanges d'informations. Ce sont les connexions du graphe qui vont déterminer d'une part si l'approche est locale ou globale et auquel cas si l'approche est centralisée ou décentralisée. On choisit de présenter ces stratégies ici car de par leur souplesse elles permettent potentiellement d'aborder le problème de façon globale et décentralisée. Des travaux de synthèse ont été effectués sur l'application de la théorie des graphes à la navigation d'une formation de véhicules et des conclusions intéressantes ont permis de corréler la topologie d'un graphe à la stabilité mais aussi à la commandabilité du système. Dans [Tanner 04] et [Lafferriere 04] l'idée consiste à faire

apparaître de façon naturelle le Laplacien du graphe $L_G = D - Q$ dans les équations de commande des agents. Dans cette formule la matrice des degrés D est diagonale et chaque composante d_{ii} représente le nombre de connexions avec le $i^{ème}$ agent. La matrice d'adjacence Q voit elle sa composante d'indices (i, j) égale à un ou nulle si les agents i et j sont ou non connectés. Dans [Tanner 04], le comportement de chaque agent suit un modèle cinématique $\dot{x}_i = u_i$ (le vecteur d'état x_i est de dimension indéterminée) et la loi de contrôle appliquée à chaque véhicule pointe dans la direction qui correspond à la position du barycentre de ses voisins :

$$u_i = -\frac{1}{d_{ii}} \sum_{j \in J_i} x_i - x_j \tag{1.43}$$

$$\Leftrightarrow u = D^{-1} L_G x \tag{1.44}$$

$$u = [u_1, \ldots, u_n]^T \,,\; x = [x_1, \ldots, x_n]^T \tag{1.45}$$

où l'ensemble J_i contient les indices des agents connectés au $i^{ème}$ véhicule. Certains véhicules jouant le rôle de leader vont alors, de par leur position, déterminer la commande des autres véhicules. Il est alors montré que la topologie du graphe détermine complètement la contrôlabilité du groupe d'agents. Dans [Lafferriere 04] l'évolution des véhicules est représentée par un modèle linéaire du second ordre où les variables d'état sont découplées :

$$\frac{d}{dt}\begin{pmatrix} x_i^1 \\ \dot{x}_i^1 \\ \vdots \\ x_i^n \\ \dot{x}_i^n \end{pmatrix} = \begin{pmatrix} \begin{bmatrix} 0 & 1 \\ a_{21}^1 & a_{22}^1 \end{bmatrix} & & \\ & \ddots & \\ & & \begin{bmatrix} 0 & 1 \\ a_{21}^n & a_{22}^n \end{bmatrix} \end{pmatrix} \begin{pmatrix} x_i^1 \\ \dot{x}_i^1 \\ \vdots \\ x_i^n \\ \dot{x}_i^n \end{pmatrix} + B \begin{pmatrix} 0 \\ u^1 \\ \vdots \\ 0 \\ u^n \end{pmatrix} \tag{1.46}$$

On considère la structure virtuelle suivante : chaque véhicule est en formation si sa position vérifie $x_i - h_i = q$, où les termes constants h_i déterminent la forme de la flotte et q représente la position courante désirée (commune à la flotte). Pour chaque véhicule i, la fonction d'erreur donnée par l'équation (1.47) est calculée à partir de la moyenne des erreurs de placement de l'ensemble des robots connectés et fait ainsi à nouveau apparaître le Laplacien du graphe (eq. 1.48).

$$e_i = (x_i - h_i) - \frac{1}{|J_i|} \sum_{j \in J_i} (x_j - h_j) \tag{1.47}$$

$$\Leftrightarrow e = D^{-1} L_G (x - h) \tag{1.48}$$

$$x = [x_1, \ldots, x_n]^T \,,\; h = [h_1, \ldots, h_n]^T \,,\; e = [e_1, \ldots, e_n]^T \tag{1.49}$$

Le contrôle est dicté par le retour d'état $u = Fe$, où la matrice F contient les gains de la commande. Un critère est développé à partir des valeurs propres du Laplacien du graphe pour déterminer l'effet de la topologie des communications sur la stabilité. La preuve est apportée que la stabilisation par retour d'état est toujours possible si le graphe de communication est connecté, c'est-à-dire qu'il existe au moins un chemin

reliant chaque paire de véhicules. De plus, le taux de convergence vers la formation est proportionnel à la plus petite valeur propre positive du Laplacien du graphe.

• ***Analogie avec une structure virtuelle :*** Ces stratégies sont très couramment proposées lorsqu'il s'agit de maintenir des formations aériennes ou pour le déplacement d'objet grâce à une coopération de robots. Le principe est de faire en sorte que le groupe de robots se comporte comme un unique robot rigide. Tous les robots de la formation maintiennent une position géométrique fixe relativement aux autres et il n'y a aucune distinction de hiérarchie entre les robots, pas de véhicule désigné comme leader. Pour cela, le contrôleur est synthétisé en trois étapes. Premièrement, la dynamique de la structure virtuelle est définie ; c'est elle qui forme l'information globale partagée par les véhicules. Ensuite, le mouvement désiré de la structure virtuelle est transposé vers les mouvements désirés pour chacun des agents qui composent la structure. Enfin, des contrôleurs individuels sont définis pour le suivi réalisé par chaque agent. Cependant, quelques observations critiques peuvent être émises à propos des structures virtuelles en général lorsque l'on considère une formation de type convoi :

i) Tout d'abord, de par sa rigidité, aucun réglage initial de la forme de la structure ne permet à l'ensemble des véhicules de suivre une trajectoire identique si cette dernière contient des courbes. L'évolution de véhicules en convoi le long d'une voie étroite n'est pas envisageable.

ii) D'autre part, la dynamique pré-établie de la structure virtuelle est une réelle contrainte dans des conditions réelles d'expérimentation. L'information partagée par les véhicules n'est pas actualisée et du coup le système n'est pas réactif si un quelconque imprévu nécessite le ralentissement, l'arrêt ou une déviation de la flotte par rapport à la trajectoire initiale.

Heureusement, dans certains travaux ces deux aspects ont été abordés. Dans [Barfoot 04], le problème i) est traité grâce à l'utilisation d'un système de coordonnées curvilignes. Plutôt que de maintenir une liaison rigide avec un leader virtuel, chaque véhicule va se positionner par rapport à une trajectoire. Pour définir cette dernière, les auteurs s'appuyent sur un modèle cinématique de type bicyclette. Il suffit alors simplement d'imposer deux fonctions qui traduisent l'évolution temporelle de la vitesse d'un point de référence C et la courbure de la trajectoire. La planification d'une trajectoire appropriée pour chaque véhicule se fait de façon similaire : un jeu de commandes en boucle ouverte est calculé afin de maintenir un écart latéral q_i et longitudinal p_i par rapport au point de référence. Comme le montre la Fig. 1.17, la formation est plus flexible et s'adapte dans les virages. On remarque aussi que le réglage $q_i = 0$ traduit une formation en convoi. Le maintien en formation est traité grâce à une planification effectuée hors-ligne, et en dehors de cette consigne de départ il n'y a plus d'information partagée entre les véhicules (ni de régulation pour compenser les erreurs de suivi de trajectoire).

Concernant le problème ii), des solutions sont apportées dans [Dierks 07] et [Gil-Pinto 07]. La structure rigide de la formation est conservée, cependant la dynamique n'est plus imposée mais dictée par un véhicule. Ces stratégies s'apparentent alors à des approches de type "leader-follower" mais présentées sous une forme globale puisque les

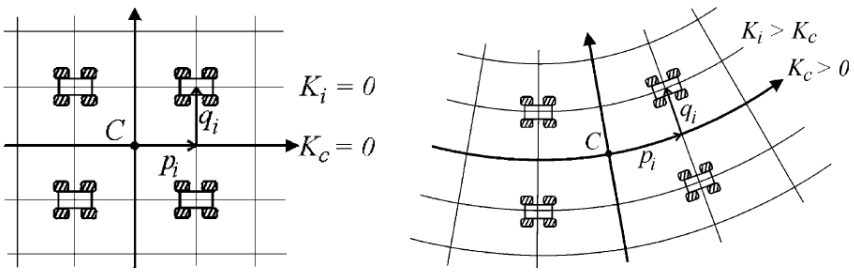


FIG. 1.17 – Formation carrée en ligne droite et en courbe

véhicules sont tous accrochés au même leader grâce à la retransmission d'informations le concernant. Aussi bien dans [Dierks 07] que dans [Gil-Pinto 07] la position désirée des suiveurs est spécifiée par un vecteur les séparant du leader. Sa longueur ainsi que son orientation, définies dans le repère local du leader, sont constantes. En représentant le comportement des véhicules par l'intermédiaire d'un modèle cinématique de type bicyclette, les auteurs obtiennent, après quelques calculs trigonométriques, un modèle d'évolution relative leader-suiveur.

Dans [Dierks 07] une première loi de commande est formulée permettant de maintenir la formation sous les hypothèses précédentes et la stabilité asymptotique du système est démontrée par la théorie de Lyapunov. Dans un deuxième temps, on suppose qu'une commande de vitesse envoyée n'est pas parfaitement prise en compte par le système. En tenant compte de la dynamique du système, une nouvelle loi est alors proposée et la stabilité asymptotique de chaque véhicule est à nouveau démontrée par la théorie de Lyapunov.

Dans [Gil-Pinto 07] on considère le maintien de la formation comme un problème de planification et le développement d'une boucle d'asservissement n'est pas abordé. L'approche est basée sur la recherche d'une trajectoire réalisable optimale, en considérant la propriété de platitude du modèle d'évolution des véhicules. A chaque instant un jeu de commande optimal est défini afin de mener les véhicules de leur configuration courante à celle désirée. Afin de simplifier le problème, l'évolution des sorties plates est approximée grâce à l'utilisation de courbes B-Splines, ce qui réduit la quantité de paramètres à optimiser au nombre de points de contrôle de la trajectoire. La convergence de l'algorihtme est prouvée, en tenant compte des saturations des véhicules (vitesse, braquage), si certaines conditions sur le déplacement du leader sont respectées.

1.2.5 Bilan

Au cours de cet état de l'art, les limites d'une commande de proche en proche, où chaque véhicule est asservi exclusivement sur le véhicule qui le précède, ont été présentées : des oscillations induites par l'accumulation des erreurs d'asservissement sont observées d'autant plus que le convoi est long. Plusieurs solutions sont envisageables

pour réduire, voire éliminer ce phénomène. On peut d'une part relâcher la contrainte d'interdistance et considérer une consigne temporelle, plus souple. En optant pour cette solution la stabilité théorique du convoi peut en effet être obtenue. Toutefois cela entraîne aussi une diminution des capacités de circulation et cette option n'est donc pas la plus intéressante pour adresser le problème des congestions urbaines dans les métropoles. Une autre façon de procéder serait de mettre en place des communications et de prendre en compte dans l'objectif de régulation l'état d'autres véhicules dans le convoi. L'utilisation d'une approche locale s'appuyant sur une architecture bidirectionnelle permettrait conjointement d'obtenir la stabilité du convoi et présenterait des propriétés de robustesse au bruit et aux erreurs de modèle. Ce qui est gênant, c'est que l'obtention de la stabilité nécessite d'imposer des gains de commande qui augmentent avec le nombre de véhicules dans le convoi. Pour l'application considérée ici, le nombre de véhicules susceptibles d'intégrer le convoi peut être élevé. Par conséquent, le parti a été pris de s'orienter vers une stratégie pouvant satisfaire le critère de stabilité forte tout en respectant le confort des passagers, et donc de mettre en place une communication globale au sein du convoi.

Afin de permettre une mise en pratique de la navigation en convoi dans des conditions réalistes et d'alléger le poids des communications dans la flotte, une architecture de contrôle décentralisée apparaît préférable. Certains travaux présentés dans les sections précédentes montrent l'intérêt de tout particulièrement prendre en compte l'état du véhicule de tête dans l'objectif de régulation. Afin de préserver la sécurité des passagers, il est également recommandé de s'informer de l'évolution du véhicule qui précède, permettant ainsi de détecter et d'anticiper un incident éventuel. Ce sont donc les deux communications que nous considérons dans ce manuscrit.

Dans la majorité des stratégies présentées précédemment, l'objectif était de réguler une distance euclidienne entre deux véhicules consécutifs. Il est tout à fait naturel de considérer une telle distance lorsque les trajectoires à suivre présentent une courbure faible (comme sur une voie de circulation rapide) ou lorsqu'il s'agit de transporter un objet grâce à une collaboration de robots mobiles. Par contre, dans le cadre d'un convoi évoluant le long de trajectoires présentant des courbures importantes, il serait plus judicieux d'exprimer l'état des véhicules vis-à-vis de la trajectoire à suivre. En effet les véhicules sont destinés à se déplacer dans des environnements urbains très contraints (voies de circulations étroites) et il est important d'assurer que chaque véhicule suive rigoureusement la même trace, ce qui n'est pas réalisable en considérant la régulation de distances euclidiennes. De plus, des comportements longitudinaux incohérents peuvent aussi survenir dans certains cas : par exemple, si deux véhicules se suivent en circulant à la même vitesse, alors leur inter-distance euclidienne va décroître dès que le premier aborde un virage serré. Ce problème peut être écarté simplement par l'utilisation de coordonnées curvilignes, en représentant la position de chaque véhicule selon son abscisse curviligne et son écart à la trajectoire.

Dans une application urbaine comme celle considérée ici, les véhicules évoluent à vitesse modeste sur des sols asphaltés. Par conséquent leur comportement peut raison-

nablement être décrit par un modèle cinématique. Il est de plus connu qu'il peut être converti sous une forme chaînée, permettant ainsi de linéariser de manière exacte le modèle et de tenir compte avec précision des effets non-linéaires induits par les courbures. Un grand intérêt de cette expression chaînée est aussi qu'elle permet de rendre indépendantes les performances de régulation latérale et longitudinale, ce qui est une propriété avantageuse dans l'optique de contrôler un convoi de véhicules.

Au LASMEA, des travaux de recherche ont déjà été effectués, développant les points pré-cités et des compétences ont été rendues opérationnelles notamment grâce à l'implication du laboratoire dans les projets de recherches cités en introduction. Ainsi cette thèse se situe dans un contexte actif et hérite d'une expérience qu'il convient de décrire ici.

1.3 Expérience capitalisée avant la thèse

Cette thèse se positionne dans la continuation des récents développements en automatique sur la navigation en convoi menés au laboratoire. Dans cette section, ces derniers sont tout d'abord présentés. Un bilan d'avant-thèse est ensuite dressé et permet de montrer l'intérêt des stratégies de localisation et de reconstruction 3D mises en œuvre pour la navigation autonome.

1.3.1 Navigation de véhicules en convoi

Au cours de ses travaux de thèse, J.Bom [Bom 06] a développé une approche adressant la mise en œuvre d'un convoi autonome de véhicules urbains électriques.

Il s'agit d'une stratégie de commande globale qui s'appuie sur des techniques de linéarisation exacte permettant de découpler les commandes latérale et longitudinale. Elle a été mise en œuvre avec la modalité capteur RTK-GPS. Néanmoins, avant de présenter ces lois de commande, il est nécessaire de bien modéliser le processus à contrôler, à savoir un convoi formé de plusieurs véhicules. Ensuite, à partir de cette modélisation, les lois de commande sont développées dans les sections suivantes.

1.3.1.1 Modélisation cinématique des véhicules

Le domaine d'application traité concerne la conduite de véhicules de faibles dimensions circulant en centre ville. De ce fait, plusieurs hypothèses simplificatrices peuvent être avancées. Nous supposons que les véhicules sont des corps rigides (distances entre les roues et inter-essieux constantes), symétriques par rapport à l'axe principal du véhicule, et se déplaçant à vitesse réduite. Il est ainsi possible de négliger les composantes dynamiques (suspensions...) et la déformation des pneumatiques. En outre, les zones de circulation dédiées aux robots sont constituées d'asphalte et le contact entre la roue et le sol peut être supposé sans glissement. Compte tenu de ces hypothèses, les lois de commande utilisées pour les véhicules se baseront sur un modèle cinématique et le mieux adapté est le modèle bicyclette (encore appelé modèle d'Ackermann). Ce véhicule est

schématisé par une roue arrière motrice et une roue avant directrice. Pour représenter le système, on introduit les notations suivantes, illustrées par les Fig. 1.18 et 1.19 :

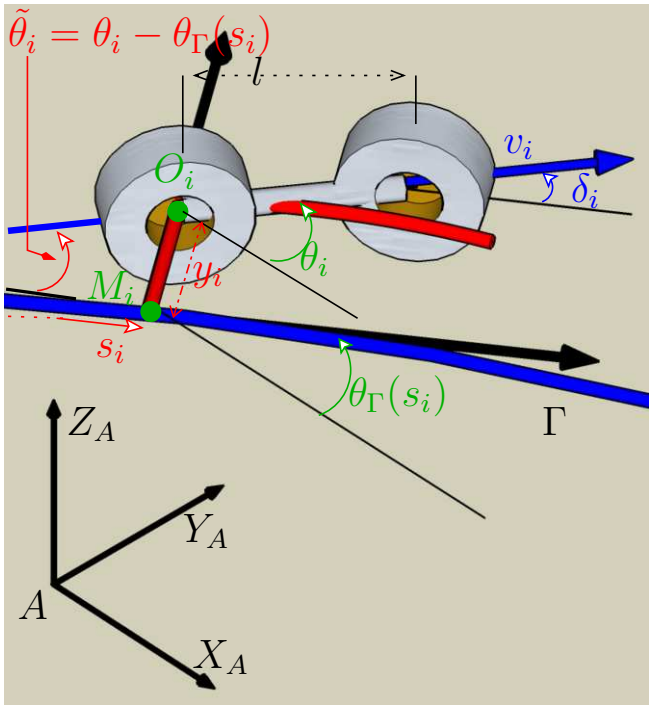


FIG. 1.18 – Modélisation d'un véhicule

Notations

- Γ est le chemin de référence, défini dans un repère absolu $[A, X_A, Y_A)$.
- O_i est le centre de l'essieu arrière, point de contrôle du i^{eme} véhicule.
- M_i est le point appartenant à Γ le plus proche de O_i. Ce point est supposé unique, ce qui se vérifie en pratique si la courbure de la trajectoire est réaliste et si le véhicule reste proche de cette dernière.
- s_i est l'abscisse curviligne du point M_i le long de Γ et $k(s_i)$ représente la courbure du chemin Γ en ce point.
- $\theta_\Gamma(s_i)$ correspond à l'orientation de la tangente à Γ en M_i, exprimée dans le repère $[A, X_A, Y_A)$.
- θ_i est l'orientation du i^{eme} véhicule dans le repère $[A, X_A, Y_A)$.
- $\tilde{\theta}_i = \theta_i - \theta_\Gamma(s_i)$ définit l'erreur angulaire du i^{eme} véhicule par rapport à Γ.
- y_i est l'erreur latérale du i^{eme} véhicule par rapport à Γ.
- δ_i est l'angle de braquage de la roue avant virtuelle du i^{eme} véhicule.

– l est l'empattement (la distance entre les centres des essieux arrière et avant).
– v_i est la vitesse linéaire du i^{eme} véhicule au point O_i.

Enfin, la Fig. 1.19 introduit la notation $d_{i,j} = s_i - s_j$ qui représente l'écart longitudinal entre les véhicules i et j au sein du convoi.

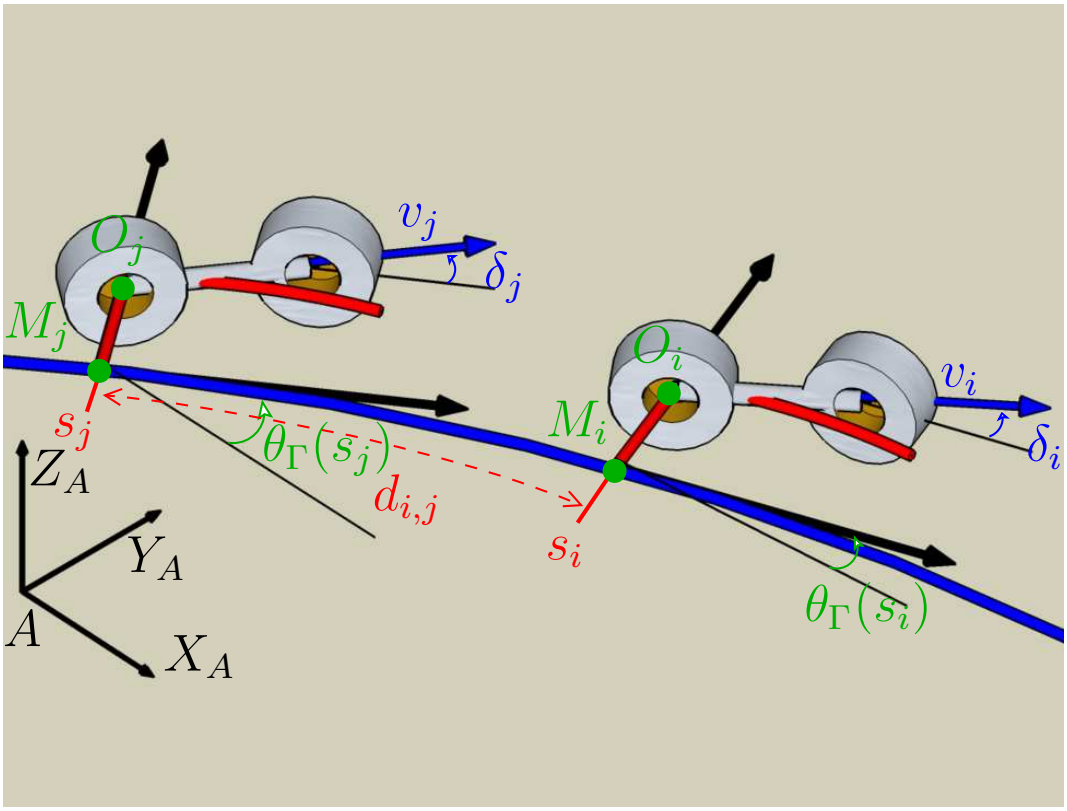


FIG. 1.19 – Ecart longitudinal

Mise en équation et modèle d'état L'état du véhicule peut être défini dans le repère de Frenet relatif à la trajectoire de référence Γ. Le vecteur d'état est alors défini par le triplet $\left[s_i, y_i, \tilde{\theta}_i\right]$ et l'équation d'état du modèle bicyclette peut s'écrire sous la forme suivante (voir [Bom 06]) :

$$\begin{cases} \dot{s}_i = \dfrac{v_i . \cos\tilde{\theta}_i}{1 - y_i . k(s_i)} \\ \dot{y}_i = v_i . \sin\tilde{\theta}_i \\ \dot{\tilde{\theta}}_i = \dfrac{v_i . \tan\delta_i}{l} - \dfrac{v_i . k(s_i) . \cos\tilde{\theta}_i}{1 - y_i . k(s_i)} \end{cases} \tag{1.50}$$

Le modèle (1.50) est clairement singulier si $y_i = \dfrac{1}{k(s_i)}$ c'est-à-dire quand le point O_i est superposé avec le centre de courbure de la trajectoire à l'abscisse s_i. Cependant, cette configuration n'est jamais rencontrée en pratique car, d'une part la courbure de la trajectoire de référence Γ est généralement assez faible et, d'autre part le véhicule est supposé proche de Γ.

1.3.1.2 Chaînage du système

Il a été établi dans [Samson 95] que les modèles cinématiques de robots non-holonomes peuvent être convertis par des changements de variables en systèmes *chaînés*, beaucoup plus simples à manipuler du point de vue de la commande. Dans cette section, l'équation (1.50) est convertie en un système chaîné. La loi de commande sera ensuite construite à partir de ce système en exploitant des outils classiques d'automatique linéaire. Nous verrons alors que le guidage le long d'une trajectoire peut être réalisé avec des performances indépendantes de la vitesse longitudinale, permettant ainsi de découpler les commandes latérale et longitudinale (décrites dans les sections 1.3.1.3 et 1.3.1.4).

Soient la transformation d'état donnée par (1.51) et la transformation de commande donnée par (1.52) :

$$\begin{aligned}\Phi(\begin{bmatrix} s_i & y_i & \tilde{\theta}_i \end{bmatrix}) &= \begin{bmatrix} a_{1i} & a_{2i} & a_{3i} \end{bmatrix} \\ &\triangleq \begin{bmatrix} s_i & y_i & (1 - y_i.k(s_i)).\tan\tilde{\theta}_i \end{bmatrix} \end{aligned} \tag{1.51}$$

$$(m_{1i}, m_{2i}) = \Xi(v_i, \delta_i) \tag{1.52}$$

avec :

$$m_{1i} \triangleq v_i \frac{\cos\tilde{\theta}_i}{1 - y_i.k(s_i)} \tag{1.53}$$

$$m_{2i} \triangleq \frac{d}{dt}((1 - y_i.k(s_i)).\tan\tilde{\theta}_i) \tag{1.54}$$

Les transformations Φ et Ξ sont inversibles sous les conditions $y \neq \dfrac{1}{k(s_i)}$ et $v_i \neq 0$ et sous la condition $\tilde{\theta}_i \neq \frac{\pi}{2}[\pi]$. Cette configuration correspond à une direction du véhicule perpendiculaire à la trajectoire, ce qui est très éloigné de la configuration souhaitée. Aussi, dans la suite, on considérera que l'écart angulaire reste dans l'intervalle $]-\frac{\pi}{2}; \frac{\pi}{2}[$.

En substituant (1.51), (1.53) et (1.54) dans (1.50), le système non linéaire (1.50) peut être réécrit sans approximation comme la forme chaînée suivante :

$$\begin{cases} \dot{a}_{1i} = m_{1i} \\ \dot{a}_{2i} = a_{3i}\, m_{1i} \\ \dot{a}_{3i} = m_{2i} \end{cases} \tag{1.55}$$

Le modèle (1.55) constitue un système chaîné, dépendant de deux variables de commande : m_{1i} qui est homogène à la vitesse d'avance $\dot{s}_i$ du véhicule le long de Γ et m_{2i}

lié à la vitesse de rotation $\omega_i = \frac{v_i.\tan\delta_i}{l}$.

Afin de construire une loi de commande avec des performances indépendantes de la vitesse d'avance v_i, il est judicieux de réécrire le système chaîné (1.55) en dérivant par rapport à l'abscisse curviligne plutôt que par rapport au temps. En notant : $a'_{ji} = \frac{da_{ji}}{da_{1i}} = \frac{da_{ji}}{ds_i}$, la forme chaînée (1.55) s'écrit alors :

$$\begin{cases} a'_{1i} = 1 \\ a'_{2i} = a_{3i} \\ a'_{3i} = \frac{m_{2i}}{m_{1i}} = m_{3i} \end{cases} \tag{1.56}$$

1.3.1.3 Commande latérale

En utilisant le système linéarisé exact (1.56), une loi de commande dédiée au suivi du chemin Γ peut être construite de la façon suivante. L'objectif du suivi de chemin est d'asservir à zéro les erreurs latérale et angulaire par rapport à Γ. D'après (1.51), cet objectif se traduit par la convergence des variables a_{2i} et a_{3i} vers zéro. Au vu de la structure du système (1.56) (équivalent à un double intégrateur), cela peut être obtenu par le choix d'un régulateur proportionnel dérivé pour la variable m_{3i} comme suit :

$$m_{3i} = -K_d a_{3i} - K_p a_{2i} \qquad (K_p,\ K_d) \in \mathcal{R}^{+*} \times \mathcal{R}^{+*} \tag{1.57}$$

où K_d et K_p sont deux réels strictement positifs fixant la dynamique de convergence du système. En injectant (1.57) dans (1.56) on constate en effet que ce régulateur permet d'aboutir à l'équation différentielle suivante :

$$a''_{2i} + K_d a'_{2i} + K_p a_{2i} = 0. \tag{1.58}$$

Les gains K_d et K_p sont donc homogènes à des gains proportionnel et dérivé d'un correcteur classique. De plus, comme la variable par rapport à laquelle on différencie l'équation (1.58) est $a_{1i} = s_i$, les gains K_d et K_p imposent une distance d'établissement et non un temps de réponse. Par conséquent, pour une erreur latérale initiale donnée, la trajectoire du véhicule est identique quelle que soit la valeur de v_i, et même si cette vitesse, tant qu'elle est non nulle, varie avec le temps [Thuilot 04]. Par inversion des transformations chaînées (1.51-1.53-1.54), l'expression non linéaire de la loi de commande s'écrit :

$$\begin{aligned} \delta_i(y_i, \tilde{\theta}_i) = \arctan\Big(l\Big[&\tfrac{\cos^3\tilde{\theta}_i}{(1-k(s_i).y_i)^2}\Big(\tfrac{dk(s_i)}{ds_i}.y_i.\tan\tilde{\theta}_i \\ &-K_d(1-k(s_i).y_i)\tan\tilde{\theta}_i \\ &-K_p y_i + k(s_i)(1-k(s_i).y_i)\tan^2\tilde{\theta}_i\Big) + \tfrac{k(s_i).\cos\tilde{\theta}_i}{1-k(s_i).y_i}\Big]\Big) \end{aligned} \tag{1.59}$$

Elle est parfaitement définie sous les trois conditions évoquées précédemment ($v_i \neq 0$, $y_i \neq \frac{1}{k(s_i)}$ et $\tilde{\theta}_i \neq \frac{\pi}{2}[\pi]$).

Réglage des gains Nous nous intéressons maintenant au réglage des gains de commande K_p et K_d. La relation (1.58) définit un système du second ordre. On peut alors poser : $K_p = \omega_n^2$ et $K_d = 2\zeta\omega_n$ où ω_n représente la pulsation naturelle du système et ζ le

facteur d'amortissement. Afin d'éviter toute oscillation de l'écart latéral autour de zéro et garantir la convergence la plus rapide possible, le système doit être amené en régime critique. Cette condition est réalisée si $\zeta = 1$. La solution de l'équation différentielle (1.58) est donnée dans ce cas par :

$$y_i(s_i) = y_i(s_{i0}) \left(1 + \omega_n(s_i - s_{i0}) + y_i'(s_{i0})(s_i - s_{i0})\right) e^{-\omega_n(s_i - s_{i0})}$$

où s_{i0} désigne l'abscisse curviligne initiale du véhicule i et $y_i(s_{i0})$ (respectivement $y_i'(s_{i0})$) l'écart latéral entre le chemin de référence et la position initiale du véhicule (respectivement sa dérivée en fonction de s_i). On considère que les objectifs de la régulation sont atteints dès que l'erreur latérale est inférieure à 5% de l'erreur initiale $y_i(s_{i0})$. La distance d'établissement d_m peut donc être définie par la relation :

$$0.05 = \left(1 + \omega_n d_m + y_i'(s_{i0}) d_m\right) e^{-\omega_n d_m} \tag{1.60}$$

On suppose que lors du démarrage, l'évolution de l'erreur latérale est nulle ($y_i'(s_{i0}) = 0$). On établit alors à l'aide d'un solveur numérique l'expression :

$$d_m \simeq \frac{4.75}{\omega_n} \tag{1.61}$$

Les valeurs des gains sont alors les suivantes :

$$K_p \simeq \frac{22.5}{d_m^2} \tag{1.62}$$

$$K_d \simeq \frac{9.5}{d_m} \tag{1.63}$$

Ces gains assurent que l'écart latéral aura diminué de 95% lorsque le véhicule aura parcouru une distance d_m le long de la trajectoire de référence.

1.3.1.4 Commande longitudinale

Deux stratégies de navigation ont été comparées, l'une est basée sur une approche locale du convoi et l'autre est référencée sur le leader. De cette étude, une stratégie globale de commande, combinant les avantages de chaque approche, a été élaborée. Dans cette section le choix de l'entrée de commande est tout d'abord exposé avant de synthétiser la loi.

Stratégies de navigation pour le convoi La stratégie locale, schématisée Fig. 1.20, repose sur une commande de proche-en-proche, c.a.d. la loi de commande de chaque véhicule dépend uniquement de son état et de celui de son prédécesseur. L'objectif est alors de garder constante la distance curviligne séparant chaque véhicule de son suiveur le long de Γ. On introduit l'erreur e_{i+1}^i définie entre les véhicules d'indice i et $i+1$ en fonction de la consigne de distance curviligne $d^\star$ par la formule (1.64) :

$$e_{i+1}^i = d_{i,i+1} - d^\star = s_i - s_{i+1} - d^\star \tag{1.64}$$

Le principal inconvénient d'une telle approche est que la stabilité d'un long convoi ne peut pas être garantie. Cela a d'ailleurs été montré précédemment dans la sous-section 1.2.2 à l'item "commandes référencées sur le véhicule précédent". Ce résultat a aussi été illustré par des simulations dans [Bom 05] : l'accumulation des erreurs crée des oscillations dans les commandes, entraînant un manque de confort et l'instabilité sur de longs pelotons.

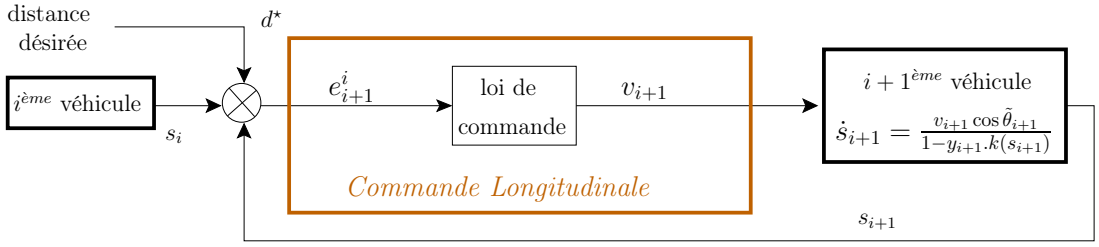


FIG. 1.20 – Contrôle longitudinal selon la stratégie locale

Pour surmonter ces problèmes, une autre approche s'appuyant sur une référence commune a été étudiée. Les distances curvilignes sont mesurées par rapport au véhicule leader (choisi arbitrairement comme référence). La variable régulée par la loi de commande est :

$$e^1_{i+1} = d_{1,i+1} - i.d^\star = s_1 - s_{i+1} - i.d^\star \tag{1.65}$$

Cependant, pour des raisons de sécurité évidentes, il est très recommandé de considérer l'état du véhicule positionné immédiatement devant le véhicule contrôlé, ce qui était réalisé par la commande locale de proche-en-proche. Au regard de ces remarques, une nouvelle variable régulée est synthétisée à partir des deux approches précédentes. Afin de donner plus ou moins d'importance à l'une ou l'autre de ces erreurs, la variable hybride, notée c_{i+1}, est construite comme suit :

$$c_{i+1} = \sigma_{i+1}(z_{i+1}).e^1_{i+1} + (1 - \sigma_{i+1}(z_{i+1})).e^i_{i+1}, \text{ avec} \tag{1.66}$$

$$z_{i+1} = e^i_{i+1} + \frac{d^\star - d_s}{2}, \tag{1.67}$$

où d_s, comme distance de sécurité, est la distance curviligne minimale qui doit toujours être observée entre deux véhicules. La fonction σ_{i+1}, à valeur dans l'intervalle $[0, 1]$, est explicitée par l'équation :

$$\sigma_{i+1}(z_{i+1}) = 0.5\left(\frac{1 - e^{-az_{i+1}}}{1 + e^{-az_{i+1}}} + 1\right) = \frac{1}{1 + e^{-az_{i+1}}} \quad \text{avec} \quad a > 0 \tag{1.68}$$

Le comportement de la fonction σ_{i+1} est illustré Fig. 1.21 avec les valeurs $a = 2.5$, $d^\star = 5m$ et $d_s = 2,\ 3$ et $4m$. Comme en témoigne l'allure de la courbe, la fonction σ_{i+1} permet de régler la prédominance de l'une ou l'autre des approches.

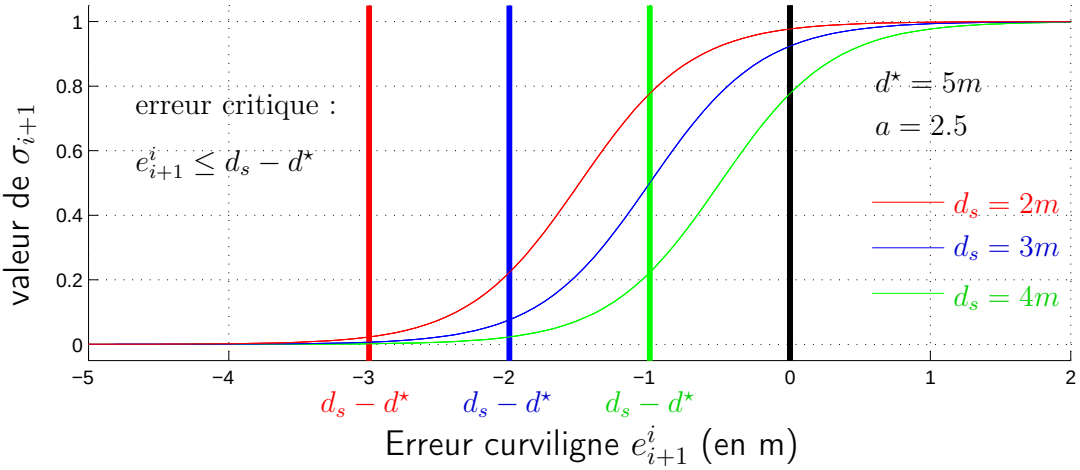


FIG. 1.21 – Fonction σ_{i+1}

- Quand le convoi est hors de danger, c.a.d. $e^i_{i+1} \gg d_s$, le confort des passagers est privilégié, soit la référence sur le leader ($\sigma_{i+1} = 1$ et donc $c_{i+1} = e^1_{i+1}$).
- En revanche, si des risques de collision apparaissent, c.a.d. e^i_{i+1} décroit $d_s - d^\star$, une commutation s'opère progressivement vers l'approche référencée sur le véhicule précédent ($\sigma_{i+1} = 0$ et donc $c_{i+1} = e^i_{i+1}$).

Le schéma de la commande longitudinale est donné par Fig. 1.22.

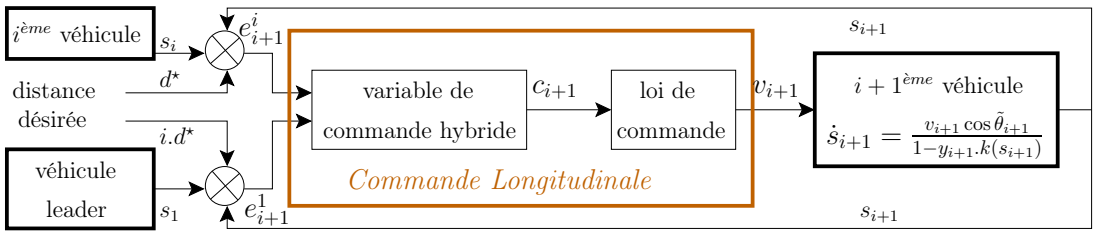


FIG. 1.22 – Commande longitudinale selon la stratégie globale

Synthèse de la loi de commande En différenciant la variable régulée c_{i+1} (équation (1.66)), on obtient :

$$\dot{c}_{i+1} = \sigma_{i+1}\dot{e}^1_{i+1} + (1-\sigma_{i+1})\dot{e}^i_{i+1} + \dot{\sigma}_{i+1}e^1_{i+1} - \dot{\sigma}_{i+1}e^i_{i+1} \tag{1.69}$$

Afin de simplifier les expressions, il est introduit :

$$A(z_{i+1}) = \frac{ae^{-az_{i+1}}}{(1+e^{-az_{i+1}})^2} \tag{1.70}$$

Ainsi, $\dot{\sigma}_{i+1}$ s'exprime simplement :

$$\dot{\sigma}_{i+1} = A(z_{i+1})\dot{e}^i_{i+1} \tag{1.71}$$

Comme dans la synthèse de la commande latérale, des techniques de linéarisation exacte sont également employées : en explicitant $\dot{e}^1_{i+1}$, $\dot{e}^i_{i+1}$ et $\dot{\sigma}_{i+1}$ dans (1.69), il est possible de linéariser exactement le comportement longitudinal en introduisant la commande auxiliaire $m_{4(i+1)}$ liée à v_{i+1} par :

$$\begin{aligned} v_{i+1} \quad = \quad & \frac{1-y_{i+1}.k(s_{i+1})}{\cos\tilde{\theta}_{i+1}[1+A(z_{i+1})(s_1-s_i-id^\star+d^\star)]} \Big(\sigma_i \frac{v_1\cos\tilde{\theta}_1}{1-y_1.k(s_1)} \\ & [1-\sigma_{i+1}+A(z_{i+1})(s_1-s_i-(i-1)d^\star)]\,\frac{v_i\cos\tilde{\theta}_i}{1-y_i.k(s_i)} - m_{4(i+1)}\Big) \end{aligned} \tag{1.72}$$

En effet, en reportant (1.72) dans (1.69), il est possible de vérifier que l'on obtient :

$$\dot{c}_{i+1} = m_{4(i+1)} \tag{1.73}$$

Ceci amène à proposer la commande auxiliaire :

$$m_{4(i+1)} = -K.c_{i+1}, \text{ avec } K \in \mathcal{R}^{+*} \tag{1.74}$$

La loi de commande longitudinale est obtenue en reportant (1.74) dans (1.72). Nous pouvons noter que des singularités peuvent se produire pour quelques situations exceptionnelles. En pratique, la probabilité est extrêmement faible. Il faudrait en effet que les premier, i^{eme} et également $(i+1)^{eme}$ véhicules soient dans une configuration spécifique. Si le cas se produisait, v_{i+1} augmenterait pour atteindre de très grandes valeurs et celles-ci seraient immédiatement corrigées par le module de supervision du véhicule (non détaillé ici).

Réglage du gain La solution de l'équation différentielle (1.73) avec la commande (1.74) est $c_{i+1}(t) = c_{i+1}(0).e^{-Kt}$. Donc le temps nécessaire pour que $c_{i+1}(t)$ soit égal à $0.05(c_{i+1}(0))$, $t_{5\%}$, peut être spécifié en réglant K selon la relation (1.75).

$$K = \frac{\ln 20}{t_{5\%}} \simeq \frac{3}{t_{5\%}} \tag{1.75}$$

1.3.2 Bilan :

Dans la section précédente, une approche adressant la mise en œuvre d'un convoi autonome de véhicules urbains électriques a été présentée. Pour des raisons discutées en section 1.2, une stratégie de commande globale décentralisée a été envisagée. Pour limiter l'architecture de communication, les échanges inter-véhicules requis sont des communications leader-véhicule et véhicule précédent-véhicule. Sur le plan des techniques de commande, les points forts de l'approche proposée sont les suivants.

Points positifs : La commande des véhicules est basée sur une modélisation cinématique plus simple que les modèles dynamiques utilisés couramment pour des stratégies dédiées à une évolution sur voie de circulation rapide. Cette modélisation est cependant tout à fait appropriée ici puisque le contexte urbain et la motorisation des véhicules permettent de négliger les composantes dynamiques dans le comportement des véhicules. Cela évite de devoir manipuler des modèles de grande dimension avec des paramètres mal connus (masse, raideur, ...).

• Puisque le modèle des véhicules est non-linéaire, des techniques de commande elles aussi non-linéaires ont été retenues pour construire un contrôleur offrant des performances élevées. Plus précisément, elles permettent de prendre en compte les non-linéarités du système explicitement dans la commande et donc de pouvoir garantir le même comportement que le véhicule soit proche ou loin de sa consigne.

• Les lois de commande sont basées sur une technique de linéarisation exacte : la transformation chaînée. Cette façon de procéder permet notamment le découplage exact entre les commandes latérale et longitudinale. Cette propriété est essentielle dans l'optique de la conduite en convoi car les performances de la commande latérale sont alors indépendantes de la vitesse des véhicules.

• Par conséquent, la vitesse peut être pilotée de façon tout à fait libre pour répondre aux attentes de la commande longitudinale. Plus précisément, cette dernière repose sur la régulation d'une variable hybride. De cette façon, lorsque le convoi est formé, si aucune difficulté n'est rencontrée pendant la navigation, chaque véhicule s'asservit par rapport au véhicule de tête, ce qui garantit la stabilité asymptotique du convoi. Si un problème intervient et que la distance entre deux véhicules devient critique, alors le véhicule suiveur va progressivement s'asservir sur son prédécesseur afin d'assurer la non collision.

• L'utilisation d'une trajectoire de référence commune à la flotte entière rend possible une évolution monotrace du convoi, quel que soit le nombre de véhicules, ce qui est souvent souhaité dans les applications pratiques. De plus cela permet de réguler des interdistances curvilignes qui restent cohérentes lors des courbes, à l'opposé de la distance euclidienne.

Points à développer : Dans ce manuscrit on a souhaité poursuivre sur l'approche qui vient d'être décrite pour tous les points forts mentionnés ci-dessus et la développer selon trois axes qui seront abordés dans les chapitres suivants :

• Dans un premier temps, on souhaite mettre au point un nouveau mode de déplacement apportant plus de liberté et une plus grande flexibilité. On souhaite que le déplacement de la flotte puisse être réalisé en fonction de la trajectoire empruntée en temps réel par le véhicule leader. Ce "mode manuel" de conduite serait en effet intéressant pour des opérations de maintenance ou pour guider un groupe de personnes sur un parcours qui n'a pas été anticipé. Un outil permettant de générer facilement une

trajectoire à partir de données de localisation doit alors être intégré à la stratégie de navigation. Cela fait l'objet du chapitre 2.

• Pour l'instant l'approche de navigation en convoi a été testée expérimentalement grâce à l'utilisation de récepteurs RTK-GPS pour la localisation des véhicules. Cependant, les signaux en provenance des satellites sont souvent masqués par les hautes structures urbaines (immeubles, bâtiments). En conséquence on souhaite changer la modalité capteur en faveur de l'utilisation de caméras. Leur faible prix les rend beaucoup plus abordables qu'un récepteur RTK-GPS et d'autre part les structures urbaines s'avèrent être très riches pour le traitement d'image.

Dans ce contexte les auteurs de [Royer 05] ont développé une stratégie de localisation dans un repère absolu à partir d'images obtenues en temps réel par vision monoculaire. Cette stratégie, décrite plus en détail dans l'annexe B, s'effectue en deux étapes. L'enregistrement d'une séquence vidéo par une caméra montée sur un véhicule alors que celui-ci se déplace en conduite manuelle permet tout d'abord une reconstruction 3D de l'environnement, on parle d'approche de type "Structure From Motion" ou plus simplement SFM. La deuxième étape est ensuite la localisation du véhicule en conduite autonome, grâce à l'appariement d'amers détectés dans l'image courante avec ceux qui constituent la reconstruction de l'environnement. Cependant, bien que l'application donne de bons résultats pour la commande latérale, voir [Royer 05], cette modalité capteur pose certains problèmes lorsque plusieurs véhicules sont à prendre en compte. Cette problématique est exposée à la fin de ce chapitre et des solutions seront apportées au chapitre 3.

• La mise en pratique d'un convoi de véhicules soulève des préoccupations d'ordre technique qu'il convient aussi d'adresser, en particulier la synchronisation des données au sein du convoi : lorsque des données issues de récepteurs RTK-GPS sont utilisées, la cohérence temporelle des données est assurée car chaque véhicule reçoit des informations de localisation datées dans un référentiel dicté par les satellites et donc commun à la flotte. Ce n'est plus le cas lors d'un asservissement par vision où les données de localisation sont datées par l'ordinateur connecté à la caméra. Ce problème sera discuté au chapitre 4.

Problématique vision Pour être capable de contrôler efficacement et en toute sécurité un groupe de robots mobiles en milieu urbain, il est nécessaire de positionner les véhicules au sein du convoi avec une précision de l'ordre du centimètre. Dans notre cas, nous souhaitons que la localisation des véhicules s'appuie sur les informations fournies au cours du temps par une unique caméra en mouvement (approche Structure From Motion). En conséquence les données de localisation s'expriment dans un monde virtuel, grossièrement recalé à la réalité métrique en estimant un facteur d'échelle global lors de la reconstruction de l'environnement. Malheureusement, même après cette transformation, le monde vision reste légèrement distordu par rapport à la réalité. Comme il est illustré sur la Fig. 1.23 l'accumulation de petites distorsions ne permet pas de joindre le point de départ au point d'arrivée d'une trajectoire bouclée parcourue dans

son intégralité. Cette jonction a été impossible malgré une forte corrélation en début et en fin de trajectoire entre les descripteurs visuels utilisés pour caractériser les points d'intérêt. Le monde 3D reconstruit se révèle incohérent dans ce cas précis dans le sens où certains points d'intérêt possèdent deux positions 3D dans le monde vision. Ainsi, si un véhicule traverse la zone où la trajectoire se boucle, il va donc virtuellement faire un (voire plusieurs) bond(s) dans l'espace de quelques mètres en une fraction de seconde...

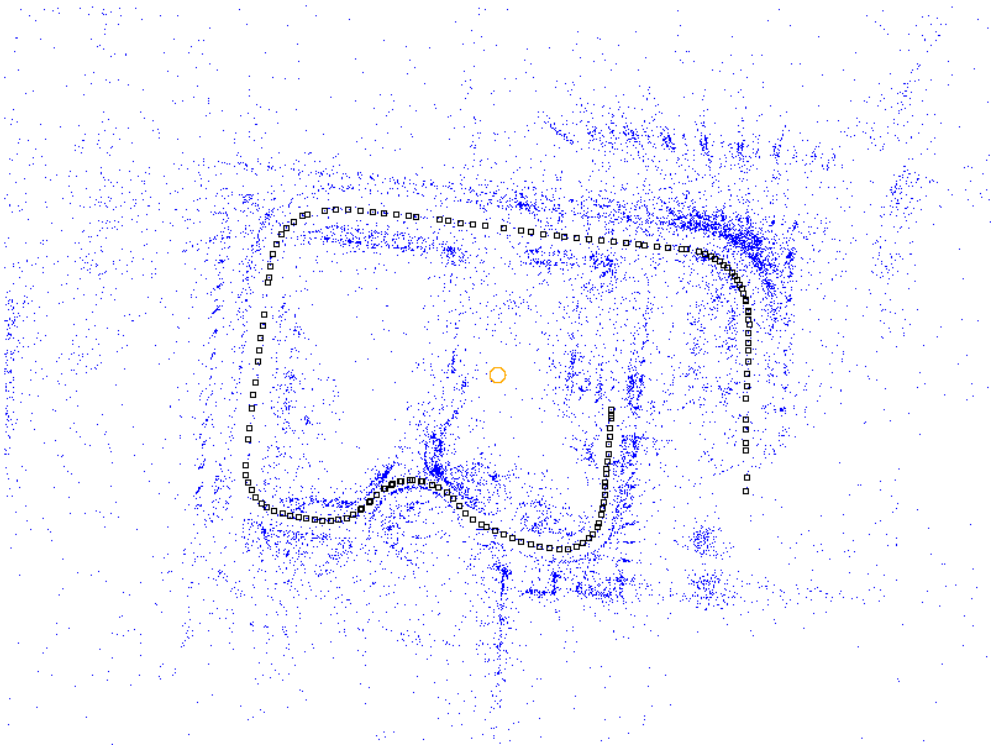


FIG. 1.23 – Reconstruction 3D d'une trajectoire bouclée

Pour approfondir la comparaison entre les deux espaces de représentations, le graphique sur la Fig. 1.24 montre une portion de la trajectoire précédente (illustrée Fig. 1.23) représentée conjointement dans le monde virtuel (partie a) et dans un monde métrique (partie b) grâce à l'utilisation d'un récepteur RTK-GPS d'une précision centimétrique. Sur la partie c, la trajectoire vision est transformée (par une translation et une rotation) de sorte que le point initial et sa tangente à la courbe correspondent pour les deux représentations. La non-superposition des courbes reflète la présence des distorsions. Cependant la direction de la tangente le long de la trajectoire semble être correctement estimée : les deux trajectoires se confondent sur la première ligne droite et apparaissent ensuite parallèles après le premier virage. Puisque la tangente, qui exprime la variation de la trajectoire selon les deux axes du plan horizontal, paraît cohérente, on peut faire l'hypothèse que les distorsions sont localement homogènes quelle que soit la direction pointée. Cette supposition, validée par la suite, paraît naturelle puisque la méthode de reconstruction est basée sur l'ajustement de faisceaux : la procédure d'optimisation

concerne des orientations (celles des rayons optiques entre la caméra courante et les points d'intérêt) et l'on s'attend ainsi à avoir de bonnes mesures sur les angles.

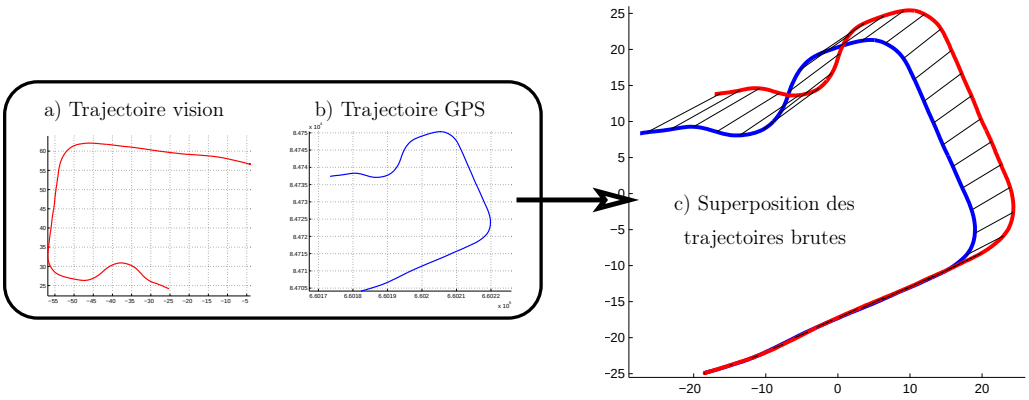


FIG. 1.24 – Superposition des deux représentations : réelle et virtuelle

Par contre, l'estimation des longueurs issue de la représentation virtuelle est sensiblement modifiée et ne permet pas d'obtenir la précision centimétrique requise. D'une part, le facteur d'échelle global ne peut pas fournir une telle précision alors que la distance couverte peut atteindre plusieurs centaines de mètres : ce dernier est en effet calculé par intégration du déplacement du véhicule. Le mouvement est estimé grâce aux données proprioceptives des odomètres et une erreur constante, même très légère dans la lecture, va inévitablement entraîner une derive dans l'estimation de la distance parcourue. D'autre part, ce qui est plus gênant encore, est que les distorsions entre monde réel et vision varient le long de la trajectoire.

Ces deux limitations sont illustrées sur la Fig. 1.25 qui compare les longueurs d'arc calculées en se basant sur la vision monoculaire et sur les données du récepteur RTK-GPS. On remarque tout d'abord qu'en fin de courbe l'évaluation du facteur d'échelle global est erronée puisque l'écart entre les deux représentations sur la longueur totale de la trajectoire atteint $1.72m$, bien que cette dernière ne dépasse pas $115m$ de long. Ensuite, on vérifie bien que les distorsions entre les deux mondes varient largement : on observe un maximum de $7.48m$ approximativement à mi-chemin sur la trajectoire.

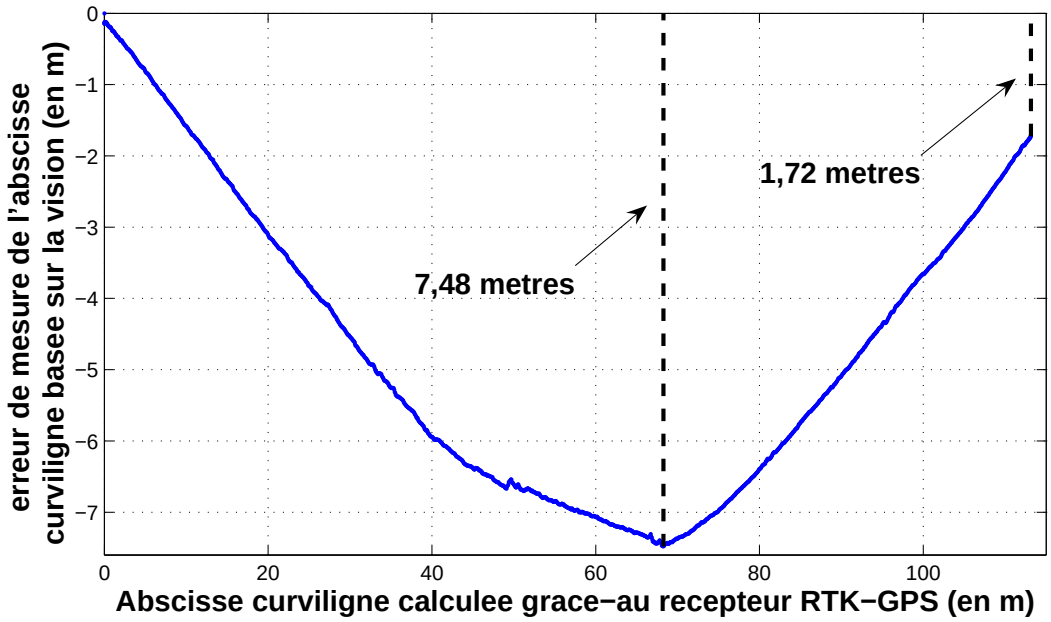


FIG. 1.25 – Erreur d'estimation de l'abscisse curviligne vision par rapport à la réelle

Chapitre 2

Génération de la trajectoire de référence

2.1 Introduction

On considère dans ce chapitre la génération d'une trajectoire de référence représentative du mouvement enregistré d'un véhicule, d'abord hors ligne, puis en ligne de façon à rendre possible un mode "manuel" de guidage d'un convoi de véhicules. Suivant ce mode, l'évolution de chaque véhicule doit suivre avec précision la trajectoire d'un véhicule leader conduit manuellement tout en respectant une consigne d'interdistance entre chaque paire de véhicules. La flexibilité et la liberté inhérente à un tel mode d'utilisation du convoi est attrayante pour des secteurs très variés dans le domaine public comme le domaine privé. Sans aucune anticipation, un parcours touristique pourrait ainsi être improvisé à travers une ville ou un quelconque site attractif. On peut aussi imaginer un groupe de personnes utiliser ponctuellement un ensemble de véhicules en libre service, guidé par l'un des utilisateurs, les autres n'ayant pas à se soucier du trajet. Cette souplesse serait aussi bénéfique pour le déplacement des personnes et/ou marchandises dans de grands espaces tels les hôpitaux, les parcs d'attraction, les grands sites industriels et autres gares et aéroports où les itinéraires de transport sont multiples.

Générer une trajectoire est une pratique courante dans le domaine de la robotique et beaucoup d'efforts ont été consacrés pour traiter ce problème. Dans la suite, une étude bibliographique présente tout d'abord les différentes approches de génération de trajectoire dans le milieu de la robotique mobile. Une stratégie de génération de trajectoire est ensuite détaillée dans laquelle deux modes de convoyage de véhicules sont adressés : le mode "tout automatique", pour lequel la trajectoire est définie à partir d'informations pré-enregistrées, et le mode "manuel" que l'on vient d'introduire.

2.2 Étude bibliographique

Dans cette étude, notre attention se focalise uniquement sur les techniques où la représentation de la trajectoire est analytique. Ce choix nous est imposé par le carac-

tère “global” de la stratégie de contrôle utilisée pour maintenir la formation en convoi et par les techniques de commande envisagées. En effet, l'approche détaillée en 1.3.1 nous impose de positionner avec précision tous les véhicules par rapport à la trajectoire de référence, ce qui n'est possible que si celle-ci est définie comme une trajectoire continue. Elle doit être, de surcroît, dérivable au moins deux fois pour permettre d'évaluer les tangentes et courbures requises dans les lois de commande. Pour nos besoins, la trajectoire de référence doit donc bien être définie de manière analytique.

De nombreuses contributions respectant cette spécification ont été développées dans la communauté “Path Planning”, permettant à des robots de passer successivement d'une configuration désirée à une autre. Les applications en robotique mobile concernent la réalisation de trajectoires admissibles sans collision, la replanification et la supervision ou routage en ligne de véhicules. Ces notions sont définies dans les paragraphes qui suivent et illustrées à travers quelques exemples. On va aborder en premier lieu le cas où une trajectoire n'est pas conçue dynamiquement mais spécifiée hors-ligne pour une utilisation ultérieure. Les travaux de génération en temps réel sont présentés dans une seconde partie.

2.2.1 Génération de trajectoire hors-ligne

On considère un robot mobile évoluant dans son espace de travail. Le problème général de planification est de déterminer le mouvement le plus pertinent selon un critère d'optimalité pour joindre une configuration initiale A donnée à une configuration finale B. Il s'agit au premier abord d'un problème combinatoire classique dans la discipline Informatique : il faut tout d'abord définir l'ensemble des séquences de configurations connexes (sans collision) qui mènent de A à B, puis choisir la solution optimale parmi les combinaisons dont le nombre croît de paire avec la complexité du système. De nombreuses contraintes sont à considérer comme les caractéristiques du robot (géométrie, cinématique, dynamique), l'environnement et le type de tâche à effectuer. Les “*sampling-based planners*” représentent les techniques les plus répandues pour de tels problèmes. Un panorama des techniques de planification est présenté dans [Gonzales-Banos 06], [Lindemann 05] et [Schwartz 88]. Dans le cadre de cette étude, on ne portera aucune attention à cette étape de création d'une *roadmap*. On s'attache à une partie restreinte du problème de planification en supposant qu'un jeu de connexions libres est déjà disponible. Sur cette base il reste donc à trouver les fonctions analytiques permettant de relier ces différentes configurations tout en respectant des contraintes données : on parle de réalisation de *local steering strategies*.

2.2.1.1 Objectif du plus court chemin

Le problème de la planification a été initialement abordé dans le cadre d'un système de robots se déplaçant dans un environnement exempt d'obstacles soumis aux seules contraintes cinématiques. Pour un robot mobile de type voiture deux contraintes sont à prendre en compte et résultent des phénomènes suivants :

- Le contact entre les roues et le sol est supposé ponctuel et sans glissement. Le

vecteur vitesse au centre des roues appartient alors au plan des roues, ce qui amène des contraintes sur le vecteur vitesse du robot. Par exemple, dans le cas d'une voiture, le vecteur vitesse au centre de l'essieu arrière est perpendiculaire à cet essieu. Sa direction est donc liée à l'orientation du véhicule. On nomme ainsi *contrainte d'orientation* cette première contrainte cinématique.

• La deuxième contrainte, nommée *contrainte de courbure*, considère la limite mécanique de l'angle entre les roues directrices et l'axe principal du véhicule. Le rayon de braquage est ainsi minoré par une constante et la courbure des chemins que le véhicule peut suivre est bornée.

Les critères qui ont été envisagés pour traiter la planification de trajectoire visent à optimiser une fonction de coût exprimée en terme de distance parcourue par le robot entre les deux configurations extrémités, de durée ou d'énergie nécessaires à l'exécution de son mouvement.

Sans donner de formule explicite, la nature des chemins optimaux pour un critère de longueur de chemin parcouru a été démontrée très tôt en 1957 par Dubins [Dubins 57]. L'objectif était de relier deux configurations (initiale et finale) identifiées par deux positions définies dans le plan ainsi que leur orientation, et cela pour des véhicules n'effectuant pas de manœuvres (i.e. seul le déplacement en marche avant est autorisé). Il a été montré que les chemins optimaux sont constitués d'au plus trois pièces dont les extrêmes sont des arcs de cercle de rayon minimal (par rapport à la contrainte de courbure) et la liaison est soit un segment soit un autre arc de rayon minimum, de sorte que le chemin final soit une courbe lisse ($\mathcal{C}^1$). Cette restriction est telle qu'au plus six ensembles de pièces élémentaires sont envisageables quelles que soient les conditions initiales et finales.

Ce résultat a été généralisé par Reeds et Shepp [Reeds 90] dans le cas où le déplacement peut être effectué dans les deux sens (marche avant et arrière). Le plus court chemin est constitué d'au plus cinq pièces (arcs de cercle de rayon minimum ou segment de droite) moyennant au plus deux renversements ou transitions marche avant - arrière. Cette formulation offre alors 48 types de chemins possibles. La trajectoire résultante, dont les pièces sont des fonctions de classe $\mathcal{C}^\infty$, assure des raccords $\mathcal{C}^1$ sauf aux points charnière à l'intersection de deux arcs.

Le grand avantage des chemins planifiés sur la base de ces travaux repose sur la simplicité des primitives considérées. Segments de droite et arcs de cercle sont en effet faciles à implémenter et limitent grandement le temps de calcul pour extraire les informations utiles au positionnement d'un véhicule sur la trajectoire de référence. De plus, les chemins proposés sont parfaitement admissibles pour des véhicules de type voiture puisque les contraintes cinématiques associées ont été prises explicitement en compte : il existe une bijection entre l'ensemble des chemins qui respectent ces contraintes et celui défini par les courbes $\mathcal{C}^1$ par morceaux de $\mathcal{R}^2$.

Cependant, ces chemins ne peuvent être suivis précisément par un véhicule expérimental qu'à condition que ce véhicule stoppe à chaque discontinuité de la courbure (transition entre arcs de cercle et segments de droite) pour réorienter ses roues direc-

trices. La fluidité du mouvement est rompue et, pour atteindre rapidement son objectif, ces arrêts qui constituent une perte de temps ne seront pas respectés. Comme les exigences de continuité de la trajectoire ne sont pas satisfaites, des effets indésirables vont survenir comme des glissements ou des déviations latérales aux points où l'ordre de continuité requis n'est pas atteint. Pour l'asservissement d'un véhicule il serait donc nettement préférable de proposer un chemin offrant une courbure continue, voire même dont la dérivée de la courbure est bornée afin de limiter la vitesse de braquage de la voiture.

2.2.1.2 Une trajectoire conçue pour la commande

Le guidage de véhicules le long de trajectoires nécessite la génération de courbes très lisses. La solution du plus court chemin n'étant plus privilégiée, de nouveaux critères de qualité représentant la régularité des chemins sont introduits. Par exemple afin de minimiser l'accélération latérale des véhicules, il convient de minimiser l'intégrale du carré de la courbure le long du chemin ; l'utilisation de courbes clothoïdales représente alors la solution optimale. De même, la variation de l'accélération latérale peut être minimisée par l'utilisation de spirales cubiques.

Ces courbes se caractérisent par une dynamique spécifique imposée à la courbure et leurs coordonnées se calculent malheureusement difficilement, puisque leur forme analytique exacte implique des calculs d'intégrale fastidieux. Dans la littérature ce type de courbes est généralement représenté par des polynômes, ces derniers pouvant être de deux types. Le premier type de polynôme représente la courbure de la courbe en fonction de son abscisse curviligne. Bien que cette représentation souffre du problème sus-cité, elle a pour avantage de traiter de manière directe les contraintes cinématiques puisque les tangentes et courbures, qui se calculent facilement le long de la courbe, peuvent se paramétrer. Le deuxième type de polynôme, plus simple à utiliser, repose lui sur une approximation et retranscrit directement la courbe en coordonnées cartésiennes.

Des trajectoires, initialement planifiées à partir de courbes simples discontinues (segments de droite et arcs de cercle), sont par exemple approchées à l'aide de figures géométriques plus complexes comme les clothoïdes dans [Kanayama 85] et des spirales cubiques dans [Kanayama 97]. Nelson [Nelson 89] lui s'appuie sur deux types de polynômes, polaire et cartésien, pour approximer des arcs de cercle et obtenir des formes du type arc-arc ou arc-ligne-arc en assurant lui aussi la continuité de la courbure.

Des trajectoires aux formes plus variées peuvent aussi être modélisées, l'unique contrainte sur leur représentation étant de s'appuyer sur des courbes lisses et de respecter la contrainte de non-holonomie (c.a.d. que la trajectoire est réalisable au sens des contraintes cinématiques). La génération de telles trajectoires est facilement accessible lorsque le modèle du robot mobile présente la propriété d'être plat.

Définition Un système dont la dynamique est représentée par un modèle différentiel est dit plat s'il existe un ensemble de sorties dites plates tel que l'état et l'entrée du

système s'expriment analytiquement en fonction des sorties plates et de leurs dérivées.

La stratégie consiste alors à choisir et imposer l'évolution des sorties plates, qui dans le cas de robots mobiles caractérisent généralement la position du système dans l'espace cartésien. De cette trajectoire, grâce à la propriété de platitude, découle l'évolution de toutes les variables du système ainsi que la commande en boucle ouverte permettant de réaliser cette évolution. Afin d'assurer la continuité au niveau des jonctions de courbes, des contraintes peuvent être facilement intégrées dans la paramétrisation de l'évolution des sorties plates en imposant des conditions sur les dérivées aux limites. D'un point de vue pratique, l'utilisation de polynômes est très répandue pour modéliser la trajectoire. Ils permettent de modeler une trajectoire selon un éventail de formes très variées et de façon très simple puisque le calcul des dérivées est donné directement en fonction des coefficients du polynôme.

En présence d'obstacles, la recherche d'une trajectoire admissible entre deux configurations est plus ardue puisque l'espace libre entre les points de passage est limité. Une courbe admissible, qui établit la liaison entre deux points, doit alors intégrer cette contrainte spatiale supplémentaire afin d'éviter les collisions. Dans [Lamiraux 00] et [Tilbury 93] la construction d'une telle trajectoire s'appuie sur la propriété de *small-time controllability* des systèmes plats. Cette dernière traduit que la trajectoire calculée entre deux configurations reste proche de la configuration initiale quand ces deux dernières se rapprochent. Cette propriété assure donc que s'il existe un chemin libre d'obstacle entre deux configurations, alors il sera toujours possible de construire une trajectoire satisfaisant les contraintes non holonomes qui reste proche de ce chemin (il suffit de fixer des points de passage proches les uns des autres sur le chemin libre), et donc qui évite les obstacles. Une méthode de guidage en boucle ouverte, exploitant cette propriété a été proposée dans [Lamiraux 00]. Elle s'appuie sur des courbes canoniques reliant les points de passage et réclame une marche arrière au niveau de points "charnière".

2.2.1.3 Techniques empruntées à la communauté CAD (Computer Aided Design)

Les courbes et surfaces polynômiales développées en CAD présentent des aspects attrayants et sont communément utilisées pour fournir des trajectoires lisses. La modélisation, basée sur des "nœuds" ou points de contrôle, est très simple. Une stratégie de combinaison de polynômes et l'ajustement des points de contrôle permettent d'imposer certaines propriétés au niveau des raccords de courbes (continuité, allure de la courbe) en fonction des conditions propres à l'application. Par la suite la notation suivante sera commune aux différents exemples : $(p_i)_{i=1...n}$ et $(P_j)_{j=1...m}$ sont respectivement les n points de passage et les m points de contrôle de la courbe paramétrée dont $Q^i(t)$ est la $i^{ème}$ portion.

Des B-Splines de degré trois sont utilisées dans le cadre de [Vazquez 94] et [Komoriya 89] pour la conception d'une trajectoire de continuité $\mathcal{C}^2$. Elles consistent en la connexion de courbes polynômiales qui sont la combinaison linéaire d'une base de polynômes. Cette base est définie de sorte que des B-Splines de degré d assurent une

continuité d'ordre $d-1$. Une description plus détaillée sur les courbes B-Splines et la base de polynômes qui les compose est donnée en section 2.3.1.

Dans [Vazquez 94] les contraintes d'interpolation concernent uniquement les points de passage. Il s'agit d'ajuster les nœuds P_j, $j = 1 \dots m$ pour que chaque portion de spline Q^i vérifie : $Q^i(0) = p_i$ et $Q^i(1) = p_{i+1}$. Un algorithme permet de répondre à cette contrainte (autrement nommée "Inverse Transformation Problem") grâce à un processus d'optimisation itératif des points de contrôle.

Afin d'augmenter la maîtrise sur la forme de la trajectoire, des conditions supplémentaires sont ajoutées dans [Komoriya 89] pour contraindre les tangentes t_i au niveau de chaque point p_i. Le principe est basé sur le fait que quatre points de contrôle alignés génèrent un segment de droite. Pour interpoler les points de passage p_i on génère ainsi des segments autour de ces derniers en alignant quatre points de contrôle sur la droite passant par les p_i et orientée selon t_i. Comme le montre la Fig. 2.1, la configuration minimale (points P_1 à P_{10}) impose que certains nœuds soient figés aux intersections de ces droites (P_4 et P_7). L'ajout de deux nouveaux points de contrôle en remplacement

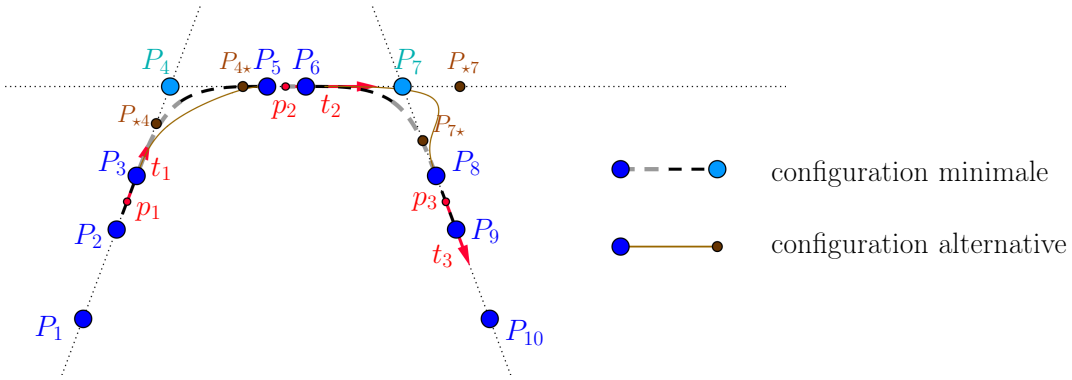


FIG. 2.1 – Exemple d'utilisation des splines définies dans [Komoriya 89]

de chacun de ces derniers permet d'obtenir une forme plus naturelle. Par exemple $P_{\star 4}$ et $P_{4\star}$ remplacent P_4 sur la Fig. 2.1 et modifient localement l'allure de la trajectoire. Cependant, même si les nouveaux points semblent bien ajustés (comme $P_{\star 4}$ et $P_{4\star}$), on ne garantit pas pour autant un meilleur comportement de la courbure (au sens de ses variations et de son amplitude maximales) sur les portions considérées.

Sur les deux travaux présentés, une reparamétrisation des B-Splines obtenues est finalement effectuée afin de définir une référence temporelle le long de la trajectoire. Il est important de signaler que la modulation de la forme de la trajectoire n'est pas considérée dans le premier cas et ne peut pas être faite de façon très satisfaisante dans le second puisqu'il faut ajouter des points de contrôle et donc occasionner inutilement des variations sur la courbure et sur la tangente à la courbe (direction mais surtout norme) même si celle-ci est reparamétrisée.

Les travaux qui suivent dans cette section appliquent une stratégie plus adroite pour faire varier la forme des courbes : des variables de paramétrisation, volontairement

laissées libres, vont permettre une plus grande flexibilité et dans la deuxième étude l'optimisation d'un critère sur la courbure.

Dans [Eren 99] les auteurs utilisent des splines d'interpolation, décrites par les équations (2.1) et (2.2), pour relier les points de passage en assurant une continuité $\mathcal{C}^1$ et en imposant la direction de la tangente à la trajectoire au niveau des points de passage.

$$Q^i(t) = \left[Q^i_x(t)\ Q^i_y(t)\right] = \left[\begin{array}{cccc} t^3 & t^2 & t & 1 \end{array}\right] \cdot M_c \cdot \left[\begin{array}{c} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{array}\right] \tag{2.1}$$

$$\text{où } M_c = \left[\begin{array}{cccc} -s & 2-s & s-2 & s \\ 2s & s-3 & 3-2s & -s \\ -s & 0 & s & 0 \\ 0 & 1 & 0 & 0 \end{array}\right] \tag{2.2}$$

Comme le montre la Fig. 2.2 (partie gauche), cette représentation permet de relier

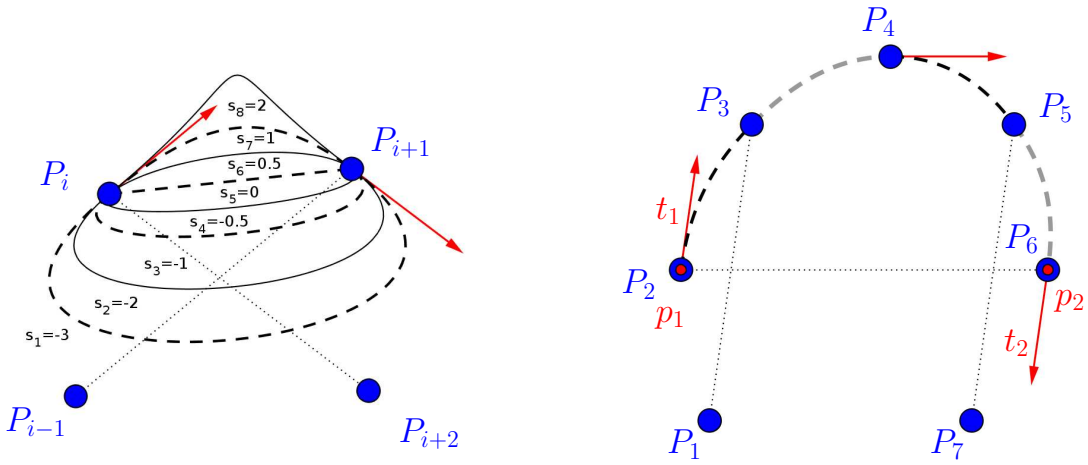


FIG. 2.2 – Exemple d'utilisation des splines définies dans [Eren 99]

le point P_i au point P_{i+1} en respectant : $\dot{Q}(0) = s\overrightarrow{P_{i-1}P_{i+1}}$ et $\dot{Q}(1) = s\overrightarrow{P_iP_{i+2}}$. Le paramètre libre s, ou paramètre de tension de la spline, permet de modeler la forme de la spline. L'effet de ce paramètre est illustré sur la Fig. 2.2 (partie gauche) avec les valeurs $-3 \leq s \leq 2$ en considérant les mêmes conditions aux extrémités pour chaque tracé. Bien que chaque courbe $Q^i(t)$ soit formée de quatre points de contrôle, sept points sont nécessaires (c.a.d. quatre splines) pour joindre deux points quelconques p_1 et p_2 du plan en considérant les contraintes aux limites :

$$\begin{cases} P_2 = p_1,\ P_6 = p_2 \\ (P_1, P_3) \text{ et } (P_5, P_7) \text{ spécifient la tangente } t_1 \text{ (en } p_1) \text{ et } t_2 \text{ (en } p_2) \text{ respectivement} \\ P_4 \text{ “contrôle” la courbure} \\ P_1 \text{ et } P_7 \text{ ne font pas partie de la trajectoire} \end{cases}$$

La jonction de deux configurations est illustrée Fig. 2.2 (partie droite). Le nœud P_4 n'est pas indispensable, il sert à modeler la courbure de la courbe afin d'obtenir une forme plus naturelle. P_4 apporte aussi une contrainte tangentielle puisque la tangente en P_4 est orientée selon la droite (P_2, P_6).

Une prise en compte très efficace de la courbure et de ses variations figure dans les travaux de Piazzi [Piazzi 02]. Les auteurs montrent que le degré d'une courbe polynômiale permettant de joindre deux configurations définies par un point, l'angle de la tangente ainsi que la courbure à la courbe en ce point, doit être au minimum cinq. Les auteurs ont alors élaboré un nouveau type de splines, dénommées "Quintic G^2-Splines", spécifié par l'équation (2.3), en considérant la configuration initiale (A, θ_A, K_A) et finale (B, θ_B, K_B).

$$Q(t) = \begin{bmatrix} Q_x(t) \\ Q_y(t) \end{bmatrix} = \begin{bmatrix} x_0 + x_1 t + \cdots + x_5 t^5 \\ y_0 + y_1 t + \cdots + y_5 t^5 \end{bmatrix} \tag{2.3}$$

Les termes x_0, $x_1 \ldots x_5$ ainsi que $y_0 \ldots y_5$ sont tous fonction des conditions limites $A, B, \theta_A, \theta_B, K_A$ et K_B et de quatre paramètres libres η_1 à η_4. Ces derniers sont judicieusement introduits de sorte que η_1 et η_2 représentent des paramètres de vitesse et η_3 et η_4 des paramètres de torsion de la courbe au niveau de ses extrémités :

$$\eta_1 = \|\dot{Q}(0)\| \tag{2.4a}$$

$$\eta_2 = \|\dot{Q}(1)\| \tag{2.4b}$$

$$\eta_3 = \frac{\ddot{Q}_x(0)\dot{Q}_x(0) + \ddot{Q}_y(0)\dot{Q}_y(0)}{\eta_1} \tag{2.4c}$$

$$\eta_4 = \frac{\ddot{Q}_x(1)\dot{Q}_x(1) + \ddot{Q}_y(1)\dot{Q}_y(1)}{\eta_2} \tag{2.4d}$$

Comme le montre la Fig.2.3, cette représentation permet alors de bonnes approxima-

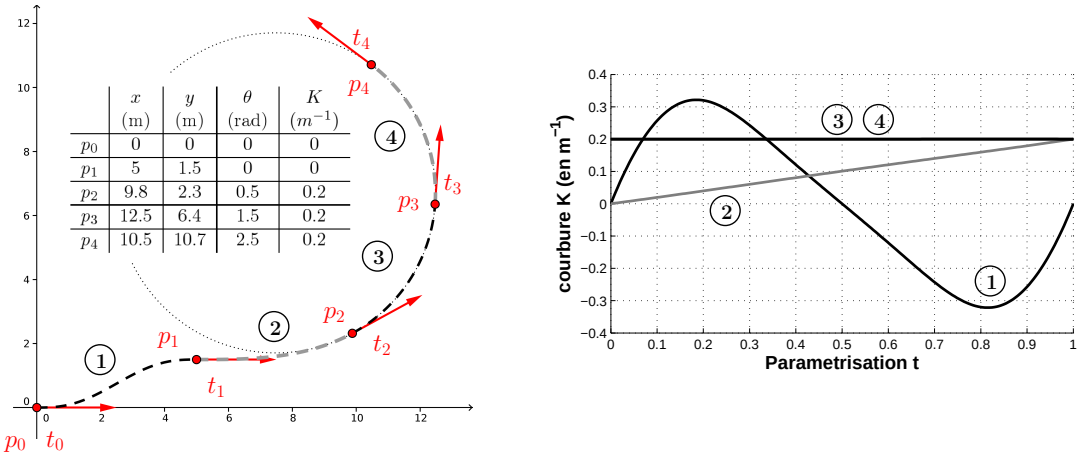


FIG. 2.3 – Exemple d'utilisation des splines définies dans [Piazzi 02]

tions de portions de droite, cercle et clothoïde en posant simplement $\eta_1 = \eta_2$, $\eta_3 = -\eta_4$.

Il faut aussi bien entendu que les configurations initiale et finale soient cohérentes avec la courbe que l'on désire obtenir (par exemple $K_A = K_B$ est nécessaire pour approximer un arc de cercle). De plus, comme les auteurs ont pour objectif de générer une trajectoire qui présente le moins de variations de courbure possible, des algorithmes de programmation quadratique ont été mis au point afin de calculer la valeur optimale des quatre paramètres selon le critère dicté par l'équation (2.5) :

$$\min_{\eta} \max_{t\in[0,1]} \left| \frac{\mathrm{d}K}{\mathrm{d}s}(t) \right| \tag{2.5}$$

Dans cette formule la paramétrisation des courbes t est fonction de l'abscisse curviligne s, qui est une variable de dérivation plus pertinente (que t) pour estimer la variation de la courbure. Moyennant quelques batteries de simulations, des tables de correspondances peuvent être construites pour utiliser cette stratégie de planification en temps réel.

On retiendra dans cette section que la propriété de platitude, qui est présentée par la plupart des modèles de robots mobiles, nous permet de modéliser une trajectoire admissible à partir de courbes lisses. Lorsque l'environnement est exempt d'obstacle, l'utilisation de polynômes est très répandue pour générer des trajectoires pour les robots mobiles. Tout particulièrement, les splines présentent des propriétés très intéressantes : la représentation par des nœuds est très simple, la continuité des courbes est assurée au niveau des jonctions où la tangente et/ou la courbure peuvent, selon le type de spline utilisé, être imposées.

2.2.2 Génération et modification en ligne de trajectoire

De très nombreuses solutions ont été avancées pour la conception de trajectoire dans le cadre statique. Un nouveau défi réside dans la prise en compte de l'environnement et des objets dynamiques qui le constituent. Le développement de stratégies de replanification de la trajectoire est alors envisagé pour répondre en ligne à la nécessité de fournir à chaque instant une trajectoire admissible libre de collision.

Dès 1997, Khatib [Khatib 97] s'intéresse à l'évolution de l'environnement et propose une stratégie qui considère localement l'espace accessible par un véhicule sans entrer en collision. Cet espace, ou bulle comme le nomment les auteurs, est représenté par une boule $B(q, r)$ centrée sur la configuration courante du robot q. La distance r, naturellement définie dans l'espace des configurations, est calculée pour être la plus grande possible, c.a.d. entre le robot et l'obstacle le plus proche. L'idée directrice est la suivante : si les bulles autour des points de passage $\{p_i\}_{i=1\ldots n}$ se recouvrent, il est alors possible de construire une trajectoire admissible sans collision de p_1 à p_n.

$$B(p_i, r_i) \cap B(p_{i+1}, r_{i+1}) \neq \emptyset,\ \forall i = 1 \ldots n-1.$$

Cette propriété est exploitée pour générer le chemin initial à partir des points de passage et doit être maintenue en chaque instant. Pour cela, le mouvement du groupe de bulles initial est imposé par la résultante de trois forces. Deux forces internes symbolisent les

interactions entre les bulles et optimisent la forme du groupe de bulles. Elles empêchent les bulles d'avoir une trop grande surface de superposition et évitent les déconnexions. Une force externe de répulsion permet enfin le maintien à distance avec les obstacles. Une fois que le groupe de bulles initial est défini, la trajectoire est préconstruite à partir de courbes "Reeds and Shepp" reliant les centres des bulles. L'utilisation de courbes de Bézier permet ensuite de lisser la trajectoire tout en la confinant dans l'espace libre : une courbe de Bézier se situant dans l'enveloppe convexe de ses points de contrôle, il suffit de positionner ceux-ci à l'intérieur des bulles. Pour chaque paire de points à relier, la portion de trajectoire correspondante est formée de trois courbes de Bézier (six points de contrôle). Les trois premiers et les trois derniers points de contrôle de chaque portion sont alignés, assurant aussi une courbure nulle aux jonctions. Le choix des longueurs $l = |\overline{p_0p_1}| = |\overline{p_4p_5}|$ et $k = |\overline{p_1p_2}| = |\overline{p_3p_4}|$ détermine la forme de la courbe de Bézier et permet de contraindre la courbure. Un reproche concernant cette stratégie peut néanmoins être formulé par rapport à la navigation du véhicule le long d'une telle trajectoire : quand les bulles se déplacent, la trajectoire peut être modifiée à l'endroit où se trouve le véhicule et l'asservissement de ce dernier risque alors d'être perturbé par ces variations.

Une autre approche pour modifier en ligne la trajectoire est proposée dans [Hwang 03]. Contrairement à [Khatib 97], elle est corrélée à la stratégie de commande des véhicules dans le sens où la position courante du véhicule est prise en compte pour ne pas modifier la trajectoire à son niveau. La commande est calculée en boucle ouverte à partir d'un modèle de robot unicycle de sorte que le véhicule suive le vecteur tangent à la trajectoire tout en assurant d'une part que la vitesse maximale des roues ne soit pas dépassée et d'autre part que la force centrifuge soit maintenue sous un seuil qui permet d'éviter les glissements. L'objectif premier est de superviser l'évolution de robots mobiles manuellement à partir d'une interface touchpad/écran, grâce à laquelle un opérateur peut tracer des trajectoires. On suppose qu'un logiciel reproduit virtuellement l'environnement dans lequel évolue le robot. Des informations 2D issues de l'interface tactile (où tout autre périphérique d'entrées 2D) sont capturées puis retranscrites pour obtenir un jeu de positions dans l'espace opérationnel. Après avoir extrait les points significatifs, ceux-ci sont interpolés à l'aide d'une trajectoire $\mathcal{C}^1$ modélisée par des courbes de Bézier cubiques :

$$Q(t) = \overrightarrow{a}t^3 + \overrightarrow{b}t^2 + \overrightarrow{c}t + P_0, \text{ avec } \begin{cases} \overrightarrow{c} = 3(P_1 - P_0) \\ \overrightarrow{b} = 3(P_2 - P_1) - \overrightarrow{c} \\ \overrightarrow{a} = P_3 - P_0 - \overrightarrow{b} - \overrightarrow{c} \end{cases} \tag{2.6}$$

Les conditions aux limites sont les suivantes :

$$\begin{cases} Q(0) = P_0,\ Q(1) = P_4 \\ \dot{Q}(0) = 3[P_1 - P_0],\ \dot{Q}(1) = 3[P_3 - P_2] \end{cases} \tag{2.7}$$

L'arrivée imprévue d'obstacles est gérée par la modification en ligne des points de contrôle. Selon l'endroit $P(t)$, $t \in [0, 1]$ où se projette le véhicule, un unique point

de contrôle va être actualisé afin que la tangente $\dot{P}(t)$ à la courbe soit dictée par une stratégie d'évitement d'obstacle et que la position courante du véhicule reste proche de la trajectoire.

Dans [Guo 07] la trajectoire n'est pas supervisée et le robot calcule de façon autonome une trajectoire à suivre libre d'obstacle avec pour objectif de joindre successivement un certain nombre de points de passage. La première étape consiste à construire une trajectoire globale joignant ces points de passage de façon naturelle. Pour cela les auteurs s'appuyent sur la stratégie décrite dans [Junkins 72] qui exploite des splines de degré trois pour générer des courbes lisses ($\mathcal{C}^2$). En réalité, cette modélisation n'est que partiellement exploitée puisque l'objectif de cette étape était uniquement de définir des points de passage assurant un comportement naturel : la trajectoire globale est perçue comme une succession de configurations dont chacune est la donnée d'un triplet (position, tangente et courbure). Une fois cette étape préalable validée, le robot doit calculer en ligne la trajectoire permettant de joindre chaque paire de configurations. Selon les équations (2.8), chaque portion de trajectoire est modelisée par des polynômes d'ordre six, soit quatorze paramètres c_0 à c_6 et d_0 à d_6.

$$\begin{aligned} Q_x(t) &= c_0 + c_1 t + \cdots + c_6 t^6 \\ Q_y(t) &= d_0 + d_1 t + \cdots + d_6 t^6 \end{aligned} \tag{2.8}$$

Comme on l'a vu précédemment dans [Piazzi 02], des polynômes d'ordre cinq suffisent pour joindre deux configurations. La trajectoire peut en effet être reformulée de sorte que les paramètres c_0 à c_5 et d_0 à d_5 s'expriment en fonction des conditions aux limites et des deux paramètres de plus haut degré c_6 et d_6. Ainsi, si le système n'est soumis à aucune perturbation extérieure, les coefficients c_6 et d_6 sont fixés égaux à zéro. Si au contraire un ou plusieurs obstacles entrent dans le champ de perception du robot, ces deux paramètres vont alors être réglés afin de satisfaire un critère d'évitement d'obstacle. Ce dernier est défini en s'appuyant sur le fait que les dimensions ainsi que la vitesse et la direction de déplacement des obstacles sont connues à chaque instant. Une zone de l'espace à éviter par la trajectoire est ainsi mise en évidence et se traduit analytiquement par deux inégalités du second ordre (une avec la variable c_6 et l'autre avec d_6). Les paramètres c_6 et d_6 sont alors choisis les plus petits possibles satisfaisant le critère d'évitement. Concernant le guidage des robots, les auteurs exploitent la propriété de platitude du modèle cinématique de type bicyclette pour calculer une commande en boucle ouverte le long de la trajectoire.

Bien que notre étude s'intègre dans le cadre de la planification de trajectoire pour robot mobile, plusieurs différences sont à prendre en compte. Concernant tout d'abord le cas hors-ligne, les approches développées précédemment s'appuyent sur un ensemble de configurations spécifiant avec précision la position des points de passage et des informations concernant l'allure de la courbe à leur niveau. Le comportement de la trajectoire entre deux configurations sera alors satisfaisant à condition que les points de passage ne soient pas trop éloignés. Dans notre cas, les points de passage sont obtenus grâce à un dispositif de localisation absolue. Comme la fréquence d'acquisition des données est

grande (comprise entre $5Hz$ et $15Hz$ selon le capteur) et que la vitesse d'un véhicule en milieu urbain est faible, deux données de localisation successives sont proches et la quantité de données pour modéliser une trajectoire serait importante, quelle que soit la représentation utilisée, s'il fallait faire passer une nouvelle courbe entre chaque point. Une solution envisageable serait de prendre des points plus espacés. Mais ilresterait encore deux difficultés : tout d'abord estimer avec précision les informations requises (tangente et courbure) aux points de passage, en dépit du bruit présent sur les données de localisation. Ces imprecisions accentuent encore plus la deuxième difficulté qui concerne le contrôle de la courbe entre les configurations désirées : on peut supposer qu'en l'absence de bruit les seules contraintes aux limites permettent de définir une trajectoire très proche des données de localisation (dans notre cas, il faudrait laisser libres certains paramètres, à la manière de [Piazzi 02] et [Eren 99] pour disposer de degrés de liberté nécessaires à l'ajustement des courbes). Mais ce n'est bien sûr plus vrai en présence de bruit. Si on se focalise maintenant sur la stratégie de génération en ligne, elle peut difficilement être spécifiée comme l'interpolation d'un jeu de configurations. La raison principale est que l'objectif ici n'est pas une modification de la trajectoire, mais une extension de cette dernière. L'attention est précisément portée sur la conservation de l'allure des courbes qui doit être représentative de l'évolution du véhicule leader. Dans ce sens il serait plus avantageux de décrire directement le problème comme étant l'approximation des données de localisation et non de le traiter comme une replanification.

2.3 Génération de la trajectoire

Dans cette section une stratégie de génération de la trajectoire, appliquée au contrôle du déplacement d'une flotte de véhicules, est présentée. Sa conception est mise en œuvre en accord avec les deux fonctionnalités de déplacement envisagées, à savoir :

• le mode "tout automatique", illustré Fig. 2.4, où un convoi se déplace le long d'une trajectoire définie à partir d'informations préalablement enregistrées.

• le mode "manuel", illustré Fig. 2.5, où le véhicule leader est conduit manuellement et définit en temps réel la trajectoire à suivre.

La représentation de la trajectoire, commune aux deux modes de fonctionnement considérés, est tout d'abord explicitée. L'optimisation d'une trajectoire exploitable en mode "tout automatique" est ensuite adressée, suivie de la stratégie employée pour l'extension en temps réel de la trajectoire en mode "manuel". Des résultats de simulation et expérimentaux sont finalement détaillés.

2.3.1 Représentation de la trajectoire

Comme il a été présenté au chapitre précédent, les variables à réguler pour opérer les contrôles latéral et longitudinal sont exprimées sous forme d'écarts à la trajectoire de référence (écart latéral y et angulaire $\tilde{\theta}$) et de mesures extraites sur cette dernière

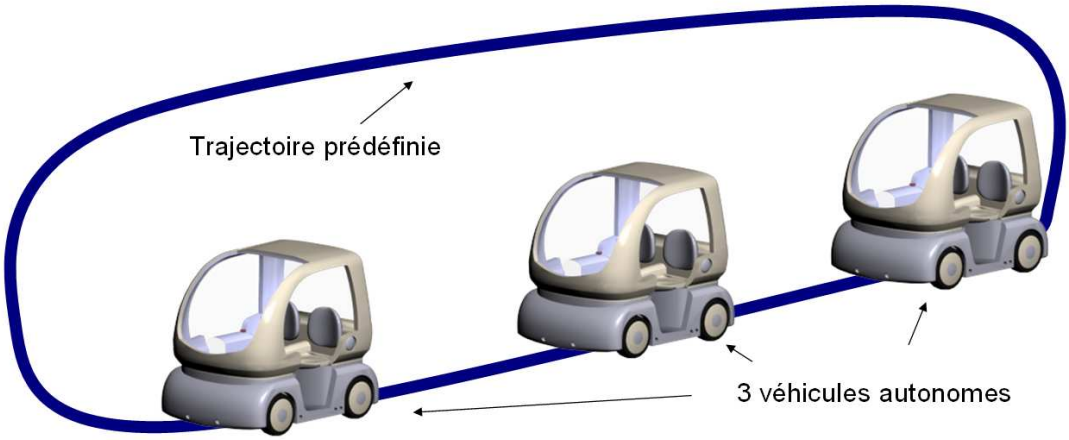


FIG. 2.4 – Exemple de guidage en mode “tout automatique”

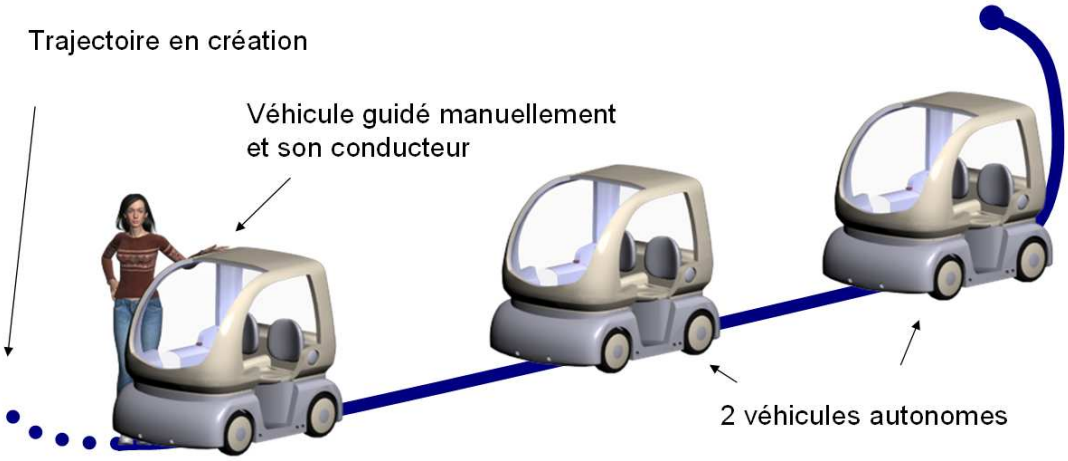


FIG. 2.5 – Exemple de guidage en mode “manuel”

(abscisse curviligne s). En conséquence, certaines caractéristiques sont imposées à la trajectoire de référence :

• Une présentation sous une forme analytique est essentielle pour positionner avec précision les véhicules sur la trajectoire et en extraire les variables nécessaires au calcul de la commande des véhicules.

• Le calcul de la courbure $c(s)$, demandé par les lois de commande, implique une trajectoire dont le degré de continuité est supérieur ou égal à deux. Des effets indésirables peuvent en effet survenir, comme des glissements ou des déviations latérales, si le degré

de continuité requis n'est pas atteint.

Comme nous n'ajoutons pas de contraintes liées à l'interpolation des points de passage, alors ces exigences peuvent être satisfaites à partir de courbes polynômiales de degré supérieur ou égal à trois. Comme elles présentent des propriétés attrayantes (la représentation par des points de contrôle est simple et la continuité des courbes au niveau des jonctions est assurée), nous avons choisi de modéliser la trajectoire grâce à l'utilisation de B-Splines. Ces dernières vont être construites à partir d'un vecteur de positions 2D, noté Σ par la suite, qui peut être constitué à partir d'informations de localisation absolue du véhicule leader (en mode "manuel") ou bien obtenu virtuellement si la trajectoire est simulée ou extraite d'une carte géoréférencée.

Pour aborder les sections qui suivent un bref rappel concernant les B-Splines et leur expression est exposé ci-dessous. Les B-Splines consistent en la connexion de courbes polynomiales notées $Q^i(t)$. Dans notre cas, ce sont des courbes sur un plan :

$$Q^i(t) = \left(\; Q_x^i(t), \quad Q_y^i(t) \; \right), \qquad 1 \leq i \leq n \text{ et } t \in [0,1]$$

Chacune de ces courbes $Q^i(t)$ est une combinaison linéaire d'une base de polynômes dont le degré d et les coefficients sont déterminés de façon à satisfaire les contraintes de continuité entre $Q^i(t)$ et $Q^{i+1}(t)$, à savoir :

$$\forall k \leq d-1, \; \forall i \in \{1,..,n-1\}, \; \frac{d^k Q^i(t)}{dt^k}_{\backslash t=1} = \frac{d^k Q^{i+1}(t)}{dt^k}_{\backslash t=0} \tag{2.9}$$

Pour les valeurs $i \in \{2,..,n-1\}$, les équations (2.9) représentent les contraintes définies aux deux extrémités des B-Splines $Q^i(t)$ (i.e. en $t=0$ et $t=1$). La base de polynômes correspondante et ses coefficients sont respectivement notés $\{B_j(t)\}_{(0\leq j\leq d)}$ et $(b_{jk})_{(0\leq j,k\leq d)}$ par la suite. Concernant les premier et dernier polynômes ($i=1$ et $i=n$), les équations (2.9) ne présentent des contraintes qu'à une extrémité. Deux autres bases de polynômes sont alors considérées permettant d'affecter les valeurs de position aux extrémités non contraintes : $Q^1(0) = p_{initial}$ et $Q^n(1) = p_{final}$. Ces bases de polynômes sont notées, $\{A_j(t)\}_{(0\leq j\leq d)}$ pour $i=1$, et $\{C_j(t)\}_{(0\leq j\leq d)}$ pour $i=n$ et leurs coefficients respectifs $(a_{jk})_{(0\leq j,k\leq d)}$ et $(c_{jk})_{(0\leq j,k\leq d)}$. La $i^{ème}$ courbe ($\forall i \in \{1,..,n\}, \forall t \in [0,1]$) s'exprime donc ainsi :

$$\begin{aligned} Q^i(t) &= \left(\; Pol_0(t) \quad \ldots \quad Pol_d(t) \; \right) \begin{pmatrix} P_x^i & P_y^i \\ \vdots & \vdots \\ P_x^{i+d} & P_y^{i+d} \end{pmatrix}, \text{ avec} \\ &\begin{cases} Pol_j(t) = A_j(t) \text{ si } i = 1, \\ Pol_j(t) = B_j(t) \text{ si } i \in \{2,..,n-1\}, \text{ et} \\ Pol_j(t) = C_j(t) \text{ si } i = n \end{cases} \end{aligned} \tag{2.10}$$

Les coefficients de la combinaison linéaire sont représentés par les points de contrôle $\left(P_x^i, P_y^i\right)_{(1\leq i\leq n)}$ qui, comme le montre la Fig. 2.6, modèlent la forme des B-Splines.

2.3.2 Optimisation de la trajectoire dans le cas hors ligne

On considère ici le cas où des positions ont été préalablement définies (stockées dans un vecteur noté Σ) dans un repère plan et constituent une référence de passage pour les véhicules. L'optimisation de la trajectoire de référence revient alors à trouver le jeu de points de contrôle $\left(P_x^i, P_y^i\right)_{(1\leq i\leq n)}$ optimal de sorte que chaque polynôme $\left\{Q^i(t)\right\}_{(1\leq i\leq n)}$ recolle au mieux à ces données pré-établies Σ.

2.3.2.1 Principe de l'optimisation

Le principe d'optimisation considère les quatre étapes suivantes :

1. La trajectoire brute, caractérisée par les données de localisation (ou de positions $2D$ virtuelles), est tout d'abord découpée en plusieurs fenêtres de longueur d'arc constante (cette longueur est notée c_2 par la suite). A ce stade les longueurs d'arc sont grossièrement estimées par la somme des distances euclidiennes entre chaque point. L'objectif est d'établir une correspondance entre la $i^{ème}$ fenêtre et le polynôme $Q^i(t)$ décrivant la trajectoire.

2. A cet effet, l'expression littérale des polynômes $\left\{Q^i(t)\right\}_{(1\leq i\leq n)}$, exprimée en fonction de ses points de contrôle, est échantillonnée : $\left\{Q_x^i(t_{i,j}), Q_y^i(t_{i,j})\right\}_{j\in J_i,\ t_{i,j}\in[0,1]}$. J_i est l'ensemble d'indices dont chaque élément cible une donnée de localisation sur la $i^{ème}$ fenêtre. Une paramétrisation uniforme est initialement utilisée pour l'échantillonnage : $\forall i \in \{1,..,n\}\,,\ \forall j \in \{1,..,card(J_i)\}\,,\ t_{i,j} = j/card(J_i)$. Les données de localisation de la $i^{ème}$ fenêtre $\left\{\Sigma_x^{i,j}, \Sigma_y^{i,j}\right\}_{j\in J_i}$ sont alors associées avec l'échantillonnage du polynôme $\left\{Q_x^i(t_{i,j}), Q_y^i(t_{i,j})\right\}_{j\in J_i,\ t_{i,j}\in[0,1]}$ correspondant.

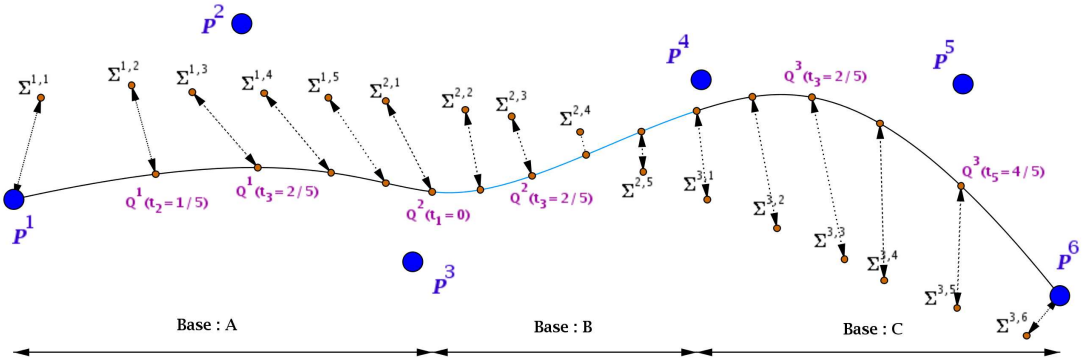


FIG. 2.6 – Echantillonnage des B-Splines

3. Comme le montre la Fig. 2.6, la traditionnelle méthode des moindres carrés est utilisée pour minimiser l'erreur de projection des données de localisation sur les

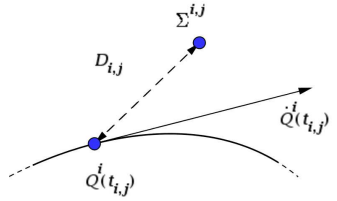
B-Splines (équation (2.11)).

$$\sum_{i=1}^{n} \sum_{j \in J_i} \left[(\Sigma_x^{i,j} - Q_x^i(t_{i,j}))^2 + (\Sigma_y^{i,j} - Q_y^i(t_{i,j}))^2 \right] \tag{2.11}$$

La valeur optimale des points de contrôle est ainsi retournée à partir du système linéaire surdéterminé obtenu par inversion du système (2.12).

$$\begin{pmatrix} \begin{pmatrix} A_0(t_1) & \dots & A_d(t_1) \\ \vdots & & \vdots \\ A_0(t_{N_1}) & \dots & A_d(t_{N_1}) \end{pmatrix} & & \\ & \ddots & \\ & \begin{pmatrix} B_0(t_1) & \dots & B_d(t_1) \\ \vdots & & \vdots \\ B_0(t_{N_i}) & \dots & B_d(t_{N_i}) \end{pmatrix} & \\ & & \ddots \\ & & \begin{pmatrix} C_0(t_1) & \dots & C_d(t_1) \\ \vdots & & \vdots \\ C_0(t_{N_n}) & \dots & C_d(t_{N_n}) \end{pmatrix} \end{pmatrix} \cdot \left(\begin{array}{c} P_x^1 \\ \vdots \\ P_x^{1+d} \\ \hline \dots \\ \dots \\ \hline P_x^i \\ \vdots \\ P_x^{i+d} \\ \hline \dots\dots \\ \hline P_x^n \\ \vdots \\ P_x^{n+d} \end{array} \right) = \left(\begin{array}{c} \Sigma_x^{1,1} \\ \vdots \\ \Sigma_x^{1,N_1} \\ \hline \dots \\ \dots \\ \hline \Sigma_x^{i,1} \\ \vdots \\ \Sigma_x^{i,N_i} \\ \hline \dots\dots \\ \hline \Sigma_x^{n,1} \\ \vdots \\ \Sigma_x^{n,N_n} \end{array} \right) \tag{2.12}$$

4. Afin d'obtenir de meilleurs résultats lors de l'étape d'optimisation, l'échantillonnage des polynômes est raffiné et permet ainsi une meilleure correspondance avec les points contenus dans chacune des fenêtres. La méthode suit les travaux décrits dans [Üstünda 05]. La Fig. 2.6 montre en effet qu'une paramétrisation uniforme n'est pas appropriée. D'une part rien ne garantit que l'espacement entre chaque paire de données brutes adjacentes $\|\Sigma^{i,j+1} - \Sigma^{i,j}\|$ est constant. D'autre part la paramétrisation elle-même ne conduit pas à un échantillonnage dont les points adjacents $Q_x^i(t_{i,j})$ et $Q_x^i(t_{i,j+1})$ sont équidistants, particulièrement lorsque les bases de polynômes A et C sont considérées. Par exemple, sur la Fig. 2.6 pour un même pas de paramétrisation $\Delta_t = 1/5$ la distance entre $P^1 = Q^1(0)$ et $Q^1(1/5)$ est plus de deux fois plus élevée que celle entre $Q^1(4/5)$ et $Q^1(1) = Q^2(0)$ sur le même polynôme. En conséquence, une nouvelle paramétrisation $\widetilde{t}_{i,j}$ est définie ainsi : $\widetilde{t}_{i,j} = t_{i,j} + \Delta t_{i,j}$ (voir Fig. 2.7).



$$\Delta t_{i,j} = \langle D_{i,j}, \frac{\dot{Q}^i(t_{i,j})}{\|\dot{Q}^i(t_{i,j})\|} \rangle / c_2,$$
$$\text{avec } D_{i,j} = \Sigma^{i,j} - Q^i(t_{i,j})$$

FIG. 2.7 – Calcul de la nouvelle paramétrisation

Sur la Fig. 2.7, la notation $\langle \cdot, \cdot \rangle$ correspond au produit scalaire de deux vecteurs. Comme $\frac{\dot{Q}^i(t_{i,j})}{\|\dot{Q}^i(t_{i,j})\|}$ est un vecteur unitaire, le terme $\langle D_{i,j}, \frac{\dot{Q}^i(t_{i,j})}{\|\dot{Q}^i(t_{i,j})\|} \rangle$ représente alors la norme de la projection de l'erreur $(\Sigma^{i,j} - Q^i(t_{i,j}))$ sur le vecteur $\dot{Q}^i(t_{i,j})$. En s'appuyant sur la Fig. 2.7, il est intuitif que plus ce produit est important, plus il faudrait décaler le paramètre $t_{i,j}$ dans la direction de $\dot{Q}^i(t_{i,j})$ afin que $\|\Sigma^{i,j} - Q^i(t_{i,j})\|$. Pour que $\Delta t_{i,j}$ soit homogène à un terme de variation dans la paramétrisation, $\langle D_{i,j}, \frac{\dot{Q}^i(t_{i,j})}{\|\dot{Q}^i(t_{i,j})\|} \rangle$ est divisé par la longueur d'arc c_2 de la courbe.

Les étapes 3. et 4. sont réitérées jusqu'à ce que le résidu de la phase d'optimisation ne dépende plus que du bruit inhérent aux données de localisation.

2.3.2.2 Résultats de simulation

Les résultats de simulation présentés ci-dessous ont pour objectifs d'étudier la sensibilité et l'influence du degré d des B-Splines et de la longueur c_2 des fenêtres sur la qualité de la trajectoire générée.

Pour réaliser des simulations représentatives des conditions réelles, des données de localisation ont été enregistrées expérimentalement par un véhicule conduit manuellement et équipé d'un récepteur RTK GPS, le long de la trajectoire présentée ci-contre Fig. 2.8.

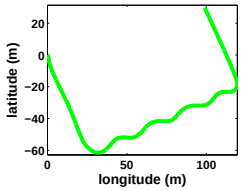


FIG. 2.8 – Trajectoire pour les simulations

Une batterie de simulations a été étudiée, explorant les valeurs des paramètres $d \in \{3, .., 5\}$ et $c_2 \in [1.5, 5]$. On rappelle que $d \geq 3$ est nécessaire pour que les B-Splines soient $\mathcal{C}^2$, voir l'équation (2.9). Chaque simulation est évaluée en fonction de la distance moyenne et maximale entre les données de localisation et leur projection sur la trajectoire de référence obtenue (c.a.d. les B-Splines). Etant donné que les variations de la courbure de la trajectoire de référence sont bornées (ce qui est naturel au vu des contraintes dynamiques sur les véhicules), il n'est pas nécessaire d'étudier de grandes valeurs pour le paramètre d. Il apparaît sur le tableau ci-contre (Fig 2.9) que $d = 3$ apporte des résultats satisfaisants. Ensuite, comme il était intuitivement attendu, les meilleurs résultats sont obtenus avec les plus petites valeurs de c_2, voir aussi Fig 2.9.

2.3.3 Génération en ligne de la trajectoire de référence

Dans ce cas, la trajectoire brute Σ s'incrémente au fur et à mesure du déplacement du véhicule leader. On note en conséquence Σ^T les données brutes disponibles à l'instant T. Idéalement, à chaque incrément, le critère (2.11) défini précédemment devrait être minimisé afin d'ajuster l'ensemble des points de contrôle présents et d'en ajouter de nouveaux quand cela devient nécessaire. Cependant le point crucial de cette procédure d'extension réside dans la stabilité des informations obtenues à partir de cette trajectoire et indispensables au calcul des commandes des véhicules. Il est en effet nécessaire que

d	c_2(m)	avec reparamétrisation		sans reparamétrisation	
		erreur maximale (m)	erreur moyenne (m)	erreur maximale (m)	erreur moyenne (m)
3	**1.5**	**0.0299**	**0.0033**	**0.0357**	**0.0045**
3	3	0.0497	0.0067	0.0638	0.0085
3	5	0.0740	0.0128	0.0962	0.0167
4	**1.5**	**0.0217**	**0.0033**	**0.0209**	**0.0036**
4	3	0.0412	0.0066	0.0383	0.0075
4	5	0.0705	0.0149	0.0996	0.0243
5	**1.5**	**0.0252**	**0.0065**	**0.0570**	**0.0079**
5	3	0.0459	0.0065	0.0570	0.0079
5	5	0.0789	0.0127	0.1158	0.0165

FIG. 2.9 – Influence de d et c_2

chaque variable (s_i, y_i, $\tilde{\theta}_i$) caractérisant les véhicules suiveurs reste cohérente, et pour cela il convient de limiter au mieux les perturbations occasionnées par l'avancée du véhicule leader sur la portion de trajectoire déjà construite (sur laquelle les véhicules suiveurs sont déjà asservis). Afin d'obtenir une trajectoire la plus stable possible, seuls les polynômes Q^i les plus récents vont être mis à jour. Cela permet aussi de borner le temps de calcul de l'optimisation, ce dernier étant lié au nombre de B-Splines, de points de contrôle et de données brutes considérés. De plus la taille de la bande passante attribuée à la retransmission en ligne de la trajectoire est maîtrisée puisqu'un nombre limité de points ajustés sont à communiquer au reste de la flotte. Plus précisément, on note n_{ac} (comme active curves) le nombre de polynômes Q^i considérés dans la procédure d'optimisation. De façon similaire à (2.11), un critère de minimisation, cette fois-ci local (2.13), est défini en ne tenant compte que des n_{ac} derniers polynômes. Ainsi chaque polynôme $(n - n_{ac} + 1 \leq i \leq n)$ est associé à un sous-espace $\Sigma^{T,i} \subseteq \Sigma^T$.

$$\sum_{i=n-n_{ac}+1}^{n} \sum_{j \in J_i} \left[(\Sigma_x^{T,i,j} - Q_x^i(t_{i,j}))^2 + (\Sigma_y^{T,i,j} - Q_y^i(t_{i,j}))^2 \right] \tag{2.13}$$

Si l'on se concentre sur le principe de construction des B-Splines, illustré Fig. 2.10, on constate que pour modifier la forme d'une courbe on a une certaine liberté sur le choix et le nombre de points de contrôle à déplacer. Ainsi, dans le critère d'optimisation local (2.13), le choix des ces points se porte naturellement vers les derniers (puisque l'on veut optimiser les derniers polynômes) mais leur nombre reste un paramètre à étudier. On note ce dernier n_{ap} (comme active points). Il est important de remarquer que des B-Splines dont l'indice est proche partagent des points de contrôle communs. De ce fait, la valeur de n_{ap} est un paramètre crucial puisque la modification d'un seul point engendre des conséquences sur un ensemble de $d+1$ polynômes adjacents Q^i. On voit clairement sur les encadrés du bas de la Fig. 2.10 que pour le degré $d = 3$ par exemple, le décalage du point P^{i+3} va déformer quatre B-Splines (Q^i à Q^{i+3}) alors qu'en degré $d = 5$, le

point P^{i+5} est actif sur six B-Splines (Q^i à Q^{i+5}). Inversement on peut remarquer que chaque polynôme Q^i est défini à partir de $d+1$ points de contrôle (encadrés du haut de la Fig. 2.10) :

- Q^1 est défini par $(P_x^1, P_y^1), \ldots, (P_x^{d+1}, P_y^{d+1})$,
- Q^i est défini par $(P_x^i, P_y^i), \ldots, (P_x^{i+d}, P_y^{i+d})$,
- $Q^{n-n_{ac}+1}$ est défini par :
  $(P_x^{n-n_{ac}+1}, P_y^{n-n_{ac}+1}), \ldots, (P_x^{n-n_{ac}+d+1}, P_y^{n-n_{ac}+d+1})$,
- Q^n est défini par : $(P_x^n, P_y^n), \ldots, (P_x^{n+d}, P_y^{n+d})$.

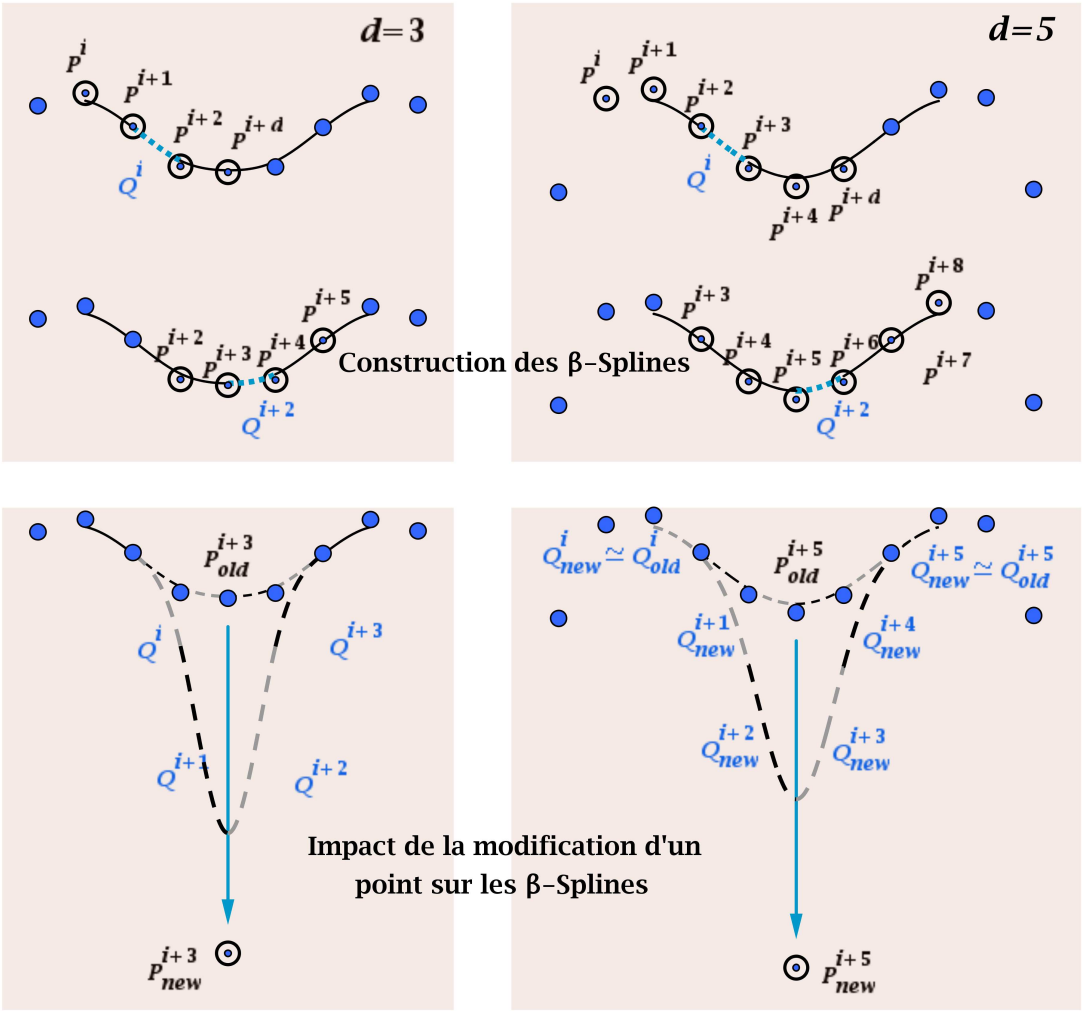


FIG. 2.10 – Construction des B-Splines

Ainsi si l'on optimise l'ensemble des points associés aux n_{ac} polynômes (c.a.d. $n_{ap} = n_{ac} + d$), alors $n_{ac} + d$ polynômes ($Q^{n-n_{ac}-d+1}$ à Q^n) vont être modifiés alors que seuls n_{ac} sont explicitement adressés dans (2.13). L'ajustement des polynômes $Q^{n-n_{ac}-d+1}$ à

$Q^{n-n_{ac}}$ n'est ainsi plus explicitement contrôlé. Les modifications engendrées sont de plus irréversibles : comme les points actifs sont amenés à évoluer au cours du processus de génération de la trajectoire, ces modifications s'accumuleraient et pourraient propager une erreur d'ajustement qui serait au final perceptible sur l'ensemble de la trajectoire.

A l'opposé, si le nombre de points actifs est minimal (c.a.d. choisi tel que $n_{ap} = n_{ac}$), alors le critère (2.13) ne peut pas être conduit à son minimum puisque les points ($P_x^{n-n_{ac}+1}$, $P_y^{n-n_{ac}+1}$) à ($P_x^{n-n_{ac}+d}$, $P_y^{n-n_{ac}+d}$) n'interviennent plus dans le critère et ne sont donc plus ajustés. Les performances de l'ajustement local, et du coup celles de l'ajustement global, pourraient donc être détériorées.

Finalement un nombre trop faible ou trop important de points de contrôle n_{ap} ne permet pas d'obtenir de bonnes performances sur le critère de minimisation. Une batterie de tests, obtenue par simulation du processus de génération de la trajectoire, a permis d'expliciter plus en détail ces phénomènes, puis de discuter du meilleur compromis (d, n_{ac}, n_{ap}). Les résultats sont présentés et commentés par la suite dans la section 2.4.

2.3.4 Détails sur l'implémentation de l'algorithme

Afin de clarifier la procédure employée pour générer la trajectoire en ligne, les différentes étapes sont illustrées Fig. 2.11 et Fig. 2.12 avec les paramètres $d = 3$, $n_{ac} = 3$ et $n_{ap} = 4$. On distingue deux phases dans l'implémentation de cet algorithme. Après avoir énoncé leur objectif respectif et délimité leur périmètre d'action, le fonctionnement général de l'algorithme est ensuite décrit.

• ***Phase d'initialisation :*** Comme son nom l'indique, il s'agit d'une phase de préparation qui s'étend sur une courte distance et qui permet de construire avec pertinence les premières courbes nécessaires au bon déroulement de l'algorithme lors de la phase nominale qui suit. Dans cet esprit, à chaque incrément de la trajectoire brute Σ, le critère global (2.11) est minimisé afin d'ajuster l'ensemble des points de contrôle existants. Cette phase, décrite Fig. 2.11, débute lorsque la première donnée de localisation est disponible. Les d premiers points de contrôle sont alors fixés égaux, superposés à celle-ci. Elle se termine quand n_{ac} courbes actives sont présentes et que l'introduction d'une nouvelle courbe devient nécessaire.

• ***Phase nominale :*** Sa mise en route correspond au moment où le critère de minimisation local (2.13) est pris en considération (voir Fig. 2.12) : on se permet alors de se focaliser sur une partie de la trajectoire en ayant l'assurance que la portion délaissée va conserver les propriétés désirées. La trajectoire contient au minimum $n_{ac} + 1$ courbes et cette phase s'achève quand la trajectoire est entièrement construite.

2.3.4.1 Structure globale de l'algorithme

Maintenant que les phases de la procédure ont été clairement présentées, on va étudier plus en détail la structure de l'algorithme (Algo. 1). La trame globale, décrite sur l'Algo. 1, décrit les actions à entreprendre quand une nouvelle donnée de localisation

est disponible. Elle se décompose en deux parties :

1) On sélectionne tout d'abord les courbes et points de contrôle actifs ainsi que le support de données (i.e. le choix des fenêtres de données brutes $\Sigma^{T,i}$) sur lesquelles les B-Splines vont s'ajuster. Cette première partie de préparation des données est capitale et c'est à ce niveau que l'agencement de la trajectoire est déterminé. C'est en effet à ce stade que les paramètres de la génération interviennent. Cette partie, qui dépend naturellement de la phase de l'algorithme, se traduit par les fonctions "p_ini" et "p_nom" apparaissant sur l'Algo. 1 et fait l'objet de la sous section qui suit.

2) Ensuite la fonction d'approximation, générique aux deux phases, retourne la valeur des nouveaux points de contrôle selon le critère d'optimisation considéré en appliquant la méthode des moindres carrés.

Fonction optimisePtsCtrl(Ptrl, sT, i_sT, new_pt, deg, c1, c2, nac, nap) :

// ***Ptrl : points de contrôle actuels***
// ***sT : vecteur contenant les données de*** Σ^T
// ***i_sT : indices permettant de retrouver les fenêtres*** $\Sigma^{T,i}$ ***dans*** Σ^T
// ***new_pt : nouvelle donnée brute de localisation***
// ***deg :*** d***, c1 :*** c_1***, c2 :*** c_2***, nac :*** n_{ac}***, nap :*** n_{ap}

// ***Ajout du nouveau point à la dernière fenêtre*** $\Sigma^{T,i}$
t_fen←add_new_pt(sT, new_pt) ;
// ***t_fen : longueur totale des fenêtres (en m)***

// ***1) Sélection des fenêtres*** $\Sigma^{T,i}$ ***et définition du nombre n_pt de points à optimiser***
// ***n_poly : nombre de B-Splines actuel***
Si (n_poly ≤ nac) **Alors**
  // ***phase d'initialisation***
  [n_poly,n_pt, sT, i_sT] ←p_init(c1,c2,nap,nac,n_poly,t_fen,sT,i_sT) ;
Sinon
  // ***phase nominale***
  [n_poly,n_pt, sT, i_sT] ←p_nom(c1,c2,nap,nac,n_poly,t_fen,sT,i_sT) ;
Fin Si

// ***2) Ajustement des n_pt points de contrôle Ptrl***
[Ptrl,sT, i_sT] ←ajuste_pt(n_pt, n_poly, deg, Ptrl, sT, i_sT) ;
Retourner ←[Ptrl,sT, i_sT]
Fin

Algorithme 1: Optimisation des points de contrôle, trame générale

2.3.4.2 Sélection des données du critère d'approximation

Le choix concernant les points de contrôle a été présenté lors de la description précédente des phases de l'algorithme : tous les points de contrôle sont actifs lors de l'initialisation, alors que seuls les n_{ap} derniers apparaissent en phase nominale dans la formule du critère local 2.13, voir les Fig. 2.11 et 2.12.

Remarque : En phase d'initialisation le nombre de points à optimiser peut devenir supérieur à la consigne n_{ap}, ce qui permet d'avoir le maximum de liberté pour contrôler la forme de la trajectoire. La variable "n_pt", qui donne le nombre de points à optimiser, a pour cette raison été introduite dans Algo. 1 : elle est déterminée pendant l'étape 1) et est utilisée au cours de l'étape 2).

La discussion sur le choix des données brutes à intégrer dans l'optimisation et leur répartition en fenêtres d'intérêt est un peu plus délicate. De façon analogue à l'optimisation hors-ligne, on souhaite que chaque B-Spline à optimiser (à savoir $Q^i_{(n-n_{ac}+1\leq i\leq n)}$ en phase nominale ou $Q^i_{(1\leq i\leq n)}$ en phase d'initialisation) soit associée à un sous-espace $\Sigma^{T,i}$ dont la longueur d'arc est constante. Comme précédemment cette valeur est notée c_2. Pour expliquer la solution adoptée on s'intéresse au moment décisif où l'on va devoir intégrer un nouveau point de contrôle : une nouvelle B-Spline va faire son apparition (voir Fig. 2.11 b), d) et Fig. 2.12 f), h)), et afin de conserver la cohérence de la trajectoire, il va falloir respecter la correspondance entre les anciennes B-Splines Q^i et les fenêtres $\Sigma^{T,i}$. Ainsi l'optimisation de la B-Spline nouvellement créée Q^{n+1} nécessite de faire apparaître une nouvelle fenêtre d'intérêt $\Sigma^{T,n+1}$. Pour ce faire, l'alternative qui paraît la plus judicieuse consiste à découper l'ancienne dernière fenêtre $\Sigma^{T,n}$ en deux parties. Afin de s'assurer qu'elle contient suffisamment d'information pour obtenir une optimisation de bonne qualité, il est nécessaire que sa taille soit supérieure à un certain seuil $c_1 > c_2 > 0$. Le découpage s'opère alors de sorte que la longueur d'arc de la nouvelle dernière fenêtre $\Sigma^{T,n+1}$ soit égale à $c_1 - c_2$. Cette opération, que l'on nommera simplement *découpage* par la suite, assure ainsi que la longueur d'arc de chaque fenêtre soit de taille c_2 exceptée la dernière. Plus concrètement le *découpage* procède à une redispersion des fenêtres (effectuée dans les fonctions "dispersionFen" et "glissementFen" des Algo. 2 et 3) : un vecteur d'indices permettant de retrouver les fenêtres $\Sigma^{T,i}$ dans Σ^T est mis à jour.

La condition qui vient à l'instant d'être mise en évidence pour réaliser l'opération de *découpage*, est utilisée pour caractériser l'instant où il faut ajouter un nouveau point de contrôle. Suivant ce principe, quand une nouvelle donnée de localisation est disponible, deux possibilités sont à envisager, en fonction de la distance curviligne le long du dernier polynôme Q^n. Cette dernière est, comme dans le cas hors-ligne, grossièrement estimée par la somme des distances euclidiennes entre les points adjacents.

• Soit la longueur d'arc du dernier polynôme Q^n est inférieure au seuil c_1. Alors, quelle que soit la phase de l'algorithme, les polynômes intervenant dans le critère de minimisation restent associés aux mêmes fenêtres sur la trajectoire brute. La dernière fenêtre $\Sigma^{T,n}$ est toutefois légèrement modifiée puisque la nouvelle donnée de localisation

y est incorporée. Il s'agit de l'opération de *remplissage* illustrée Fig. 2.11 a), c) et e) Fig. 2.12 g).

• Soit la longueur d'arc du dernier polynôme Q^n dépasse le seuil c_1. Alors un nouveau point de contrôle va être ajouté puis les fenêtres d'intérêt $\Sigma^{T,i}$ vont être réarrangées selon la phase de l'algorithme :

$\rightsquigarrow$ *En phase d'initialisation*, on commence par vérifier la condition de sortie de la phase : si le nombre de polynômes actuel est égal au nombre de courbes actives, on exécute les instructions de la phase nominale (voir tiret suivant). Sinon on procède simplement à l'opération de *découpage*.

$\rightsquigarrow$ *En phase nominale*, n_{ac} et n_{ap} étant constants, la première fenêtre qui intervenait dans le critère de minimisation local (2.13) ne sera plus utilisée par la suite. Afin de ne pas garder en mémoire des données inutiles, l'algorithme ne conserve pas l'historique complet des données brutes. En conséquence la première fenêtre est toujours indexée $\Sigma^{T,1}$ même s'il s'agit en réalité de l'ensemble $\Sigma^{T,n-n_{ac}+1}$ dans la notation du critère d'optimisation local (2.13). On va donc simplement décrémenter les indices des fenêtres, ce qui a pour conséquence de faire disparaître la première fenêtre. Combinée au *découpage* de la dernière fenêtre, cette opération de *glissement*, illustrée Fig. 2.12 f) et h) permet de conserver perpétuellement un nombre n_{ac} de fenêtres d'intérêt équilibrées en terme de quantité et de dispersion de données.

```
Fonction p_init(c1,c2,nac,nap,n_poly,t_fen,sT,i_sT) :
    // On vérifie si il faut réarranger les fenêtres Σ^{T,i}
    Si (t_fen<(nac-1)*c2+c1) Alors
        n_pt←n_poly+deg ;
    Sinon
        // On vérifie si il faut faire glisser les fenêtres Σ^{T,i}
        Si (n_poly=nac) Alors
            // Glissement et redispersion des fenêtres Σ^{T,i}
            [sT,i_sT] ←glissementFen(c1,c2,sT,i_sT) ;
            n_poly←n_poly+1 ;
            n_pt← nap ;
        Sinon
            // redispersion des fenêtres Σ^{T,i}
            i_sT←dispersionFen(c1,c2,sT,i_sT) ;
            n_poly ← n_poly+1 ;
            n_pt ← n_poly+deg
        Fin Si
    Fin Si
    Retourner ←[n_poly, n_pt, sT,i_sT]
Fin
```

Algorithme 2: Arrangement des fenêtres : phase d'initialisation

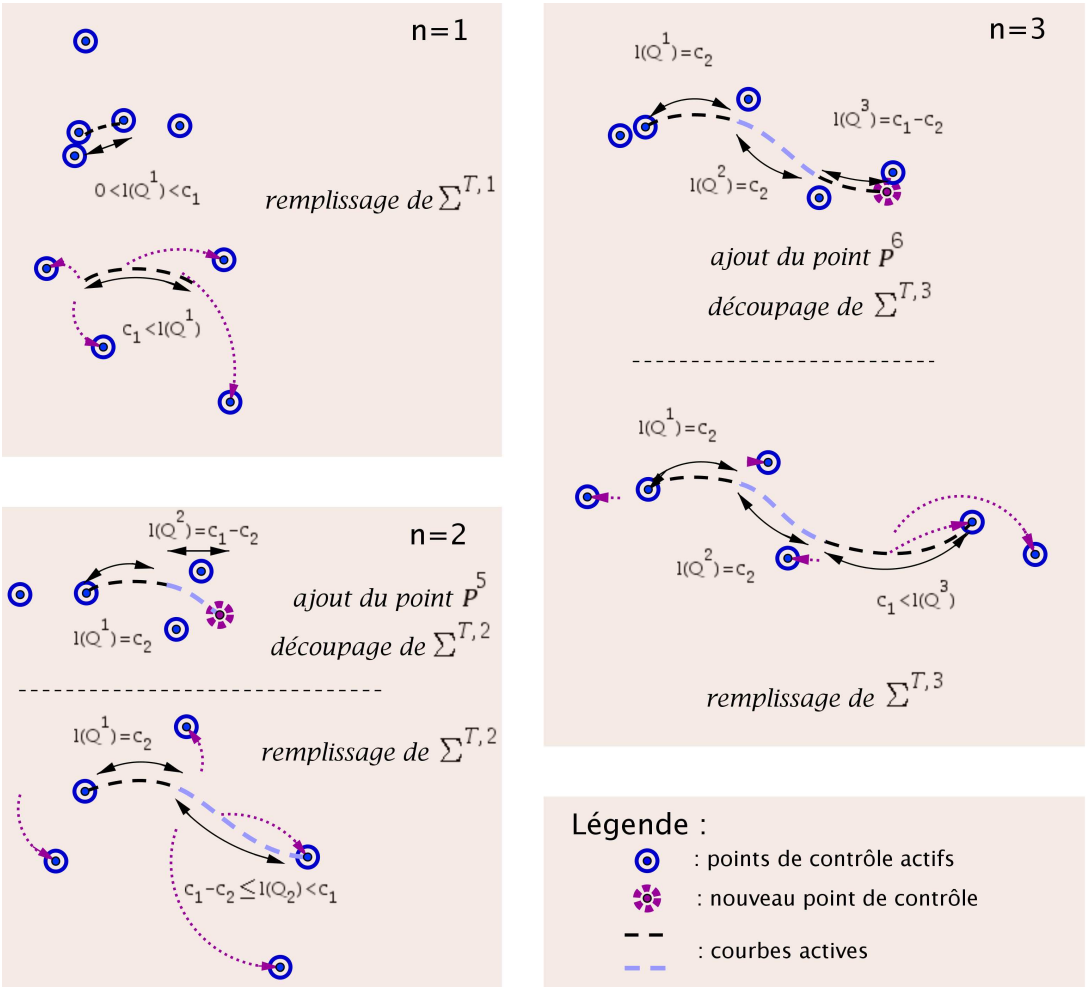


FIG. 2.11 – Extension des B-Spline : phase d'initialisation

Fonction p_nom(c1,c2,nac,nap,n_poly,t_fen,sT,i_sT) :
    n_pt=nap ;
    // ***On vérifie si il faut réarranger les fenêtres*** $\Sigma^{T,i}$
    Si (t_fen<(nac-1)*c2+c1) **Alors**
        // ***Glissement et redispersion des fenêtres*** $\Sigma^{T,i}$
        [sT,i_sT] ←glissementFen(c1,c2,sT,i_sT) ;
        n_poly ← n_poly+1 ;
    Fin Si
    Retourner ←[n_poly, n_pt, sT,i_sT]
Fin

Algorithme 3: Arrangement des fenêtres : phase nominale

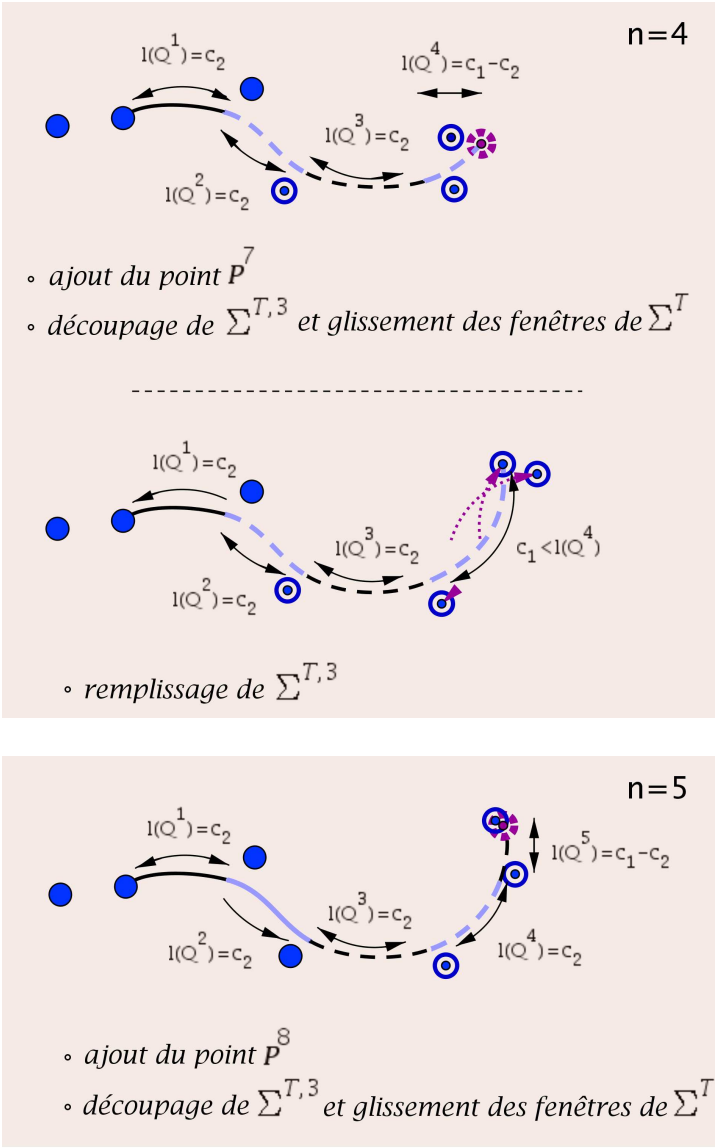


FIG. 2.12 – Extension des B-Spline : phase nominale

Afin de simultanément réduire le temps de calcul et traiter le cas statique (le véhicule qui génère la trajectoire n'est pas en mouvement à un moment donné au cours du processus de création), une nouvelle donnée brute de localisation est prise en compte, c.a.d ajoutée à Σ^T, uniquement si sa distance euclidienne à la dernière donnée intégrée est supérieure à un certain seuil.

Finalement, afin de calculer l'abscisse curviligne s_1 du véhicule leader jusqu'à la prochaine acquisition de sa localisation, le déplacement de ce véhicule est prédit sur un court horizon et les positions prédites sont elles aussi intégrées à Σ^T lors de l'optimisation des points de contrôle.

2.4 Résultats de simulation pour la génération en ligne de la trajectoire

Cette section est divisée en deux parties : des résultats de simulation sont tout d'abord présentés pour tester la sensibilité des principaux paramètres de génération en ligne de la trajectoire et les valeurs les plus appropriées pour ces paramètres sont identifiées. Un indicateur est ensuite proposé pour quantifier la qualité de la trajectoire en fonction de l'ensemble des paramètres.

2.4.1 Création en ligne de la trajectoire

A des fins de comparaison avec les résultats obtenus dans le cas hors-ligne et aussi pour se placer dans des conditions réelles, la trajectoire brute utilisée est la même que celle qui a été considérée lors des simulations dans le cas hors-ligne. On rappelle que cette dernière, présentée ci-contre Fig. 2.13, a été enregistrée expérimentalement par un véhicule conduit manuellement et équipé d'un récepteur RTK GPS.

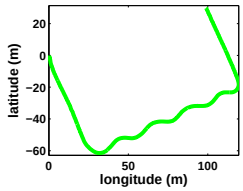


FIG. 2.13 – Trajectoire pour les simulations

Comme pour le cas hors-ligne, une batterie de simulations a été étudiée, explorant cette fois-ci les valeurs des paramètres $d \in \{3, .., 5\}$, $c_2 \in [1.5,\ 5]$, $n_{ac} \in \{2, .., 8\}$ et $n_{ap} \in \{n_{ac}, .., n_{ac} + d\}$.

2.4.2 Critères de qualité de la trajectoire

Au vu des objectifs de la génération, deux critères ont été mis en exergue pour étudier la qualité d'une trajectoire. Les variables qui en découlent sont explicitées ci-dessous puis récapitulées Fig. 2.14.

• ***Fidélité de la représentation :*** Comme il a été spécifié auparavant, cette procédure en ligne a pour premier objectif de générer une trajectoire qui décrit avec justesse les positions de passage du véhicule de tête. Des mesures d'écartement à la trajectoire sont ainsi calculées et l'on évalue ce critère de qualité en fonction des distances moyenne

et maximale entre les données brutes de localisation et leur projection sur la trajectoire de référence obtenue (c.a.d. la B-Spline).

• ***Perturbations occasionnées au cours de la génération :*** L'intérêt d'une telle procédure est bien évidemment de pouvoir simultanément construire et exploiter en ligne la trajectoire. On doit par conséquent s'assurer de la cohérence des informations extraites de la trajectoire pour le calcul de la commande. A cet effet les variations locales qui se manifestent sur les polynômes actifs sont quantifiées selon plusieurs critères à travers le protocole suivant : tout au long de la génération, lorsque la trajectoire vient juste d'être optimisée, une série de points uniformément répartis sur les polynômes actifs est sélectionnée. On évalue alors l'orientation de la tangente et la courbure à la trajectoire en ces points. Dès que la trajectoire brute Σ est incrémentée, on mesure la distance euclidienne entre la série de points sur les anciennes B-Splines et sa projection sur les B-Splines nouvellement optimisées. On calcule aussi la différence d'orientation et de courbure au niveau des points de projection. On dispose alors d'informations caractéristiques des perturbations entre deux itérations de la procédure d'extension. A chacune de ces itérations les erreurs maximales sont répertoriées. Ces erreurs vont finalement être moyennées sur l'ensemble des itérations de la simulation pour constituer un indicateur représentatif sur la globalité d'une simulation.

fidélite de la représentation :		**perturbations occasionnées :**		
erreur maximale (m)	erreur moyenne (m)	moyenne des variations absolues sur la : distance (m)	direction (rad)	courbure (m^{-1})

FIG. 2.14 – Variables représentatives de la qualité de la trajectoire

Avant d'aborder l'analyse des paramètres de l'algorithme de génération, quelques résultats préliminaires sont à souligner concernant les relations entre ces variables représentatives de la qualité de la trajectoire. Comme il est illustré Fig. 2.15 et 2.17 les trajectoires générées présentant les meilleurs résultats de fidélité sont aussi celles au cours desquelles les plus petites perturbations sont recensées. En ciblant notre attention sur les résultats qui ne présentent pas de divergence de recollement trop importante (erreur moyenne inférieure à 10cm) on observe même sur la Fig. 2.15 une tendance linéaire entre l'erreur moyenne de fidélité et la moyenne des variations absolues sur la distance. Il est également rassurant de constater que l'amplitude des erreurs de fidélité est supérieure à celle des perturbations, de plus du double au vu de la Fig. 2.15. Cela assure qu'au cours de la génération il n'y a pas de "bond" (d'une itération à l'autre) de part et d'autre de la trajectoire réellement empruntée. En fin de simulation, le décalage que l'on mesure (par la variable de l'erreur moyenne) provient donc uniquement d'une accumulation de petites dérives locales, principalement lors des courbes (généralement vers l'intérieur de la courbe) ou lorsque des variations de courbure sont imposées par le véhicule. On observe aussi Fig. 2.15 que pour des performances de fidélité égales, les perturbations de la trajectoire augmentent avec une tendance inverse à la valeur du degré d. Le degré des B-Splines joue donc un rôle dans la stabilité de la trajectoire lors de la création.

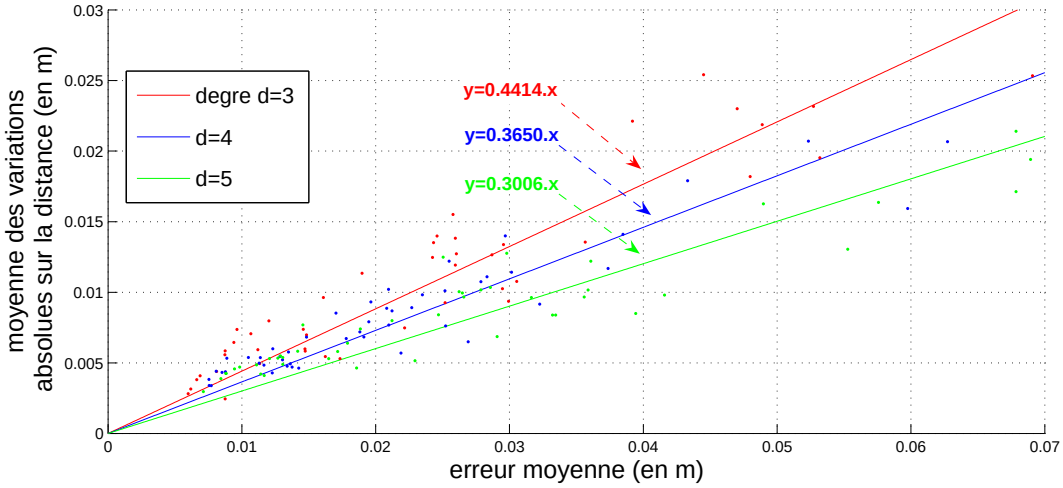


FIG. 2.15 – Relation entre les variations absolues de la trajectoire et la fidélité

2.4.3 Analyse des paramètres

L'analyse de l'influence des paramètres de génération n'est pas chose évidente : ces derniers sont très corrélés et il est difficile pour la plupart d'isoler l'incidence de chacun sur la qualité de la trajectoire. Par souci de clareté, les résultats de l'analyse de chaque paramètre (ou groupe de paramètres) vont tout d'abord être présentés indépendemment, en suivant un ordre de complexité croissant. On cherchera en premier lieu à établir une *correspondance directe* entre les valeurs du (des) paramètres et la qualité de la trajectoire. Puis, pour une valeur donnée du (des) paramètres, nous étudierons sur quelle plage peut varier la qualité de la trajectoire lorsque l'on modifie les autres paramètres (*corrélations*). Finalement, après avoir décomposé les influences des différents paramètres, une analyse plus fine est mise en place en explorant simultanément l'ensemble des paramètres. Cette étude permet de faire ressortir des indicateurs sur la qualité de la trajectoire et ainsi d'extraire les paramètres optimaux à utiliser au cours des expérimentations.

- La taille des fenêtres d'intérêt c_2 :

$\rightsquigarrow$ Impact direct. Assez logiquement les conclusions à tirer pour ce paramètre suivent celles du cas hors-ligne. Les meilleurs résultats sont obtenus avec les plus petites valeurs de c_2. Cependant, les bénéfices procurés par l'utilisation de courbes B-Spline est perdu si la valeur de c_2 est choisie trop petite. Dans ce cas, le nombre de points de contrôle grandit rapidement avec la taille de la trajectoire, et par conséquent la représentation devient lourde lorsque la distance parcourue est grande. Par ailleurs, avec de petites valeurs de c_2 l'effet du bruit sur les données de localisation se ressent fortement sur l'estimation de la courbure. Ce phénomène est illustré Fig. 2.16 : les tracés de la courbure le long de la trajectoire, obtenus selon la méthode hors-ligne avec les valeurs $c_2 = 0.6m$ et $c_2 = 3m$,

sont comparés. Les valeurs de courbures affichées pour $c_2 = 0.6m$ ne peuvent clairement pas être envisagées pour alimenter le calcul des lois de commande des véhicules. Pour nos applications, le seuil minimal pour obtenir des mesures de courbure représentatives de la réalité est $c_2 = 1.5m$.

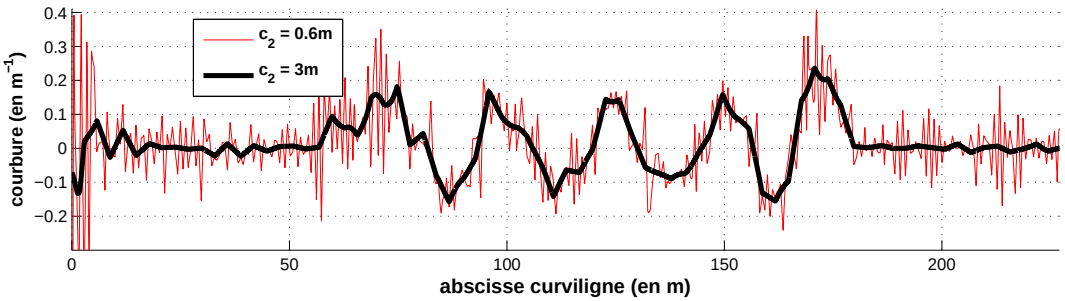


FIG. 2.16 – Effet du bruit sur la courbure en fonction de la valeur de c_2

c_2(m)	d	erreur maximale (en m)	erreur moyenne (en m)	variation de distance moyenne (en m)
1.5	3	0.042	0.006	0.003
1.5	4	0.038	0.008	0.004
1.5	5	0.046	0.007	0.003
3	3	0.068	0.012	0.006
3	4	0.059	0.010	0.005
3	5	0.060	0.014	0.004
5	3	0.146	0.024	0.012
5	4	0.086	0.019	0.008
5	5	0.111	0.025	0.008

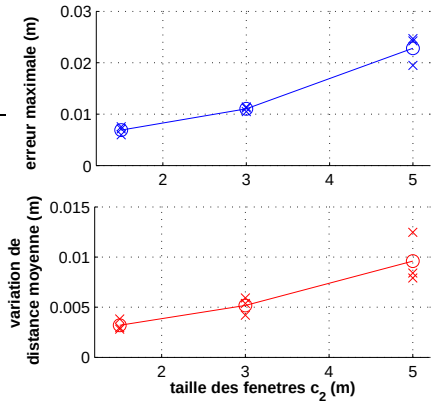


FIG. 2.17 – Influence du paramètre c_2 : présentation des meilleurs résultats selon c_2 et d

$\rightsquigarrow$ Corrélations. On constate que si l'on augmente la valeur de c_2 alors les nombres n_{ac} et n_{ap} de courbes et points actifs nécessaires pour afficher des résultats optimaux sont plus élevés. En effet, si c_2 est élevé, chaque polynôme est susceptible de modéliser une dynamique plus importante (l'intégrale des variations absolues de la courbure est plus importante sur une longue distance). Il est donc profitable d'augmenter les nombres n_{ac} et n_{ap} qui donnent plus de contrôle sur la trajectoire. Comme il est illustré Fig. 2.18, cette valeur de n_{ac} dépend aussi du degré d. Ces valeurs seuil (pour n_{ac} et n_{ap}) seront par la suite mises en valeur, en blanc sur fond bleu.

d	c_2(m)	valeur optimale pour n_{ac}	valeur optimale pour n_{ap}
3	1.5, 3 et 5	5, 5 et 6 resp	5, 5 et 6 resp
4	1.5, 3 et 5	6, 6 et 7 resp	6, 6 et 7 resp
5	1.5, 3 et 5	> 7...	> 7...

FIG. 2.18 – Seuil de qualité pour n_{ac} selon d et c_2

• Le degré d : Le choix de l'intervalle de test pour ce paramètre se base sur les mêmes critères que pour le cas hors-ligne. Pour effectuer le calcul de la courbure sans avoir de saut de continuité, le degré minimal à respecter est $d = 3$. Ensuite, étant donné que les variations de la courbure de la trajectoire de référence sont bornées (ce qui est naturel au vu des contraintes dynamiques sur les véhicules), il n'est pas nécessaire d'étudier de grandes valeurs pour le paramètre et $d \leq 5$ convient.

$\rightsquigarrow$ Impact direct. Le degré d ne semble pas présenter d'influence directe sur la qualité de la trajectoire. En effet, si l'on considère les meilleurs résultats recensés au cours de ces simulations, on constate qu'ils se répartissent entre les trois valeurs de d. Pour s'en convaincre, on peut observer que les meilleures performances selon chaque degré sont très proches tant sur le critère de fidélité que sur les perturbations occasionnées (voir les valeurs en gras Fig. 2.19). On note tout de même que le nombre de configurations présentant de bons résultats est plus important si la valeur de d est petite : ainsi le nombre de tests ayant une erreur moyenne inférieure à 1cm est de 10 pour la valeur $d = 3$ alors qu'il se réduit à 7 puis à 6 pour les valeurs $d = 4$ et $d = 5$ respectivement. Ceci montre que si le choix se porte sur de grandes valeurs pour d, alors les contraintes imposées sur la sélection des autres paramètres seront plus importantes. Des éléments de réponse sont donnés au cours de l'étude qui suit sur les corrélations.

	d	n_{ap}	erreur maximale (en m)	erreur moyenne (en m)	variation de distance moyenne (en m)
	3	3	0.0621	0.0088	0.0025
sélection	3	4	0.0559	0.0087	0.0056
des	3	5	**0.0466**	**0.0060**	**0.0028**
meilleurs	3	6	0.0453	0.0069	0.0041
résultats	3	7	0.0432	0.0062	0.0031
en	4	5	0.0357	0.0077	0.0034
fonction	4	6	**0.0383**	**0.0075**	**0.0038**
des	4	7	0.0387	0.0076	0.0034
valeurs	5	6	0.0483	0.0088	0.0043
de d et	5	7	0.0519	0.0084	0.0039
n_{ap}	5	8	**0.0459**	**0.0071**	**0.0030**

FIG. 2.19 – Influence de d et corrélation avec n_{ap}

$\rightsquigarrow$ Corrélations. Pour chaque valeur prise par d on constate l'existence d'un nombre seuil de points actifs à partir duquel les résultats sont proches des performances optimales, en blanc sur rouge. Ainsi les performances deviennent très satisfaisantes en degré $d = 3$ dès $n_{ap} = 3$ alors que le seuil minimal est de $n_{ap} = 4$ et $n_{ap} = 6$ pour les valeurs respectives $d = 4$ et $d = 5$. Le fait que le seuil soit plus élevé si d est grand paraît assez naturel puisqu'un degré d plus élevé des B-Splines nécessite plus de points de contrôle (exactement $d+1$) pour contrôler entièrement chaque B-Spline. Puisque le nombre n_{ac} de courbes actives est lié au nombre de points actifs n_{ap} ($n_{ap} - d \leq n_{ac} \leq n_{ap}$), il découle logiquement qu'un seuil similaire sur les courbes actives soit requis pour l'obtention de bonnes performances. Ce dernier va être plus élevé pour les plus grandes valeurs du degré d, comme illustré sur la Fig. 2.18.

• Le nombre de fenêtres et de points actifs n_{ac} et n_{ap} : Les deux paramètres sont très corrélés et il convient de les étudier en parallèle.

$\rightsquigarrow$ Impact direct. Pour des valeurs fixées de d et c_2, les erreurs moyennes et maximales montrent clairement que plus grand est le nombre n_{ac} de courbes actives (et par suite le seuil minimal pour n_{ap}), plus précise sera l'optimisation en ligne des B-Splines sous la condition que n_{ap} soit choisi de façon judicieuse (voir corrélations). En effet si cette valeur atteint son maximum (n_{ac} correspond au nombre total de B-Splines matérialisant la trajectoire), alors l'approximation suivant le critère local (2.13) est strictement équivalente à celle selon le critère global (2.11). Pourtant, et fort heureusement, au delà d'un certain seuil, le fait d'augmenter la valeur de n_{ac} n'améliore plus de façon significative la qualité d'approximation. En degré $d = 3$, la Fig. 2.20 illustre assez clairement le comportement de l'algorithme en fonction de n_{ac}. La trajectoire est complètement incohérente quels que soient les paramètres si $n_{ac} < 2$. Une nette amélioration des résultats est ensuite visible jusqu'à la valeur $n_{ac} = 5$ pour ensuite se stabiliser pour les valeurs de n_{ac} supérieures. Ainsi, même si l'on ne considère qu'une petite portion de la trajectoire brute Σ, la qualité de l'approximation en-ligne est à la hauteur de celle obtenue de façon hors-ligne. Cette dernière remarque ne considère bien sûr que le critère de fidélité.

	d	n_{ac}	erreur maximale (en m)	erreur moyenne (en m)	variation de distance moyenne (en m)
	3	1	∞	∞	∞
sélection	3	2	0.1546	0.0343	0.0123
des	3	3	0.0621	0.0088	0.0025
meilleurs	3	4	0.0559	0.0066	0.0056
résultats	3	5	**0.0466**	**0.0060**	**0.0028**
selon n_{ac}	3	6	0.0453	0.0069	0.0041
	3	7	0.0432	0.0062	0.0031

FIG. 2.20 – Mise en évidence d'un seuil caractéristique pour n_{ac} en degré $d = 3$

$\rightsquigarrow$ Corrélations. Comme il a déjà été signifié dans les corrélations du paramètre c_2, le seuil caractéristique pour n_{ac} dépend des valeurs des paramètres c_2 et d, voir Fig. 2.18. Une fois que ce seuil pour n_{ac} a été franchi, alors les meilleures approximations sont obtenues quand $n_{ap} = n_{ac}$ (on suppose bien entendu ici que les paramètres d et c_2 sont fixés). Cela signifie assez logiquement que pour obtenir une configuration optimale, il n'est pas recommandé de modifier des polynômes $Q^i(t)$ qui n'apparaissent pas explicitement dans le critère local (2.13) (cas où $n_{ap} > n_{ac}$). En degré $d = 3$ et $d = 4$ on remarque par exemple sur la Fig. 2.21 que les résultats sur la qualité de la trajectoire se dégradent en même temps que la différence $n_{ap} - n_{ac}$: quand il n'y a pas assez de courbes actives, il est alors souhaitable d'augmenter de degré de liberté sur celles-ci. Des valeurs $n_{ap} > n_{ac}$ sont plus facilement tolérées si le degré d est élevé. On peut aussi remarquer qu'en deça du seuil caractéristique pour n_{ac}, les meilleurs résultats sont souvent obtenus pour des valeurs $n_{ap} > n_{ac}$, voir Fig. 2.21 pour $d = 5$. On verra par la suite que plus le degré est élevé, moins les points de contrôle qui impactent les polynômes n'entrant pas dans l'optimisation locale (2.13) ont d'effet (l'indice de ces points de contrôle est compris entre $n_{ac} + 1$ et n_{ap}).

d	n_{ac}	n_{ap}	erreur maximale (en m)	erreur moyenne (en m)	variation de distance moyenne (en m)
3	5	5	**0.0466**	**0.0060**	**0.0028**
3	5	6	0.0427	0.0094	0.0064
3	5	8	0.5520	0.1304	0.0281
4	6	6	**0.0383**	**0.0075**	**0.0038**
4	6	7	0.0543	0.0085	0.0043
4	6	10	4.2546	0.5018	0.1284
5	7	7	0.0483	0.0094	0.0046
5	7	8	**0.0459**	**0.0071**	**0.0030**
5	7	9	0.0643	0.0129	0.0055

FIG. 2.21 – Influence de n_{ac} et n_{ap} sur l'optimisation en ligne

Maintenant que l'on a étudié l'influence des différentes configurations (choix de d, c_2, n_{ac} et n_{ap}) sur la qualité de la trajectoire, il est plus aisé de définir, à partir de ces variables, un indicateur caractéristique de la qualité de la trajectoire générée en ligne.

2.4.4 Recherche d'un indicateur sur la qualité de la trajectoire

Pour faire aboutir cette recherche, la précédente analyse des paramètres a mis en évidence trois règles :

(i) Augmenter la taille des fenêtres c_2 diminue la qualité de la trajectoire (voir impact direct du paramètre c_2).

(ii) Augmenter la liberté d'action sur les dernières B-Splines améliore la qualité de la trajectoire (voir impact direct des paramètres n_{ac} et n_{ap}, et aussi les corrélations

des paramètres c_2 et d).

(iii) Minimiser les perturbations sur les courbes non actives améliore la qualité de la trajectoire (voir corrélations des paramètres n_{ac} et n_{ap}).

Concernant le point (ii), la liberté d'action dont on dispose sur l'optimisation d'une courbe B-Spline se mesure par l'influence des points qui la contrôlent et sur lesquels on peut agir. En s'appuyant sur le fait que chaque B-Spline est une combinaison linéaire des polynômes de la base (A, B où C) prenant les points de contrôle pour pondération (voir 2.10), on propose la définition suivante :

Définition L'action d'un point de contrôle sur une courbe, notée $a(P^j, Q^i)$ est mesurée par la moyenne sur l'intervalle de définition de la B-Spline ($[0;1]$) du polynôme de la base associé au point, voir l'équation (2.14).

$$\forall i \in \{1,..,n\}, \forall j \in \{i,..,i+d\},$$
$$a(P^j, Q^i) = \int_0^1 Pol_{j-i}(t)dt, \text{ avec } \begin{cases} Pol_{j-i}(t) = A_{j-i}(t) \text{ si } i = 1, \\ Pol_{j-i}(t) = B_{j-i}(t) \text{ si } i \in \{2,..,n-1\}, \\ Pol_{j-i}(t) = C_{j-i}(t) \text{ si } i = n \end{cases} \tag{2.14}$$

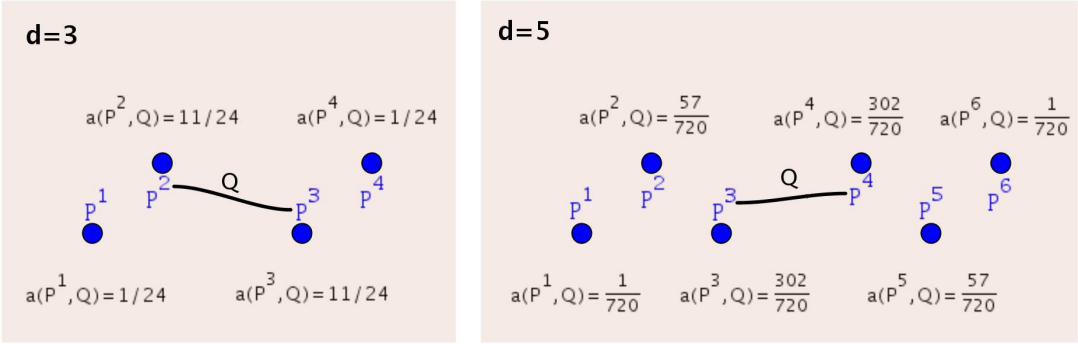


FIG. 2.22 – Coefficients $a(P^j, Q)$ en base B pour les degrés $d = 3$ et $d = 5$

La figure Fig. 2.22 représente ces coefficients pour les valeurs $d = 3$ et $d = 5$ avec la base de polynômes B. Ainsi si les k premiers points d'une courbe d'indice i et de degré d sont fixés, alors la liberté d'action sur le modelage de cette courbe va être mesurée selon l'équation (2.15) :

$$a(Q^i) = \sum_{j=i+k}^{i+d} a(P^j, Q^i) \tag{2.15}$$

Remarques :

- La somme des coefficients $\sum_{j=i}^{i+d} a(P^j, Q^i)$ d'une courbe Q^i, vaut 1.
- La liberté d'action sur une B-Spline est évaluée au regard des contraintes qui

proviennent des points de contrôle inactifs. Cependant, ce calcul n'est pas parfaitement réaliste dans notre cas, car une contrainte majeure n'est pas prise en compte. En effet, puisqu'un point de contrôle est effectif sur un ensemble de $d+1$ courbes, son action va se disperser sur le sous-ensemble de ces courbes qui sont actives. Il faudrait alors intégrer la contrainte issue de l'ajustement des courbes voisines.

D'après la deuxième remarque, chaque coefficient $a(P^j, Q^i)$ est surestimé. Pour réajuster ces derniers, il suffit simplement de connaître le rapport entre 1) l'action du $j^{ème}$ point sur la $i^{ème}$ courbe et 2) son action sur l'ensemble des courbes actives sur lesquelles il est effectif. En conséquence, un indicateur non biaisé $\hat{a}(P^j, Q^i)$ peut être calculé selon l'équation (2.16). Une représentation visuelle du calcul de cette estimation est proposée par l'illustration Fig. 2.23. Pour cet exemple $\hat{a}(P^j, Q^i)$ est calculé avec la configuration : $d = 3$, $n_{ac} = 3$ et $n_{ap} = 5$.

$$\forall i \in \{1, .., n\}, \forall j \in \{i, .., i+d\}, \ \hat{a}(P^j, Q^i) = a(P^j, Q^i) * \frac{a(P^j, Q^i)}{\sum\limits_{\substack{k \leq n \\ j-d \leq k \leq j}} a(P^j, Q^k)} \tag{2.16}$$

Comme précédemment, si les k premiers points d'une courbe d'indice i et de degré d

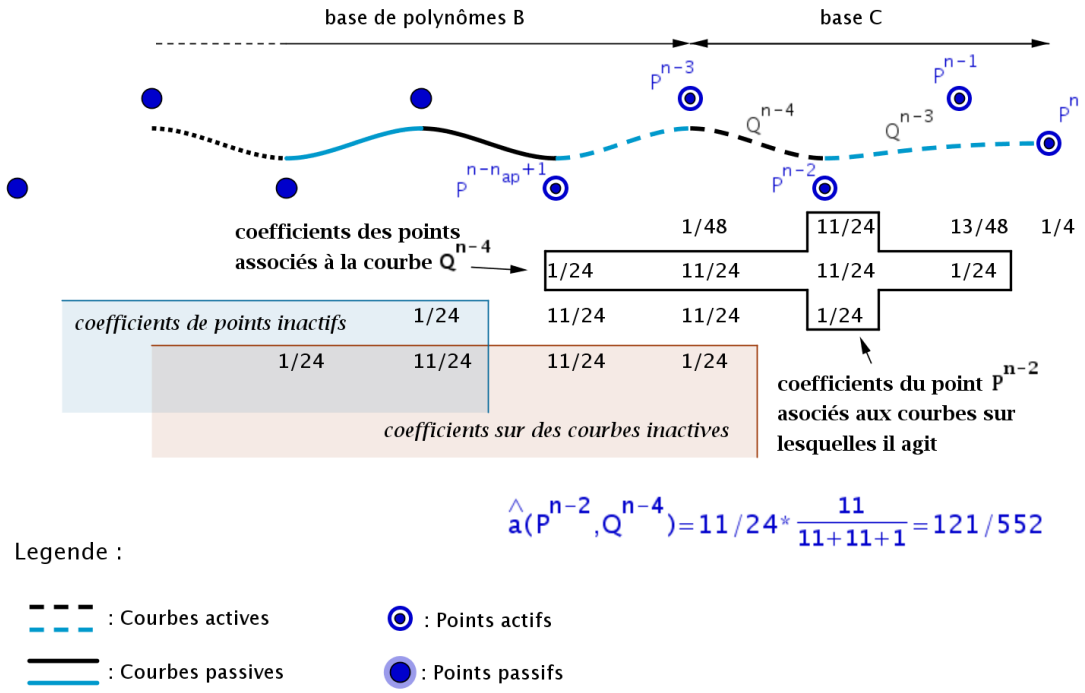


FIG. 2.23 – Exemple de calcul de coefficient $\hat{a}(P^j, Q^i)$ avec $d = 3$, $n_{ac} = 3$ et $n_{ap} = 5$

sont fixés, alors la liberté d'action (sans biais) sur le modelage de cette courbe va être mesurée selon l'équation (2.17) :

$$\hat{a}(Q^i) = \sum_{j=i+k}^{i+d} \hat{a}(P^j, Q^i) \tag{2.17}$$

Remarque : Pour chaque courbe Q^i, la somme de ces coefficients $\sum_{j=i}^{i+d} \hat{a}(P^j, Q^i)$ vaut 1 uniquement si $\forall i \leq j \leq i+d,\ \hat{a}(P^j, Q^i) = a(P^j, Q^i)$ et donc si le nombre de courbes actives $n_{ac} = 1$.

Finalement, l'indicateur qui va nous permettre de caractériser la liberté d'action sur les k premières B-Splines actives, comme préconisé par le point (ii), est formulé selon l'équation (2.18).

$$\forall k \leq n_{ac},\ action_coef(k) = \sum_{i=n-k+1}^{n} \hat{a}(Q^i) \tag{2.18}$$

Concernant le point (iii), les perturbations occasionnées sur les courbes non actives se mesurent par l'influence (néfaste) de l'optimisation des points actifs sur les courbes inactives (qui ne sont pas explicitées dans le critère local (2.13)). Les courbes modifiées au cours de l'optimisation sont les n_{ap} dernières, ainsi les courbes inactives perturbées sont indexées de $n - n_{ap} + 1$ à $n - n_{ac}$ et l'indicateur caractéristique des perturbations peut être formulé selon (2.19).

$$disturbance_coef = \sum_{i=n-n_{ap}+1}^{n-n_{ac}} a(Q^i) \tag{2.19}$$

En accord avec les points (i), (ii) et (iii), la qualité de la trajectoire est bonne si :

(i) - c_2 est petit

(ii) - $\sum_{i=n-n_{ap}+1}^{n-n_{ac}} a(Q^i)$ est petit

(iii) - $\sum_{i=n-n_{ac}+1}^{n} \hat{a}(Q^i)$ est grand

A partir de ces trois tendances, il serait intéressant d'établir une relation liant la fidélité de la représentation, à travers la variable d'erreur moyenne, avec un indicateur prenant en compte ces trois variables. On cherche une relation linéaire du type : $erreur\ moyenne = \alpha.qual$. Une intuition concernant la formule d'un indicateur de

qualité de la trajectoire est donnée par l'équation (2.20).

$$qual = c_2 * \frac{\sum_{i=n-n_{ap}+1}^{n-n_{ac}} a(Q^i)}{\sum_{i=n-n_{ac}+1}^{n} \hat{a}(Q^i)} \tag{2.20}$$

Afin d'obtenir une corrélation linéaire entre l'indicateur et les résultats de recollement obtenus lors des simulations, quelques retouches doivent être effectuées (équation (2.21)) comme l'exposant sur les variables retenues dans la formule. Le nombre "1" a été ajouté au numérateur pour permettre de ne pas court-circuiter l'indicateur sur la liberté d'action (qui se trouve au dénominateur) lorsque l'indicateur de perturbation est nul. Enfin, le coefficient d'action est calculé à partir des trois dernières B-Splines uniquement.

$$qual = \alpha.(c_2)^{0.89} * \frac{\left(1 + \sum_{i=n-n_{ap}+1}^{n-n_{ac}} a(Q^i)\right)^{2.242}}{\left(\sum_{i=n-\underline{\mathbf{3}}+1}^{n} \hat{a}(Q^i)\right)^{2.124}} \tag{2.21}$$

La valeur de α est finalement retrouvée par régression linéaire en sélectionnant les meilleurs résultats, à savoir ceux dont la valeur moyenne de recollement à la trajectoire n'excèdent pas 10cm. Les autres tests présentent en effet une divergence trop forte et incontrôlable pour pouvoir déterminer une estimation sur les mesures. Ainsi 151 tests peuvent être exploités sur les 226 initialement simulés. Les résultats de la régression sont présentés Fig. 2.24. On constate que seuls une dizaine d'entre eux s'éloignent de la droite d'équation $y = x$ et le coefficient de corrélation, d'une valeur supérieure à 0.85 atteste de la pertinence de l'indicateur.

Cet indicateur constitue donc un outil adapté pour prédire les performances de l'algorithme de génération de la trajectoire en fonction des ses paramètres.

2.5 Conclusion

Dans ce chapitre le problème de la génération d'une trajectoire, destinée au guidage d'un convoi de véhicules urbains, a été adressé. La modélisation retenue s'appuie sur l'utilisation de courbes B-Spline qui offrent une représentation simple de la trajectoire puisqu'elle est caractérisée par des points de contrôle. Ces courbes sont de plus $\mathcal{C}^\infty$ et des contraintes de continuité au niveau des jonctions sont naturellement intégrées. Aussi en fixant le degré des courbes supérieur à trois, cette représentation permet de calculer les variables nécessaires aux lois de commande qui ont été définies au chapitre 1 pour opérer le suivi de trajectoire et la régulation longitudinale.

La procédure de création de la trajectoire considère deux modes de déplacement. Le mode "tout automatique" où un convoi se déplace le long d'une trajectoire définie à

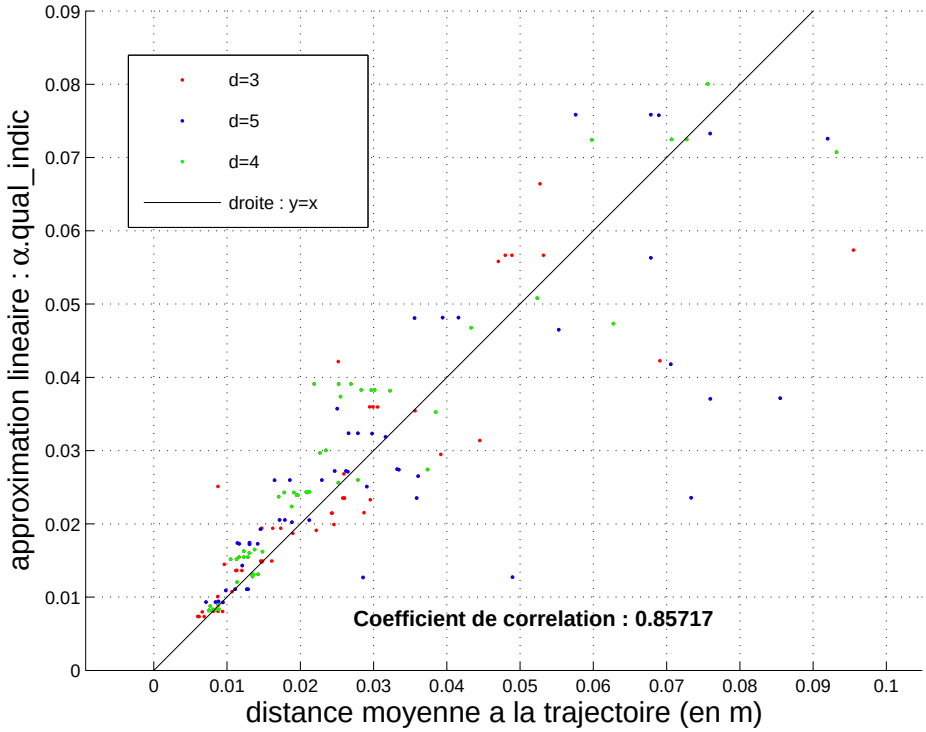


FIG. 2.24 – Comparaison entre l'indicateur de qualité et les résultats de fidélité obtenus

partir d'informations préalablement enregistrées, alors que le mode "manuel" adresse le guidage de la flotte à partir d'un véhicule leader conduit manuellement et construisant la trajectoire au fur et à mesure de son évolution.

Le deuxième mode est particulièrement difficile à mettre en œuvre, car il demande de définir une procédure d'extension de la trajectoire en temps réel qui permette à chaque véhicule de suivre avec précision la trace du leader et ceci bien que les de données de localisation de celui-ci soient bruitées. Pour assurer la cohérence des variables de commande, il faut aussi que la trajectoire soit stable, c.a.d que la création de nouvelles portions de trajectoire ne perturbe pas celles qui ont été précédemment construites. Ainsi, pour répondre à ces exigences, la génération est traitée de façon itérative et abordée comme un problème d'optimisation. Un critère de minimisation local est défini et permet, lorsqu'une nouvelle donnée de localisation est disponible, d'approximer les points de contrôle qui modélisent l'extrémité finale de la trajectoire. En conséquence, la trajectoire n'est modifiée que sur une courte distance, le temps de calcul est borné et la bande passante occupée pour transmettre les informations de la trajectoire aux autres véhicules n'est pas saturée. Des résultats de simulation, obtenus à partir de données de localisation réelles (mesurées par un récepteur RTK-GPS) ont prouvé l'efficacité de la

méthode et ont permis de mettre en évidence l'influence des paramètres qui entrent en jeu. La trajectoire construite au cours de chaque simulation est évaluée selon sa fidélité par rapport au chemin emprunté et aux perturbations engendrées à chaque itération d'optimisation. En choisissant correctement les paramètres, la stratégie de génération affiche des résultats aussi satisfaisants que lorsque la trajectoire est construite avec un critère de minimisation global, tout en assurant de faibles perturbations.

Chapitre 3

Localisation par vision monoculaire pour la conduite en convoi

3.1 Introduction

Pour réaliser l'asservissement d'un convoi de véhicules, on a vu au chapitre 1 qu'il était préférable de s'appuyer sur une stratégie globale, c'est-à-dire que chaque véhicule soit contrôlé à partir d'informations partagées par l'ensemble de la flotte. On évite ainsi de propager des erreurs d'asservissement, introduites par le bruit des capteurs et/ou le délai des actionneurs, et s'accumulant du véhicule leader jusqu'au dernier de la chaîne. Une telle stratégie, reposant sur le partage d'information entre les véhicules et s'appuyant sur des techniques de contrôle non-linéaires a été proposée dans [Bom 05] et a fait l'objet d'une présentation détaillée dans la section 1.3.1 du chapitre 1. Cette approche a été éprouvée expérimentalement grâce à l'utilisation de récepteurs RTK-GPS pour la localisation des véhicules. Cependant ces capteurs ne sont pas fiables pour des applications urbaines car les signaux en provenance des satellites peuvent être masqués par les hautes structures urbaines (immeubles, bâtiments). L'utilisation de caméras pour localiser les véhicules paraît plus attractive pour plusieurs raisons : d'une part le faible prix d'une caméra la rend beaucoup plus abordable qu'un récepteur RTK-GPS et d'autre part les structures urbaines s'avèrent cette fois être un avantage, puisqu'elles sont très riches pour le traitement d'image.

Dans ce contexte, les auteurs de [Royer 05] ont développé une stratégie de localisation dans un repère absolu à partir d'une séquence d'images obtenue par vision monoculaire. Cette séquence est fournie par une caméra montée sur le véhicule alors que celui-ci se déplace et permet une reconstruction 3D de l'environnement, on parle d'approche de type "Structure From Motion" ou plus simplement SFM. Une description des étapes menant à ce résultat se trouve dans l'annexe B. On y souligne notamment que, de par les principes de géométrie projective employés pour extraire l'information retranscrite par la caméra, les données de localisation sont fournies à un facteur d'échelle près par rapport au monde réel. Hélas ce dernier n'est pas parfaitement constant : il évolue en fonction de la position du véhicule dans l'environnement. En conséquence, la localisation

absolue provenant de l'algorithme de vision est exprimée dans un monde vision virtuel, comportant des distorsions par rapport au monde réel, nommé monde métrique dans la suite. L'estimation des distances est alors affectée (voir section 1.3.2 du chapitre 1). Et donc, si l'asservissement longitudinal est alimenté directement avec ces données, les performances dans le monde vision seront bien sûr satisfaisantes, mais ce ne sera pas le cas dans le monde métrique, alors que c'est évidemment dans le monde métrique que l'on souhaite que les distances inter-véhicule soient correctement régulées.

Dans ce chapitre, on adresse ce problème, c'est-à-dire on souhaite mettre en œuvre l'approche de commande globale développée dans [Bom 05] avec la localisation par vision monoculaire issue de [Royer 05] tout en maintenant le niveau de performance sur ce qu'il était avec une localisation par RTK-GPS. La suite du chapitre s'articule ainsi : un état de l'art est tout d'abord présenté sur les différentes façons de calculer des distances métriques à partir d'un dispositif visuel, notamment par vision monoculaire. La section suivante permet de cibler la problématique avec plus de précision. Une analyse plus fine des distorsions entre les mondes vision virtuel et métrique permet alors de proposer une une stratégie de correction hors-ligne des distorsions. Deux sections sont ensuite consacrées à l'élaboration de stratégies de correction en ligne, la première basée sur l'utilisation de données télémétriques et la deuxième sur des données odométriques. Des résultats de simulation sont finalement présentés afin de comparer les différentes méthodes de correction et d'examiner la sensibilité de chacune en fonction de leur principaux paramètres.

3.2 État de l'art : localisation d'un véhicule par vision, estimation du facteur d'échelle

Se localiser est une tâche incontournable qu'un véhicule doit être capable de réaliser pour accomplir un quelconque déplacement de manière autonome. En fonction des contraintes environnementales et des exigences formulées en terme de commande, cette phase de localisation a été abordée sous divers angles dans la littérature.

3.2.1 Localisation partielle

On peut tout d'abord remarquer que la commande d'un véhicule ne nécessite pas toujours de se localiser au sens strict du terme, c'est-à-dire de fournir à chaque instant la pose complète du véhicule. En vision, il est également possible de suivre un chemin représenté par une séquence d'images clefs, on parle de chemin sensoriel. Au cours de son déplacement la pose du robot n'est pas calculée : dans cette approche, des primitives extraites sur la vue courante et l'image clef la plus proche sont utilisées pour définir le mouvement du robot. Les travaux présentés dans [Argyros 05] et [Goedemé 05] par exemple visent à stabiliser un robot mobile autour d'une succession de poses caractérisées par les images clefs. Le guidage se fait à partir d'une heuristique simple résultant de la mise en correspondance d'amers entre deux images panoramiques. Cependant, pour

notre application on ne souhaite pas que le véhicule s'arrête à chaque image clef.

Le parcours autonome d'un chemin sensoriel peut également être traité sans point d'arrêt en définissant une tâche de suivi de chemin ; la vitesse linéaire est alors généralement constante. Dans [Chen 06] un guidage rudimentaire s'effectue en comparant deux vecteurs calculés dans l'image courante et l'image clef : la commande en braquage est choisie parmi les valeurs suivantes $\{-w, 0, w\}$, où w est fixe. Des expérimentations sont menées avec un robot Pioneer AT3 équipé d'une webcam embarquée. Dans un environnement intérieur, les auteurs obtiennent une précision de l'ordre de $35cm$ le long d'un chemin de $10m$ en forme de U, alors qu'en extérieur, le robot suit une trajectoire de $140m$ de long en boucle sur un parking avec une erreur maximale de $2.2m$. Dans [Rivlin 03] et [Diosi 07] la vitesse angulaire est choisie proportionnelle à une erreur dépendant des positions des points dans l'image courante et désirée. Des expérimentations sont menées avec un CyCab le long d'une trajectoire de $740m$ constituée de lignes droites et de deux virages. Le véhicule roule à une vitesse comprise entre $0.3m.s^{-1}$ et $0.8m.s^{-1}$ selon la courbure. Finalement dans [Courbon 09], le chemin de référence pour atteindre l'image intermédiaire à rejoindre est défini comme la droite qui pointait dans la direction du robot lors de l'enregistrement des images clefs. La loi de commande latérale basée sur les systèmes chaînés, rappelée dans la section 1.3.1 du chapitre 1, permet alors de suivre ce chemin à partir des données d'écart angulaire à la trajectoire et de valeurs de courbure et d'écart latéral obtenues à un facteur d'échelle près. Les auteurs ont ainsi pu faire naviguer un véhicule de type CyCab sur $750m$ dans un environnement urbain avec un écart latéral moyen de $25cm$ par rapport au chemin réalisé lors de l'apprentissage, avec une médiane s'élevant à $10cm$ et un écart type de $34cm$ (valeurs estimées à l'aide d'un récepteur RTK-GPS).

Le principal avantage des méthodes qui utilisent un chemin sensoriel est qu'une connaissance détaillée de l'environnement n'est pas exigée. La perception de l'environnement n'étant représentée que par un jeu de primitives extraites des images clefs, la localisation du véhicule se traduit simplement par la mise en correspondance des primitives avec l'image courante, limitant ainsi le temps de calcul sur le traitement des images. Si l'on se concentre sur la navigation d'un unique robot, les inconvénients communs à ces stratégies sont d'une part de ne pas pouvoir s'écarter de la trajectoire apprise et d'autre part de ne pas maîtriser la trajectoire 3D : la régulation d'une information dans l'image ne conduit pas forcément à une trajectoire 3D naturelle.

Si l'on considère maintenant une application en convoi, un inconvénient supplémentaire, qui se révèle bloquant, est que la localisation du véhicule n'est pas fournie dans un monde métrique, mais dans un monde virtuel. Dans un tel référentiel, une stratégie de commande globale dont l'objectif est d'assurer des distances métriques entre les véhicules ne peut clairement pas être mise en œuvre. C'est pourquoi dans la suite de cet état de l'art, nous allons désormais nous intéresser aux stratégies qui permettent d'évaluer une position du véhicule dans un référentiel se rapprochant d'un référentiel métrique.

3.2.2 Localisation absolue

Dans cette section on recense les stratégies de vision qui permettent d'aboutir à une carte 3D de l'environnement et de localiser avec une contrainte de temps réel le robot dans cet environnement. L'attention est tout particulièrement portée sur la qualité de la carte 3D en vue de notre application pour le convoi de véhicules. Par contre on ne s'attachera pas dans la suite à la phase de détection et d'appariement des amers qui est à la base des algorithmes de vision.

Le problème d'établir une carte 3D a été abordé par plusieurs communautés et il convient au préalable de définir le vocabulaire et les notions propres à chacune. Dans la communauté robotique, de nombreux travaux sont regroupés sous le terme SLAM (de l'anglais Simultaneous Localization And Mapping) que l'on peut traduire par cartographie et localisation simultanée. Le principe consiste, en partant d'une position initiale sans connaissance de l'environnement, à construire une carte à partir des observations qui sont faites et aussitôt qu'un début de carte est disponible à s'en servir pour se localiser. Au fur et à mesure des observations, la carte est enrichie de nouveaux amers et la position des anciens amers est affinée en tenant compte des nouvelles informations. Le principe est très général et peut être utilisé avec de nombreux capteurs (télémètre, radar, caméra,...). L'expression "Structure From Motion", plus courante dans la communauté de la vision par ordinateur, désigne quasiment le même problème que le SLAM. La seule différence entre les deux problèmes est que le SLAM doit forcément être fait de manière incrémentale alors que pour le problème de la reconstruction 3D, on peut parfois disposer dès le départ de l'ensemble des images de la séquence. Certains travaux de reconstruction 3D tendent à se rapprocher du SLAM en procédant de manière incrémentale et en temps réel. Ils donnent alors lieu à des algorithmes d'odométrie visuelle.

Dans la suite on va tout d'abord s'intéresser aux stratégies basées sur la vision monoculaire seule, c'est-à-dire sans information supplémentaire concernant l'environnement, l'état du robot ou son déplacement au cours de l'acquisition de la séquence vidéo. On verra ensuite comment de telles informations peuvent être exploitées pour raffiner la carte 3D.

3.2.2.1 L'approche SLAM basée sur la vision monoculaire

Les premières publications sur le SLAM datent de la deuxième moitié des années 1980, avec les travaux de Smith et Cheesman [Smith 87] et de Moutarlier et Chatila [Moutarlier 89] qui s'appuyent sur des données télémétriques. L'algorithme le plus fréquemment utilisé pour traiter ce problème est basé sur l'utilisation d'un filtre de Kalman étendu [Kalman 60], on parle alors d'EKF-SLAM. Dans cette section on va tout d'abord introduire cette méthode de façon générale. Les spécificités de l'algorithme lorsqu'il est adapté pour la vision monoculaire seront ensuite présentées pour pouvoir aborder la discussion sur cette approche.

Principes de l'EKF-SLAM Dans cette approche, l'ensemble des paramètres à évaluer est représenté par un vecteur d'état $\mathbf{X}$. Ce dernier contient à la fois la pose courante

du robot $\mathbf{X}_V$ et la position de l'ensemble des amers présents dans la carte $\mathbf{X}_M$:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_V \\ \mathbf{X}_M \end{bmatrix} \tag{3.1}$$

Pour employer un filtre EKF, le vecteur $\mathbf{X}$ est modélisé par une variable aléatoire suivant une loi gaussienne. Une matrice de covariance est alors associée au vecteur d'état et permet de quantifier la confiance que l'on a sur la position du robot et des amers :

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_V & \mathbf{P}_{VM} \\ {\mathbf{P}_{VM}}^T & \mathbf{P}_M \end{bmatrix} \tag{3.2}$$

$\mathbf{P}_V$ représente la matrice de covariance associée au vecteur $\mathbf{X}_V$ et $\mathbf{P}_M$ la matrice de covariance associée à $\mathbf{X}_M$. Les éléments de la matrice $\mathbf{P}_{VM}$ représentent les corrélations entre les éléments de la carte et les états du véhicule. Cette modélisation de la carte et du véhicule est dite stochastique et a été introduite initialement dans l'article [Smith 87].

A chaque itération de l'algorithme, trois étapes de calcul sont réalisées.

• Tout d'abord, avant l'apparition d'une nouvelle information extéroceptive, une phase de prédiction est réalisée en s'appuyant sur un modèle non-linéaire $\mathbf{f}$ de l'évolution de l'état $\mathbf{X}$. L'état prédit à l'instant $k+1$ sur la base des informations disponibles à l'instant k, noté $\hat{\mathbf{X}}(k+1|k)$, est donné par l'équation (3.3), avec $\hat{\mathbf{X}}(k|k)$ et $\mathbf{U}(k)$ l'estimation de l'état et de la commande à l'instant k.

$$\hat{\mathbf{X}}(k+1|k) = \mathbf{f}(\hat{\mathbf{X}}(k|k), \mathbf{U}(k)) \tag{3.3}$$

La matrice de covariance associée au vecteur d'état $\hat{\mathbf{X}}(k+1|k)$ est aussi calculée :

$$\mathbf{P}(k+1|k) = \mathbf{F}_X.\mathbf{P}(k|k).(\mathbf{F}_X)^T + \mathbf{F}_U.\mathbf{Q}_U.(\mathbf{F}_U)^T \tag{3.4}$$

où

- $\mathbf{Q}_U$ est la matrice de covariance associée au bruit sur la commande,
- $\mathbf{F}_X$ est la matrice jacobienne de la fonction $\mathbf{f}$ par rapport à l'état,
- et $\mathbf{F}_U$ est la matrice jacobienne de la fonction $\mathbf{f}$ par rapport aux commandes.

• Dans un second temps, une fois que la prédiction $\hat{\mathbf{X}}(k+1|k)$ a été établie, il est possible de projeter la position des amers de la carte prédite dans le repère associé à la position prédite du véhicule. On utilise pour cela la fonction d'observation $\mathbf{h}$:

$$\hat{\mathbf{Z}}(k+1|k) = \mathbf{h}(\hat{\mathbf{X}}(k+1|k)) \tag{3.5}$$

Le but est de comparer, dans un repère commun, la position des amers de la carte estimée avec les mesures recueillies. La matrice de covariance associée à cette observation est donnée par :

$$\mathbf{P}_Z(k+1|k) = \mathbf{H}_X.\mathbf{P}(k+1|k).{\mathbf{H}_X}^T \tag{3.6}$$

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

• Ensuite, lorsqu'une nouvelle acquisition de données est disponible, certains amers présents dans les données sont répertoriés dans la carte. Ces derniers sont alors mis

en correspondance alors que de nouveaux amers permettent d'enrichir la carte et de cartographier des zones qui n'avaient pas été explorées. Après la mise en correspondance des mesures et des éléments de la carte, une innovation $\mathcal{V}$ est calculée. Elle correspond à la différence entre les positions estimées $\hat{\mathbf{Z}}(k+1|k)$ et mesurées $\mathbf{Z}_{mesure}(k+1)$ d'une même entité de la carte :

$$\mathcal{V}(k+1) = \mathbf{Z}_{mesure}(k+1) - \hat{\mathbf{Z}}(k+1|k) \tag{3.7}$$

L'innovation est associée à une matrice de covariance $\mathbf{S}$:

$$\mathbf{S} = \mathbf{P}_Z(k+1|k) + \mathbf{R} \tag{3.8}$$

où $\mathbf{R}$ est la matrice de covariance associée aux mesures extéroceptives.

Au final, une correction de l'estimation de l'état est réalisée à partir de l'innovation $\mathcal{V}$. Cette correction est pondérée par un gain $\mathbf{W}$ permettant de prendre en compte les incertitudes sur les variables d'état et sur les mesures. On obtient alors l'estimation de l'état à l'instant $k+1$, notée $\hat{\mathbf{X}}(k+1|k+1)$, connaissant les informations à l'instant $k+1$:

$$\hat{\mathbf{X}}(k+1|k+1) = \hat{\mathbf{X}}(k+1|k) + \mathbf{W}.\mathcal{V}(k+1) \tag{3.9}$$

La matrice de covariance associée à l'état est aussi mise à jour :

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k) - \mathbf{W}.\mathbf{S}.\mathbf{W}^T \tag{3.10}$$

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

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

Comme le montrent les équations (3.9) à (3.11), chaque nouvelle observation permet d'une part de localiser le robot et d'autre part de mettre à jour la position des amers pour tenir compte de l'observation. En général chaque nouvelle observation permet de réduire l'incertitude sur les amers.

Spécificités du SLAM par vision monoculaire La grande majorité des solutions existantes utilisent des capteurs qui observent l'environnement en fournissant à chaque instant directions et distances aux amers perçus. Lors de l'utilisation de caméras vidéo, ces distances ne sont plus disponibles. L'une des spécificités de la vision dans l'algorithme classique de SLAM, c'est le fait qu'un seule observation d'un amer ne suffit pas à connaître sa position 3D. Il faut alors attendre d'avoir au moins deux observations d'un amer depuis deux positions différentes pour ajouter ce dernier à la carte, la contrainte étant que sa variance soit suffisamment petite pour que les jacobiennes utilisées dans l'algorithme EKF soient localement représentatives des fonctions réelles.

Davison a été le premier à mettre en œuvre une stratégie de SLAM par vision monoculaire [Davison 03]. Lorsqu'un nouvel amer est détecté, celui-ci n'est pas directement intégré à la carte, mais il suit une phase d'initialisation qui s'achève lorsque l'estimation sur sa localisation (c.a.d. la profondeur de l'amer) est suffisamment précise. La

procédure d'initialisation s'appuie sur une méthode probabiliste et fonctionne sous l'hypothèse que les amers détectés se trouvent à proximité (c'est-à-dire dans un intervalle de $0.5m$ à $5m$). Plus précisément un filtre particulaire, totalement indépendant du reste de la carte, est mis en place lorsqu'un nouvel amer est détecté et chaque particule matérialise une position possible de l'amer sur l'intervalle considéré le long du rayon optique. Les hypothèses sur chaque particule sont testées sur les images successives jusqu'à ce qu'une probabilité gaussienne puisse être finalement associée à la profondeur. A ce moment, l'amer est incorporé au vecteur d'état de l'EKF. La limitation de cette approche réside bien sûr dans l'hypothèse qui est émise quant à la proximité des points d'intérêt, ce qui limite la reconstruction 3D à des environnements en intérieur.

Plus récemment, il a été montré [Montiel 06] qu'il était possible d'avoir une initialisation gaussienne des amers à partir d'une seule observation. La méthode proposée consiste à paramétriser les amers de la carte en fonction de l'inverse de leur profondeur. Les nouveaux amers sont immédiatement intégrés dans la carte avec une profondeur supposée infinie lors de l'insertion puis raffinée ensuite. De surcroit cette approche permet de prendre en compte des points d'intérêt distants et donc de fonctionner en extérieur.

La principale limite de ces algorithmes est l'accroissement du temps de calcul de la mise à jour du filtre de Kalman lorsque la taille de la carte augmente. La complexité de la mise à jour est en $O(N^2)$, où N est la taille du vecteur d'état. Cela signifie que le SLAM temps réel avec l'approche décrite plus haut n'est possible que dans un environnement de taille réduite où le nombre d'amers utilisés est limité. Pour aller au delà, il est possible d'utiliser des sous-cartes. On peut se contenter de mettre à jour une carte locale de l'environnement immédiat du robot et changer de sous carte lorsque le robot en sort. Il faut alors gérer la cohérence entre les sous cartes comme le proposent Newman et Leonard [Newman 03].

3.2.2.2 L'approche "Structure From Motion" basée sur la vision monoculaire

Lorsqu'on dispose d'un ensemble de points homologues donnés dans une séquence d'images, la détermination de la localisation des caméras et la détermination de la position 3D des points observés sont deux problèmes intimement liés sur lesquels la communauté de la vision par ordinateur a porté une grande attention. En effet, dès que le positionnement relatif des caméras est connu, il devient possible de calculer la position 3D des points par simple triangulation (voir annexe A). De même, si l'ensemble des points 3D est connu, alors le calcul de la pose de chaque caméra est un problème simple. Chacun de ces problèmes peut être traité par de multiples méthodes et variantes en s'appuyant sur les principes de la géométrie projective et épipolaire.

Remarque : Les stratégies de reconstruction basées sur ces principes utilisent des algorithmes qui apportent des informations à un facteur d'échelle près. Les principaux algorithmes sont décrits dans l'annexe A. Ainsi, dans les exemples qui suivent, la reconstruction 3D obtenue est fournie elle aussi à un facteur d'échelle près, qui peut comme pour les stratégies de SLAM par vision monoculaire être identifié par exemple

grâce à la connaissance d'un objet particulier reconnu dans l'image.

D'une séquence d'images à un modèle tridimensionnel Une solution élégante a par exemple été proposée dans [Nistér 03a]. Les auteurs font partie des premiers à avoir mis en œuvre en temps réel un algorithme permettant d'estimer le mouvement d'une caméra à partir des seules données visuelles, sans aucune connaissance a priori de la scène ou du mouvement. La procédure de reconstruction fonctionne selon les étapes suivantes :

1. Les amers sont d'abord suivis sur un certain nombre d'images. Puis la pose relative est estimée à un facteur d'échelle près entre trois d'entre elles grâce à l'utilisation de l'algorithme des 5 points [Nistér 03a] selon le processus de RANSAC [Nistér 03b] (voir l'annexe A pour plus de détails sur ces deux algorithmes).
2. Les amers observés sont ensuite triangulés afin d'obtenir leur position 3D à un facteur d'échelle près. La première et la dernière image sur lesquelles ils étaient présents ainsi que les images permettant une triangulation optimale selon l'erreur directionnelle sont utilisées à cet effet. Parmi les points d'intérêt, ceux qui étaient présents dans la reconstruction à l'itération précédente sont utilisés pour estimer le facteur d'échelle qui lie la pose courante de la caméra avec sa trajectoire aux instants précédents. Cette étape va alors permettre d'intégrer à la reconstruction la position 3D des amers qui ont pu être correctement triangulés et qui n'appartiennent pas encore à la carte.
3. La pose de la caméra n'est intégrée qu'ultérieurement : après suivi des amers sur quelques images, la pose de la caméra est obtenue à partir des points 3D de la reconstruction en utilisant l'algorithme des 3 points [Haralick 94] (voir annexe A) selon le processus de RANSAC.
4. Les points 3D sont alors re-triangulés en utilisant la première et la dernière image sur lesquelles ils étaient présents. La procédure est répétée un certain nombre de fois à partir de l'étape 3 afin de raffiner l'estimation de la pose caméra et des positions 3D.
5. La procédure est répétée un certain nombre de fois à partir de l'étape 1.
6. Afin de stopper la propagation d'erreur, un "firewall" est introduit et la procédure est répétée à partir de l'étape 1. Cette étape est primordiale si l'on considère le système dans sa globalité. En effet si la pose d'une caméra est mal estimée, alors des points 3D seront mal positionnés. Cette erreur affectera aussi l'estimation des poses suivantes et par suite le système entier sera altéré sans jamais pouvoir recouvrir ses erreurs. La solution proposée est donc de mettre un firewall après lequel les poses relatives sont estimées comme si le système avait été remis à zéro. L'état du système avant le firewall affecte alors uniquement le choix du système de coordonnées des futures poses. Ce concept protège donc le système de grosses erreurs. Cependant les auteurs ont aussi remarqué que de petites erreurs de construction avaient un effet boule de neige qui est aussi évité de cette manière.

Cependant malgré les multiples raffinements ou l'introduction de firewalls, on remarque que le calcul du mouvement de la caméra obtenu par les méthodes décrites précédemment présente des dérives importantes. D'ailleurs, les auteurs de [Nistér 03a] se sont tournés par la suite vers une stratégie en vision stéréo [Nistér 06], qu'ils considèrent plus fiable et plus commode pour reconstruire avec précision un environnement 3D et estimer la pose du dispositif. La stratégie et les résultats associés seront décrits et discutés dans la section suivante. D'autres stratégies plus récentes en vision monoculaire ont permis d'obtenir de meilleurs résultats en choisissant de s'appuyer sur un algorithme d'ajustement de faisceaux [Royer 05] et [Mouragnon 06].

Raffinement par ajustement de faisceaux Il s'agit d'un processus de minimisation basé sur l'algorithme de Levenberg-Marquardt. Il est utilisé pour affiner une reconstruction 3D initiale fournie par les algorithmes décrits ci-dessus et permet de limiter grandement les accumulations d'erreur. L'optimisation par ajustement de faisceaux concerne toutes les variables de la reconstruction 3D, c.a.d. à la fois les poses des caméras et les coordonnées 3D des points d'intérêt. La fonction de coût est généralement la somme sur toutes les images des carrés des erreurs de reprojection de tous les points visibles et correctement identifiés sur chaque image. Plus de détails sur l'ajustement de faisceaux sont donnés dans l'annexe B.

L'algorithme basé sur un ajustement de faisceaux, développé dans les travaux d'Eric Royer [Royer 05] [Royer 06] (et utilisé dans le cadre de mes travaux de thèse), permet d'améliorer significativement la reconstruction hors-ligne des scènes 3D à partir des seules données visuelles, sans aucune autre connaissance a priori de la scène ou du mouvement. Une solution initiale au problème de reconstruction 3D est tout d'abord obtenue par un algorithme de calcul de pose incrémental, décrit à l'annexe A, s'appuyant sur les principes de la géométrie projective et épipolaire. Ensuite, l'ajustement de faisceaux n'est pas utilisé une seule fois sur la totalité de la séquence, mais est utilisé tout au long de la reconstruction afin d'éviter que l'accumulation des erreurs n'éloigne trop la solution initiale de la vraie solution et empêche la convergence de l'ajustement de faisceaux. Comme il serait trop coûteux de faire un ajustement après chaque nouvelle image, une solution plus rapide, décrite dans [Hartley 00], est utilisée et consiste à faire un ajustement hiérarchique. Les différentes étapes pour y parvenir sont décrites dans l'annexe B.

Dans [Mouragnon 06] les outils employés pour la reconstruction sont identiques à ceux utilisés dans [Royer 05]. Cependant l'utilisation d'une méthode incrémentale, qui raffine uniquement les paramètres estimés en fin de séquence, permet la réalisation de la reconstruction dans un cadre temps réel tout en gardant des performances similaires. Lors de la reconstruction 3D incrémentale, lorsqu'une nouvelle image clef est sélectionnée, de nouveaux points appariés sont triangulés et la pose courante est estimée grâce à la géométrie épipolaire. Ensuite, comme un ajustement de faisceaux classique est trop coûteux en temps pour affiner la solution, l'idée consiste à réduire le nombre de pa-

ramètres à optimiser et éviter par la même occasion les redondances dans les calculs. Ainsi seuls les paramètres des n dernières poses et les coordonnées des points 3D vus dans les n dernières images clefs sont retenus dans la procédure de minimisation. Pour assurer la cohérence des informations calculées à chaque incrément (c'est-à-dire que les nouveaux paramètres soient compatibles avec ceux estimés avant), les reprojections de points dans les n_1 dernières images clefs avec $n_1 > n$ sont prises en compte.

Les techniques basées sur un ajustement de faisceaux sont à l'heure actuelle reconnues comme les meilleures solutions pour la précision des reconstructions 3D lorsque la vision seule est considérée. Cependant, lorsque le résultat de ces techniques est comparé à la vérité terrain, généralement à l'aide de récepteurs GPS de précision centimétrique, une erreur de reconstruction qui peut être grande sépare les deux trajectoires. Cette erreur globale, mesurée par la distance séparant les extrémités finales des deux trajectoires après avoir fait correspondre les extrémités initiales (à l'aide d'une rotation et d'une translation), n'est pas un problème si l'on cherche juste à se localiser par rapport à la trajectoire. Par contre ce qui est un problème, c'est la présence de distorsions locales le long de la trajectoire. Dans le cadre d'une application de navigation en convoi, il a été mis en évidence au chapitre 1 que la précision de tels algorithmes était insuffisante pour maintenir de façon satisfaisante une consigne d'interdistance métrique entre chaque véhicule.

3.2.2.3 Vision augmentée par des informations additionnelles

L'obtention de distances évaluées avec un maximum de précision étant un point fondamental pour notre application, il n'est pas envisageable, au vu des précédentes conclusions, de n'exploiter que la vision monoculaire pour localiser les véhicules. Dans la suite de cet état de l'art, nous présentons les stratégies de reconstruction qui ont intégré des informations additionnelles pour raffiner le résultat final.

Intégration d'un capteur supplémentaire Une solution toute naturelle repose sur l'utilisation d'une autre caméra. La stéréovision présente effectivement plusieurs avantages dont le principal est de fournir dès la première paire d'images des informations 3D recalées dans un référentiel métrique, grâce à la connaissance de la pose relative entre les deux caméras. De plus, le cas où il n'y a pas ou peu de mouvement, qui pose des difficultés en vision monoculaire, se traite sans aucun problème avec un dispositif multi-caméras. Suivant cette logique, un dispositif stéréo a été proposé dans [Nistér 06] pour localiser des robots mobiles en temps réel. Cette méthode incrémentale s'appuie tout d'abord sur la géométrie épipolaire classique pour fournir l'estimation d'une nouvelle pose du dispositif stéréo puis raffine ensuite cette pose par ajustement de faisceaux. Les nouveaux points 3D sont finalement triangulés à partir de la paire stéréo et ajoutés au modèle. La stratégie a été comparée expérimentalement sur plusieurs sites à l'aide d'un dispositif couplant un système de localisation par GPS avec des données inertielles. Les résultats sont très satisfaisants puisqu'après avoir effectué trois boucles et parcouru 184m sur l'une des expériences, l'erreur de positionnement finale était seulement de

4.1m. Malheureusement l'objectif des auteurs est concentré sur l'estimation du parcours du véhicule et peu d'intérêt est porté à l'obtention d'une carte 3D réutilisable pour une localisation ultérieure. Effectivement, à chaque nouvelle itération de la méthode, les amers retenus pour l'estimation de la pose et pour la création de la carte sont ceux communs aux images des deux caméras, ce qui restreint le champ de perception à l'environnement proche et diminue fortement le potentiel de la reconstruction. Lors d'une localisation ultérieure, le véhicule devrait alors rester proche de la trajectoire pour qu'une correspondance puisse toujours être établie entre les points d'intérêt de la vue courante et ceux de la trajectoire d'apprentissage. La localisation serait aussi sensible à toute modification de l'environnement proche.

Dans [Clipp 08], ce problème a été abordé et les auteurs proposent de s'appuyer sur un dispositif multi-caméras sans recouvrement des champs de vue. La théorie est formulée avec deux caméras : chaque nouvelle pose est évaluée classiquement à un facteur d'échelle près en utilisant les points d'intérêt reconnus sur deux images consécutives d'une des deux caméras. La connaissance de la baseline (pose relative entre les deux caméras du dispositif) permet alors de retrouver le facteur d'échelle à partir d'une seule correspondance obtenue entre deux images consécutives sur la deuxième caméra. Un certain nombre de configurations critiques est cependant mis en évidence et le facteur d'échelle ne peut s'évaluer que dans peu de situations puisque le déplacement en ligne droite n'est par exemple pas autorisé. Les performances d'estimation du facteur d'échelle ont été évaluées lors d'expérimentations avec un dispositif contenant non moins de huit caméras sans recouvrement des champs de vue. Le rapport entre les normes des translations réelle et estimée $|T_{est}|/|T_{true}|$ dans les configurations non-singulières affiche une valeur moyenne de 0.9 et un écart type de 0.28, ce qui n'est malheureusement pas mieux qu'avec un ajustement de faisceaux global en utilisant la vision monoculaire.

Certains auteurs ont proposé d'ajouter d'autres types de capteurs et de fusionner leurs données avec les données vision pour apporter des informations sur la la trajectoire de la caméra. Dans [Ikeda 07], un ajustement de faisceaux original intègre par exemple les données d'un récepteur GPS. La fonction de coût à minimiser est alors la somme pondérée, en fonction de la confiance sur les capteurs, de deux erreurs : l'erreur de reprojection classique et l'erreur entre la position du récepteur GPS exprimée dans le repère estimé de la caméra courante et sa position déduite de la translation (connue) entre la caméra et le récepteur GPS. Dans [Hol 07] et [Nuetzi 10] des informations inertielles précises permettent de robustifier une approche d'EKF-SLAM basée sur la vision monoculaire. L'état du système comporte alors un nombre conséquent de variables : accélération, vitesse et position sont fournies selon les trois directions ainsi que l'orientation et la vitesse angulaire représentées par des quaternions. Malheureusement, pour une application en milieu urbain, le problème principal de ces solutions reste le prix et la complexité d'intégration de ce type de matériel sur un véhicule grand public. En particulier, ces coûts ont tendance à augmenter avec la précision de localisation recherchée. Ces capteurs ne sont aussi pas tous adaptés à la ville, à l'image du GPS dont les signaux en provenance des satellites sont facilement masqués par les hautes structures urbaines.

Intégration d'une contrainte Pour pallier ces difficultés, différents auteurs ont développé des stratégies pour ajouter des connaissances à faible coût en intégrant des contraintes sur la scène ou sur le déplacement de la caméra. Plusieurs méthodes ont par exemple étudié la mise en corrélation des données visuelles avec un modèle. Ainsi dans [Levin 04] les auteurs "alignent" la trajectoire estimée de la caméra sur un tracé réalisé à la main sur une carte et décrivant le parcours réalisé. Pour assurer la cohérence globale de la scène, un ajustement de faisceaux global est finalement appliqué. Cependant, lors de cette deuxième étape, la carte n'est plus exploitée et des déformations peuvent survenir à nouveau. Une autre limite de la méthode est que la précision de la carte est inconnue et peut difficilement être contrôlée, celle-ci étant dessinée à la main.

Dans [Lothe 10] les auteurs proposent une méthode pour corriger la reconstruction a posteriori en fusionnant les informations d'une caméra avec un modèle 3D simple de l'environnement. La stratégie s'applique dans un cadre urbain et le modèle 3D est constitué d'un ensemble de plans verticaux qui représentent les façades des bâtiments d'un quartier de ville. Par conséquent, les trajectoires dans un tel environnement sont composées de segments de droite qui forment des angles droits. La méthode se déroule en trois étapes. Tout d'abord, la trajectoire de la caméra est fragmentée au niveau des intersections des segments de droite, puis chacun des points 3D reconstruits est associé à un des segments obtenus. Ensuite, une transformation de type articulée est utilisée pour aligner grossièrement la reconstruction avec le modèle 3D : les lignes droites de la trajectoire reconstruite sont considérées comme des éléments rigides et des articulations sont placées à chaque virage. Ainsi, les transformations retenues sont des similitudes par morceaux avec contraintes de jointure aux extrémités. Finalement, un ajustement de faisceaux considérant à la fois la cohérence entre les observations 2D et les points 3D reconstruits et la cohérence géométrique des points 3D avec le modèle de l'environnement est proposé. Plutôt que de minimiser la somme pondérée des deux erreurs, l'ajustement de faisceaux est effectué à partir de nouveaux points 3D cette fois-ci localisés sur le modèle 3D. L'idée consiste à faire converger les rayons optiques issus de l'observation d'un point d'intérêt vers un point de la surface du modèle 3D. Cela revient à limiter au maximum la dispersion des points 3D qui correspondent à l'intersection des rayons optiques d'un point d'intérêt vu depuis plusieurs images avec la surface du modèle. Dans l'ajustement de faisceaux on considère alors le barycentre de ces points projetés sur le modèle 3D : l'erreur mesurée entre les observations et la reprojection de ces barycentres est utilisée comme fonction de coût. La méthode a été testée sur des séquences de synthèse et une séquence réelle de grande échelle obtenue lors d'un parcours de $1.5km$ dans Versailles. Un modèle 3D de la ville d'une précision de $2m$ est alors utilisé. De bons résultats sont obtenus puisque l'erreur moyenne s'élève à $14cm$ sur les séquences de synthèse et la superposition des séquences réelles avec l'image satellite associée semble correspondre. Concernant cette stratégie, un réel inconvénient réside dans la contrainte que l'on impose sur la position des points 3D : ces derniers doivent nécessairement appartenir à un plan pour pouvoir être recalés au modèle. Cela peut paraître réaliste si l'on se déplace le long de rues quadrillées dans une grande ville, mais dans d'autres environnements cette contrainte serait rédhibitoire.

De façon générale, une limite importante de ces stratégies pour une application de navigation réside dans l'existence, même grossière, d'une carte 3D. On aimerait effectivement s'affranchir de la connaissance de la scène, puisque sinon le système de navigation serait alors restreint aux zones de circulation cartographiées. Une deuxième contrainte majeure vient du caractère hors-ligne de ces méthodes. On a vu précédemment que la construction d'environnement 3D pouvait être réalisée en temps réel, il est dommage que l'information fournie par le modèle ne puisse pas être utilisée elle-aussi en parallèle.

Une autre classe de stratégies propose d'intégrer des contraintes sur le déplacement de la caméra. Dans [Fraundorfer 10] les auteurs simplifient grandement le problème de reconstruction en introduisant des hypothèses sur le mouvement de la caméra lorsque celle-ci est montée sur un véhicule. Le mouvement de la caméra est tout d'abord supposé planaire, ce qui permet de réduire l'estimation de pose à trois paramètres au lieu de six. Ensuite la procédure de reconstruction est scindée en deux étapes. On cherche en premier lieu à obtenir la pose relative à un facteur d'échelle près entre les images clefs. Ce calcul s'effectue de façon précise sans avoir recours à un ajustement de faisceaux car on suppose que la caméra effectue localement un mouvement circulaire. Cette hypothèse permet d'exprimer la matrice essentielle ainsi que la direction de la translation $\overrightarrow{t_i}$ entre deux positions successives P_i et P_{i+1} en fonction d'un unique paramètre : l'orientation relative θ_i entre les deux poses. Chaque amer mis en correspondance entre les deux images clefs va alors fournir une estimation de la pose relative. L'information obtenue par l'ensemble des amers est ensuite représentée à l'aide d'un histogramme où l'on visualise le nombre d'amers pour lesquels l'estimation d'angle appartient à un même intervalle. L'estimation retenue est celle qui correspond au pic de l'histogramme. L'orientation des caméras, ainsi que la direction de la translation entre les positions successives obtenues au cours de cette étape resteront inchangées. Ainsi la $i+1^{ème}$ position P_{i+1} est logiquement située sur une droite passant par la position précédente P_i et dont les paramètres sont θ_i et $\overrightarrow{t_i}$. Seules les distances $d_i = \|P_{i+1} - P_i\|$ restent inconnues et sont estimées au cours de la deuxième étape à travers un ajustement de faisceaux. Si N images clefs ont été retenues, le nombre de paramètres à estimer sera donc également N. La solution initiale pour débuter l'ajustement de faisceaux est définie par $d_i = 1,\ \forall 1 \leq i \leq N$ et la position des points 3D est re-calculée avant chaque itération de l'optimisation. Plusieurs reproches peuvent être énoncés concernant cette stratégie. Tout d'abord le système dynamique, tel qu'il est modélisé, est très limité puisqu'entre deux images clefs le véhicule est censé avoir un mouvement circulaire. Du point de vue de la commande c'est irréalisable, à moins que le véhicule s'arrête à chaque image clef pour changer de braquage, puisque l'on a une discontinuité de courbure. Le modèle d'Ackermann sur lequel s'appuient les auteurs caractérise effectivement à chaque instant un centre de rotation instantané, mais la trajectoire qui en découle peut avoir des formes plus complexes (clothoïdale, etc.). Si l'on veut prendre plus rigoureusement ce modèle en considération, d'autres paramètres seraient vraisemblablement à prendre en compte et il n'est pas dit que l'ajustement de faisceaux qui en découlerait compterait moins de paramètres à estimer que la méthode classique. Il est aussi regrettable que

cette stratégie n'ait pas un fonctionnement en temps réel, alors que des approches qui considèrent plus de paramètres y parviennent [Mouragnon 06]. Finalement, à la vue des résultats expérimentaux présentés, des distorsions notables sont toujours présentes. Il s'agit d'une remarque qualitative puisque les performances en terme de précision sont évaluées simplement en superposant la trajectoire reconstruite avec celle estimée par odométrie.

Au cours de cet état de l'art, différentes techniques ont été proposées pour localiser un véhicule par vision monoculaire. Les approches de localisation partielle ont été écartées au profit de celles qui permettent d'évaluer des informations 3D proches de celles que l'on pourrait obtenir dans un référentiel métrique. L'approche de commande globale pour un convoi de véhicules, décrite en section 1.3.1, repose en effet sur la capacité des véhicules à se positionner avec précision par rapport au véhicule leader qui partage son état avec le reste de la flotte. Par conséquent la mise en œuvre du convoi nécessite de pouvoir se localiser dans un référentiel métrique et estimer des distances entre les véhicules.

Le problème d'établir une carte 3D a été abordé selon deux types d'approche : la première s'appuie sur les principes de l'EKF-SLAM, tandis que la deuxième propose d'affiner une première reconstruction obtenue par les principes de la géométrie projective et épipolaire à l'aide d'un ajustement de faisceaux. Quand un filtre de Kalman (EKF) est utilisé, il est nécessaire qu'il soit bien initialisé, c.a.d. que l'on doit disposer d'une bonne estimation de l'état du robot et de points d'intérêt au démarrage de l'application. Dans notre cas, nous souhaitons que l'application en convoi soit la plus flexible possible et n'implique aucune connaissance a priori sur la scène. L'initialisation d'un filtre de Kalman se révèlerait alors problématique. La technique par ajustement de faisceaux est la plus coûteuse en temps de calcul. Cependant on a vu dans [Mouragnon 06] que l'optimisation pouvait être réalisée localement sans pour autant diminuer la qualité de la carte. Par ailleurs, une fois la carte reconstruite, le temps de localisation d'un robot est identique quelle que soit la méthode utilisée. Aussi l'optimisation par ajustement de faisceaux, qui n'est pas sensible au biais sur le modèle d'évolution du robot, permet d'obtenir les meilleurs résultats en terme de précision. Pour ces raisons nous avons choisi d'utiliser la technique de reconstruction et de localisation élaborée dans [Royer 05] pour mettre en œuvre la stratégie de commande en convoi de véhicules rappelée en section 1.3.1.

Malgré l'optimisation globale, réalisée sur l'ensemble des poses de la caméra et des points d'intérêt, cette stratégie présente tout de même une dérive du facteur d'échelle et des distorsions locales sont perceptibles le long de la trajectoire. En section 3.2.2.3, on a vu que le problème de reconstruction peut être raffiné lorsque les données vision sont fusionnées avec celles d'autres capteurs ou d'une carte de l'environnement. Malheureusement une carte n'est pas toujours disponible et les autres capteurs sont chers et/ou inadaptés au cadre urbain de notre application. L'intégration d'une contrainte qui prend en compte le déplacement de la caméra a aussi été envisagée mais les performances ne sont pas meilleures qu'un ajustement de faisceaux classique. Dans cette thèse nous proposons ci-après plusieurs méthodes pour apporter des corrections à la

trajectoire et mettre en œuvre efficacement un convoi de véhicules.

3.3 Analyse des distorsions

L'objectif de cette section est de cerner plus précisément la nature des distorsions et leur influence sur l'estimation des informations 3D requises pour commander un système robotique. Dans notre cas, on considère en particulier les variables de distance, orientation et courbure mesurées le long d'une trajectoire obtenue dans le monde vision. Cette étape préliminaire est nécessaire par la suite afin de concevoir une approche analytique pour corriger les informations fournies par les techniques de localisation par vision monoculaire développées dans [Royer 05], pour ensuite les exploiter dans les algorithmes de conduite en convoi et obtenir des performances aussi satisfaisantes qu'avec une localisation par RTK-GPS. Certaines hypothèses, formulées ci-dessous, ont assez logiquement été émises au vu des résultats des approches détaillées à la section 3.2. La première sous-section est consacrée à la validation de ces hypothèses. Les deux qui suivent s'attachent respectivement à modéliser de façon analytique le problème et à formuler une stratégie de correction hors-ligne performante.

Hypothèse 1. *Le phénomène de distorsion est répétable et dépend uniquement de la qualité de la reconstruction 3D générée.*

Hypothèse 2. *Les distorsions sont localement homogènes.*

3.3.1 Analyse empirique

Une première façon pour caractériser les distorsions entre le monde réel (c.a.d. le monde métrique) et le monde vision consiste simplement à examiner localement la valeur du facteur d'échelle existant entre les longueurs mesurées dans les deux mondes. Pour ce faire, la première expérimentation consiste en un enregistrement en parallèle des données de localisation fournies par vision et par un récepteur RTK-GPS lors de guidages le long de la trajectoire représentée dans le coin supérieur gauche de la Fig. 3.1. Cette trajectoire mesure approximativement $115m$ de long. Plusieurs guidages ont été réalisés avec plusieurs véhicules, chacun disposant de sa propre caméra. Afin d'estomper au mieux l'effet du bruit sur les données sans trop lisser les résultats et perdre de la pertinence, les facteurs d'échelle sont calculés sur des tronçons successifs d'une longueur de $2m$. Les résultats obtenus lors de deux expérimentations sont présentés sur la Fig. 3.1, mais les résultats obtenus lors des autres expérimentations sont tout à fait semblables. Compte-tenu du type d'expérimentation réalisée, les facteurs d'échelle présentés sur la Fig. 3.1 sont représentatifs de distorsions selon la tangente à la trajectoire. On peut remarquer que les profils d'évolution des facteurs d'échelle sont vraiment similaires, attestant une forte répétabilité des distorsions au cours des différentes expérimentations. On valide ainsi l'**Hypothèse 1** : les distorsions ne proviennent pas d'un phénomène aléatoire lié à l'algorithme de localisation ni à un mauvais réglage ou d'un problème de

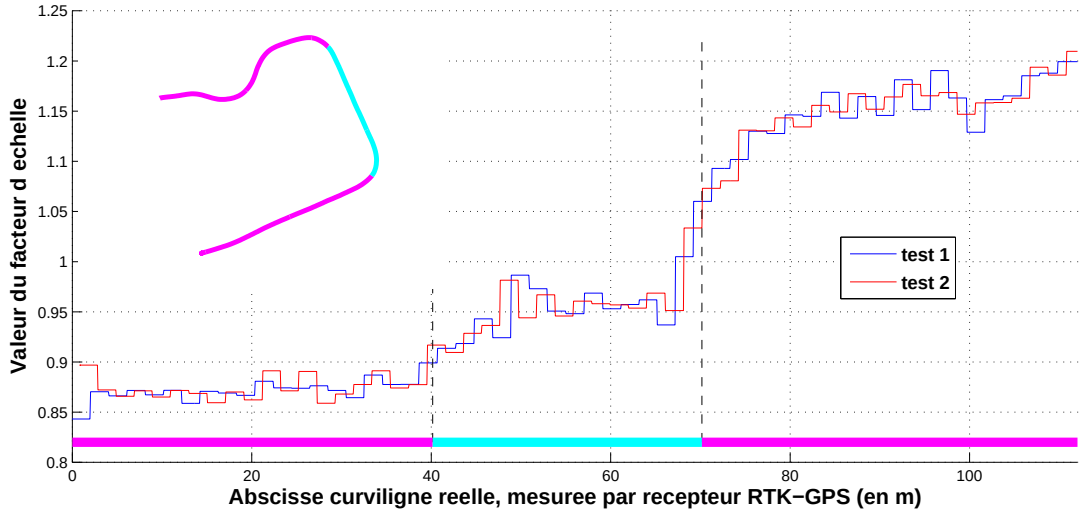


FIG. 3.1 – Première estimation des facteurs d'échelle le long de la trajectoire

calibration des caméras.

On peut aussi remarquer que le facteur d'échelle est quasi-constant le long des lignes droites, comme sur la première portion de couleur magenta sur la Fig. 3.1, et varie de manière assez abrupte en courbe. Ce phénomène s'explique en partie par le changement d'axe de vue de la caméra lors des virages pendant la phase d'apprentissage : le champ de perception se modifie et les points d'intérêt communs entre plusieurs images successives se raréfient, ce qui naturellement diminue la précision de la procédure d'optimisation pour reconstruire l'environnement 3D.

Un second ensemble d'expérimentations a été réalisé afin de valider l'**Hypothèse 2** sur l'homogénéité des facteurs d'échelle. On souhaite vérifier que le jeu de facteurs obtenu précédemment ne soit pas uniquement représentatif des distorsions selon la tangente à la trajectoire mais qu'il soit aussi exploitable pour retrouver une valeur de distance métrique entre deux points 3D quelconques situés dans un voisinage de cette trajectoire. Pour cela l'idéal serait de comparer des distances du monde réel et du monde virtuel dans une direction perpendiculaire à la trajectoire. Cependant, ce type d'expérience est complexe à mettre en œuvre : compte tenu des contraintes de non holonomie du véhicule il est en effet nécessaire d'utiliser un dispositif expérimental constitué de deux caméras (ou plus) pour calculer des longueurs perpendiculairement à la trajectoire. Aussi, pour éviter les difficultés de calibration (pour estimer la transformation rigide qui lie deux caméras fixées au véhicule) et de synchronisation (ou recalage temporel) des données sur les deux images, nous avons préféré procéder à l'aide de simulations en suivant la procédure suivante :

- Un environnement, constitué de blocs texturés, a tout d'abord été généré à l'aide du

logiciel 3DS MAX, pour représenter des bâtiments et délimiter une trajectoire circulaire dont le rayon est $113m$. Deux images prises dans cet environnement se trouvent sur la Fig. 3.2 a) et b). Cette forme particulière est choisie pour la trajectoire afin de faire apparaître des distorsions (très faibles en ligne droite), tout en restant suffisamment simple pour pouvoir simuler facilement l'évolution d'un dispositif à deux caméras liées par une transformation rigide.

• Ensuite, dans cet univers simulé, plusieurs enregistrements vidéos ont été réalisés en émulant le passage d'une caméra le long de la trajectoire. Chaque enregistrement est fait sur quatre tours, en maintenant une distance latérale de $0m$ ou $4m$ d'un côté ou de l'autre du centre de la voie, voir Fig. 3.2 a) et b). On note ces séquences S_0, S_4 et S_{-4}. Les enregistrements sont effectués de sorte que deux images S_j^i et S_k^i portant le même indice i sur deux séquences différentes correspondent à deux positions de la caméra alignées selon un axe perpendiculaire à la tangente à la trajectoire au même point d'intersection. 300 images sont ainsi enregistrées sur chaque séquence, émulant des poses de caméra approximativement espacées de $9.5m$ pour la séquence S_0. Cet écart peut être cohérent avec la variation du facteur d'échelle le long d'une telle trajectoire régulière et permet de diminuer l'effet du bruit sur les mesures de localisation vision.

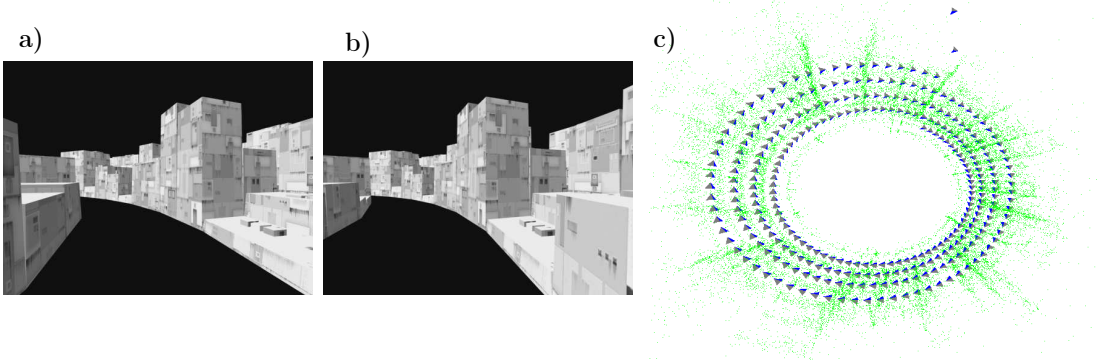


FIG. 3.2 – a) Image S_{-4}^{10} : la trajectoire de la caméra est déportée de $4m$ sur la gauche. b) Image S_4^{10} : la trajectoire de la caméra est déportée de $4m$ sur la droite. c) Vue de dessus de la reconstruction 3D de la séquence de référence S_0

• La reconstruction 3D, représentée sur la Fig. 3.2 c), est finalement générée à partir de la séquence de référence S_0 qui est alignée au centre de la voie. Comme on peut le voir, des distorsions sont bien présentes puisque la forme de la trajectoire reconstruite, reflétée par la position des caméras (triangles), n'est pas un cercle mais une spirale. Finalement une localisation est effectuée sur chaque image des autres séquences à partir des points 3D reconstruits sur la séquence de référence. L'évolution de la caméra émulée étant parfaitement connue, le facteur d'échelle peut alors facilement être mesuré tangentiellement et latéralement à la trajectoire. Au niveau de chaque image S_0^i, on réalise les calculs suivants :

Estimation tangentielle Le facteur d'échelle est mesuré comme le rapport des dis-

tances réelle et vision entre les images S_0^{i-1} et S_0^{i+1}.

Estimation latérale Le facteur d'échelle est mesuré comme le rapport des distances réelle et vision entre les images S_{-4}^{i} et S_4^{i}.

Les résultats sont présentés sur la Fig. 3.3. Sur l'image de gauche sont comparés les facteurs d'échelle évalués selon les composantes tangentielle et latérale. On constate qu'ils sont vraiment très similaires, ce qui montre que l'**Hypothèse 2** est parfaitement vérifiée. Les facteurs d'échelle obtenus par l'estimation tangentielle sont ensuite utilisés pour estimer la courbure de la trajectoire à partir des données vision : les positions vision sont localement approximées par un polynôme, puis la valeur de courbure calculée au niveau de la position de chaque caméra est divisée par la valeur du facteur d'échelle. Comme le montre la Fig. 3.3, l'évaluation de la courbure à partir des données vision ainsi corrigées est très proche de la réalité.

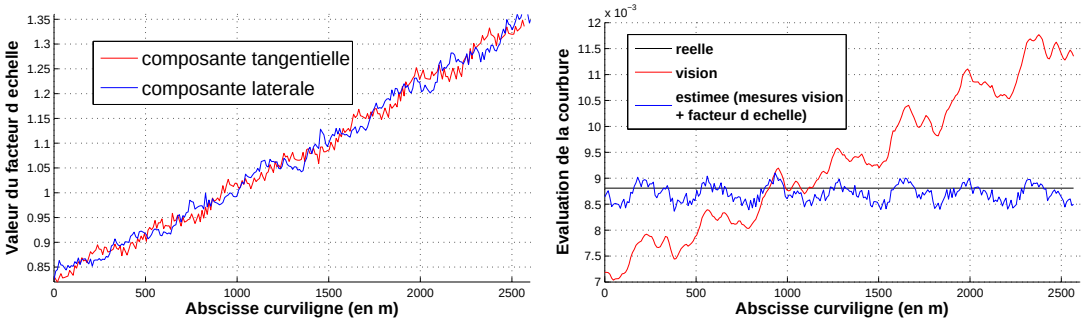


FIG. 3.3 – Evaluation des facteurs d'échelle selon plusieurs directions

Conclusions

• Un jeu de facteurs d'échelle calculé le long d'une trajectoire lors d'une expérimentation peut, de par sa répétabilité, être utilisé pour convertir des distances du monde vision en distances métriques lors d'expérimentations ultérieures, menées avec d'autres véhicules et/ou d'autres caméras.

• Chaque facteur d'échelle, associé à une portion (la plus petite possible) de la trajectoire, permet grâce à la propriété d'homogénéité de corriger les distances du monde vision dans un voisinage autour de cette portion de trajectoire.

Aussi, si l'on dispose d'un tel jeu de facteurs d'échelle, il est donc possible à partir des données visuelles d'estimer la valeur des informations requises pour le calcul des commandes latérale et longitudinale. Un guidage précis de véhicules en convoi est donc tout à fait envisageable par simple vision monoculaire.

3.3.2 Influence des distorsions sur les performances des lois de commande

On dispose à présent de tous les éléments pour exprimer les lois de commande directement avec les variables fournies par la localisation par vision monoculaire. Pour cela on introduit les notations suivantes :

Notation Il est nécessaire désormais de distinguer la valeur des variables lorsqu'elles sont exprimées d'une part dans le monde vision et d'autre part dans le monde réel. Soit une variable "x " à valeur dans le référentiel métrique réel. Alors la valeur de cette variable dans le référentiel vision sera notée "x^v ". Et si la valeur de x dans le référentiel métrique est estimée à partir de variables exprimées dans le référentiel vision, alors cette estimée sera notée "$\hat{x}^v$ ".

Représentation du facteur d'échelle Par définition, le facteur d'échelle entre le monde vision et le monde réel est le ratio de la mesure d'une longueur dans l'un des mondes par la mesure de la même longueur dans l'autre monde. Pour l'analyse empirique présentée sur la Fig. 3.1, des segments de $2m$ avaient été utilisés pour évaluer à grosse maille l'évolution du facteur d'échelle le long de la trajectoire. Si on veut maintenant définir proprement le facteur d'échelle local dans le voisinage d'un point de la trajectoire, il convient de faire tendre les longueurs des segments vers un infiniment petit, ou ce qui est équivalent, calculer le ratio des vitesses instantanées le long de la trajectoire exprimées dans les deux mondes. Aussi, dans la suite, le facteur d'échelle dans un voisinage de la trajectoire autour de l'abscisse s_i^v est défini par la fonction suivante :

$$\lambda(s_i^v) = \dot{s}_i \,/\, \dot{s}_i^v \tag{3.12}$$

Remarque : la fonction λ peut s'exprimer de deux façons différentes : en fonction de l'abscisse curviligne réelle ($\lambda(s)$) ou de celle mesurée dans le référentiel vision ($\lambda(s^v)$). On choisit arbitrairement ici de s'adapter à l'espace de référence de l'application et de conserver l'expression obtenue dans le référentiel vision.

A partir de l'équation (3.12), on déduit immédiatement que l'abscisse curviligne exprimée dans le repère métrique peut être calculée à partir de celle exprimée dans le repère vision par :

$$s_i = \int_0^{s_i^v} \lambda(\xi) d\xi \tag{3.13}$$

où ξ est la variable courante d'abscisse curviligne exprimée dans le repère vision. Pour rester cohérent avec la représentation de la trajectoire proposée au chapitre précédent, si $\Gamma(\tau) = (\,\Gamma_x(\tau),\, \Gamma_y(\tau)\,)$ désigne l'équation 2D paramétrique de la trajectoire de référence Γ dans le référentiel vision, alors l'équation (3.13) devient :

$$s_i = \int_0^{\tau(s_i^v)} \lambda(\tau) \left|\left| \frac{d\Gamma}{d\tau}(\tau) \right|\right| d\,\tau \tag{3.14}$$

où $\tau(s_i^v)$ représente la valeur du paramètre de la courbe 2D $\Gamma(\tau)$ associée à l'abscisse curviligne s_i^v et par abus de notation $\lambda(\tau)$ désigne la valeur de λ pour l'abscisse curviligne correspondant au paramètre τ de la courbe Γ.

Les distorsions étant localement homogènes (*Hypothèse 2*), alors :

- les écarts latéraux évalués dans le monde réel et dans le monde vision autour du point d'abscisse s_i^v sont dans le rapport :

$$y_i = \lambda(s_i^v) y_i^v \tag{3.15}$$

- les écarts angulaires évalués dans les deux mondes sont identiques (car localement autour du point d'abscisse s_i^v, les longueurs dans les deux directions du plan sont affectées par le même facteur d'échelle $\lambda(s_i^v)$) :

$$\tilde{\theta}_i = \tilde{\theta}_i^v \tag{3.16}$$

- enfin, les courbures évaluées dans les deux mondes autour du point d'abscisse s_i^v sont dans le rapport $1/\lambda(s_i^v)$ (puisqu'une courbure est homogène à l'inverse d'une distance) :

$$k(s_i) = k^v(s_i^v)/\lambda(s_i^v) \tag{3.17}$$

Ces équations vont maintenant nous permettre d'analyser l'effet des distorsions de la reconstruction 3D sur la régulation latérale et longitudinale des véhicules.

Régulation latérale La régulation latérale a été présentée à la section 1.3.1.3. L'expression analytique de la loi de commande est donnée par l'équation (1.59), que l'on peut découper comme suit :

$$\delta_i(X_i) = \arctan\left(l\left[f(X_i)(m_{3i}(X_i) + g_1(X_i)) + g_2(X_i)\right]\right) \tag{3.18}$$

où X_i désigne le vecteur d'état du véhicule : $X_i = (s_i, y_i, \tilde{\theta}_i)$

$$f(X_i) = \frac{\cos^3 \tilde{\theta}_i}{(1 - k(s_i).y_i)^2} \tag{3.19}$$

$$g_1(X_i) = \frac{dk(s_i)}{ds_i}.y_i.\tan\tilde{\theta}_i + k(s_i)(1 - k(s_i).y_i)\tan^2\tilde{\theta}_i \tag{3.20}$$

$$g_2(X_i) = k(s_i).\frac{\cos\tilde{\theta}_i}{1 - k(s_i).y_i} \tag{3.21}$$

$$m_{3i}(X_i) = -K_d(1 - k(s_i).y_i)\tan\tilde{\theta}_i - K_p y_i \tag{3.22}$$

Cette loi de commande est particulièrement intéressante car la dérivée seconde de l'écart latéral par rapport à s_i (rappel : cette dérivée est notée par un prime) est tout simplement :

$$y_i'' = m_{3i}(X_i) \tag{3.23}$$

Le choix particulier pour $m_{3i}(X_i)$ (éq. (3.22)) impose alors la loi de comportement ci-dessous (voir (1.58)) :

$$y_i'' + K_d y_i' + K_p y_i = 0 \tag{3.24}$$

L'équation (3.24) assure que l'écart latéral y_i converge vers zéro, avec des performances que l'on peut ajuster en spécifiant les gains (K_p, K_d).

Si on reporte maintenant le vecteur d'état X_i^v à la place de X_i dans la loi de commande (3.18), la loi de comportement (3.24) n'est plus satisfaite. En effet, si on reporte (3.13), (3.15), (3.16) et (3.17) dans (3.19) à (3.22), on vérifie aisément que $f(X_i^v) = f(X_i)$, mais que :

$$g_1(X_i^v) \neq g_1(X_i) \tag{3.25}$$

$$g_2(X_i^v) \neq g_2(X_i) \tag{3.26}$$

$$m_{3i}(X_i^v) \neq m_{3i}(X_i) \tag{3.27}$$

En reprenant les calculs qui permettent d'établir la relation (3.23), on montre que si X_i^v est utilisé à la place de X_i, alors :

$$y_i'' = m_{3i}(X_i^v) + g_1(X_i^v) - g_1(X_i) + f(X_i)(g_2(X_i^v) - g_2(X_i)) \tag{3.28}$$

La courbure ou la dérivée de la courbure sont en facteur dans les expressions de $g_1(X_i)$ et $g_2(X_i)$ (voir (3.20) et (3.21)). Comme dans les applications pratiques les courbures sont en général faibles, les erreurs commises en remplaçant X_i par X_i^v sont faibles devant $m_3(X_i^v)$, de sorte que l'on peut approcher l'équation (3.28) par :

$$y_i'' \simeq m_{3i}(X_i^v) \tag{3.29}$$

En reportant (3.15), (3.16), (3.17) dans (3.22), on obtient finalement la loi de comportement ci-dessous pour l'erreur latérale :

$$y_i'' + K_d y_i' + \frac{K_p}{\lambda(s_i^v)}.y_i = 0 \tag{3.30}$$

Puisque $\lambda(s_i^v) > 0$, alors l'équation (3.30) assure toujours que y_i converge vers zéro. En conclusion, les distorsions ne compromettent pas la régulation latérale, ce qui est très positif. Elles modifient juste les performances de cette régulation, puisqu'elles se traduisent par une modification du gain sur la loi de comportement (3.30). La convergence asymptotique à 0 de y_i et $\tilde{\theta}_i$ est d'ailleurs plutôt satisfaisante si l'on regarde les résultats de suivi de trajectoire obtenus dans [Royer 05].

Régulation longitudinale La régulation longitudinale a été présentée à la section 1.3.1.4. L'expression analytique de la loi de commande est donnée par les équations (1.72)-(1.74). Pour mettre en évidence, de la façon la plus simple possible, les difficultés introduites par les distorsions sur la régulation longitudinale, on se place ci-dessous dans le cas d'étude où le convoi est en régime nominal, c'est-à-dire tous les véhicules sont espacés de $d^\star$, comme désiré. Dans cette situation il n'y a aucun danger immédiat pour le véhicule $i+1$. Par conséquent, dans l'équation (1.66), la fonction sigmoïde prend la valeur

un, de sorte que la variable régulée c_{i+1} est directement e^1_{i+1}, l'erreur de régulation longitudinale entre le véhicule $i+1$ et le leader :

$$e^1_{i+1} = s_1 - s_{i+1} - id^\star \tag{3.31}$$

La commande longitudinale est dans ce cas une commande référencée sur le leader. La loi de commande (1.72) se simplifie alors comme suit :

$$v_{i+1}(X_1, X_{i+1}) = \frac{1 - y_{i+1}.k(s_{i+1})}{\cos\tilde{\theta}_{i+1}} \left(\frac{v_1 \cos\tilde{\theta}_1}{1 - y_1.k(s_1)} - m_{4(i+1)} \right) \tag{3.32}$$

Cette loi de commande est particulièrement intéressante car elle impose (voir (1.73)) que :

$$\dot{c}_{i+1} = \dot{e}^1_{i+1} = m_{4(i+1)} \tag{3.33}$$

et le choix (1.74) pour $m_{4(i+1)}$ conduit à

$$\dot{e}^1_{i+1} = -Ke^1_{i+1} \tag{3.34}$$

ce qui assure un convergence exponentielle vers zéro de l'erreur longitudinale e^1_{i+1}.

Si on reporte maintenant les vecteurs d'état X^v_1 et X^v_{i+1} à la place de X_1 et X_{i+1} dans la loi de commande (3.32), alors la structure de la loi de comportement (3.34) est complètement détruite. En effet si on reporte (3.13), (3.15), (3.16) et (3.17) dans (3.32), alors la relation (3.33) est toujours satisfaite, mais la relation (3.34) s'écrit désormais :

$$\dot{e}^1_{i+1} = -K(s^v_1 - s^v_{i+1} - id^\star) \tag{3.35}$$

Pour faire réapparaître la quantité e^1_{i+1} dans le membre de droite de l'équation (3.35), il faut ajouter et soustraire à ce membre la quantité $K(s_1 - s_{i+1})$. L'équation (3.35) se ré-écrit alors comme :

$$\dot{e}^1_{i+1} = -Ke^1_{i+1} - K(s^v_1 - s^v_{i+1} - s_1 + s_{i+1}) \tag{3.36}$$

ou encore, en utilisant (3.13) :

$$\dot{e}^1_{i+1} = -Ke^1_{i+1} - K(s^v_1 - s^v_{i+1} - \int_{s^v_{i+1}}^{s^v_1} \lambda(\xi)d\xi) \tag{3.37}$$

L'équation (3.37) montre clairement que l'erreur longitudinale n'est plus régulée à zéro : les distorsions entre le monde vision et le monde réel, qui n'ont aucune conséquence sérieuse sur la commande latérale (si ce n'est de modifier légèrement les gains de commande et donc les performances de la régulation), conduisent à l'inverse à des erreurs de régulation longitudinale qui peuvent être significatives. Par exemple, dans le cas particulier où la fonction $\lambda(s^v_i)$ est constante (ce qui est le cas lors d'un déplacement en ligne droite, comme mis en évidence sur la Fig. 3.1), alors les écarts longitudinaux exprimés dans les deux mondes sont liés par la relation :

$$s_1 - s_{i+1} = \int_{s^v_{i+1}}^{s^v_1} \lambda(\xi)d\xi = \lambda.(s^v_1 - s^v_{i+1}) \tag{3.38}$$

En reportant (3.38) dans (3.35), la loi de comportement longitudinale se ré-écrit :

$$\dot{e}^1_{i+1} = -K.(\frac{1}{\lambda}(s_1 - s_{i+1}) - id^\star) \tag{3.39}$$

ce qui montre que l'écart entre le véhicule $i+1$ et le véhicule 1 se stabilise sur la valeur $i\lambda d^\star$ au lieu de se stabiliser sur la valeur attendue $id^\star$.

Pour illustrer ce problème dans un cadre plus général, le comportement d'un convoi de trois véhicules a été simulé. L'évolution du facteur d'échelle est celle relevée lors des

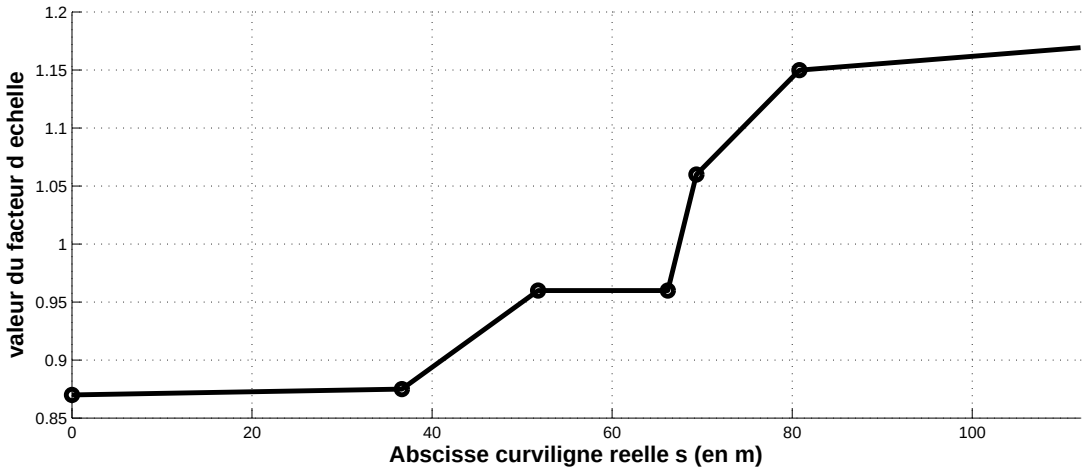


FIG. 3.4 – Dynamique simulée du facteur d'échelle

expérimentations décrites à la section 3.3.1, c'est-à-dire l'évolution est celle présentée sur la Fig. 3.1. Plus précisément, pour les besoins de la simulation, cette évolution a été approchée par des segments de droite comme décrit sur la Fig. 3.4, leur jonction formant une fonction continue sur l'ensemble de la trajectoire. Les véhicules sont initialement espacés de $5m$ conformément à la consigne $d^\star$ et commandés en utilisant les données exprimées dans le monde vision. Aucun bruit n'a été ajouté sur les données, les seules perturbations sont celles introduites par les distorsions. Les résultats sont rapportés sur la Fig. 3.5. Comme on peut le constater les erreurs sont flagrantes et s'élèvent approximativement à $0.8m$ pour le second véhicule et $1.60m$ pour le troisième.

Conclusion Le problème posé par les distorsions a été clairement identifié : elles n'ont pas de conséquences sérieuses sur la commande latérale, mais elles introduisent des erreurs rédhibitoires sur la commande longitudinale. Pour recouvrer de bonnes performances de commande, il faut donc évaluer la fonction $\lambda(s_i^v)$ qui décrit l'évolution du facteur d'échelle local le long de la trajectoire. Une première méthode, utilisable uniquement hors ligne, est présentée à la section 3.3.3. Elle servira de base de comparaison pour apprécier les stratégies en ligne à la section 3.4 et 3.5.

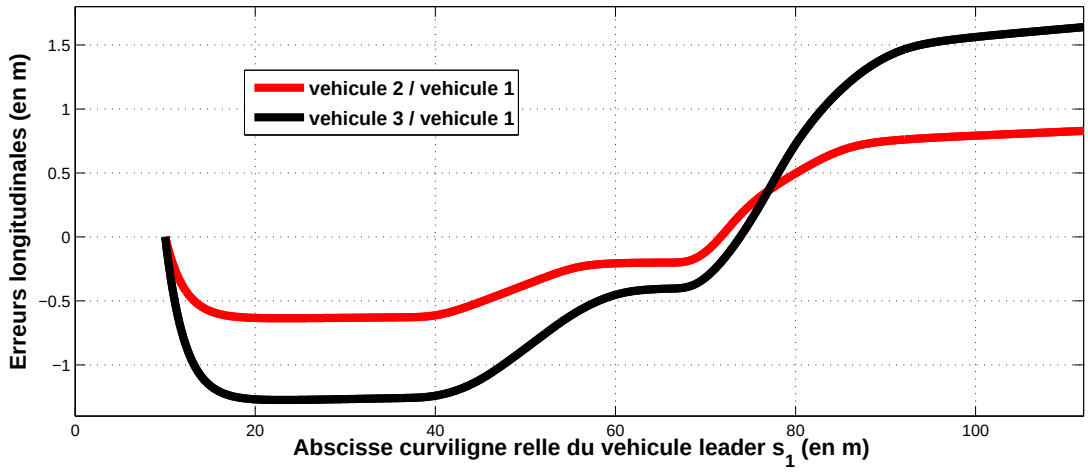


FIG. 3.5 – Performances longitudinales d'un convoi de 3 véhicules

3.3.3 Évaluation hors-ligne des facteurs d'échelle

En vue d'une représentation optimale du phénomène de distorsion, le récepteur RTK-GPS, qui est le capteur à disposition le plus précis pour estimer des longueurs du monde réel, va de nouveau être utilisé. En accord avec la modélisation du facteur d'échelle, formulée par l'équation (3.12), on souhaite que ce dernier soit évalué le plus ponctuellement possible. Jusqu'à présent, lors de l'analyse empirique en section 3.3.1, une représentation discrète du facteur d'échelle a été proposée, en évaluant de manière indépendante sa valeur moyenne sur des portions de la trajectoire découpée selon des intervalles de $2m$ de long (voir Fig. 3.1). On constate que si on diminue cette longueur, le bruit présent tant sur les données de localisation par vision que sur celles mesurées par le récepteur RTK-GPS, va se répercuter et altérer l'estimation des facteurs d'échelle. En plus de cette limitation, qui conduit à masquer les variations les plus locales, cette technique est contestable car les calculs reposent uniquement sur les données de postion aux extrémités des portions de $2m$ sur la trajectoire, délaissant ainsi la majeure partie de l'information puisque les positions contenues dans ces intervalles ne sont pas utilisées.

Afin d'estimer le facteur d'échelle de façon plus ponctuelle, une nouvelle stratégie est proposée. Comme précédemment (en section 3.3.1) on souhaite obtenir une représentation discrète du facteur d'échelle, c'est-à-dire évaluer un ensemble de facteurs λ_i effectifs sur des intervalles $\Gamma^v_{\lambda_i}$ qui forment un recouvrement disjoint de la trajectoire, voir la Fig. 3.6. L'effet du bruit sur les données de localisation peut être estompé de la manière suivante. Plutôt que d'évaluer directement chaque facteur d'échelle λ_i par le ratio des longueurs des intervalles $\Gamma^v_{\lambda_i}$ et Γ_{λ_i} des trajectoires vision et réelle, on exploite l'intégralité des données de localisation par vision et leur correspondance dans le référentiel réel. Pour cela, la valeur λ_i d'un facteur d'échelle est déterminée à partir de la comparaison de distances (monde vision / monde réel) sur un ensemble de portions (et non une unique portion) de la trajectoire. Ces portions, dont le nombre peut être

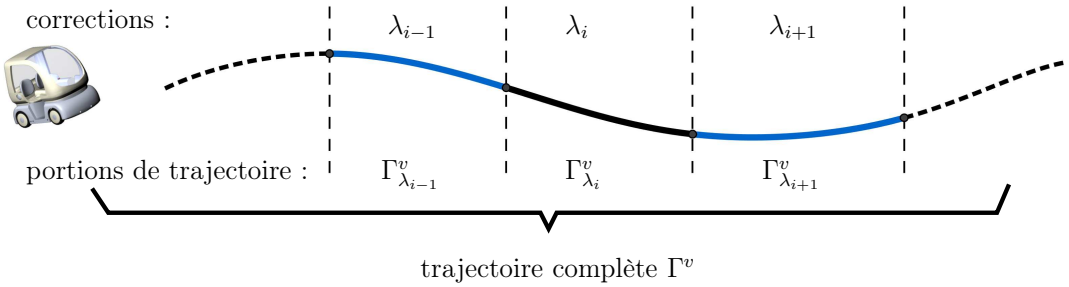


FIG. 3.6 – Découpage de la trajectoire en tronçons

élevé, peuvent se chevaucher et être de taille différente. Elles constituent la base de données à partir de laquelle sont estimés les facteurs d'échelle. On note par la suite Γ^v_k ces portions et Δs^v_k leur longueur d'arc dans le monde vision. Γ_k et Δs_k sont les mêmes données dans le monde réel. L'ensemble de facteurs d'échelle optimal sur la trajectoire est celui qui permet de minimiser un critère qui exprime la relation entre les distances Δs_k du monde réel et les distances correspondantes Δs^v_k du monde vision en fonction des facteurs λ_i.

Description des étapes La procédure d'évaluation se décompose en trois étapes de la façon suivante :

• On détermine tout d'abord les tronçons de trajectoire $\Gamma^v_{\lambda_i}$ sur lesquels les facteurs d'échelle λ_i que l'on souhaite évaluer sont effectifs. D'après les remarques précédentes, il serait préférable de choisir pour ces tronçons une taille $\Delta s^v_{\lambda_i}$ la plus petite possible. La limite est imposée par les incertitudes sur les mesures de distance. Nous avons observé que pour obtenir une estimation répétable des facteurs d'échelle, il faut se limiter à $\Delta s^v_{\lambda_i} = 0.5m$. On rappelle que la trajectoire Γ^v est modélisée par des courbes paramétrées lisses (voir section 2.3.1) et ainsi la longueur curviligne du $i^{ème}$ tronçon s'obtient facilement par le calcul $\Delta s^v_{\lambda_i} = \int_{\Gamma^v_{\lambda_i}} ds^v$.

• On détermine ensuite les portions Γ^v_k sur lesquelles l'ajustement des facteurs λ_i est optimisé. Pour obtenir des résultats satisfaisants, c'est-à-dire qui réduisent efficacement l'effet du bruit, il faut considérer un grand nombre de portions et aussi différentes tailles pour ces dernières. Dans la pratique on a choisi des portions dont la longueur d'arc vaut $0.5m$, $2m$, $10m$ et $40m$ afin que les valeurs λ_i soient représentatives du facteur d'échelle tant sur des petites que sur des grandes distances. Pour simplifier la formulation du critère d'optimisation, défini au point suivant, on suppose que la formule (3.40) est vérifiée.

$$\Delta s^v_k = \sum_{j \in J_k} \Delta s^v_{\lambda_j} \tag{3.40}$$

L'ensemble des indices J_k correspond aux tronçons de trajectoire $\Gamma^v_{\lambda_i}$ couverts par les

$k^{ièmes}$ mesures Δs_k^v et Δs_k.

Remarques :

- Pour que la formule (3.40) soit vérifiée, il faudrait que pour chaque portion Γ_k^v, les points initial et final correspondent respectivement au point initial d'un tronçon $\Gamma_{\lambda_{i_1}}^v$ et au point final d'un tronçon $\Gamma_{\lambda_{i_2}}^v$. Comme ce n'est bien sûr pas le cas, l'expression (3.40) de Δs_k^v utilise une longueur partielle pour les tronçons $\Gamma_{\lambda_{i_1}}^v$ et $\Gamma_{\lambda_{i_2}}^v$.

- Afin d'établir une correspondance entre les distances Δs_k^v et Δs_k, les positions réelles sont interpolées aux mêmes dates que les positions fournies par l'algorithme de vision.

• Les facteurs d'échelle locaux λ_i sont finalement obtenus par la méthode des moindres carrés en minimisant le critère (3.41) :

$$\sum_k \left(\Delta s_k - \sum_{j \in J_k} \lambda_j . \Delta s_{\lambda_j}^v \right)^2 \tag{3.41}$$

Résultats Les résultats obtenus avec cette stratégie sont tout d'abord comparés "visuellement" à ceux obtenus avec la précédente méthode proposée dans la section 3.3.1. Comme le montre la Fig. 3.7, les facteurs d'échelle le long de la trajectoire utilisée pour présenter l'analyse empirique ont été estimés de façon plus ponctuelle sur des tronçons de $0.5m$ sans être perturbés par le bruit sur les données de localisation. Cependant il est difficile de discerner si les oscillations sont l'expression du bruit ou représentent les variations courantes du facteur d'échelle.

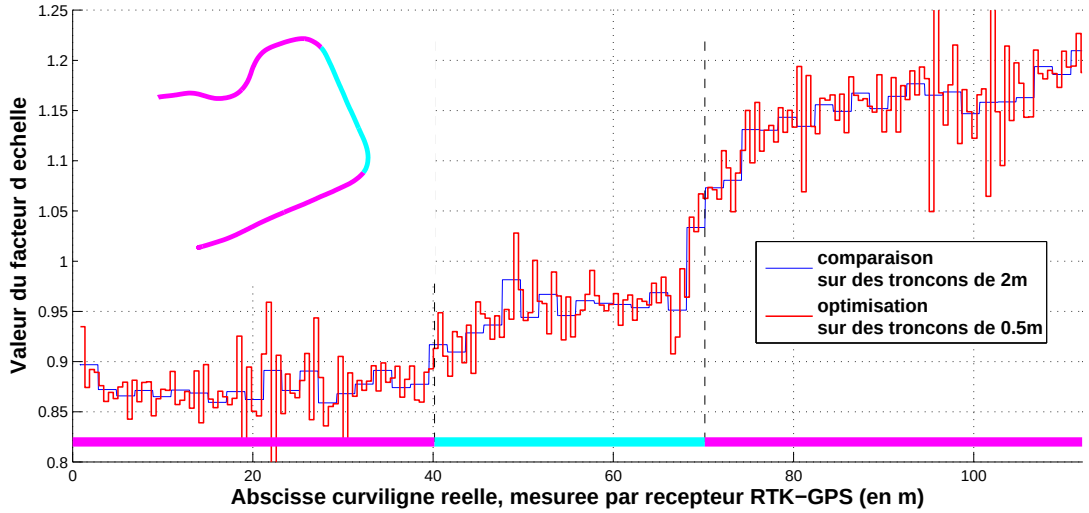


FIG. 3.7 – Estimation hors-ligne des facteurs d'echelle le long de la trajectoire

Pour quantifier les bénéfices de la stratégie hors-ligne, des simulations exploitant

les résultats fournis par les deux méthodes d'évaluation des facteurs d'échelle ont été réalisées. On souhaite comparer les valeurs d'abscisse curviligne réelle obtenues à partir de la localisation vision lorsqu'un véhicule se déplace le long d'une trajectoire selon que l'on utilise les facteurs d'échelle fournis par l'approche de la section 3.3.1 (ratio de longueurs sur des portions de $2m$) ou par la méthode des moindres carrés sur des tronçons de $0.5m$. L'évolution théorique du facteur d'échelle est la représentation avec des segments de droite illustrée Fig. 3.4. Afin de refléter les conditions réelles d'expérimentation, le véhicule simulé se déplace à la vitesse de $1m.s^{-1}$, et enregistre des données de localisation à une fréquence de $10Hz$. Des bruits d'écart-type $\sigma_{gps} = 0.01m$ et $\sigma_v = 0.02m$ ont été ajoutés pour émuler respectivement les données de localisation fournies par un récepteur RTK-GPS et par l'algorithme de vision. Ces données nous permettent alors d'évaluer les facteurs d'échelle à l'aide des deux méthodes, puis de calculer l'abscisse curviligne réelle à partir de l'abscisse vision (pour évaluer uniquement la représentativité du facteur d'échelle, l'abscisse curviligne réelle est calculée à partir de l'abscisse curviligne vision non bruitée). Les erreurs d'estimation sont illustrées sur la Fig. 3.8. On constate que l'estimation fournie par optimisation conduit localement à des variations dont l'amplitude est plus faible. Lors d'expérimentations en convoi l'entrée de commande longitudinale sera donc plus lisse si l'on utilise cette méthode (On note sur la Fig. 3.8 une erreur statique sur l'estimation de l'abscisse avec les facteurs d'échelle calculés à partir de portions de $2m$, due uniquement à l'initialisation de la méthode. Cette erreur statique n'est cependant pas préjudiciable pour la commande longitudinale d'un convoi, puisque seuls les écarts d'abscisse curviligne sont injectés dans les lois de commande).

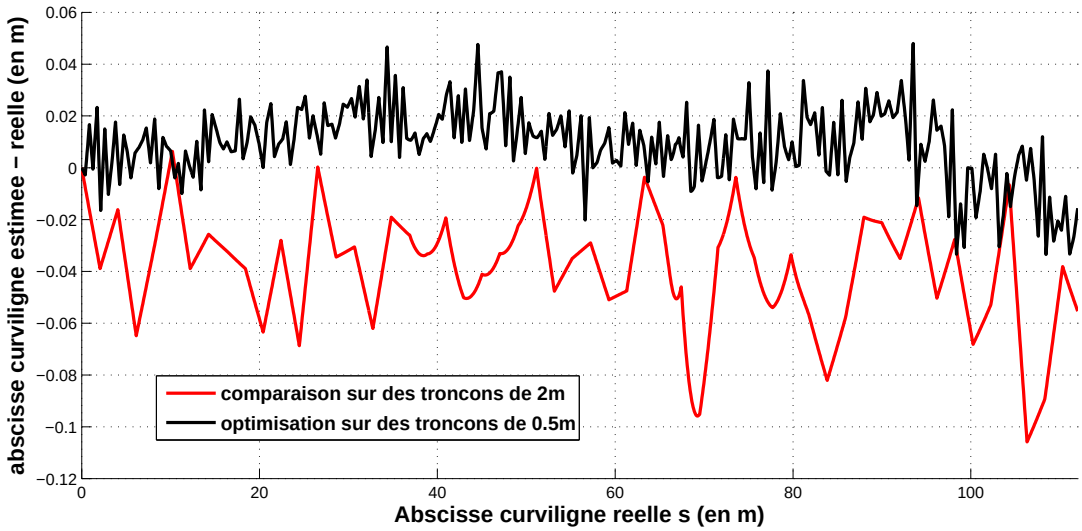


FIG. 3.8 – Comparaison des méthodes hors-ligne

Le comportement d'un convoi de trois véhicules a finalement été simulé, dans le cas où l'abscisse curviligne reportée dans les lois de commande était obtenue par vision

et corrigée par le jeu de facteurs d'échelle présenté dans cette section. Pour les deux véhicules suiveur, les erreurs longitudinales par rapport au véhicule leader estimées par cette approche et comparées aux erreurs réelles calculées par le simulateur, sont reportées sur la Fig. 3.9. Les résultats obtenus sont très satisfaisants puisque l'erreur n'excède pas $3cm$ pour les deux véhicules. Par la suite cette procédure de calcul de

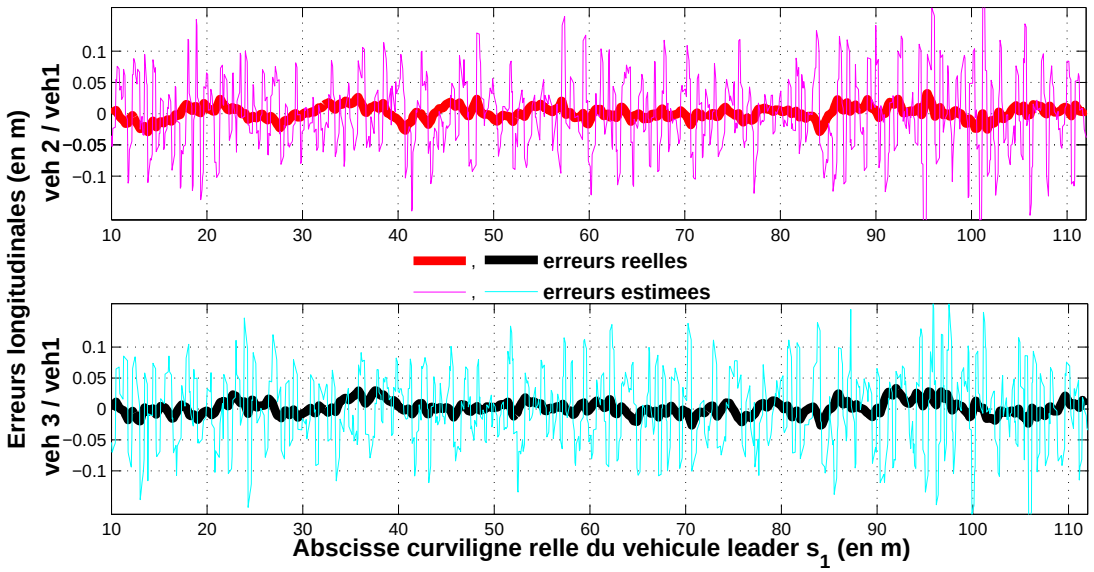


FIG. 3.9 – Simulation des performances longitudinales d'un convoi de trois véhicules

facteurs d'échelle servira de méthode de référence pour les autres stratégies présentées dans les sections 3.4 et 3.5 à venir et sera désignée par la dénomination de procédure hors-ligne d'estimation des facteurs d'échelle.

Conclusion Une première solution au problème posé par les distorsions a été proposée. Il a été montré qu'en exploitant les données de localisation fournies par un récepteur RTK-GPS, il était possible d'associer à la trajectoire exprimée dans le monde vision un jeu de facteurs d'échelle locaux pertinent qui rend alors possible une navigation en convoi précise, à partir d'une localisation des véhicules par vision monoculaire. Ce résultat est déjà encourageant mais reste insatisfaisant pour plusieurs raisons.

D'une part l'utilisation de récepteurs RTK-GPS n'est pas adaptée à la ville, puisque les signaux en provenance des satellites sont facilement masqués par les hautes structures urbaines. L'enregistrement de données GPS le long de certaines trajectoires ne sera donc pas possible, empêchant l'évaluation du jeu de facteurs d'échelle.

Sur un autre plan, pour apporter un maximum de liberté au guidage de véhicules en convoi, il serait très attrayant que celui-ci soit réalisé sans apprentissage préalable de la trajectoire. On pourrait ainsi envisager une conduite en convoi selon un mode manuel (défini en section 2.3), très flexible et pratique pour accomplir des opérations de maintenance ou des changements imprévus d'itinéraire. Dans [Mouragnon 06], les travaux

présentés permettent de réaliser en temps réel la reconstruction 3D de l'environnement et de procéder en parallèle à la localisation du véhicule. La qualité de la reconstruction 3D est similaire à celle obtenue par une stratégie hors-ligne comme celle utilisée dans cette thèse [Royer 05]. Par conséquent, pour pouvoir exploiter par la suite les travaux de [Mouragnon 06], et guider une flotte de véhicules par vision en mode manuel, il faudrait être capable d'évaluer en-ligne les distorsions le long de la trajectoire.

Ces raisons justifient l'étude de stratégies de correction des distorsions en-ligne, basées sur des capteurs adaptés au milieu urbain.

3.4 Stratégie de correction en-ligne : approche télémétrique

Pour cette approche, les données métriques du récepteur RTK-GPS sont remplacées par l'utilisation d'un unique télémètre laser monté sur le deuxième véhicule de la flotte. Le choix de ce capteur est justifié car il fonctionne parfaitement en condition urbaine et il est capable de fournir des mesures de distance avec une précision centimétrique. Par ailleurs il n'introduit pas de coût supplémentaire puisque de toute façon il est indispensable lors d'expérimentations en conditions réelles pour des questions de sécurité, afin de superviser les situations de collision avec des obstacles imprévus (personnes, autres véhicules, etc.). La stratégie se décompose en deux étapes détaillées dans les deux sous-sections qui suivent.

3.4.1 Suivi du véhicule de tête

L'arrière du véhicule de tête est tout d'abord isolé parmi les données télémétriques lues par le deuxième véhicule : une zone de recherche est définie autour de la position obtenue à l'itération précédente et ensuite, selon le type de véhicule de tête (c'est-à-dire CyCab ou RobuCab), une ellipse ou un segment de droite sont recherchés parmi les points de la zone, en utilisant la méthode des moindres carrés. Dans le cas de l'ellipse, l'approche proposée par A. Fitzgibbon [Fitzgibbon 99] est exploitée. Le détail de la méthode est présenté à l'annexe C. Enfin, la position du centre de l'essieu arrière O_1 du véhicule de tête est déduite des paramètres obtenus pour l'ellipse ou le segment de droite. O_1 est représenté en rouge sur les Fig. 3.10 et 3.11.

La distance euclidienne exprimée dans le monde métrique entre le véhicule de tête et son premier suiveur est donc connue à la fréquence du télémètre laser, c'est-à-dire $60Hz$. Cette même distance, exprimée dans le monde vision, n'est disponible qu'à la fréquence d'échantillonnage de la commande (c'est-à-dire $10Hz$), puisqu'à cet instant les états des véhicules considérés dans la commande longitudinale sont actualisés (voir section 4.2). Par conséquent, on n'utilise que les données télémétriques les plus récentes au moment du calcul de la commande, que l'on va comparer aux données vision disponibles à cet instant.

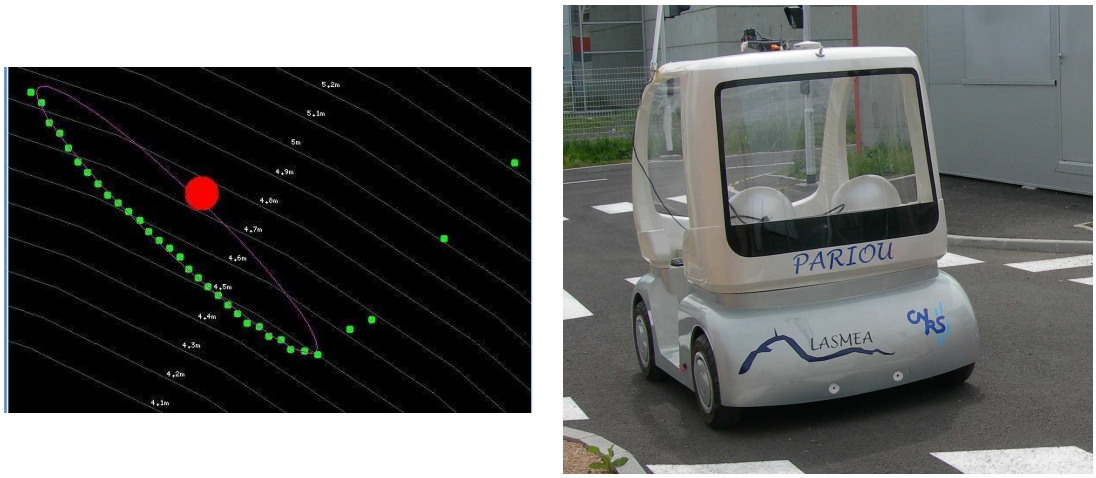


FIG. 3.10 – Arrière du CyCab sur les données télémétriques

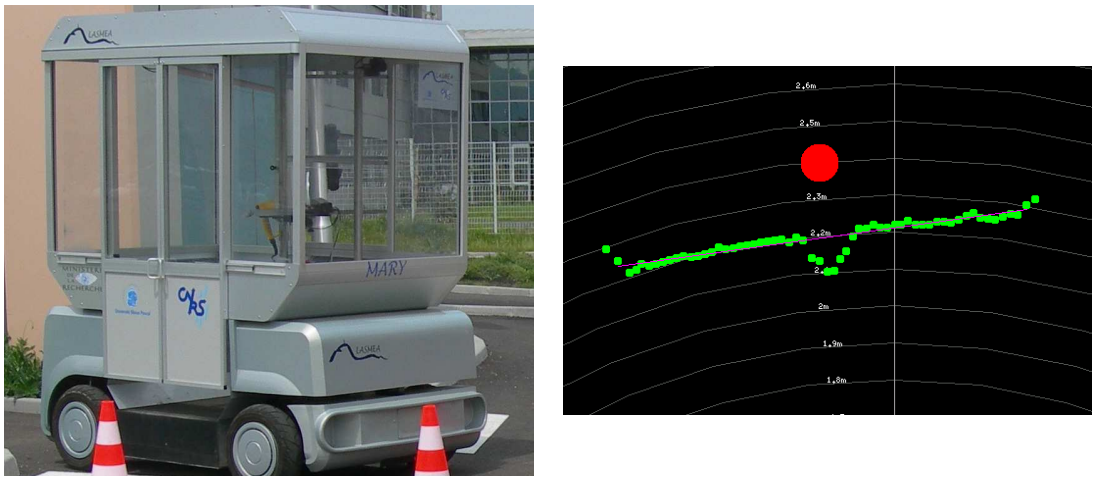


FIG. 3.11 – Arrière du RobuCab sur les données télémétriques

3.4.2 Evaluation des facteurs d'échelle

Comme pour la procédure hors-ligne, la stratégie s'appuie toujours sur la comparaison de distances évaluées parallèlement dans le référentiel métrique réel et dans son homologue virtuel et les facteurs d'échelle le long de la trajectoire sont calculés via la minimisation d'un critère. Cette stratégie présente cependant certaines différences majeures. Il s'agit, en premier lieu, d'une stratégie locale : les facteurs d'échelle sur chaque portion de la trajectoire sont calculés successivement, en utilisant uniquement les mesures disponibles dans un voisinage de cette portion de trajectoire. D'autre part, l'utilisation de distances obtenues via un télémètre réduit la variété dans les données prises en compte par le critère d'optimisation : la régulation longitudinale des véhicules impose que la valeur des distances télémétriques soit proche de $d^\star$ alors que des intervalles de différentes longueurs étaient considérés dans le cas hors-ligne. Pour présenter le principe de cette stratégie on introduit les notations ci-dessous. Certaines d'entre elles ont déjà été définies lors de la présentation de la stratégie hors-ligne.

Notations

• La trajectoire Γ^v est toujours la trajectoire exprimée dans le monde vision. Elle est à nouveau décrite comme la réunion de tronçons de trajectoire disjoints $\Gamma^v_{\lambda_j}$, associés chacun à une valeur de facteur d'échelle λ_j (Fig. 3.12).

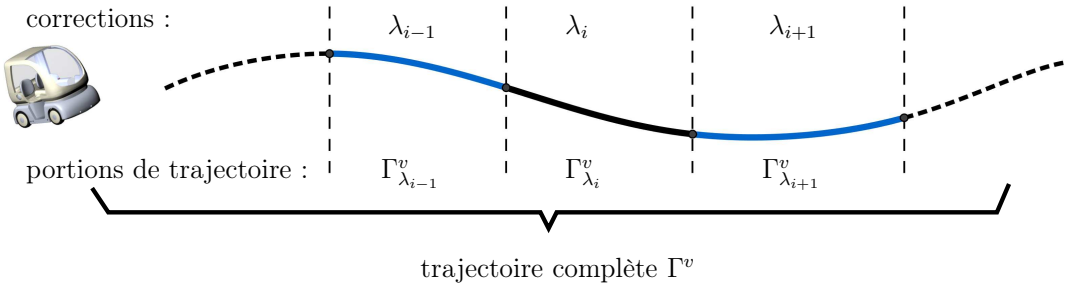


FIG. 3.12 – Découpage de la trajectoire en tronçons

• On note Γ^v_k la portion de trajectoire séparant les deux véhicules de tête lorsque la $k^{ème}$ donnée télémétrique a été stockée. Au vu des contraintes mécaniques qui lient les véhicules, on peut faire l'hypothèse que Γ^v_k est la projection du $k^{ème}$ tir télémétrique sur la trajectoire Γ^v. En effet, comme le montre la Fig. 3.13, la distance désirée entre deux véhicules consécutifs dans un convoi est trop courte pour qu'il en soit autrement.

• En opposition à la stratégie hors-ligne, les données recueillies dans le monde réel sont des distances euclidiennes (et non curvilignes). On note d_k la $k^{ème}$ distance enregistrée à partir des données télémétriques et d^v_k la même distance évaluée dans le monde vision à partir des données de localisation des deux véhicules. Cependant, l'objectif est toujours de corriger les mesures d'interdistance curviligne fournies par les algorithmes

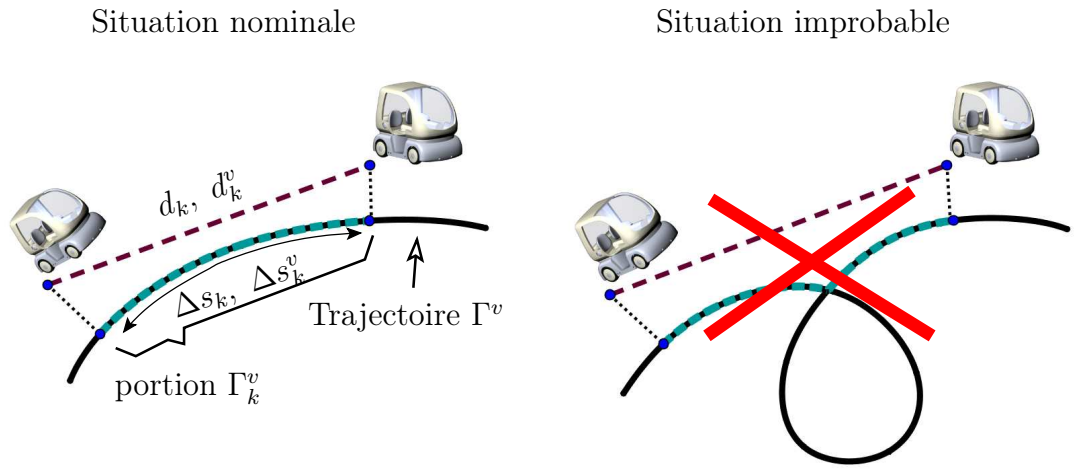


FIG. 3.13 – Γ_k^v est la projection de d_k sur Γ^v

de vision, dans le but de ne pas dégrader les performances de la commande longitudinale. Par conséquent, il serait plus naturel de s'appuyer, comme pour la stratégie hors-ligne, sur un jeu de données correspondant à des distances curvilignes. On note Δs_k^v la $k^{ème}$ distance curviligne entre les deux véhicules de tête évaluée dans le monde vision, et on propose d'estimer cette même distance dans le monde réel par le simple calcul $\Delta s_k = (d_k/d_k^v).\Delta s_k^v$. En effet, puisque l'on a admis que les distorsions étaient localement homogènes, il est naturel de faire l'hypothèse que le ratio des distances curvilignes et des distances euclidiennes est identique. En conséquence, la distance curviligne Δs_k entre les deux premiers véhicules, exprimée dans le monde réel, peut bien être évaluée à partir des données d_k, d_k^v et Δs_k^v disponibles à chaque itération.

Détails de l'algorithme Les corrections sont calculées de manière complètement indépendante comme suit. Lorsque la $j^{ème}$ portion de trajectoire $\Gamma_{\lambda_j}^v$ vient juste d'être dépassée par le second véhicule, on considère que l'on dispose de suffisamment d'information pour déclencher le calcul de la $j^{ème}$ correction λ_j. Les données télémétriques qui vont être utilisées sont celles dont la projection sur la trajectoire intersecte la portion à corriger (voir Fig. 3.14). Les indices de ces données télémétriques sont regroupés dans l'ensemble I_j^{mes} :

$$I_j^{mes} = \{k \ / \ \Gamma_{\lambda_j}^v \cap \Gamma_k^v \neq \emptyset\} \tag{3.42}$$

On voit sur la Fig. 3.14 que les tirs télémétriques couvrent largement le tronçon $\Gamma_{\lambda_j}^v$. Sur une aussi longue distance, on ne peut clairement pas assimiler le facteur d'échelle à une constante : la valeur locale du facteur, que l'on souhaite associer au court tronçon $\Gamma_{\lambda_j}^v$, serait alors occultée par un filtrage moyen sur une distance qui serait, dans le meilleur des cas, au moins égale à la consigne de distance longitudinale $d^\star$. Par expérience, l'évolution du facteur d'échelle n'est pas très élevée et une description du facteur d'échelle à l'aide d'un polynôme de degré deux est assez réaliste. Cette hypothèse est confortée par le fait que la variation du facteur d'échelle est liée à la courbure de la

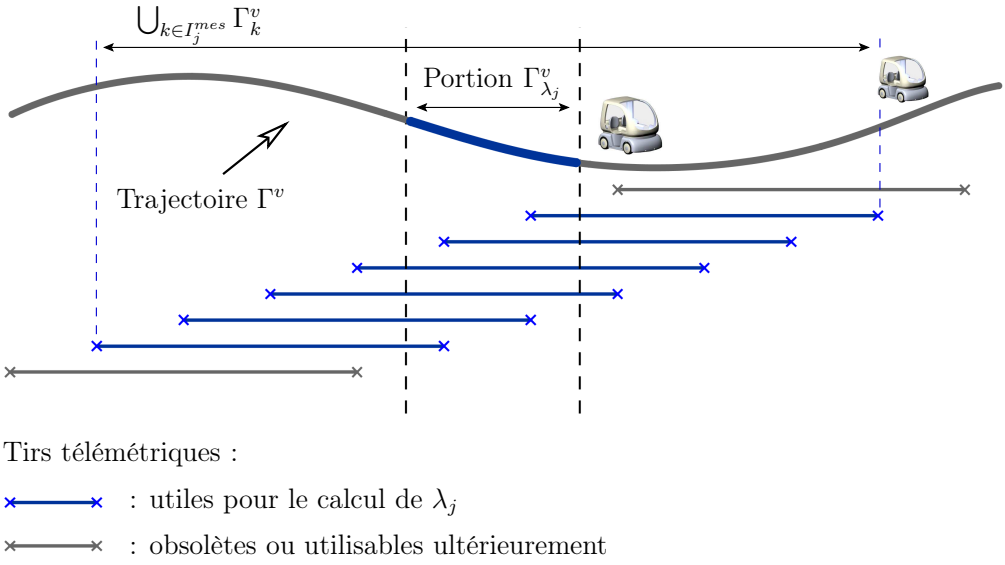


FIG. 3.14 – Mesures télémétriques retenues pour le calcul de la correction locale λ_j

trajectoire (voir la section 3.3.1), qui est limitée au vu des contraintes mécaniques imposées aux véhicules réels. Ce polynôme, noté $\breve{\lambda}_j(s^v)$, est défini sur la zone couverte par les tirs télémétriques retenus ($\cup_{k\in I_j^{mes}}\Gamma_k^v$) comme suit :

$$\breve{\lambda}_j(s^v) = a_j.s^{v2} + b_j.s^v + c_j \tag{3.43}$$

Les coefficients du polynôme sont évalués par la méthode des moindres carrés en minimisant le critère (3.44).

$$\sum_{k\in I_j^{mes}} \left(\Delta s_k - \int_{\Gamma_k^v} \breve{\lambda}_j(s^v) ds^v \right)^2 \tag{3.44}$$

La correction locale λ_j est ensuite calculée en moyennant les valeurs du polynôme $\breve{\lambda}_j(s^v)$ le long du tronçon concerné $\Gamma_{\lambda_j}^v$:

$$\begin{aligned} &\lambda_j = (\Delta s_{\lambda_j}^v)^{-1} \textstyle\int_{\Gamma_{\lambda_j}^v} \breve{\lambda}_j(s^v) ds^v \text{ avec} \\ &\Delta s_{\lambda_j}^v = \textstyle\int_{\Gamma_{\lambda_j}^v} ds^v \end{aligned} \tag{3.45}$$

Le fonctionnement de l'algorithme est illustré sur les Fig. 3.15 et 3.16. L'évolution du facteur d'échelle est celle déjà présentée sur la Fig. 3.4 et rappelée ici en trait noir en prenant cette fois pour abscisse s^v et non plus s. Elle est modélisée par des segments de droite dont la jonction forme une fonction continue sur l'ensemble de la trajectoire. Sur la Fig. 3.15 sont représentées les étapes de calcul permettant d'obtenir la valeur du facteur

d'échelle en trois points distincts de la trajectoire. Les mesures télémétriques utilisées au cours du calcul sont symbolisées par des segments de droite bleus qui matérialisent l'abscisse curviligne vision des deux véhicules de tête (s_1^v et s_2^v). Puis, pour les trois calculs, les valeurs des polynômes $\breve{\lambda}_i(s^v)$ sont retranscrites en magenta sur leur intervalle de définition, et la valeur de la correction que l'on retient finalement est indiquée en rouge. La Fig. 3.16 présente avec le même code couleur les résultats de correction finaux lorsque l'ensemble de la trajectoire a été parcouru.

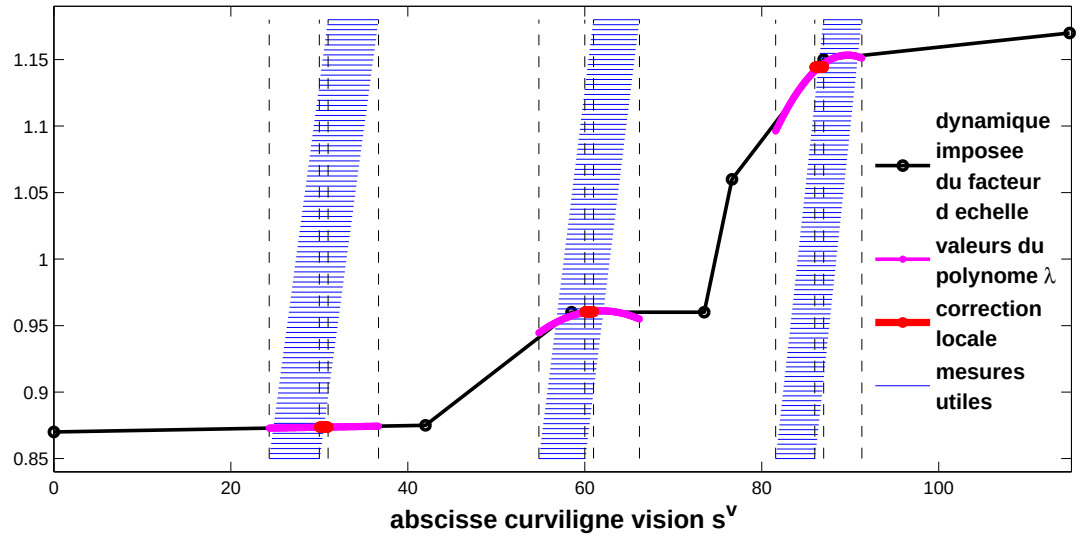


FIG. 3.15 – Illustration du calcul du facteur d'échelle en trois points de la trajectoire

Les performances quantitatives de cette stratégie en ligne sont étudiées par le biais de simulations à la section 3.6. Cependant, sur son principe même, cette stratégie présente plusieurs inconvénients que l'on peut lister dès à présent. Tout d'abord elle n'est pas très pratique car elle requiert un algorithme de suivi de véhicule pour acquérir des distances dans le monde réel. Des contraintes sur les conditions de fonctionnement de l'approche en découlent. D'une part, pour que le suivi par télémètre du véhicule de tête renvoie des mesures de distance avec une précision satisfaisante ($\leq 5cm$), la distance entre les deux véhicules doit nécessairement être inférieure à une limite. La distance de consigne inter-véhicule $d^\star$ ne peut donc être choisie que sur un intervalle restreint, dépendant du matériel utilisé (télémètre) et de la qualité de l'algorithme de suivi. D'autre part, la vitesse des véhicules doit être limitée afin d'assurer que sur une portion $\Gamma^v_{\lambda_j}$ de longueur raisonnable (on utilise des arcs de $10cm$ à $40cm$ pour pouvoir représenter finement le facteur d'échelle, voir section 3.6) il y aura suffisamment d'informations fournies par la localisation vision et le télémètre laser pour obtenir une valeur λ_j représentative. Finalement le schéma de communication au sein du convoi est légèrement plus complexe : puisque les corrections sont "détenues" par le deuxième véhicule, il est nécessaire que ce dernier les partage en temps réel avec tous les autres

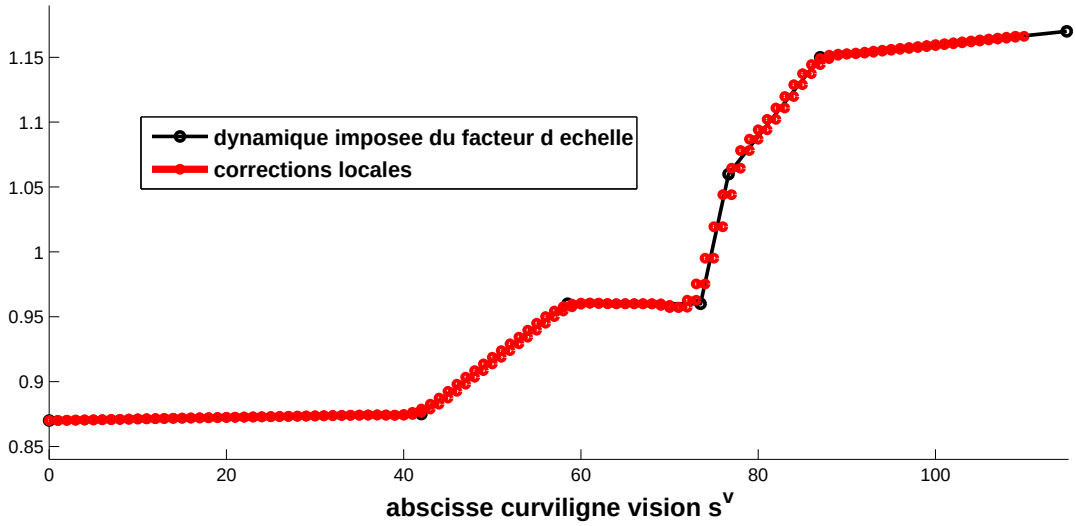


FIG. 3.16 – Illustration des corrections sur l'ensemble de la trajectoire

véhicules, accompagnées de la position du véhicule leader puisqu'il est le seul à pouvoir le positionner avec précision sur la trajectoire : l'abscisse curviligne du véhicule de tête est estimée à partir du ratio des distances euclidiennes courantes d et d^v exprimées dans les deux mondes et des abscisses curvilignes mesurées sur la trajectoire vision au moment du calcul : $s_1 = s_2 + (d/d^v).(s_1^v - s_2^v)$. En phase nominale, la flotte est donc référencée sur le véhicule leader mais s'appuie sur les informations de son suiveur immédiat pour être commandée.

Toutes ces raisons motivent ainsi la recherche d'une nouvelle stratégie, palliant les défauts ci-dessus.

3.5 Stratégie de correction en-ligne : approche odométrique

Une approche alternative, s'appuyant uniquement sur des données odométriques, est considérée dans la suite. Elle est basée sur un observateur et le modèle d'évolution du véhicule (1.50).

3.5.1 Modèles d'évolution : monde réel / virtuel

Pour cette approche, la mesure de référence dans le monde métrique qui sera utilisée pour évaluer des facteurs d'échelle locaux est la vitesse linéaire v_i fournie par les odomètres. En effet, puisque l'on ne désire pas faire intervenir de capteurs supplémentaires, les seules données définies dans le référentiel métrique que l'on peut recueillir ne peuvent provenir que des odomètres.

On propose de caractériser le phénomène de distorsion à travers son effet sur le comportement du véhicule, perçu dans le référentiel vision. Au vu des équations (3.15) et

(3.17), on a $y_i^v.k^v(s_i^v) = y_i.k(s_i)$. Ensuite l'équation (3.16) assure $\tilde{\theta}_i^v = \tilde{\theta}_i$. En reportant ces deux éléments dans la première composante du modèle cinématique (1.50), on obtient l'équation (3.46-gauche). La formule (3.46-droite) est ensuite donnée par la modélisation de facteur d'échelle (3.12) qui lie $\dot{s}_i$ et $\dot{s}_i^v$.

Monde métrique réel	Monde virtuel vision
$\dot{s}_i = \dfrac{v_i.\cos\tilde{\theta}_i^v}{1 - y_i^v.k^v(s_i^v)}$	$\dot{s}_i^v = \dfrac{v_i.\cos\tilde{\theta}_i^v}{\lambda(s_i^v).(1\ -\ y_i^v\,k^v(s_i^v))}$

(3.46)

L'intérêt de ces représentations est de décrire l'évolution des véhicules uniquement à partir de variables disponibles en cours d'expérimentation ; à savoir l'état du véhicule dans le monde virtuel, fourni grâce à l'algorithme de localisation par vision, ainsi que la vitesse linéaire et l'angle de braquage des roues avant dans le monde métrique, issus des odomètres.

3.5.2 Principe de la méthode

L'objectif ici est de concevoir un observateur pour estimer le paramètre λ à partir d'un modèle d'évolution du véhicule. Puisque la commande latérale est conçue pour maintenir chaque véhicule sur la trajectoire, il est attendu que les variables y_i et $\tilde{\theta}_i$ restent faibles. Par conséquent le comportement longitudinal du véhicule est nécessairement plus significatif que le comportement latéral. L'information fournie par la première composante du modèle cinématique (1.50), décrite ici par (3.46), est donc nettement plus riche que pour les deux autres. C'est donc uniquement sur l'équation d'évolution de l'abscisse curviligne, ainsi que sur les données mesurées à partir de l'algorithme de localisation, que l'on se focalise.

Le choix du modèle est assez simple : puisque les mesures s'effectuent dans le référentiel vision, il est nécessaire de sélectionner un modèle exprimé dans le monde virtuel, soit donc (3.46-droite). Lorsque la variable à observer n'est pas déjà présente dans le vecteur d'état du système étudié, la méthode la plus classique consiste à l'y ajouter, avec sa loi d'évolution, puis à bâtir l'observateur sur une structure classique de type Luenberger. Malheureusement, pour la fonction de distorsion que l'on souhaite observer ici, il est impossible de proposer une expression analytique pour son évolution : c'est le résultat d'une procédure complexe d'optimisation (ajustement de faisceaux) qui est imprédictible. Il faut donc envisager d'autres structures pour construire l'observateur.

Conception de l'observateur On propose, comme dans [Lenain 07], de construire l'observateur comme une loi de commande, dont le but serait ici que la sortie du modèle d'évolution du véhicule exprimée dans le référentiel vision, converge vers les mesures réalisées dans ce même repère. Plus précisément, en copiant le modèle (3.46), on définit

le modèle d'observation comme suit :

$$\dot{\hat{s}}_i^v = \frac{v_i . \cos \tilde{\theta}_i^v}{u_i.(1 \, - \, y_i^v \, k^v(s_i^v))} \tag{3.47}$$

où $\hat{s}_i^v$ représente l'abscisse curviligne observée dans le monde virtuel vision, y_i^v, $\tilde{\theta}_i^v$ et $k^v(s_i^v)$ sont des quantités mesurées dans ce même référentiel, et u_i est la variable de commande qu'il reste à définir.

Le principe de l'observateur repose sur l'assertion suivante : si la variable de commande u_i du modèle d'observation peut être définie de sorte que la variable $\hat{s}_i^v$ du modèle d'observation (3.47) converge vers l'abscisse curviligne s_i^v mesurée dans le repère vision, alors au vu des équations (3.46) et (3.47), la commande u_i est représentative du facteur d'échelle local $\lambda(s_i^v)$ au niveau de l'abscisse curviligne vision s_i^v. En suivant cette idée, il est naturel de définir l'erreur d'observation ϵ comme suit :

$$\epsilon = (\hat{s}_i^v - s_i^v) \tag{3.48}$$

Puisque la dynamique de $\hat{s}_i^v$, est régie par une équation différentielle d'ordre un (équation (3.47)), on cherche alors à établir un profil de convergence exponentiel pour l'erreur ϵ. On souhaite que :

$$\dot{\epsilon} = \dot{\hat{s}}_i^v - \dot{s_i^v} = -K \cdot \epsilon, \text{ où } K \ \in \mathcal{R}^{+*}. \tag{3.49}$$

Cependant, la quantité $\dot{s_i^v}$, qui intervient dans l'expression de la dynamique de l'erreur (3.49) n'est pas une mesure a priori disponible. En effet, l'algorithme de localisation par vision ne fournit, à chaque itération, une mesure que de l'abscisse curviligne s_i^v exprimée dans le référentiel vision. Malgré le bruit enregistré sur les signaux de mesure, dont l'écart-type a été évalué à $2cm$ lors des expérimentations, il est possible d'accéder numériquement à une mesure réaliste de la dérivée $\dot{s_i^v}$ de l'abscisse curviligne dans le monde vision, en lissant les données à l'aide d'un filtre passe-bas à très courte constante de temps (le choix de ce paramètre est discuté en section 3.6). L'utilisation pratique d'un tel filtre, qui occasionne nécessairement un retard sur la valeur à estimer, est présentée à la section suivante 3.5.3. Pour l'instant on suppose que $\dot{s_i^v}$ est mesuré avec précision. Le profil désiré de convergence (3.49) peut alors facilement être imposé en proposant pour u_i l'expression suivante :

$$u_i = \frac{v_i . \cos \tilde{\theta}_i^v}{(\dot{s}_i^v - K.\epsilon)(1 \, - \, y_i^v . k^v(s_i^v))} \tag{3.50}$$

Singularités L'équation (3.50) permet d'estimer avec précision le facteur d'échelle local à l'abscisse curviligne s_i^v. Cependant cette équation présente plusieurs singularités :

• Une première singularité intervient si les valeurs de $\dot{s}_i^v$ sont proches de celles de $K.\epsilon$, c'est-à-dire si la vitesse du véhicule est comparable au produit du gain et de

l'erreur d'observation. A priori, puisque l'erreur d'observation est censée être faible, cette situation est rare. Cette singularité ne se rencontre que dans deux cas de figure :

- En premier lieu, à chaque démarrage / arrêt du véhicule, la valeur de vitesse (partant ou atteignant zéro selon le cas) va nécessairement croiser celle de l'erreur d'observation. Il convient donc de mettre en place un "chien de garde" qui inhibe l'estimation du facteur d'échelle dès que la vitesse du véhicule passe sous une valeur seuil qui dépend de K. Ce n'est pas une véritable limitation de l'algorithme, car le facteur d'échelle est défini par rapport à l'abscisse curviligne le long de la trajectoire, et donc si le véhicule est arrêté, la valeur du facteur d'échelle ne bouge pas.

- Cette même singularité peut d'autre part se produire si la localisation s_i^v du véhicule est sujette, à un moment donné, à une importante erreur. L'observation $\hat{s}_i^v$ serait très différente de la mesure s_i^v et le produit $K \cdot \epsilon$ pourrait être du même ordre de grandeur que la vitesse. Cependant ces erreurs larges sont identifiées au niveau de l'algorithme de localisation qui ne les transmet pas au module de commande : cette sorte de singularité ne peut donc pas apparaître dans l'équation (3.50) .

• Une dernière singularité apparaît dans le cas de figure : $1 - y_i^v.k^v(s_i^v) = 0$. D'un point de vue géométrique, cela correspond au cas où le point O_i situé au centre de l'essieu arrière des véhicules se trouve sur le centre de courbure à la trajectoire Γ^v. Cette condition, qui a déjà été discutée dans la présentation du modèle (1.50), est en pratique toujours évitée : de par les dimensions des véhicules urbains et des limites sur l'angle de braquage, le rayon de courbure $\frac{1}{k^v(s_i^v)}$ ne peut être inférieur à quelques mètres ($3m$ dans le cas des CyCab et RobuCab) alors que l'écart latéral y_i excède rarement $1m$.

Le principe de cette méthode est illustré Fig 3.17. La boucle d'observation, implémentée sur le véhicule leader permet de calculer, parallèlement à la boucle d'asservissement, les corrections à apporter. Ces corrections sont ensuite partagées avec l'ensemble de la flotte qui peut alors convertir dans le repère métrique les données de localisation initialement fournies dans le repère vision, ce qui permet aux lois de commande latérale et surtout longitudinale d'atteindre leur objectif.

3.5.3 Estimation de la vitesse curviligne dans le monde vision

Comme mentionné dans la section précédente 3.5.2, une valeur précise du facteur d'échelle, donnée par l'équation (3.50), requiert de connaître une mesure réaliste de la vitesse curviligne estimée dans le référentiel vision. On propose ici d'introduire un filtre de Butterworth passe-bas du premier ordre pour lisser l'estimée de la vitesse curviligne $\dot{s}_1^v$ (ici l'indice du véhicule est $i = 1$, car en pratique c'est le leader qui corrige le facteur d'échelle), puisque cette dernière est obtenue par une dérivée numérique sur les mesures s_1^v.

Dérivée numérique et filtrage : Soit T_e la période d'acquisition des images, et $(s_1^v)_{[k]}$ l'abscisse curviligne du premier véhicule fournie par la localisation vision à l'ins-

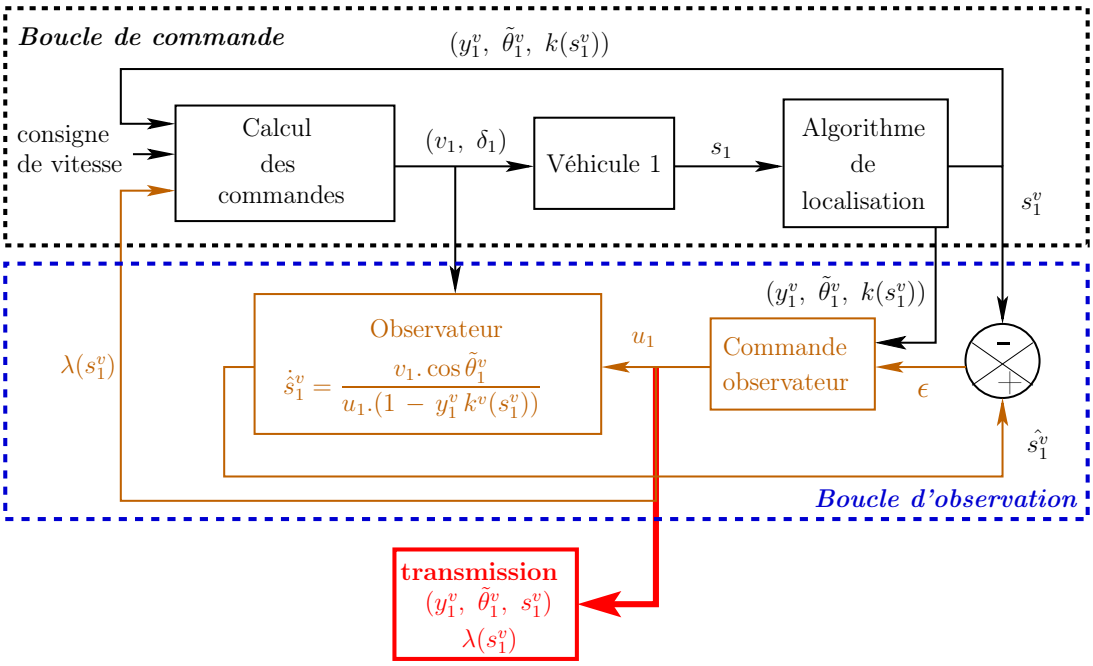


FIG. 3.17 – Schéma d'estimation par observation des corrections

tant $k.T_e$. Une estimée de sa vitesse curviligne $(\dot{s}_1^v)_{[k]}$ au même instant $k.T_e$ peut être obtenue en "dérivant" s_1^v à partir des deux dernières valeurs disponibles :

$$\left(\dot{s}_1^v\right)_{[k]} = \frac{\left(s_1^v\right)_{[k]} - \left(s_1^v\right)_{[k-1]}}{T_e} \tag{3.51}$$

Lorsque les mesures $(s_1^v)_{[k]}$ sont bruitées, l'estimée de la vitesse curviligne $(\dot{s}_1^v)_{[k]}$ obtenue par dérivée numérique (3.51) est alors très bruitée et peu représentative de la véritable valeur de la vitesse curviligne, si bien qu'on est dans l'obligation de filtrer cette estimée. Si on se tourne vers un filtre de Butterworth du premier ordre, alors l'estimée filtrée de la vitesse curviligne à l'instant kT_e, notée $\left(\dot{\hat{s}}_1^v\right)_{[k]}$ s'obtient à partir des estimées bruitées $(\dot{s}_1^v)_{[k]}$ et $(\dot{s}_1^v)_{[k-1]}$ par la relation suivante :

$$\left(\dot{\hat{s}}_1^v\right)_{[k]} = \alpha\left(\dot{\hat{s}}_1^v\right)_{[k-1]} + \frac{1-\alpha}{2}\left(\dot{s}_1^v\right)_{[k]} + \frac{1-\alpha}{2}\left(\dot{s}_1^v\right)_{[k-1]},\ \alpha \in [0;1] \tag{3.52}$$

Plus α est proche de zéro, plus l'estimée brute $(\dot{s}_1^v)_{[k]}$ est prise en compte dans le calcul de l'estimée filtrée $\left(\dot{\hat{s}}_1^v\right)_{[k]}$: cette dernière présente donc peu de retard par rapport à $(\dot{s}_1^v)_{[k]}$, mais peu de bruit a été éliminé. On a bien sûr les conclusions inverses si α est proche de 1.

Prise en compte du retard introduit : Pour estimer le retard occasionné par le filtrage de la vitesse curviligne $\dot{s}_1^v$, on observe le comportement du filtre lorsque le signal

d'entrée, noté ici S_{ig}, est une rampe de pente un et de valeur initiale nulle :

$$\Big(S_{ig}\Big)_{[k]} = k.T_e \tag{3.53}$$

En régime stabilisé, le retard T de la sortie du filtre $\hat{S}_{ig}$ par rapport à l'entrée est devient constant. On a donc :

$$\Big(\hat{S}_{ig}\Big)_{[k]} = \Big(S_{ig}\Big)_{[k]} + T = k.T_e + T \tag{3.54}$$

En effet, la valeur T représente l'offset sur la sortie, mais est aussi le retard temporel du filtre puisque la rampe a été fixée à un. En reportant (3.54) dans (3.52) on a alors :

$$k.T_e - T = \alpha\Big((k-1)T_e - T\Big) + \frac{1-\alpha}{2}k.T_e + \frac{1-\alpha}{2}(k-1).T_e \tag{3.55}$$

ce qui donne après résolution :

$$T = \frac{T_e}{2} \cdot \frac{1+\alpha}{1-\alpha} \tag{3.56}$$

A partir de l'estimation de ce retard, il est maintenant possible de retrouver une cohérence temporelle sur les variables présentes dans l'équation (3.50) qui donne le facteur d'échelle. L'obtention du facteur d'échelle le long de la trajectoire se fait ensuite de la façon suivante :

- Pour expliquer simplement la démarche, on considère de nouveau que les données vision sont parfaites. Dans ce cas, le filtrage de la vitesse curviligne n'est pas nécessaire (c.a.d. $\alpha = 0$) et l'estimation du facteur d'échelle $\lambda_{[k]}$ obtenue à l'instant $k.T_e$, s'exprime en fonction de l'état du véhicule $\big(X_1^v\big)_{[k]}$, de l'erreur d'observation $\epsilon_{[k]}$ et de l'estimation de la vitesse curviligne $\big(\dot{\hat{s}}_1^v\big)_{[k]}$ à l'instant $k.T_e$, soit encore en reprenant (3.50) :

$$\lambda_{[k]} = f\Big(\big(X_1^v\big)_{[k]},\ \epsilon_{[k]},\ \big(\dot{\hat{s}}_1^v\big)_{[k]}\Big) \tag{3.57}$$

$$f(X^v,\ \epsilon,\ \dot{s}^v) = \frac{v.\cos(\tilde{\theta}^v)}{(\dot{s}^v - K.\epsilon)(1 - y^v.k^v(s_i^v))} \tag{3.58}$$

  La valeur du facteur d'échelle $\lambda_{[k]}$ obtenue est alors active sur la portion $\big[(s_1^v)_{[k-1]};\ (s_1^v)_{[k]}\big]$.

  Remarque : Même en l'absence de filtrage, la formule (3.56) nous indique qu'un retard temporel, qui s'élève à $\frac{T_e}{2}$, est présent sur l'estimation de la vitesse curviligne vision $\big(\dot{\hat{s}}_1^v\big)_{[k]}$. Ce dernier n'est pas explicitement pris en compte dans l'estimation du facteur d'échelle, cependant on remarquera que l'intervalle $\Gamma^v_{\lambda_{[k]}}$ n'est pas centré autour de $(s_1^v)_{[k]}$ mais décalé sur une distance qui est celle parcourue par le véhicule pendant la période $\frac{T_e}{2}$. Ceci permettant de compenser d'une certaine manière le retard introduit.

– Si l'on considère maintenant la présence de bruit, alors $\alpha \neq 0$ et la valeur de T donnée par (3.56) peut être élevée. Il est alors important de tenir compte explicitement de ce retard dans l'observateur afin de préserver la cohérence temporelle des données dans la boucle d'observation et d'estimer avec précision la valeur du facteur d'échelle. Pour ce faire, la valeur estimée de la vitesse curviligne vision $\left(\dot{\hat{s}}_1^v\right)_{[k]}$ doit être comparée dans l'équation (3.50) aux valeurs d'état $\left(X_1^v\right)_{[k-r]}$ et d'erreur $\epsilon_{[k-r]}$ obtenues r itérations précédentes. La valeur utilisée pour r est simplement donnée comme suit :

$$r = E\left(\frac{1+\alpha}{2(1-\alpha)}\right) \tag{3.59}$$

où la fonction $E(\cdot)$ retourne la partie entière du nombre en entrée. On propose alors d'estimer, à l'instant $k.T_e$, le facteur d'échelle $\lambda_{[k]}$ par la formule suivante :

$$\lambda_{[k]} = f\Big(\left(X_1^v\right)_{[k-r]},\ \epsilon_{[k-r]},\ \left(\dot{\hat{s}}_1^v\right)_{[k]}\Big) \tag{3.60}$$

La valeur du facteur d'échelle $\lambda_{[k]}$ obtenue est alors active sur la portion $\left[(s_1^v)_{[k-r-1]};\ (s_1^v)_{[k-r]}\right]$.

3.6 Simulations

Dans cette section des résultats de simulation sont présentés afin de comparer les méthodes de correction en ligne et examiner la sensibilité de chacune en fonction de leurs principaux paramètres. Afin d'être au plus proche des conditions d'expérimentation, les batteries de tests ont été lancées avec des paramètres représentatifs des conditions réelles :

- Des facteurs d'échelle très proches de ceux calculés par la méthode hors-ligne (voir Fig. 3.7) ont été générés à partir de segments de droite dont la jonction forme une fonction continue sur l'ensemble de la trajectoire. Ils sont représentés sur la Fig. 3.4.

- Les données de localisation par vision sont fournies à la fréquence de $15Hz$, qui est la fréquence maximale de la caméra (variable en fonction de la luminosité) et qui coïncide aussi avec le double du temps de traitement ($30ms$) d'une image. Des données de localisation parfaites ainsi que des données bruitées sont considérées. Dans le dernier cas la valeur de l'écart type s'élève à $\sigma_v = 0.02m$ qui est une valeur réaliste pour l'algorithme de localisation par vision monoculaire considéré.

- On suppose que les données télémétriques sont fournies en même temps que les données de localisation. Comme pour les données vision on considère deux valeurs d'écart type sur les données : $\sigma_t = 0m$ dans le cas parfait et $\sigma_t = 0.01m$ pour refléter le bruit d'un suivi du véhicule de tête à l'aide du télémètre laser.

- L'écart type sur la lecture odométrique de la vitesse linéaire est de $\sigma_o = 0.015m.s^{-1}$.

Evaluation des méthodes Elle se fait par comparaison entre les distances curvilignes le long de la trajectoire simulées dans le monde métrique et les mêmes distances

simulées dans le monde vision puis corrigées par le facteur d'échelle calculé en ligne par la méthode que l'on souhaite évaluer. Les deux mesures ont été comparées sur plusieurs sections de la trajectoire de longueur $5m$ (dans le référentiel métrique) qui est la valeur de consigne $d^\star$ lors des expérimentations. La longueur de cette section paraît adéquate, car des longueurs plus petites ne permettraient pas d'évaluer l'intérêt de ces stratégies de correction en ligne pour des applications de guidage en convoi (où les inter-distances à mesurer sont de l'ordre de $d^\star$), et si on considérait des longueurs plus grandes, alors au vu des valeurs simulées pour le facteur d'échelle le long de la trajectoire, les résultats obtenus pourraient être sans signification. En effet, sur la première moitié de la trajectoire le facteur d'échelle est inférieur à un puis supérieur à un sur la seconde, ainsi l'erreur sur l'évaluation d'une section qui chevauche la première moitié s'annule partiellement dans le cas où cette même section chevauche également la deuxième moitié. En revanche des intervalles trop petits ($\leq 5m$) ne permettraient pas de mettre en valeur la mise en œuvre d'un convoi de véhicules où les interdistances peuvent être grandes. Enfin, les erreurs d'estimation sont quantifiées par la moyenne et l'écart type de l'erreur commise sur l'ensemble des sections sélectionnées tout au long de la trajectoire.

3.6.1 Optimisation locale avec les données télémétriques

Les résultats les plus significatifs sont répertoriés sur la Fig. 3.18 en fonction des paramètres influents qui sont : la vitesse de déplacement du convoi qui n'est autre que celle du véhicule leader v_1, la consigne de distance entre les véhicules $d^\star$, et la longueur d'arc Δs^v_λ des tronçons auxquels sont attachées les valeurs λ_j du facteur d'échelle. Certaines configurations ont par ailleurs été mises en valeur et les résultats d'optimisation correspondants sont illustrés sur la Fig. 3.19

	v_1 (m.s^{-1})	$d^\star$ (m)	Δs^v_λ (m)	erreur moyenne (10^{-3}m)		ecart-type (10^{-3}m)	
				$\sigma_v = 0m$	$= 0.02m$	$\sigma_v = 0m$	$= 0.02m$
				$\sigma_t = 0m$	$= 0.01m$	$\sigma_t = 0m$	$= 0.01m$
(1)	0.5	2	0.1	0.0769	4.0296	0.2068	2.9720
(2)	1	2	0.1	0.0767	5.4979	0.2062	3.9156
(3)	3	2	0.1	0.0847	8.1975	0.2312	5.9429
(4)	1	5	0.1	1.0057	3.0993	2.5935	2.3503
(5)	1	7	0.1	2.0748	2.8135	4.8512	4.5980
(6)	1	5	0.2	1.0163	2.6895	2.6168	2.7767
(7)	1	5	0.4	1.0390	2.6142	2.6618	2.4983
(8)	1	5	2	1.4577	2.4323	3.5311	3.3117

FIG. 3.18 – Influence des paramètres sur les performances de l'algorithme d'estimation en ligne du facteur d'échelle utilisant les données télémétriques

$\rightsquigarrow$ influence de la vitesse du leader v_1 : Comme la vitesse des véhicules doit être bornée en milieu urbain, les tests ont été limités aux valeurs de vitesse $v_1 \leq 3m.s^{-1}$ (la vitesse de pointe des CyCabs est de $4m.s^{-1}$). Les résultats (1)-(2)-(3) présentés sur

la Fig. 3.18 montrent clairement que ce paramètre n'a pas d'impact significatif sur la qualité de l'approximation locale dans le cas où des données parfaites sont injectées dans l'algorithme. On le constate aussi clairement sur les Fig. 3.19 a) et c), puis e) et g). Par contre, lorsque les données sont entachées de bruit, une vitesse faible est préférable, voir Fig. 3.19 b) et d). Cela est tout à fait logique : le nombre de mesures qui alimentent le critère de minimisation (3.44) à chaque itération est moindre quand la vitesse devient élevée. Les perturbations introduites par le bruit sont donc moins bien lissées et leur effet sur la qualité d'optimisation s'en fait ressentir. A ces vitesses, pour améliorer les performances, il faudrait pouvoir augmenter $d^\star$ et/ou Δs^v_λ, de manière à faire intervenir plus de mesures dans le critère (3.44). Cela peut être observé sur les résultats (4) à (8) de la Fig. 3.18 et est discuté ci-après.

$\rightsquigarrow$ influence de la consigne d'inter-distance $d^\star$: Dans une application pratique de conduite en convoi, il est attendu que les véhicules respectent une distance de sécurité (aussi une consigne $d^\star = 2m$ paraît une valeur minimale) tout en restant proches (sinon les bénéfices d'une conduite en convoi sont perdus, $d^\star = 7m$ paraît une valeur maximale). Les résultats présentés sur la Fig. 3.18 permettent de comparer les performances de l'approximation locale pour différentes valeurs de $d^\star$ variant dans cet intervalle $[2; 7]m$. Les conclusions sont complètement différentes suivant que les données sont bruitées ou non.

• Si les données vision ne sont pas bruitées, les meilleurs résultats sont obtenus pour la plus petite valeur de $d^\star$ (résultats (2)-(4)-(5) de la Fig. 3.18). Cela est tout à fait logique puisque les plus petites valeurs de $d^\star$ permettent d'estimer le polynôme $\lambda(s^v)$ sur une courte distance. Le degré deux choisi pour ce polynôme est alors suffisant pour que $\lambda(s^v)$ reflète convenablement les variations du facteur d'échelle. Ceci est clairement visible sur les Fig. 3.19 : la qualité de l'approximation locale lorsque $d^\star = 2m$ (Fig. 3.19 a) et c)) est meilleure que lorsque $d^\star = 5m$ (Fig. 3.19 e) et g)).

• A l'inverse, si les données vision sont entachées de bruit, les meilleurs résultats sont cette fois obtenus pour les grandes valeurs de $d^\star$ (toujours résultats (2)-(4)-(5) de la Fig. 3.18). On peut avancer deux justifications :

- Le nombre de données qui alimentent le critère de minimisation (3.44) à chaque itération est plus grand si $d^\star$ l'est aussi, ce qui a pour effet de lisser les perturbations introduites par le bruit.

- D'autre part, puisque la précision des capteurs (vision ou télémètre) est indépendante (ou presque indépendante) de la distance mesurée, une meilleure précision relative sera obtenue pour des mesures de distance longue. Si $d^\star$ est choisi grand, alors les mesures apparaissant dans le critère (3.44) sont plus longues, donc moins bruitées (en relatif) et donc il est attendu que le résultat de l'optimisation soit plus précis. Ces résultats sont illustrés sur la Fig. 3.19 : la qualité de l'approximation est améliorée lorsque la consigne de distance passe de $d^\star = 2m$ à $d^\star = 5m$, voir Fig. 3.19 b) et f) et Fig. 3.19 d) et h).

$\rightsquigarrow$ influence de la taille Δs^v_λ du tronçon à corriger : Ce dernier paramètre est logi-

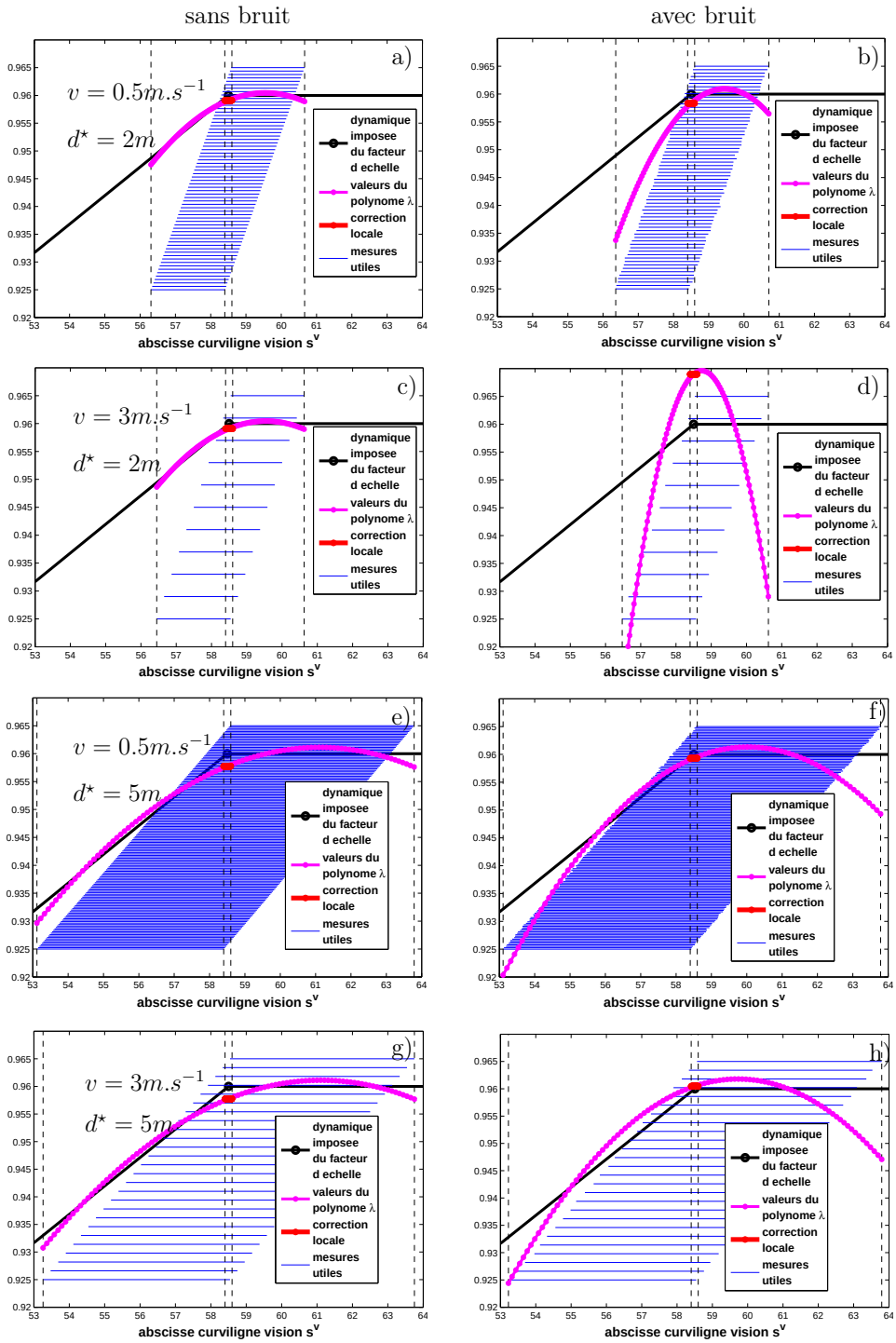


FIG. 3.19 – Influence des paramètres de l'estimation basée télémètre : quelques exemples

quement choisi de sorte que $\Delta s_\lambda^v < d^\star$. Il représente l'intervalle sur lequel le polynôme $\lambda(s^v)$ est moyenné. Quatre valeurs $\Delta s_\lambda^v = 0.1m$, $0.2m$, $0.4m$, $2m$ ont été simulées dans le cas où $v_1 = 1m.s^{-1}$ et $d^\star = 5m$ (résultats (4)-(6)-(7)-(8) de la Fig. 3.18)). On observe sans surprise qu'en l'absence de bruit, il vaut mieux que la valeur Δs_λ^v soit petite pour avoir l'estimation la plus ponctuelle possible. Si les données sont bruitées, les perturbations introduites par le bruit sont lissées en considérant des valeurs plus grandes, mais l'amélioration est assez légère, surtout pour de grandes valeurs de $d^\star$.

3.6.2 Estimation à partir de données odométriques

A nouveau, trois paramètres exercent une influence sur les performances de l'estimation de l'abscisse curviligne. On recense toujours la vitesse v_1 du véhicule concerné (que l'on suppose ici être celui de tête). Le deuxième paramètre concerne la valeur du gain K de l'observateur et, dans le cas ou l'on simule des données de localisation bruitées, le troisième paramètre α caractérise le filtre passe-bas récursif de Butterworth qu'il est nécessaire d'introduire pour lisser l'estimée de la vitesse curviligne $\dot{s}_1^v$ obtenue par dérivée numérique.

Le meilleur compromis pour les valeurs de K et α, ainsi que l'influence de la vitesse v_1 ont été étudiés au travers de simulations. Les résultats les plus significatifs en fonction de ces paramètres sont répertoriés sur la Fig. 3.20. Les meilleures valeurs (en terme d'erreur moyenne mais aussi d'écart-type de l'erreur) pour le couple (α, K) selon la vitesse sont mises en valeur, en blanc sur fond bleu. Seuls les résultats avec des données bruitées sont présentés, car en l'absence de bruit, les erreurs et écarts-type sont parfaits (proches de $10^{-14}m$).

	v_1 (m.s^{-1})	K	α	erreur moyenne (10^{-3}m) $\sigma_v = 0.02m$ $\sigma_o = 0.015m.s^{-1}$	écart-type (10^{-3}m) $\sigma_v = 0.02m$ $\sigma_o = 0.015m.s^{-1}$
(1)	0.5	0.4	0.94	10.1869	11.1847
(2)	0.5	0.5	0.70	85.3467	29.8355
(3)	0.5	0.5	0.94	10.3081	11.0556
(4)	1	0.5	0.70	21.5789	15.3462
(5)	1	0.5	0.90	10.9390	13.2481
(6)	1	0.8	0.90	10.5295	12.0055
(7)	1	1.1	0.90	10.7259	11.7735
(8)	3	2	0.94	18.9091	31.7719
(9)	3	2	0.82	12.2912	16.1353
(10)	3	0.2	0.74	12.9140	13.3171

FIG. 3.20 – Influence des paramètres sur les performances d'estimation en exploitant les données odométriques

Influence des paramètres On constate d'une part que les performances de l'estimation diminuent pour les valeurs de vitesse du véhicule les plus grandes (résultats (8) à (10) de la Fig. 3.20). Cela était attendu : le facteur d'échelle $\lambda_{[k]}$ calculé à l'instant kT_e est associé à la portion de trajectoire $\left[(s_1^v)_{[k-r-1]};\ (s_1^v)_{[k-r]}\right]$ et plus la vitesse v_1 du véhicule augmente, plus cette portion de trajectoire s'allonge, on s'écarte donc de la correction très locale qui était recherchée pour le facteur d'échelle. La détérioration reste tout de même très faible puisqu'il y a moins de $3mm$ d'erreur moyenne de différence entre les différents choix de v_1 (amplitude maximale pour les résultats (1) et (10)).

A contrario, lorsque la vitesse est très faible, l'estimation brute de la vitesse curviligne devient très bruitée (car $(s_1^v)_{[k-1]}$ et $(s_1^v)_{[k]}$ dans (3.51) ont alors des valeurs très proches). Comme le montrent les résultats (1) à (3), il est alors nécessaire d'augmenter α pour limiter ce bruit. Or, une trop grande valeur peut aussi dégrader l'estimée de la vitesse curviligne $\left(\dot{\hat{s}}_1^v\right)$ injectée dans le calcul (3.60) du facteur d'échelle : même si le retard introduit par le filtre est pris en compte, les variations les plus rapides de $\dot{s}_1^v$ sont masquées par un effet de lissage, appauvrissant l'information portée par $\dot{s}_1^v$ et au final l'estimation de la valeur de l'abscisse curviligne dans le référentiel métrique.

Ce compromis pour le paramètre de filtrage est étudié plus en détail sur la Fig. 3.21 avec les cas particuliers où le gain est fixé à $K = 0.5$ et la vitesse à $0.5m.s^{-1}$ et $1m.s^{-1}$. La colonne de gauche présente la valeur moyenne des erreurs pour chaque valeur donnée à α dans la plage $[0.5, 0.98]$, et celle de droite l'écart-type de ces erreurs. On remarque très nettement sur ces courbes qu'à vitesse faible ($v_1 = 0.5m.s^{-1}$) de bonnes performances ne peuvent s'obtenir que si un filtrage fort $\alpha > 0.9$ est effectué (résultats (1)-(3) par rapport à (2)) alors qu'en augmentant de très peu la vitesse ($v_1 = 1m.s^{-1}$), de bons résultats peuvent être obtenus avec un filtrage plus modéré : $0.7 < \alpha < 0.90$, voir les résultats (4)-(5). Les valeurs optimales, par rapport à l'erreur moyenne ou à l'écart-type (légèrement différentes) sont représentées sur la Fig. 3.21 et apparaissent dans le tableau de la Fig. 3.20 (résultats (3)-(5)).

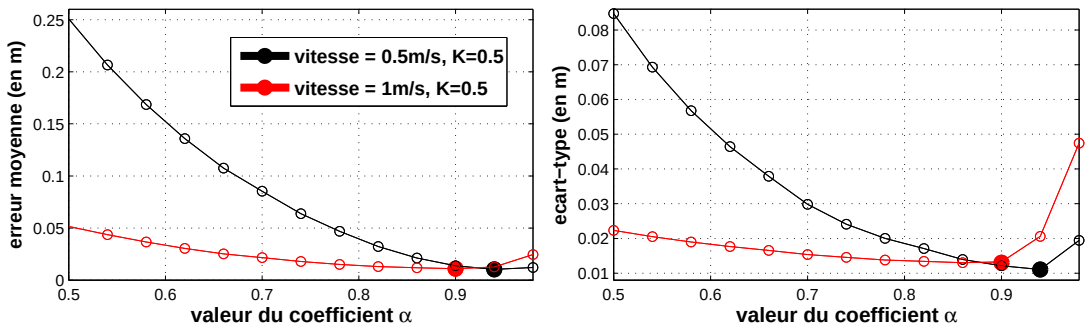


FIG. 3.21 – Influence du paramètre de filtrage α : quelques exemples

Concernant le gain K, il contrôle en théorie la vitesse de convergence de l'observateur. Ainsi, si les données sont parfaites, une valeur de K élevée permet de faire tendre plus rapidement l'erreur ϵ à zéro. En revanche, dans notre cas, les données vision et

odométriques sont bruitées, et l'utilisation d'un filtre altére l'estimation de la vitesse curviligne $(\dot{s}_1^v)_{[k]}$, et donc le modèle d'évolution, et peut conduire à faire osciller l'erreur ϵ. Dans notre cas ce type de perturbation apparaît essentiellement à faible vitesse puisque le bruit sur l'estimation de $(\dot{s}_1^v)_{[k]}$ est plus important. Ce phénomène est illustré sur la Fig. 3.22. L'influence du gain est présentée pour les deux vitesses $v_1 = 0.5m.s^{-1}$ et $v_1 = 3m.s^{-1}$, et la valeur du paramètre α est alors celle qui donne les meilleures performances pour ces vitesses, à savoir $\alpha = 0.94$, $\alpha = 0.82$ et $\alpha = 0.74$ (voir Fig. 3.20). Comme l'attestent les courbes de la partie supérieure de la Fig. 3.22, pour les deux critères de performance (valeur moyenne des erreurs ou l'écart-type de ces erreurs), le bruit présent à faible vitesse impose l'utilisation d'un gain faible et cette tendance est vérifiée quelle que soit la valeur du paramètre de filtrage α. Lorsque la vitesse est plus élevée, les courbes de la partie inférieure de la Fig. 3.22 montrent que le gain n'a pas beaucoup d'influence si le paramètre de filtrage α est bien réglé, ce qui explique la valeur de K plutôt faible (10) au regard des autres configurations (notamment (8)-(9)). En revanche, si le gain est surestimé, alors une valeur de gain K élevée permet de limiter les erreurs.

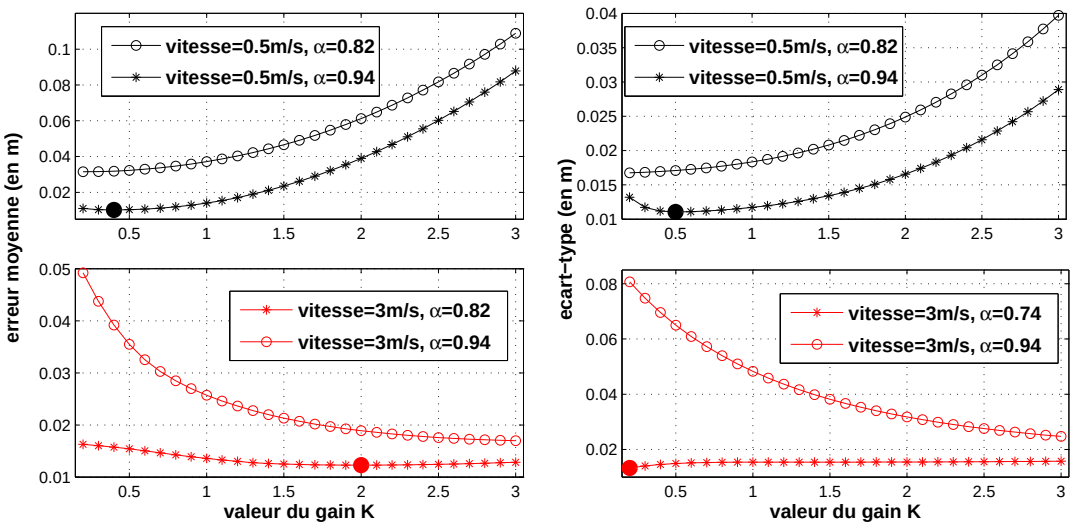


FIG. 3.22 – Influence du gain K : quelques exemples

Prise en compte du retard La nécessité de prendre en compte le retard introduit par le filtrage de la vitesse curviligne vision $\dot{s}_1^v$ est mise en avant sur les Fig. 3.23. Les courbes des meilleurs résultats pour les valeurs de la moyenne des erreurs et l'écart-type de ces erreurs sont respectivement tracées à gauche et à droite de la Fig. 3.23. On constate que les performances d'estimation sont meilleures lorsque le retard est pris en compte, notamment lorsque la vitesse est élevée. Ce résultat est logique puisque le retard porte sur l'estimation d'une vitesse, ici $\dot{s}_1^v$.

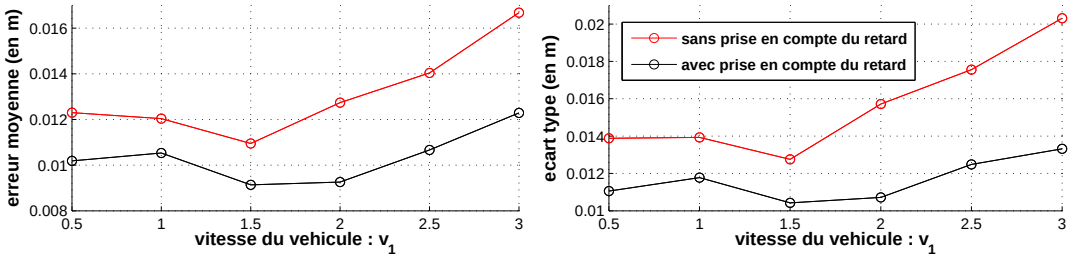


FIG. 3.23 – Meilleurs résultats en fonction de la vitesse v_1, avec et sans prise en compte du retard

3.7 Conclusion

Dans ce chapitre, le problème des distorions présentes dans le monde virtuel obtenu par vision monoculaire a été étudié. Ainsi, la mise en œuvre de l'approche de commande globale développée dans [Bom 05] avec la localisation fournie dans le monde vision a été adressée.

Plus précisément, une analyse empirique a permis de montrer que les distorsions relevées étaient localement homogènes. Cette propriété est très intéressante en pratique : cela signifie qu'à partir d'un jeu de facteurs d'échelle estimés localement le long de la trajectoire de référence, il est possible à partir des données visuelles d'estimer la valeur des informations requises pour le calcul des commandes latérale et longitudinale.

Plusieurs stratégies ont été proposées pour obtenir les facteurs d'échelle et recouvrer les informations d'état des véhicules dans un référentiel métrique. Une stratégie hors-ligne, permet d'estimer ponctuellement la valeur des facteurs d'échelle sur des tronçons de $0.5m$ en exploitant, selon un processus d'optimisation, des distances obtenues par vision monoculaire et les mêmes quantités mesurées dans le monde réel à l'aide d'un récepteur RTK-GPS. Dans la perspective de réaliser un guidage manuel du convoi avec une localisation par vision monoculaire, deux autres méthodes ont été proposées pour évaluer les facteurs d'échelle en-ligne cette fois, à l'aide de capteurs adaptés au milieu urbain. Pour la première approche, les données métriques du récepteur RTK-GPS sont remplacées par l'utilisation d'un unique télémètre laser monté sur le deuxième véhicule de la flotte et opérant le suivi du véhicule de tête dans l'image télémétrique. Les distorions sont alors évaluées à partir d'une approximation polynômiale du facteur d'échelle sur une portion de la trajectoire. Une approche alternative, s'appuyant uniquement sur des données odométriques, est considérée pour la deuxième approche. Elle est basée sur un observateur et le modèle d'évolution du véhicule. Pour ces deux approches, des batteries de simulations ont été réalisées pour évaluer l'influence des paramètres et mettre en évidence les meilleurs configurations à utiliser en vue d'applications expérimentales.

Chapitre 4

Expérimentations

Après avoir présenté les contributions théoriques de cette thèse, la phase suivante est logiquement consacrée à l'implantation et à la validation des algorithmes de commande sur des véhicules électriques prévus pour la conduite urbaine. Ce chapitre a pour objectif de valider par expérimentations les stratégies développées au cours des deux chapitres qui précèdent, à savoir :

• la navigation en convoi en *mode manuel*, décrite au chapitre 2, au cours de laquelle un véhicule leader se comporte librement (et peut ainsi être guidé par un utilisateur) et les autres véhicules sont alors asservis sur la trajectoire empruntée par ce véhicule.

• la navigation en convoi par *modalité capteur vision*, et plus précisément la vision monoculaire, pour laquelle plusieurs approches ont été proposées au chapitre 3.

Dans ce chapitre, une première section décrit le dispositif expérimental nécessaire pour la mise en œuvre d'une conduite en convoi, à savoir la plateforme dédiée aux expérimentations, les véhicules ainsi que les capteurs requis. La section qui suit est consacrée à la problématique de la cohérence temporelle des informations utilisées. La dernière section reporte finalement les performances obtenues en convoi au cours des différentes expérimentations qui ont été réalisées.

4.1 Contexte expérimental

Dans cette section, le contexte de la mise en œuvre sur véhicules urbains est présenté. Pour commencer, la plateforme d'expérimentation PAVIN est introduite, suivie de la description des véhicules urbains, nommés CyCab et RobuCab. Puis, les organes sensoriels installés sur ces robots pour leur permettre de se localiser dans leur environnement de travail, sont exposés. L'usage de communications sans fil, nécessaire pour situer chaque véhicule au sein du convoi, est ensuite détaillé. Enfin, une architecture logicielle propre au convoi est proposée. Celle-ci a pour rôle essentiel de coordonner et gérer l'interaction entre les différents membres du convoi au moyen des communications inter-véhicules.

4.1.1 Plateforme expérimentale : PAVIN

Inaugurée en septembre 2008 sur le campus universitaire de Clermont-Ferrand, la plateforme PAVIN - comme Plateforme d'Auvergne pour les Véhicules INtelligents - est un site d'expérimentation, de développement et de mise au point de véhicules entièrement automatiques. Cet équipement permet de réaliser des tests intensifs en milieu sécurisé, en intégrant les recherches scientifiques sur des démonstrateurs réalistes. Ce projet a été porté par la Fédération de Recherche TIMS qui réunit plusieurs laboratoires clermontois dont le LASMEA pour conduire des projets pluridisciplinaires, aux interfaces de l'informatique, de la mécanique et de l'automatique ; notamment sur le thème "Véhicules et Infrastructures Intelligents". Il a bénéficié d'une forte dynamique locale entre les acteurs scientifiques, institutionnels et industriels.



FIG. 4.1 – Plateforme PAVIN vue du LASMEA

Plus précisément, la plateforme PAVIN, illustrée Fig. 4.1, permet de tester les méthodes de guidage et de localisation sur la flotte de véhicules à disposition, dans des environnements réalistes "urbain" d'une part avec $317m$ de voies goudronnées entre façades d'immeubles, avec carrefour, rond point, zone de montée de passagers, et "rural" d'autre part avec $264m$ de pistes stabilisées avec zone de devers, sol instable ("bac à sable"), anneau de vitesse ($15km/h$ max). Le site est également équipé de feux de signalisation tricolores, d'éclairages, de passages piétons, de bornes d'appel et de bornes "Energie".

4.1.2 Véhicules expérimentaux : CyCabs et RobuCabs

S'appuyant sur les innovations développées pour le programme Praxitèle, l'INRIA est à l'origine de la conception de petits véhicules électriques, nommés CyCab et RobuCab (voir Fig. 4.2), destinés à une utilisation en libre-service. Ces véhicules sont actuellement fabriqués de façon industrielle par la société Robosoft[1], une société de technologie innovante issue de l'INRIA. Ces véhicules sont spécifiquement conçus pour les zones où la circulation automobile doit être fortement restreinte (centre urbain, ga-

[1] http ://www.robosoft.fr



FIG. 4.2 – Les véhicules expérimentaux : deux CyCab et un RobuCab sur PAVIN

re/aérogare, campus universitaire, site touristique) pour lesquelles leurs faibles dimensions (longueur 1.90 m, largeur 1.20 m) sont des avantages. Les prototypes actuels, d'un poids de 300 kg, sont équipés d'un moteur électrique de 1 kW sur chacune des quatre roues motrices. Ces 4 moteurs sont alimentés par 8 batteries au plomb, conférant une autonomie de 2 heures pour une utilisation en continu. Un ordinateur embarqué permet de piloter le robot manuellement via l'usage d'un joystick ou de manière entièrement autonome. Les exemplaires dont est doté le LASMEA peuvent transporter deux individus (CyCab) ou quatre individus (RobuCab) à une vitesse maximale $v_{max} = 18km/h$. La configuration cinématique utilisée pour nos expérimentations est celle d'une voiture (c'est-à-dire, seules les roues avant sont directrices).

4.1.3 Capteurs utilisés

Localisation Deux types de capteurs ont été utilisés pour fournir la localisation des véhicules dans un repère absolu. D'une part le GPS cinématique (RTK-GPS : Real-Time Kinematic Global Positioning System), utilisé lors des expérimentations de navigation en conduite manuelle, et d'autre part une caméra utilisée bien évidemment pour les expérimentations de navigation en convoi par modalité vision. Dans ce dernier cas, l'information fournie par la caméra peut être fusionnée avec celle d'un télémètre laser ou des odomètres selon la stratégie de correction des distorsions employée, comme il a été présenté au chapitre 3. Enfin, l'information fournie par le GPS cinématique est également enregistrée, non plus pour commander les véhicules, mais pour disposer d'une vérité terrain, puisque ce capteur est celui qui offre le plus de précision.

- ***RTK-GPS :*** Ces capteurs sont basés sur des technologies de positionnement par satellites. Ils permettent d'obtenir des performances de localisation de l'ordre du centimètre à une fréquence élevée, $10Hz$ dans notre cas. Pour obtenir cette précision, le dispositif utilise tout d'abord des corrections différentielles qui exploitent la propriété que, si deux récepteurs observent les mêmes satellites, alors ils font des erreurs de

mesure d'autant plus semblables que ces récepteurs sont proches. En conséquence, un récepteur de référence est placé sur une position parfaitement connue. En comparant les mesures fournies par les signaux GPS à sa position théorique, il est possible de connaître l'erreur courante sur la localisation GPS dans un voisinage du récepteur de référence. Cette erreur est ensuite transmise par liaison UHF aux récepteurs mobiles pour leur permettre d'affiner leur propre localisation GPS. Ensuite, la phase des signaux GPS envoyés sur deux fréquences est également utilisée pour atteindre une précision de localisation centimétrique.

Les récepteurs utilisés sont de type Z-Max, conçus par Thales. Chaque capteur, monté sur un des éléments du convoi, transmet ses informations par le biais d'une liaison série RS-232 reliée au PC haut-niveau du véhicule. L'antenne GPS est fixée à la verticale du milieu de l'essieu arrière du véhicule de façon à pouvoir délivrer directement le positionnement du point de contrôle O du véhicule. En fonctionnement optimal, le récepteur offre une précision horizontale (longitude et latitude) de 1 cm et de 2 cm en altitude pour une fréquence d'acquisition de 10 Hz.

• ***Caméra :*** Il s'agit d'une caméra MARLIN de technologie CMOS noir et blanc. Elle produit des images à une fréquence allant jusqu'à 15 images par seconde avec une résolution 1024×768 pixels. Traiter des images de cette taille serait trop lent. Aussi, elles sont rééchantillonnées par logiciel à une taille de 512×384. La procédure de rééchantillonnage a été programmée en utilisant le jeu d'instruction SSE. Le niveau de gris d'un pixel de l'image finale est le résultat de la moyenne des niveaux de gris de 4 pixels contigus de l'image originale. Le rééchantillonnage prend moins de $2ms$.

L'objectif utilisé est un objectif à très grand angle : sa focale est de $3.5mm$, avec une caméra disposant d'une zone photosensible dont la diagonale mesure 1/2" soit $12.7mm$. Si on néglige la distorsion, le champ angulaire (dans la diagonale) est $\alpha = 2\tan^{-1}\left(\frac{12.7}{3.5\times 2}\right)$, soit $\alpha = 122°$. En réalité, à cause des effets de distorsion de la caméra, le champ réel est plus grand que cela et atteint 130° dans la diagonale de l'image. L'utilisation d'un objectif à très grand angle est très intéressante car cela permet de réduire les risques d'occultation. Par exemple, une personne située à deux ou trois mètres de la caméra occupe une toute petite partie de l'image. Dans ce cas, elle masque très peu de points d'intérêt, et donc sa présence est sans conséquence sur le résultat de la localisation.

La caméra monoculaire est directement connectée sur le port FireWire du PC haut-niveau. Celui-ci est idéal puisque la vitesse de transmission via un bus IEEE 1394 peut atteindre des débits de plusieurs dizaines de Mo/s et offre ainsi une bande passante suffisante pour l'enregistrement de flux vidéo.

Autres capteurs Les capteurs odométriques et télémétriques interviennent également dans la mise en œuvre du convoi. Les deux sont directement impliqués dans les stratégies de navigation en convoi par vision et correspondent aux deux modalités proposées pour évaluer le facteur d'échelle le long de la trajectoire. Mais, ils sont aussi utilisés en dehors des stratégies de navigation : le télémètre laser pour détecter la présence d'obstacles à proximité du véhicule (géré par le PC bas-niveau) et les odomètres

pour recaler temporellement l'état des véhicules (voir 4.2).

• ***Télémètre laser :*** Il s'agit d'un SICK LMS 221 dont le champ d'action est de 180°. La fréquence d'acquisition s'élève à 60 Hz et la résolution angulaire est de 0.5°. Pour une portée de 20 m, les mesures sont entachées d'une erreur systématique de 35 mm et d'une erreur statistique de 15 mm. Une liaison série RS422 connecte le télémètre au PC bas niveau, sur lequel est implantée la routine de détection d'obstacles. Les données peuvent ensuite être remontées au PC haut niveau via une liaison Ethernet.

• ***Odomètres :*** Sur chaque "moteur-roue" l'axe du rotor est équipé d'un codeur optique de type HEDS-9140 qui permet de calculer la vitesse de la roue. La direction est actionnée par un vérin électrique et la mesure de braquage s'opère à l'aide d'un codeur optique associé à un capteur de déplacement rectiligne de type RECTIP12 de MCB industrie. Les signaux des codeurs optiques de chaque train (avant et arrière) du véhicule sont ensuite gérés par une carte microcontrôleur Motorola MPC555 POWER-PC qui est connectée au PC bas niveau par un bus CAN.

4.1.4 Communication sans fil

Les capteurs présentés à la section précédente permettent à chaque véhicule de connaître sa localisation absolue avec une haute précision. Pour l'application visée (convoi de véhicules), il faut de surcroît que ces informations soient partagées entre les différents membres du peloton. Dès lors, une communication sans fil est incontournable.

Un réseau local sans fil (WLAN : Wireless Local Area Network) au sein du convoi a donc été mis en place. Les caractéristiques d'un tel réseau sont décrites par la norme ieee802.11 qui concerne les couches basses du modèle OSI, c'est-à-dire d'une part la couche physique, qui définit la modulation des ondes radio-électriques et les caractéristiques de la signalisation pour la transmission de données, et d'autre part la couche liaison de données qui définit l'interface entre le bus de la machine et la couche physique. Dans la pratique, cette norme permet de créer une liaison à haut débit sur plusieurs centaines de mètres en environnement ouvert.

Dans notre cas, les différents PC haut-niveau montés sur les CyCabs sont reliés via un réseau WiFi (nom donné à la certification des matériels répondant à la norme Wi-Fi : Wireless Fidelity) suivant la révision ieee802.11g qui permet d'obtenir un débit théorique de 54 $Mbits/s$ pour une portée d'environ 75 m (mais qui peut aller jusqu'à 400 m pour des débits inférieurs). Les caractéristiques du réseau en terme de débit et d'espace couvert sont donc correctement dimensionnées pour un convoi de véhicules, et en font une solution adéquate pour l'application considérée ici.

Les informations transitent alors grâce au protocole de communication IPv4 (Internet Protocol), qui appartient à la couche réseau du modèle OSI et assure le service d'adressage et d'acheminement des données, et grâce au protocole UDP (User Datagram Protocol) appartenant lui à la couche transport et qui gère la transmission intègre des

données de manière simple : il n'y a pas de procédure de connexion préalable à l'envoi des données ni de garantie de bonne livraison. Avec cette configuration, les délais moyens de communication entre les véhicules ont été mesurés à 47 ms.

4.1.5 Architecture informatique du convoi

Tous les éléments matériels (véhicules expérimentaux, capteurs,...) ont été présentés ci-dessus. L'interaction entre ceux-ci se fait via une architecture décentralisée, adaptée à l'application convoi, détaillée ci-dessous par couche hiérarchique et illustrée sur la Fig. 4.3.

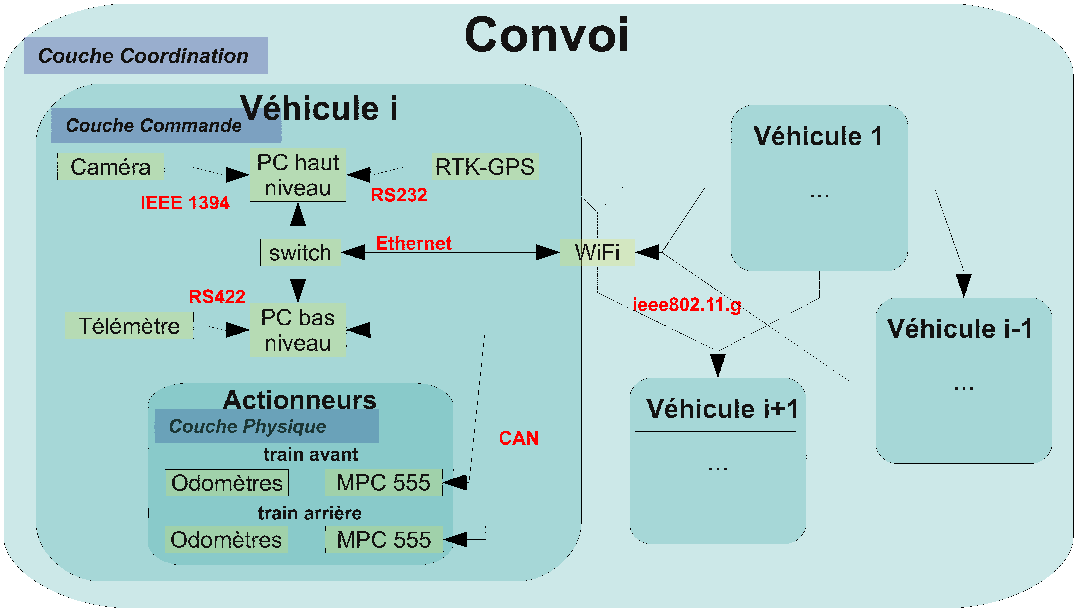


FIG. 4.3 – Architecture informatique du convoi

Couche coordination La couche de coordination collecte les données provenant d'autres véhicules du convoi, au moyen de passerelles WiFi décrites ci-dessus. Plus concrètement, un point d'accès WiFi est situé sur le véhicule leader et chaque véhicule est connecté au réseau via le PC haut-niveau muni d'un adaptateur interne.

Couche commande La couche commande a pour objectif de générer la commande à appliquer au véhicule. Elle se situe au niveau de l'ordinateur portable haut-niveau. A partir des mesures des capteurs implantés localement (RTK-GPS relié au PC portable par une liaison série classique RS232, ou vision par le biais d'une liaison IEEE 1394 (FireWire)), des données du télémètre et des odomètres relayées par le PC bas-niveau, et de l'état du convoi provenant de la couche supérieure (couche *coordination*), les consignes en commandes longitudinale et latérale sont calculées et transmises à la couche inférieure : la couche *physique*.

Couche physique Finalement, les actionneurs du véhicule sont asservis à l'aide d'un contrôleur proportionnel intégral (PI), géré par le PC bas niveau de sorte que le mouvement engendré corresponde aux valeurs de commande calculées par la couche supérieure.

4.2 Cohérence temporelle et synchronisation des informations du convoi

La stratégie de contrôle que nous avons choisie ici (voir chapitre 1) réclame que les informations du véhicule leader soient retransmises à toute la flotte, alors que les autres véhicules envoyent leurs informations uniquement à leur suiveur direct, voir Fig. 4.4. Les données sont transférées de façon asynchrone dès qu'elles sont disponibles, à une fréquence ainsi similaire à celle du dispositif de localisation. A chaque nouvelle acquisition (image ou trame GPS), la quantité d'information à transférer s'élève à 97 *Octets* (données de localisation et odométriques).

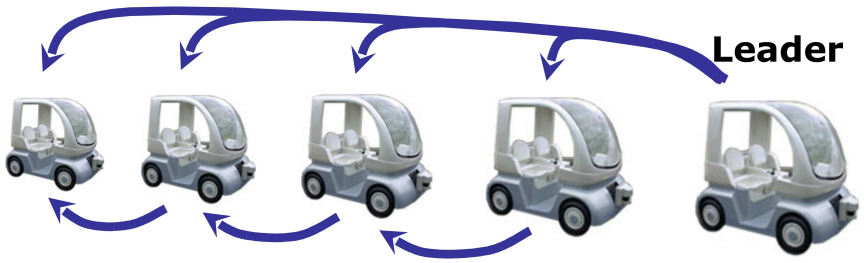


FIG. 4.4 – Schéma de communication

4.2.1 Acquisition asynchrone des données de commande

Pour que chaque véhicule soit convenablement contrôlé, les lois de commande latérale et longitudinale doivent être évaluées à fréquence constante. Pourtant, les données à partir desquelles les lois sont calculées, sont obtenues de façon asynchrone : les données à exploiter sont donc temporellement incohérentes. Cette difficulté apparaît à deux niveaux, comme le montre la Fig. 4.5.

• Sur la partie supérieure de la Fig. 4.5, on s'intéresse à la représentation de l'*état du véhicule*. Ce dernier est obtenu à partir de données provenant de capteurs extéroceptifs et proprioceptifs. La localisation absolue est donnée par un récepteur RTK GPS ou par vision monoculaire, alors que la vitesse et l'angle de braquage des roues directrices sont fournis par odométrie. Chaque capteur possède sa propre latence et éventuellement son propre temps de calcul (pour la localisation par vision monoculaire, le temps de calcul moyen est de $30ms$, voir [Royer 04]), si bien que les informations d'état, obtenues par un capteur et valables au moment de l'acquisition des données brutes de ce capteur,

ne le sont plus une fois les données transmises et/ou traitées. Sur la Fig. 4.5, le délai correspondant au capteur qui fournit la localisation est représenté par la variable Δt_{loc}. D'autre part, la période d'acquisition des données est différente selon chaque capteur (elle est notée T_{loc} pour la localisation) et ne s'accorde pas avec le cadencement de la tâche qui consiste au calcul et à l'envoi des commandes, représentée par des losanges noirs sur la Fig. 4.5 et caractérisée par sa période T_{cde}. Ainsi les données propres à chaque véhicule ne sont déjà pas cohérentes temporellement.

• L'*état du convoi*, étudié sur la partie inférieure de la Fig. 4.5, est obtenu en collectant les états des véhicules cibles (leader et précédent) au travers des communications inter-véhicules. Le retard induit par le temps de transmission, noté Δt_{com} sur la Fig. 4.5, a été mesuré dans l'intervalle $[20.6ms\,,\,98.4ms]$ avec une valeur moyenne de $47ms$. Les incohérences temporelles relevées sont donc encore plus importantes que pour l'état propre des véhicules. De plus ce retard peut aussi occasionnellement être augmenté par des pertes de paquets durant la transmission ou par d'autres problèmes inhérents aux systèmes de communication sans fil.

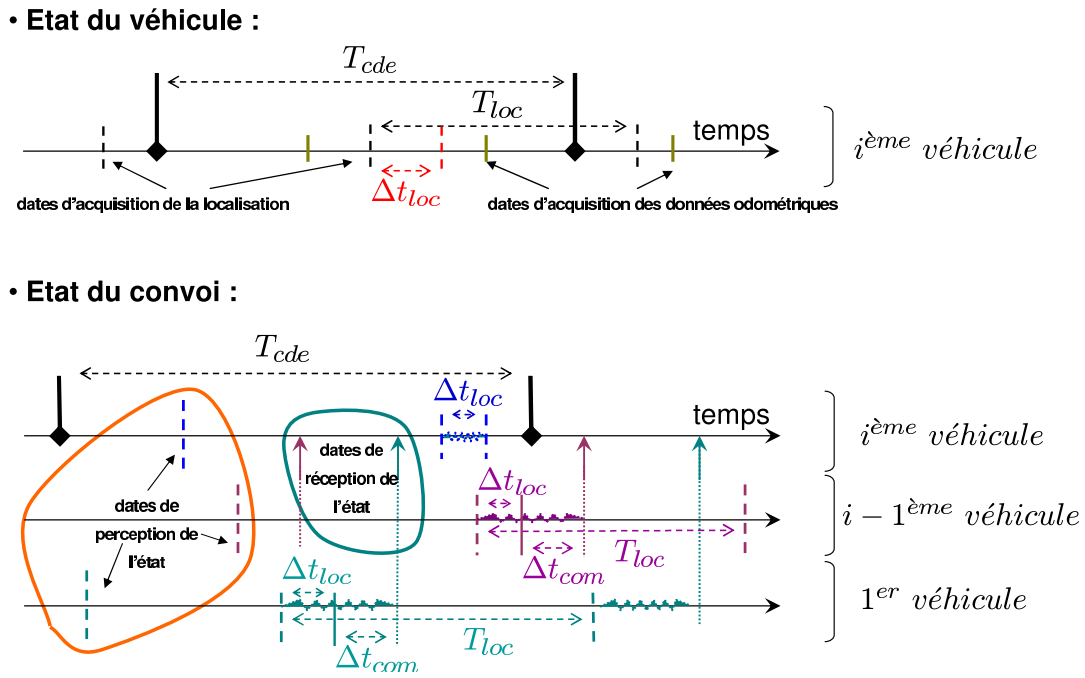


FIG. 4.5 – Acquisition des données pour la commande

4.2.2 Mise à jour de l'état du convoi

Pour contrôler convenablement les véhicules, les données à exploiter dans les lois de commande doivent refléter l'état de chaque véhicule dans le convoi au même instant.

Pour répondre à cette exigence, la structure logicielle AROCCAM [Tessier 06] est actuellement utilisée. Des modules (ou pilotes), responsables de la communication avec les capteurs, collectent les informations en parallèle dans différents *threads*. Une procédure de datation permet d'estampiller les données : les latences et autres temps de calcul pour faire remonter l'information sont pris en compte afin de refléter avec précision la date de perception des capteurs. Chaque véhicule est donc capable de dater avec précision les informations délivrées par ses propres capteurs. De cette façon, dès qu'une nouvelle localisation est disponible, le véhicule j envoie ses informations d'état, estampillées à la date t^i_{loc} de la perception (voir Fig. 4.6), aux véhicules concernés.

Ensuite, pour que chaque véhicule puisse exploiter les données des autres véhicules, il faut que tous partagent la même référence temporelle. Pour cela, un simple client NTP (Network Time Protocol) est utilisé pour configurer l'horloge des systèmes des PCs haut-niveau sur l'horloge d'un serveur. Cette actualisation, représentée par un marqueur circulaire à l'instant t_{ref} sur la Fig. 4.6, est lancée à une fréquence de $1Hz$ sur les PCs haut-niveau de chaque véhicule après avoir défini le serveur de référence. Les données émanant de tous les véhicules sont alors datées de façon cohérente.

• Etat du convoi pour le véhicule i : $(X_i,\ X_1,\ X_{i-1})$

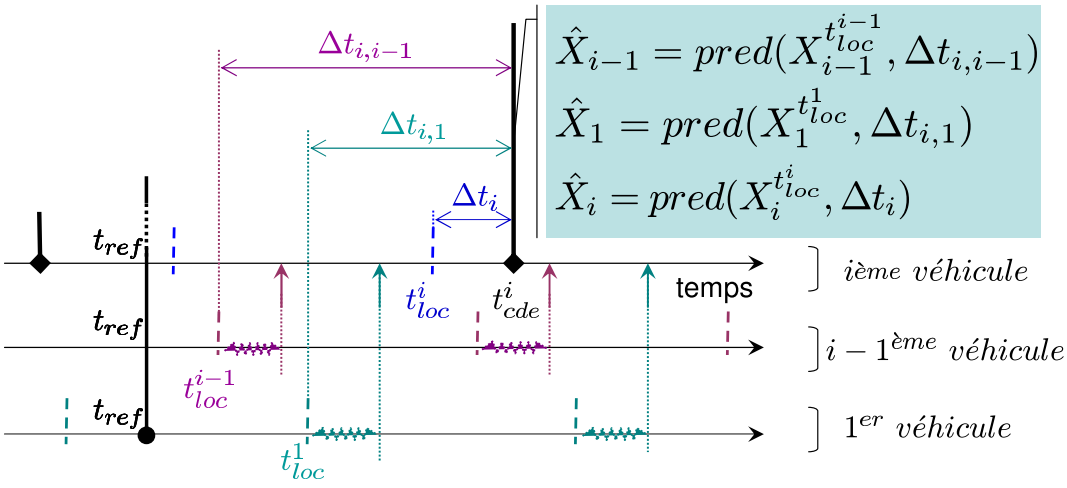


FIG. 4.6 – Recalage temporel des données pour la commande

Lorsque le *thread* AROCCAM de plus haute priorité, associé à un timer de $10Hz$, est sollicité pour le calcul des lois de commande, l'état du convoi est alors actualisé : le modèle cinématique (1.50) des véhicules est intégré par la méthode classique de Runge-Kutta d'ordre quatre entre l'instant où ont été fournies les données sur un véhicule et l'instant courant, de façon à pouvoir disposer d'une prédiction de l'état global du convoi à l'instant courant. Plus concrètement, l'état du convoi perçu par le $i^{ème}$ véhicule est constitué du triplet $(X_i,\ X_1,\ X_{i-1})$, où X_j est l'état du véhicule j. Comme le montre la Fig.4.6, lorsque la commande du véhicule est calculée, à l'instant t^i_{cde}, les données

d'état disponibles sont celles perçues aux instants t^i_{loc}, t^{i-1}_{loc} et t^1_{loc} respectivement par les véhicules i, $i-1$ et par le leader. Leur mise à jour est par conséquent effectuée en intégrant le modèle d'évolution sur les intervalles de temps $\Delta t_i = t^i_{cde} - t^i_{loc}$, $\Delta t_{i,i-1} = t^i_{cde} - t^{i-1}_{loc}$ et $\Delta t_{i,1} = t^i_{cde} - t^1_{loc}$. Ces prédictions d'état, notées $\hat{X}_i$, $\hat{X}_1$ et $\hat{X}_{i-1}$ sur la Fig.4.6, sont cohérentes sur le plan temporel et sont reportées dans le calcul des lois latérale et longitudinale. Les valeurs obtenues par ces algorithmes sont ensuite envoyées aux actionneurs. Bien sûr, un chien de garde est mis en place afin de commuter les véhicules dans une phase d'arrêt si certaines données n'ont pas été acquises sur une période relativement longue (à cause de problème de communication par exemple). Si le problème disparaît, le véhicule, autorisé à repartir, est commuté dans sa phase nominale.

Les variables d'état du convoi dont l'évolution dans le temps est la plus rapide, et qui sont par conséquent les plus sensibles aux incohérences temporelles, sont les abscisses curvilignes s_i des véhicules le long de la trajectoire. La Fig. 4.7 présente la mise à jour de ces variables calculée par le troisième véhicule lors d'expérimentations en convoi.

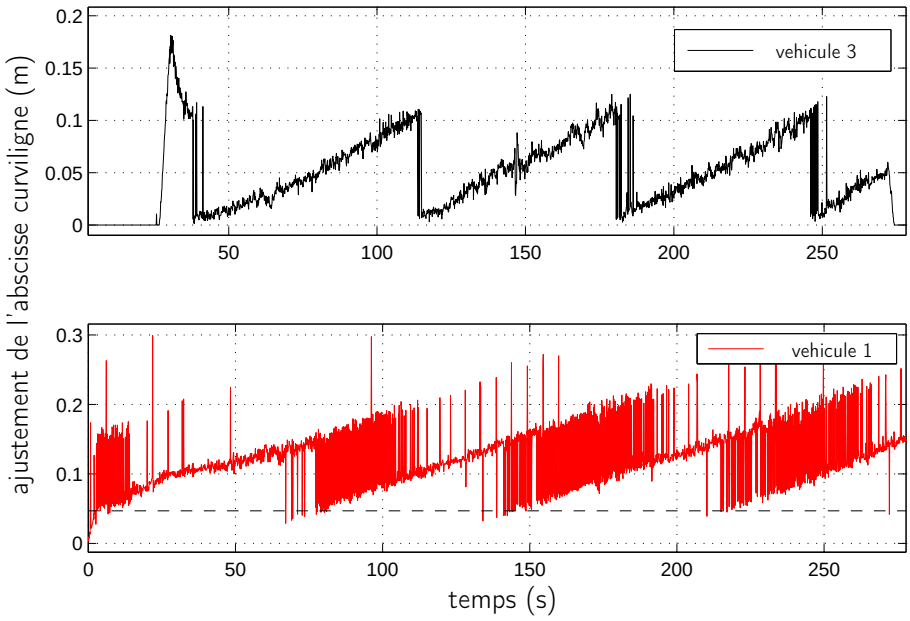


FIG. 4.7 – Mise à jour de l'abscisse curviligne effectuée par le troisième véhicule

Plus précisément, l'image du haut montre l'actualisation de la distance curviligne courante (c'est-à-dire celle du troisième véhicule), et l'image du bas affiche l'actualisation de la même variable mais concernant le leader. On peut constater que ces mises à jour sont non-négligeables. Sur la figure du haut, l'ordonnancement des tâches AROCCAM

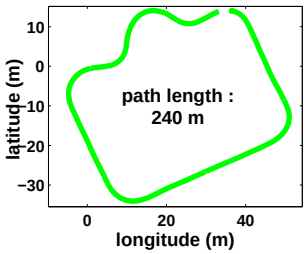
est parfaitement visible : la différence entre les abscisses curvilignes enregistrée et actualisée varie linéairement, à cause de la différence quasi-constante entre la fréquence du timer qui régit l'envoi des commandes et la fréquence d'acquisition des données de localisation. La mise à jour maximale atteint $10cm$, ce qui est tout à fait logique : dans l'expérimentation reportée sur la Fig. 4.7, la vitesse du convoi est $1m.s^{-1}$ et la commande est calculée avec une fréquence de $10Hz$. Par conséquent, le véhicule peut effectivement parcourir jusqu'à $10cm$ entre deux itérations de commande. Des valeurs plus élevées sont présentes au début du graphe car la vitesse du véhicule est légèrement supérieure au cours de cette période, pour permettre au troisième véhicule d'atteindre la configuration d'inter-distance désirée pour le convoi. La figure du bas montre que l'actualisation de variables obtenues par le biais de la communication sans fil présente des valeurs bien plus grandes : à cause des délais de communication, la différence d'abscisse curviligne peut atteindre $30cm$ et est supérieure à $4.7cm$ tout au long de l'expérimentation. Cette dernière valeur, représentée par la ligne discontinue, correspond à la distance couverte pendant le délai de communication moyen qui est de $47ms$, comme mentionné précédemment.

4.3 Résultats expérimentaux

Dans cette section les expérimentations concernant les différentes stratégies mises en œuvre pour la navigation en convoi sont détaillées et les performances des loi de commande sont discutées. Les expérimentations dans un mode de navigation manuel sont tout d'abord présentées, suivies de celles par modalité capteur vision.

4.3.1 Guidage manuel du convoi : localisation par récepteur RTK-GPS

Description de l'expérimentation Pour cette expérimentation, le convoi se compose de quatre véhicules, deux CyCabs en tête puis deux RobuCabs. La consigne d'interdistance est $d^\star = 6m$ alors que le leader est conduit manuellement à une vitesse constante de $1m.s^{-1}$ le long d'une trajectoire de $280m$ présentée ci-contre. Comme on peut le voir, la trajectoire comporte aussi bien des lignes droites que des virages à forte courbure.



Les véhicules suiveurs ne peuvent bien sûr être commandés que s'ils peuvent se localiser par rapport à la trajectoire du leader qui sert, en mode manuel, de trajectoire de référence. Par conséquent, l'asservissement en convoi se déroule suivant deux phases. La première, dite "d'accrochage", est illustrée Fig. 4.8. Les véhicules suiveurs (les 2, 3 et $4^{ème}$) attendent dans l'ordre et sans bouger que le leader les dépasse (parties **a)** et **b)** de la figure). Lorsque la position d'un véhicule suiveur peut être projetée sur la trajectoire du leader, et que bien sûr l'abscisse curviligne de ce dernier est supérieure à

celle du suiveur, les commandes longitudinale et latérale du suiveur sont alors activées. Cependant, en conformité avec la commande longitudinale, les suiveurs ne bougeront que si le véhicule précédent n'est pas à l'arrêt juste devant eux. Donc les véhicules n'entameront leur déplacement que lorsque le leader sera positionné devant la flotte (partie **c)** de la figure). Une fois cette étape terminée, les véhicules entrent dans la phase nominale, illustrée Fig. 4.9, au cours de laquelle les vitesses des véhicules se stabilisent ainsi que les distances curvilignes entre chacun.



FIG. 4.8 – Phase d'accrochage du convoi en mode manuel

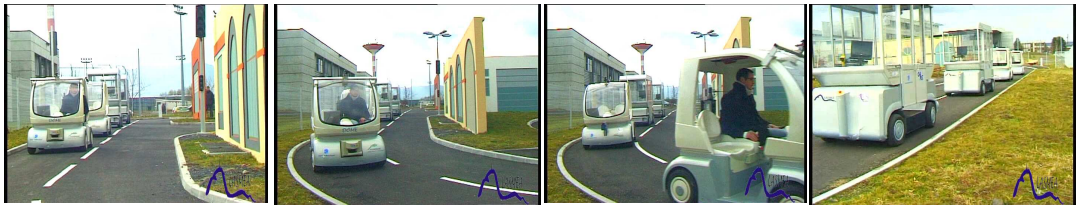


FIG. 4.9 – Phase nominale du convoi en mode manuel

Qualité de la trajectoire Les paramètres (d, c_2, n_{ac}, n_{ap}) utilisés pour la création en ligne de la trajectoire de référence lors de cette expérimentation sont ceux qui apparaissent les plus pertinents au terme des simulations présentées au chapitre 2, à savoir $d = 3$, $c_2 = 1.5m$, $n_{ac} = n_{ap} = 5$. La précision de la trajectoire de référence générée en ligne est évaluée Fig. 4.10 : les valeurs moyenne et maximale des erreurs mesurées entre les données brutes de localisation du leader et leurs projections sur les courbes B-Spline qui servent de trajectoire de référence aux suiveurs, sont respectivement $0.76cm$ et $5.49cm$. Elles sont tout à fait conforme aux valeurs obtenues lors des simulations présentées au chapitre 2, voir Fig. 2.21. Afin de comparer la qualité de la trajectoire en ligne avec celle qui aurait été utilisée en mode de navigation "tout automatique", les données brutes de localisation enregistrées sur le leader ont été utilisées pour générer une nouvelle trajectoire de référence selon le protocole hors ligne, c'est-à-dire en utilisant le critère d'optimisation global. La précision de cette seconde trajectoire de référence est évaluée sur la Fig. 4.11, de la même manière que pour la trajectoire calculée en ligne.

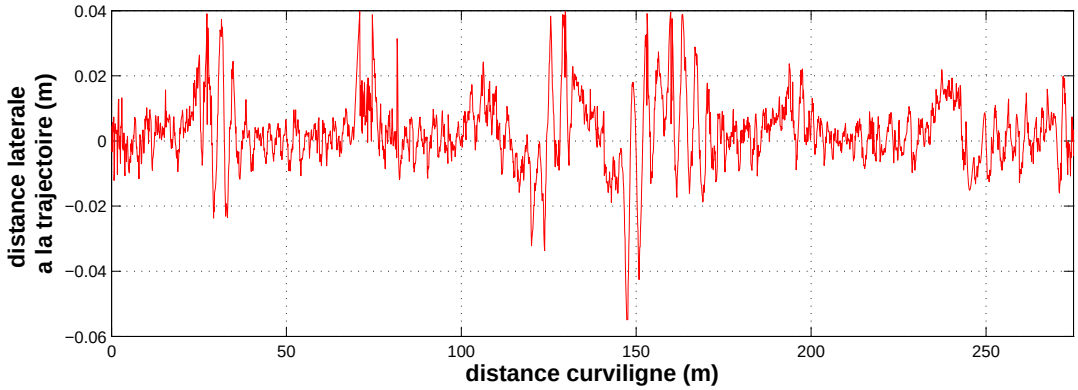


FIG. 4.10 – Qualité de la trajectoire générée en cours d'expérimentation : optimisation locale

Les résultats apparaissent logiquement légèrement meilleurs, avec des valeurs pour les erreurs moyenne et maximale de $0.43cm$ et $3.41cm$, mais restent globalement proches des résultats obtenus lors de la génération en ligne de la trajectoire de référence. Le protocole d'optimisation local proposé au chapitre 2, et les paramètres retenus, permettent donc de générer en ligne une trajectoire de référence très proche de la trajectoire du leader et donc rend possible une évolution monotrace du convoi.

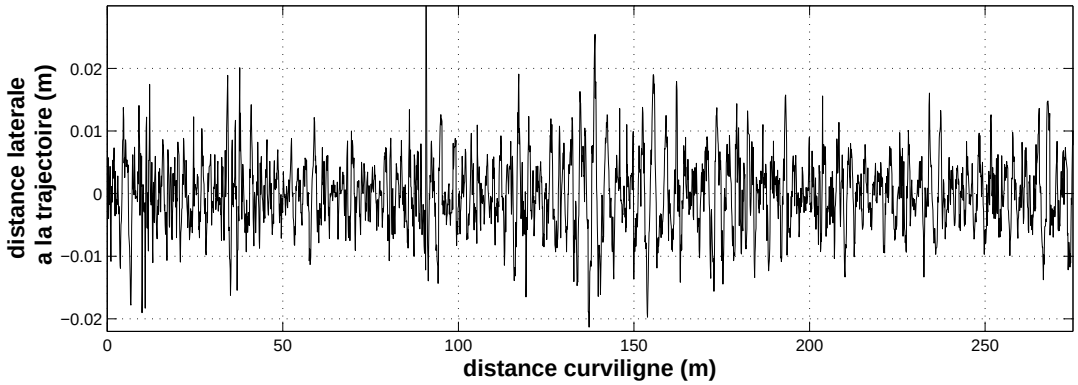


FIG. 4.11 – Qualité de la trajectoire générée hors-ligne : optimisation globale

Performances latérales Comme pour l'étude sur la qualité de la trajectoire, la précision du guidage latéral des véhicules 2, 3 et 4, illustrée sur la Fig. 4.12, est évaluée par rapport aux deux trajectoires : celle générée en ligne lors de l'expérimentation et celle calculée hors-ligne au terme de l'expérimentation.

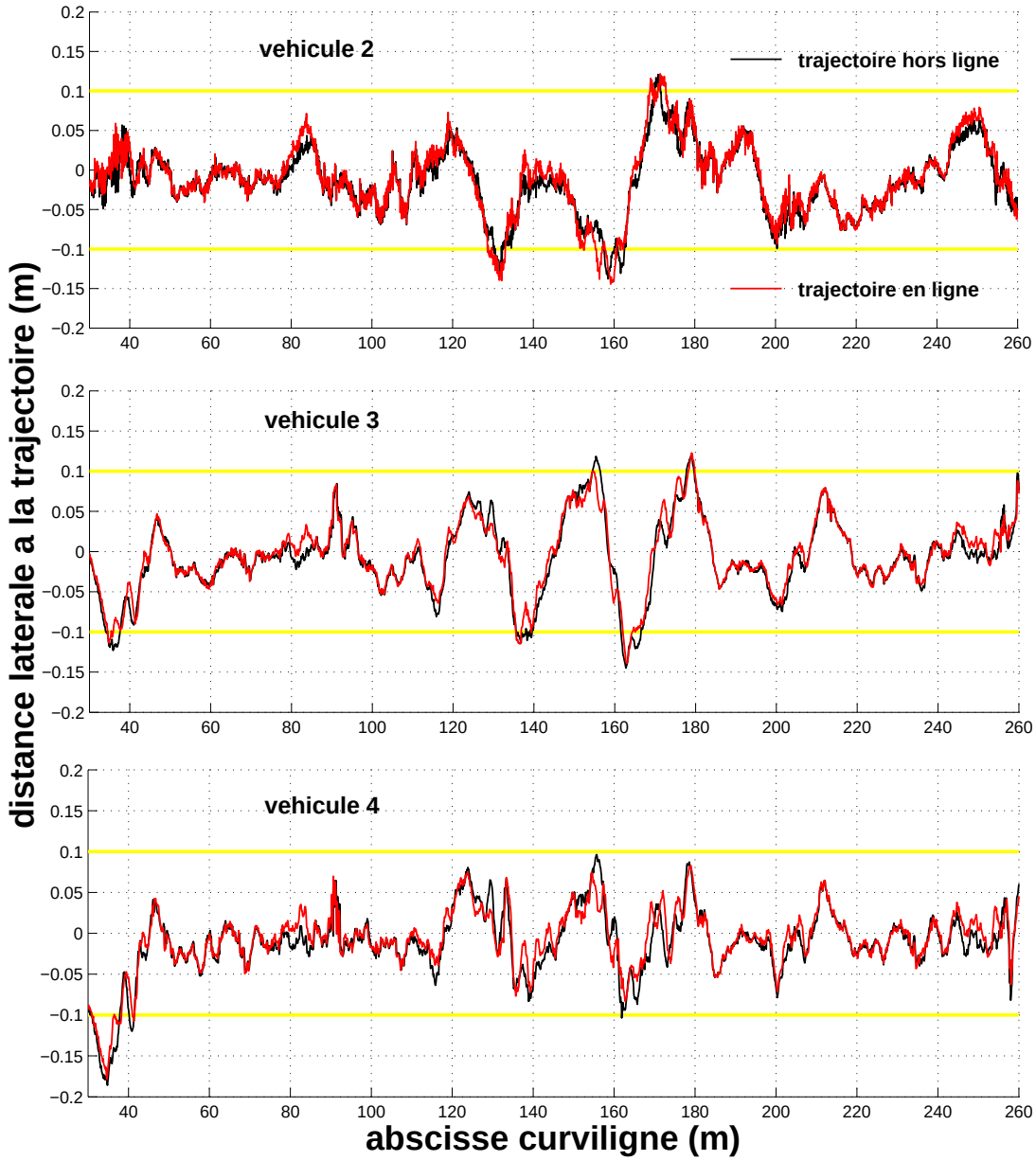


FIG. 4.12 – Performances latérales d'un convoi manuel de quatre véhicules

On constate tout d'abord que la précision du guidage latéral des véhicules 2, 3 et 4 est similaire à celle obtenue lors des travaux précédents, lorsque tous les véhicules étaient guidés par rapport à une trajectoire spécifiée par avance (voir [Thuilot 04]). On remarque aussi que, pour un même véhicule, la différence entre les écarts latéraux par rapport à l'une et l'autre des trajectoires est très faible. Plus précisément, pour le quatrième véhicule l'erreur latérale maximale s'élève respectivement à $8cm$ et $10cm$ par rapport aux trajectoires de référence en ligne et celle hors ligne. Pour le second et le troisième véhicule, ces mêmes valeurs atteignent $14cm$. Pourtant, au vu des paramètres c_2, n_{ap} et de la consigne longitudinale $d^\star$, on aurait pu s'attendre à des performances dégradées pour le deuxième véhicule : ce dernier, qui est régulé par la commande longitudinale à la distance de consigne $d^\star = 6m$ par rapport au leader, se trouve en effet en permanence sur l'une des deux dernières courbes polynômiales qui sont modifiées à chaque incrément de la trajectoire. En effet, pour cette configuration, la longueur de la plage où la trajectoire de référence n'est pas complètement figée est $n_{ap}.c_2 = 5 \times 1.5 = 7.5m$ et les deux dernières courbes s'étendent sur la portion distante du leader de $(n_{ap} - 2).c_2 = 4.5m$ à $n_{ap}.c_2 = 7.5m$, comme illustré sur la Fig. 4.13. Donc le véhicule 2 évolue bien en permanence sur une partie non figée de la trajectoire de référence. Si l'on regarde attentivement l'allure des courbes d'évolution de l'écart latéral sur la Fig. 4.12, on constate que les véhicules 3 et 4 ont un comportement semblable qui diffère légèrement de celui du véhicule 2 : l'ajustement des premiers points de contrôle altère légèrement la trajectoire de référence à son niveau et ainsi sa position relative à la trajectoire connaît de légers sauts. Pourtant, comme le montre la Fig. 4.12, l'impact de ce phénomène sur les performances latérales reste très limité.

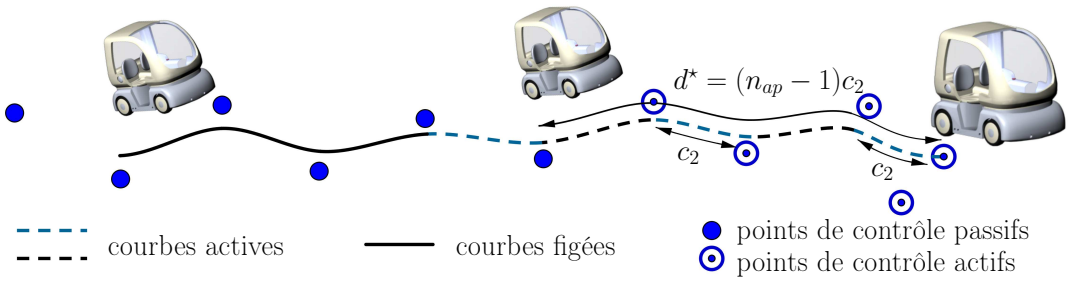


FIG. 4.13 – Projection des véhicules sur la trajectoire au cours de l'expérimentation en guidage manuel

Performances longitudinales Finalement, la précision du contrôle longitudinal est examinée sur la Fig. 4.14. Les erreurs longitudinales ont été calculées par rapport à la trajectoire de référence générée hors-ligne, en optimisant l'ensemble des points de contrôle. Une fois le convoi en mode nominal (c.a.d. chaque véhicule a atteint la distance inter-véhicule de consigne), le comportement est identique à ce qui était observé dans les travaux précédents lorsque les véhicules étaient guidés par rapport à une trajectoire spécifiée par avance (voir [Bom 05]), à savoir une précision de $10cm$. La génération de la trajectoire en ligne ne perturbe donc pas les performances longitudinales.

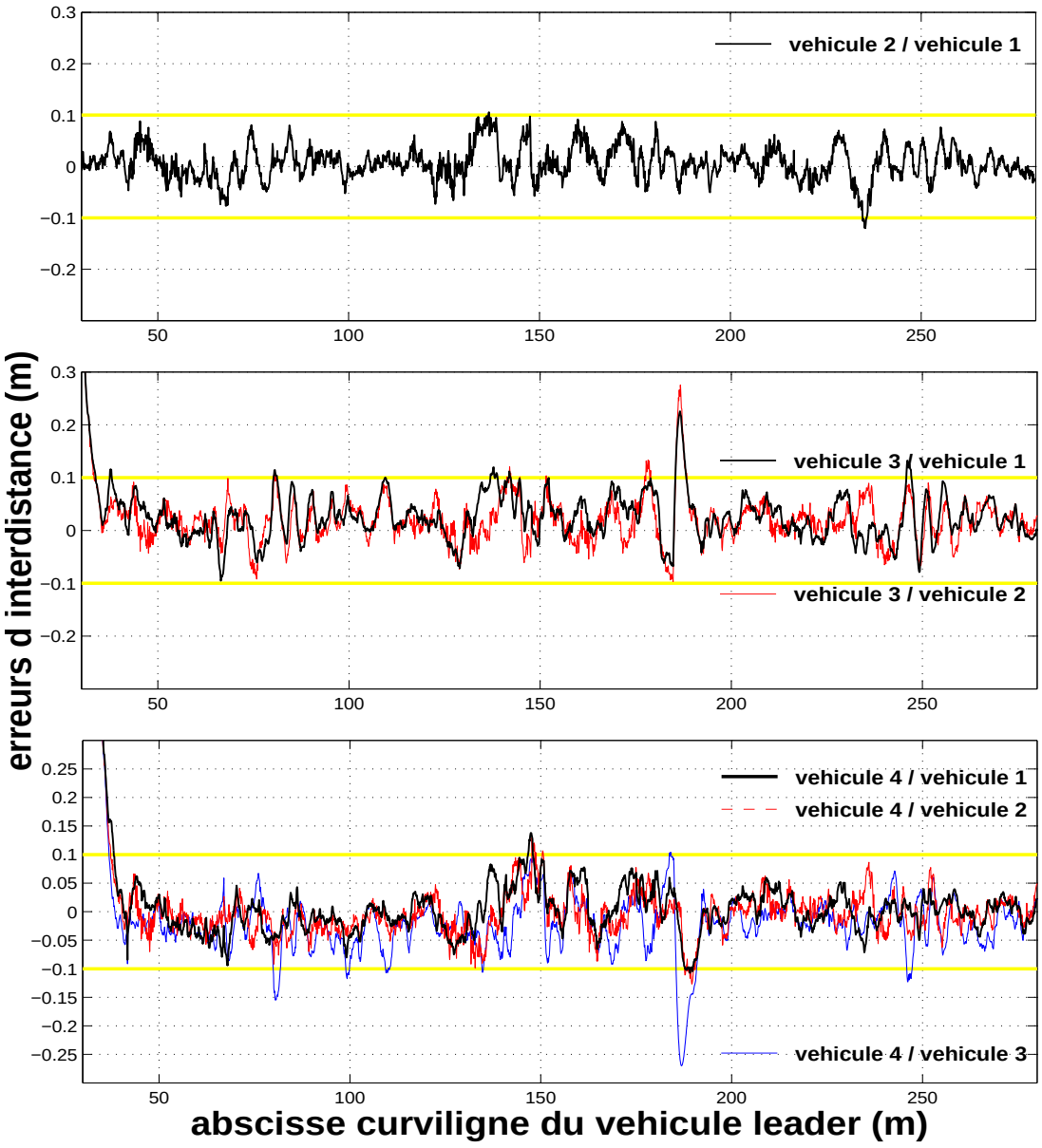


FIG. 4.14 – Performances longitudinales d'un convoi manuel de quatre véhicules

4.3.2 Guidage du convoi en mode tout automatique : localisation par vision monoculaire

Pour ces expérimentations, le convoi se compose de trois véhicules, un CyCab en tête évoluant à la vitesse constante de $1m.s^{-1}$ puis deux RobuCabs. Des tests de guidage par vision ont été effectués sur deux sites différents. Une première trajectoire, d'une longueur de $115m$ et présentée sur la Fig. 4.15 partie a), a tout d'abord été reconstruite à partir d'une séquence vidéo obtenue sur la plateforme PAVIN. Une deuxième, formant un "U" de $220m$ et présentant de longues lignes droites, a aussi été retenue pour les expérimentations et se situe sur le campus universitaire des Cézeaux à Clermont-Ferrand, voir Fig. 4.15 partie b).

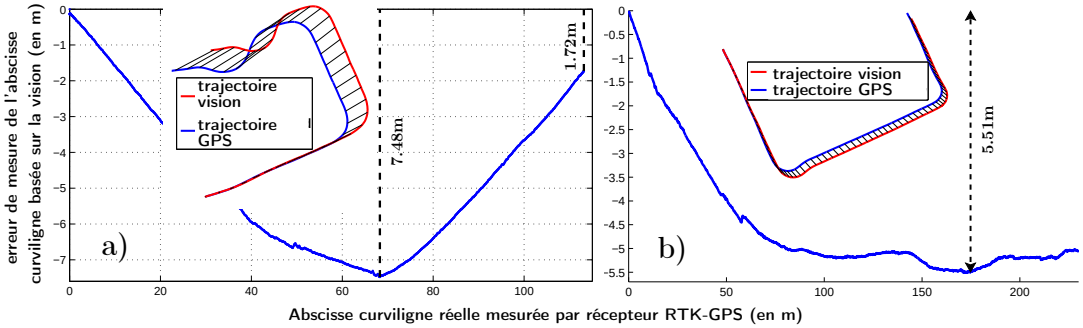


FIG. 4.15 – Trajectoires expérimentales, comparaison avec la vérité terrain :
a) site expérimental PAVIN
b) campus universitaire des Cézeaux

Sur cette figure, les trajectoires sont représentées conjointement dans le monde vision et dans un monde métrique grâce à l'utilisation d'un récepteur RTK-GPS. Comme on peut le constater, dans chaque cas les deux courbes ne se superposent pas, ce qui reflète la présence de distorsions. Les graphes associés présentent l'écart entre l'abscisse curviligne évaluée dans le monde vision et celle évaluée dans le monde métrique le long des deux trajectoires. On observe clairement que l'estimation des longueurs dans la représentation vision ne permet pas d'obtenir la précision centimétrique requise pour le guidage du convoi. Tout d'abord, on note que le facteur d'échelle global, utilisé lors de la phase de reconstruction, est imparfait. Cela était attendu, puisqu'il est calculé à partir d'une estimation de la distance totale parcourue par le véhicule, évaluée avec les données odométriques. Avec ces capteurs, une dérive est inévitable : elle s'élève ici à $1.72m$ sur la trajectoire a) et environ $5m$ sur la trajectoire b) qui est deux fois plus longue que la trajectoire a). Cependant, pour notre application, ce phénomène n'est pas le plus préoccupant. En effet, pour obtenir de bonnes performances longitudinales en convoi, ce qui est primordial c'est d'estimer avec précision les distances qui séparent les véhicules, soit de petites longueurs d'arc mesurées le long de la trajectoire. De ce point de vue, la dérive introduite par les odomètres a peu d'impact, puisque son effet

se répartit de manière uniforme le long de la trajectoire vision, ne détériorant ainsi que faiblement l'estimation des petites distances sur cette dernière. En revanche, ce qui est rédhibitoire dans notre cas, ce sont les distorsions locales que l'on observe entre monde réel et vision le long des trajectoires. On remarque en effet sur la Fig. 4.15 que la dérive maximale sur chaque trajectoire est importante, et dépasse même de beaucoup la dérive finale sur la trajectoire a), puisqu'elle s'élève à $7.48m$. Elle est de $5.51m$ sur la trajectoire b), ce qui est proche de la valeur de dérive finale. De telles distorsions ne permettent pas d'envisager la mise en œuvre d'une conduite en convoi avec les seules données de localisation vision.

4.3.2.1 Evaluation hors-ligne du facteur d'échelle

Pour les deux trajectoires considérées, l'évolution du facteur d'échelle obtenue par la méthode hors ligne (c'est-à-dire par minimisation du critère global (3.41)), à partir des données issues du traitement de la séquence de référence vision et des mesures RTK-GPS enregistrées en parallèle) est représentée sur la Fig. 4.16. Comme précédemment, les parties a) et b) correspondent respectivement aux expérimentations sur la plateforme PAVIN et à celles sur le campus des Cézeaux.

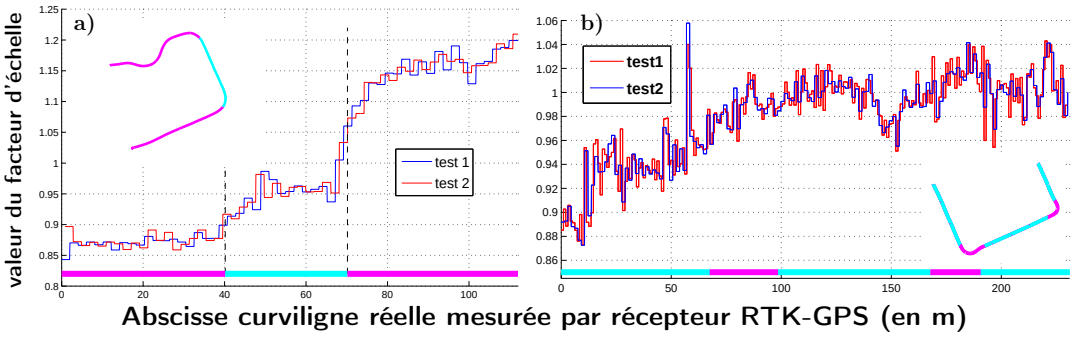


FIG. 4.16 – Estimation hors-ligne des facteurs d'échelle :
a) site expérimental PAVIN
b) campus universitaire des Cézeaux

Concernant les essais sur la plateforme PAVIN, on peut observer que l'évolution du facteur d'échelle est faible en ligne droite, comme sur la première portion de couleur magenta de la Fig. 4.16 a), et varie de manière assez abrupte en courbe. Cela s'explique par l'environnement très riche, du point de vue vision, dans lequel a évolué le véhicule : la plateforme PAVIN comporte beaucoup de façades de bâtiment, donc les points d'intérêt détectés dans les images clef sont uniformément répartis, et ceci quel que soit l'endroit où l'on se trouve sur la trajectoire. En conséquence, lorsque le véhicule avance en ligne droite, les points d'intérêt utilisés sur plusieurs images clef consécutives sont les mêmes, ce qui permet au facteur d'échelle de rester constant. C'est uniquement en courbe,

lorsque les points d'intérêt présents dans le champ de perception changent, que le facteur d'échelle varie.

Par opposition, l'environnement visuel du campus des Cézeaux est moins riche : il contient moins de structures (bâtiments et autres motifs qui procurent des amers "fiables"), de sorte que la reconstruction est faite avec moins de points d'intérêt, et ceux-ci ne sont plus répartis de façon homogène dans le champ de perception. Ainsi, bien que la trajectoire soit composée de longues lignes droites, des variations du facteur d'échelle sont occasionnées par la répartition particulière des points d'intérêt détectés dans les images. Par exemple la reconstruction des 30 premiers mètres de la trajectoire, qui présente le plus de distorsions, est impactée par la présence d'un arbre riche en amers mais présent uniquement un court instant car très proche du passage. De même, la variation en pleine ligne droite aux alentours de $s = 150m$ est due à la présence, à cet endroit précis, de voitures de part et d'autre de la voie empruntée.

4.3.2.2 Comparaison des méthodes d'évaluation en ligne du facteur d'échelle

Plusieurs approches ont été envisagées au chapitre 3 pour évaluer en ligne des corrections de facteur d'échelle, à partir de capteurs adaptés au milieu urbain. Pour la première approche, les données métriques du récepteur RTK-GPS sont remplacées par l'utilisation d'un unique télémètre laser monté sur le deuxième véhicule de la flotte. La seconde approche est basée sur un modèle d'évolution du véhicule et s'appuie uniquement sur les données odométriques classiques. Pour pouvoir évaluer les deux méthodes sans être gêné par les problèmes spécifiques amenés par le capteur télémétrique, la comparaison n'a été effectuée que sur le site du campus des Cézeaux (trajectoire b) sur les Fig. 4.15 et 4.16). En effet sur la plateforme PAVIN, la densité des objets présents dans l'environnement (façades de bâtiments, bordures, panneaux de signalisation,...) conduit à des images télémétriques avec de très nombreux échos, et l'algorithme de tracking de véhicule présenté en annexe C peut parfois être pris en défaut. De même, lors de virages à angle droit entre les façades de PAVIN, l'arrière du véhicule précédent peut être transitoirement perdu si $d^\star$ n'est pas très petit. Par conséquent, pour que la première méthode de correction en ligne puisse être mise en œuvre sur PAVIN, il faudrait au préalable raffiner l'algorithme de tracking pour le rendre plus robuste. Comme l'objectif ici est avant tout de comparer les performances des deux méthodes, nous avons préféré nous placer dans des conditions expérimentales moins sévères.

Estimations en ligne des facteurs d'échelle La Fig. 4.17 représente l'évolution du facteur d'échelle évalué hors ligne (en rouge), en ligne avec les données télémétriques (en bleu), et en ligne avec les données odométriques alimentant un observateur (en vert). Dans ce dernier cas, le gain choisi pour l'observateur est $K = 0.5$. Pour l'approche utilisant le télémètre, la consigne d'interdistance et la longueur d'un tronçon de trajectoire associé à une correction sont respectivement $d^\star = 5m$ et $\Delta s_\lambda = 0.1m$. Pour les deux méthodes en ligne, une phase d'initialisation est requise au cours de laquelle les distorsions sont supposées nulles, c'est-à-dire l'estimation du facteur d'échelle est fixée

à 1. Plus précisément, pour l'approche basée sur les données télémétriques, le second véhicule doit parcourir une distance supérieure à $d^\star$ afin que suffisamment de données télémétriques soient disponibles pour procéder à la première minimisation du critère local (3.44). Concernant la deuxième approche, l'observateur n'est activé qu'au bout de $15m$, lorsque le véhicule atteint la vitesse désirée ($v_1 = 1m.s^{-1}$). En effet, autour de la vitesse nulle, l'estimation du facteur d'échelle par observateur présente une singularité, et à faible vitesse, elle serait polluée par le bruit alors important sur les données odométriques. Par conséquent, on attend que le véhicule ait atteint sa vitesse nominale pour "lancer" l'observateur.

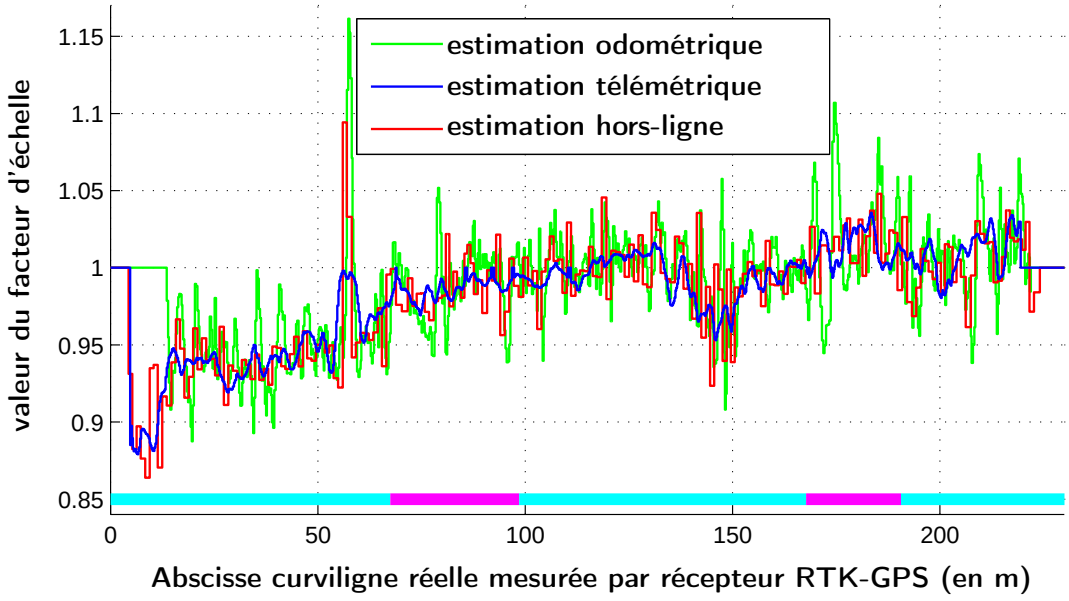


FIG. 4.17 – Estimation en ligne des facteurs d'échelle

On peut tout d'abord remarquer que les facteurs d'échelle calculés en ligne avec l'observateur ou les données télémétriques sont aussi satisfaisants que ceux calculés hors-ligne et donc très proches de la réalité terrain.

En examinant le graphe de façon plus approfondie, on constate que la courbe d'évolution de l'estimation télémétrique du facteur d'échelle est la plus lisse. Ceci est tout à fait logique au vu des outils employés pour cette méthode. On procède en effet par approximations polynômiales successives en utilisant à chaque fois de nombreuses mesures de distance par télémètre. Le degré deux choisi pour le polynôme interdit toute évolution abrupte du facteur d'échelle. Cela a pour effet de moyenner les mesures de distance entrant dans le critère local (3.44) et conduit à une évolution lisse. Cette propriété n'est pas forcément un avantage. Par exemple, on relève sur la Fig. 4.17 un pic de distorsion pour $60 \leq s_1 \leq 70$. Ce pic, bien identifié par l'estimation hors ligne et l'estimation

odométrique, est a contrario largement sous-estimé dans l'estimation télémétrique pour les raisons développées ci-dessus : la variation très ponctuelle du facteur d'échelle se matérialise alors sur un intervalle plus large, avec une ampleur moindre. On verra, lors du commentaire sur la Fig 4.20, que cela a des répercussions sur les performances de la loi de commande longitudinale.

A l'inverse la courbe d'évolution de l'estimation odométrique du facteur d'échelle apparaît comme la plus bruitée. On peut cependant vérifier que ces variations ne découlent pas de mauvaises performances de l'observateur : on constate en effet sur la Fig. 4.18 que l'erreur absolue d'observation reste majoritairement inférieure à $5cm$, avec une valeur moyenne et maximale respectivement de $2.23cm$ et $16.24cm$.

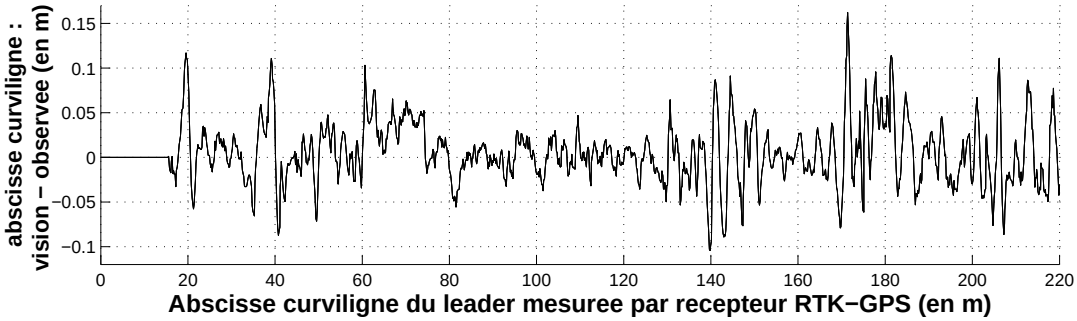


FIG. 4.18 – Erreur d'observation $\epsilon = s_1^v - \hat{s}_1^v$

Ensuite, pour expliquer l'existence de zones où les variations sur l'erreur d'observation sont plus importantes, on a tracé sur la Fig. 4.19 l'évolution du nombre de points de correspondance (inliers) entre l'image courante et l'image clef la plus proche utilisés pour localiser le véhicule au cours de son déplacement. On constate que les variations observées sur l'estimation du facteur d'échelle apparaissent lorsque l'algorithme de localisation par vision monoculaire est placé dans des situations difficiles. Ainsi, sur le passage $s_1 \in [70, 120]m$, un nombre élevé d'inliers (environ 140) est utilisé. Dans ce cas, l'erreur d'observation reste très faible, comme le montre la Fig. 4.18. Au contraire, aux alentours de $s_1 = 140m$, le nombre d'inliers descend jusque 30 et c'est bien sûr sur cette zone que l'erreur d'observation varie le plus. La qualité d'observation du facteur d'échelle est directement corrélée à la qualité de la localisation par vision.

Performances longitudinales Finalement, les performances de la loi de commande longitudinale sont évaluées sur la Fig. 4.20 en fonction du type de corrections apportées sur les données vision. Sur les parties supérieure et médiane sont présentés les écarts longitudinaux respectivement du second et troisième véhicule par rapport au véhicule leader. La partie inférieure présente l'écart longitudinal entre les deux véhicules suiveurs. Ces erreurs d'inter-distance ont été calculées grâce aux mesures RTK-GPS enregistrées en parallèle au cours des expérimentations et correctement datées par l'architecture

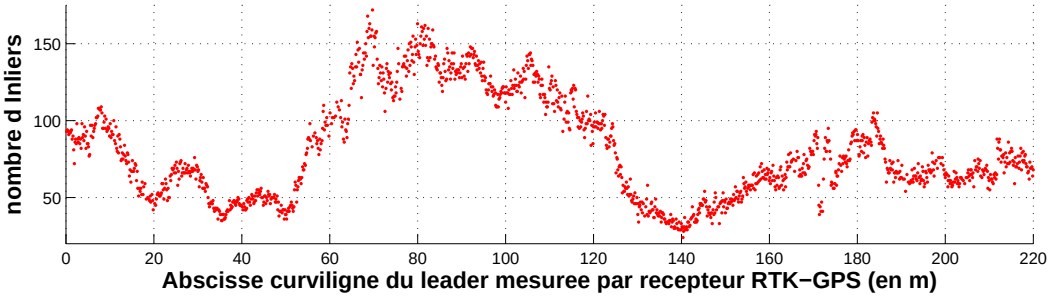


FIG. 4.19 – Nombre de points de correspondance 2D entre l'image courante et l'image clef la plus proche

logicielle AROCCAM.

Lorsque les erreurs d'inter-distance sont calculées à partir de données vision brutes, le contrôle longitudinal est largement erroné, comme l'attestent les courbes noires de la Fig. 4.20. Ces larges erreurs, à savoir $40cm$ entre le véhicule 2 et le véhicule 1 et $70cm$ entre le véhicule 3 et le véhicule 1, montrent clairement la nécessité de corriger les données vision. Comme attendu, les meilleures performances sont obtenues en utilisant les corrections calculées hors-ligne : les erreurs longitudinales sont alors du même ordre de grandeur que précédemment quand les véhicules étaient localisés à l'aide de récepteurs RTK-GPS, voir [Bom 05] : quel que soit le véhicule, ces erreurs sont très satisfaisantes puisqu'elles restent inférieures à $10cm$.

Lorsque les corrections en ligne sont utilisées, les performances restent identiques pour le deuxième véhicule et sont légèrement dégradées sur l'intervalle $s_1 \in [70, 80]m$ correspondant au pic de distorsion déjà mentionné en commentant la Fig. 4.17. Les erreurs d'inter-distance n'excèdent toutefois pas $16cm$ et $21cm$ respectivement avec l'observateur et les corrections télémétriques. Les performances supérieure obtenues sur ce passage avec l'observateur montrent l'intérêt de sa "réactivité" déjà évoquée lors de l'analyse de la Fig. 4.17. Contrairement à l'approche télémétrique, l'estimation basée observateur a pu saisir la variation rapide et locale du facteur d'échelle, ce qui permet de limiter l'erreur longitudinale transitoire. Cependant ces expérimentations montrent que les deux stratégies de correction en ligne du facteur d'échelle proposées au chapitre 3 sont effectivement pertinentes pour contrôler avec précision un convoi de véhicules par vision monoculaire.

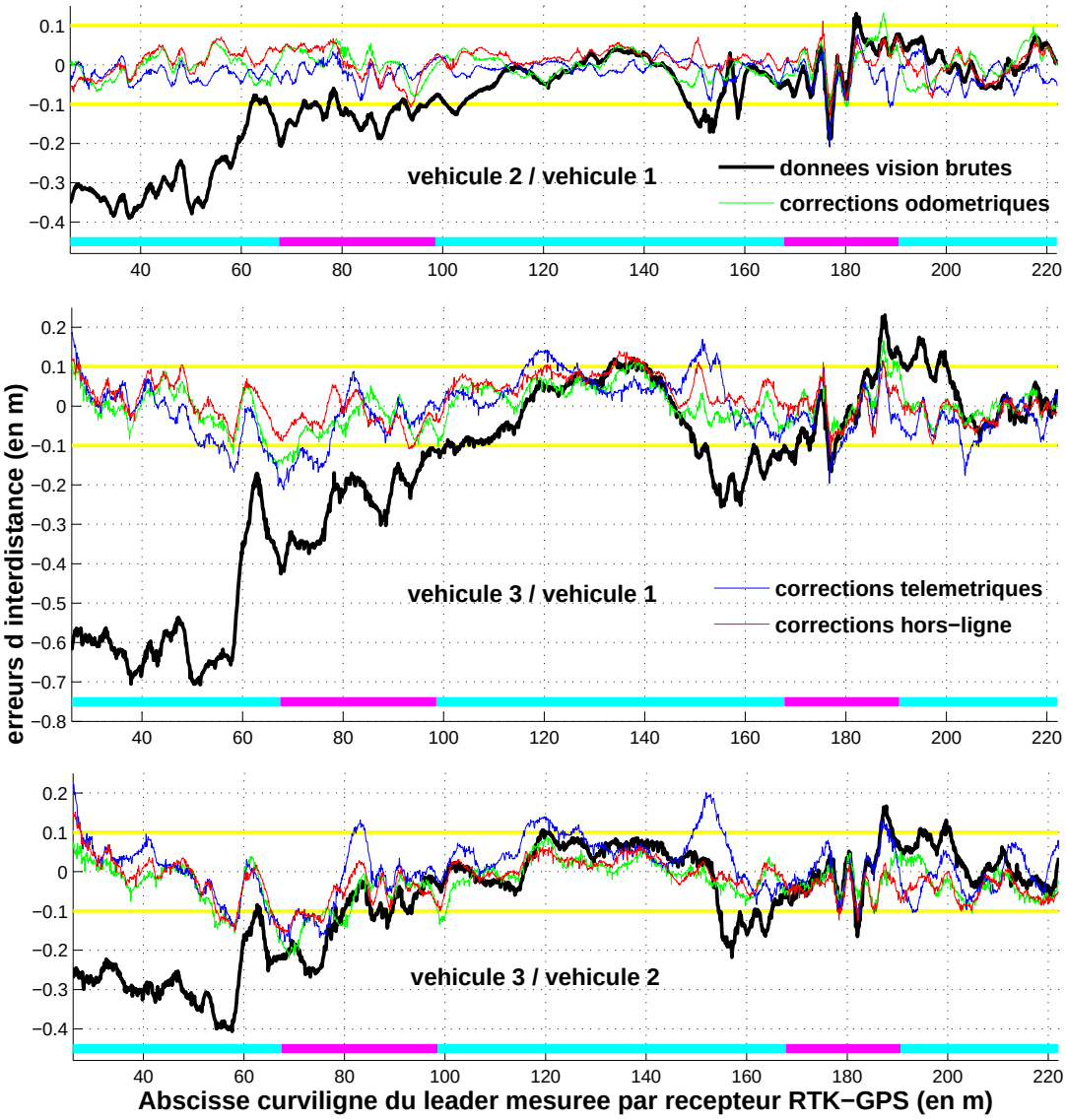


FIG. 4.20 – Performances longitudinales d'un convoi de trois véhicules guidés par vision

4.3.2.3 Validation du convoi vision sur la plateforme PAVIN

Finalement, des expérimentations de conduite en convoi par vision monoculaire ont été menées avec trois véhicules sur la plateforme PAVIN, qui représente parfaitement un environnement urbain. Pour les raisons détaillées en introduction de la section 4.3.2.2, la stratégie de correction en ligne du facteur d'échelle basée sur des données télémétriques a été écartée : seule la stratégie basée sur un observateur a été mise en œuvre et est présentée ci-dessous.

Performances longitudinales sans correction Une première expérimentation de conduite en convoi a été réalisée en utilisant les données vision brutes. Les résultats sont présentés sur la Fig. 4.21. Les distorsions sur la trajectoire du site PAVIN étant beaucoup plus importantes que sur celle du campus (le facteur d'échelle varie entre 0.84 et 1.21 sur le site PAVIN, voir Fig. 4.16), sans surprise les erreurs longitudinales sont plus importantes sur PAVIN que sur le campus des Cézeaux : on relève sur la Fig. 4.21 jusque $80cm$ entre le véhicule 1 et le véhicule 2, et $1.5m$ entre le véhicule 1 et le véhicule 3. Cela illustre bien que les fortes distorsions entre le monde vision et le monde métrique, si elles ne sont pas corrigées, conduisent à des performances inacceptables pour la commande longitudinale.

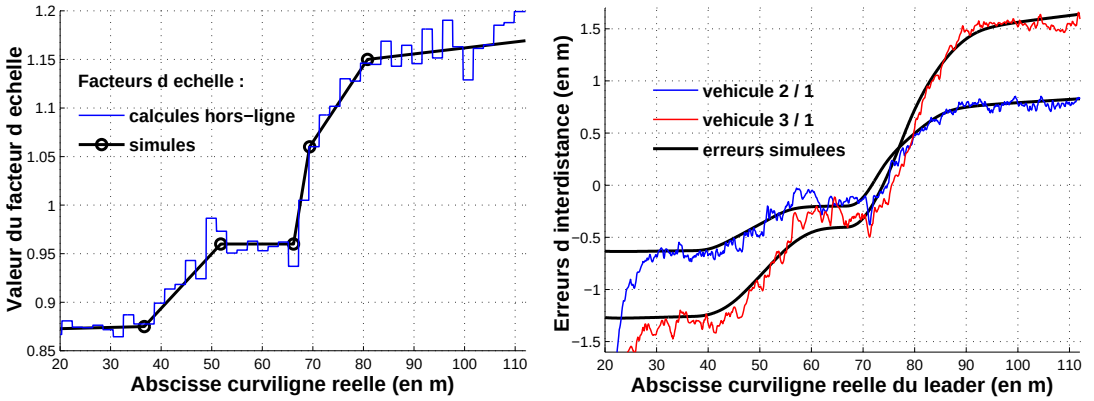


FIG. 4.21 – Comparaison des performances simulées / en conditions réelles

Cette expérimentation permet également de vérifier que les simulations présentées au chapitre 3 reflètent parfaitement la réalité terrain : pour que ces simulations soient représentatives de conditions réelles, l'évolution du facteur d'échelle avait été décrite par une jonction de segments de droite suivant grossièrement l'estimation hors ligne du facteur d'échelle le long de la trajectoire du site PAVIN, voir Fig. 3.4 et partie gauche de la Fig. 4.21. La partie droite de la Fig. 4.21 montre que les erreurs longitudinales obtenues en simulation avec cette description simplifiée pour l'évolution du facteur d'échelle, et les erreurs longitudinales enregistrées lors de l'expérimentation de conduite en convoi sur cette même trajectoire du site PAVIN sont parfaitement superposées. Ceci renforce

a posteriori la pertinence de l'ensemble des simulations présentées au chapitre 3.

Estimations en ligne des facteurs d'échelle L'évolution du facteur d'échelle calculée en ligne à partir de données odométriques est représentée sur la Fig. 4.22. Le gain choisi pour l'observateur est toujours $K = 0.5$. La moyenne des facteurs d'échelle sur des sections de $2m$ a été tracée en rouge afin de faciliter la comparaison avec les facteurs d'échelle évalués hors-ligne, reportés en bleu. On constate à nouveau, en dépit d'une plage plus étendue pour les distorsions et de la présence de deux zones où leur variation est très brusque ($s_1 \in [40, 50]m$ et $s_1 \in [70, 80]m$), que les facteurs d'échelle calculés en ligne sont aussi satisfaisants que ceux calculés hors-ligne et donc très proches de la réalité terrain obtenue par les mesures fournies par les récepteurs RTK-GPS.

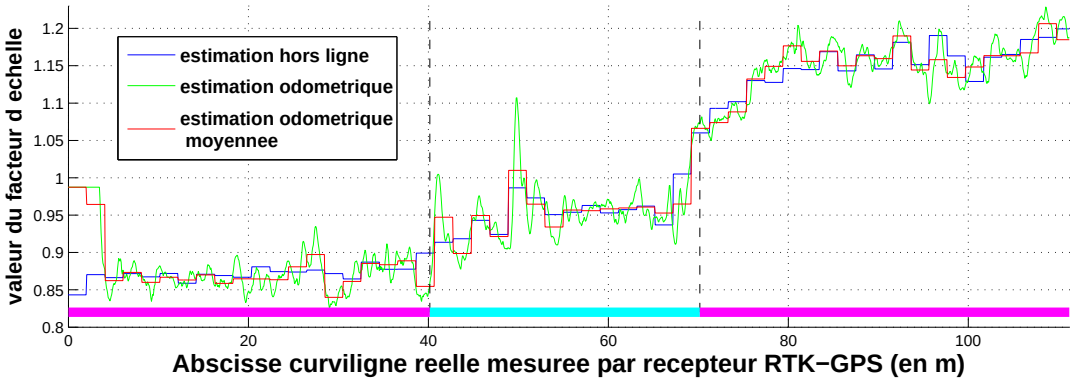


FIG. 4.22 – Estimation en ligne des facteurs d'échelle sur PAVIN

Performances longitudinales Les performances de l'asservissement longitudinal avec les données vision corrigées sont présentées sur la Fig. 4.23. On observe tout d'abord que l'estimation en ligne des facteurs d'échelle permet aux erreurs longitudinales évaluées par vision (courbe rouge) de présenter des valeurs cohérentes avec l'erreur longitudinale mesurée à l'aide des récepteurs RTK-GPS (courbe noire). En conséquence, les performances de la commande longitudinale par modalité vision sont similaires, comme il était souhaité, à celles obtenues précédemment par modalité RTK-GPS [Bom 05]. Les erreurs longitudinales restent de façon satisfaisante inférieures à $10cm$. Les performances sont juste légèrement dégradées sur l'intervalle $s_1 \in [70, 80]m$ sur lequel le facteur d'échelle varie de façon abrupte. Cependant, même dans ce cas, les erreurs d'inter-distance n'excèdent pas $14cm$ entre le véhicule 1 et le véhicule 2, et $17cm$ entre le véhicule 1 et le véhicule 3.

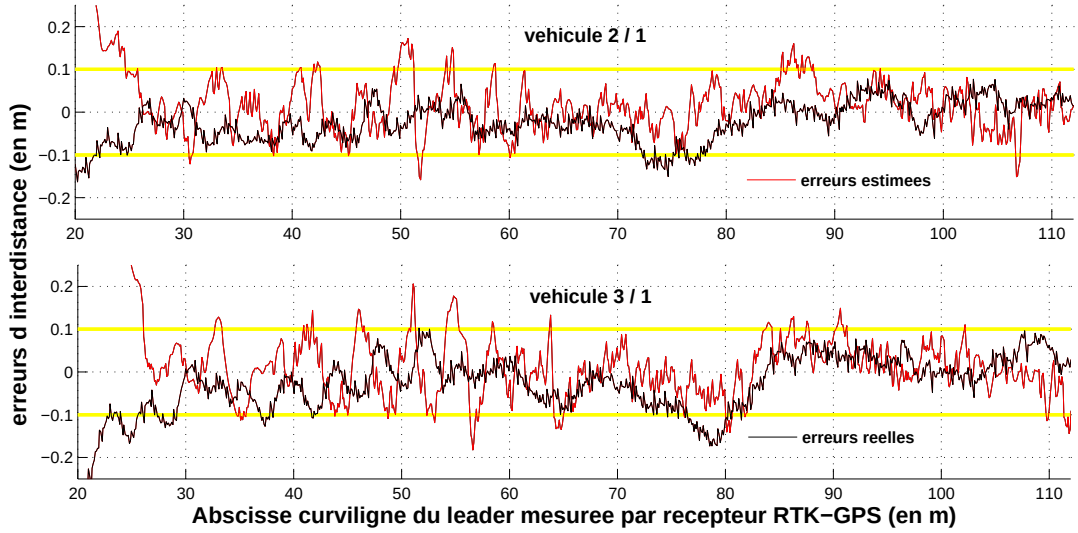


FIG. 4.23 – Performances longitudinales d'un convoi de trois véhicules guidés par vision sur PAVIN

Chapitre 5

Conclusion

Mes travaux de thèse s'inscrivent dans le cadre des développements sur la mobilité urbaine. Afin d'adresser les problèmes de pollution et d'engorgement des centre ville, une alternative de transport intéressante consiste en la mise en place d'une flotte de véhicules électriques autonomes accessibles en libre service depuis des stations relais. Ce système pourrait aussi servir pour la desserte de sites dédiés (aéroports, parcs, ...). La fonctionnalité de navigation en convoi permettrait alors une maîtrise accrue de la circulation et de pouvoir adapter le service à la demande. Dans ce contexte, des travaux ont déjà été développés au LASMEA. Une stratégie de commande globale a été définie. Elle repose sur le partage des informations d'état du véhicule leader avec l'ensemble de la flotte. La stabilité et la sécurité du convoi sont assurées conjointement par la régulation d'une variable de commande longitudinale hybride qui traduit l'erreur de positionnement avec le leader et aussi avec le véhicule qui précède directement dans le convoi. Cette stratégie a été validée expérimentalement avec trois véhicules en mode tout automatique. La localisation de chaque véhicule dans un référentiel métrique était alors fournie par des capteurs RTK-GPS.

Contributions

Afin d'apporter plus de flexibilité à l'application en convoi, mes premiers travaux ont été axés sur la mise en place d'un nouveau mode de navigation, dans lequel le premier véhicule est cette fois conduit manuellement et entraîne dans son sillage les autres véhicules, qui doivent alors suivre précisément sa trace. Dans cette optique, la représentation de la trajectoire de référence devrait être révisée pour permettre de définir une procédure d'extension en ligne décrivant le déplacement du véhicule leader. Nous avons montré l'intérêt de se tourner vers des courbes B-Splines. Tout d'abord, cette description est compacte (la trajectoire est complètement caractérisée par la donnée des points de contrôle) et permet de calculer très facilement les variables utiles pour les lois de commande (abscisse curviligne, écarts, courbure, ...). Elle est donc intéressante, y compris lorsque la trajectoire de référence est spécifiée au préalable (mode tout automatique). Nous avons ensuite proposé une stratégie de création en ligne de la trajectoire

pour caractériser à tout moment le parcours du véhicule leader et établir le chemin de référence aux autres membres du convoi. La trajectoire est construite de manière incrémentale à partir des positions du véhicule de tête, en considérant une fenêtre glissante sur ces données. Pour étendre la trajectoire, sans pour autant perturber les portions déjà construites sur lesquelles les véhicules suiveurs sont engagés, la procédure d'extension a été traitée de façon itérative et abordée comme un problème d'optimisation. Plus précisément, un critère de minimisation local est défini et permet, lorsqu'une nouvelle donnée de localisation est disponible, d'approximer uniquement les points de contrôle qui modélisent l'extrémité de la trajectoire. La méthode a été éprouvée en simulation : des batteries de tests ont mis en évidence l'influence des paramètres de l'algorithme d'extension et ont permis de caractériser les meilleurs d'entre eux selon des critères de fidélité de la trajectoire et de perturbations occasionnées sur celle-ci au cours de la création.

Ensuite, le capteur RTK GPS utilisé jusqu'ici n'était clairement pas adapté au milieu urbain (en raison du risque élevé de masquage des signaux satellite) ni à l'application visée (son coût est prohibitif pour équiper une flotte de véhicules). A l'inverse, les caméras paraissent bien appropriées, en raison de la richesse des structures urbaines pour le traitement d'images et leur faible coût. Au LASMEA, une stratégie de localisation dans un repère absolu avait été développée, basée sur la vision monoculaire, reposant sur une phase d'apprentissage au cours de laquelle une reconstruction 3D de l'environnement est effectuée. Cependant, la localisation absolue provenant de l'algorithme de vision est exprimée dans ce monde reconstruit qui comporte des distorsions locales par rapport au monde réel et affecte l'estimation des distances. Cela gêne peu la navigation autonome d'un véhicule seul, puisque les distorsions agissent comme une modification de gain pour la commande latérale. Mais dans le cas d'un convoi, l'erreur commise sur l'évaluation des distances inter-véhicule est rédhibitoire. Par conséquent, il n'est pas envisageable d'utiliser directement les informations fournies par l'algorithme pour alimenter les lois de commande du convoi. Une analyse empirique de ces distorsions a été effectuée par comparaison de données de localisation obtenues dans le monde vision et dans le monde réel. De cette manière il a pu être montré que les distorsions relevées étaient localement homogènes et qu'il était possible de redonner un sens métrique aux distances mesurées dans le monde vision à partir d'un jeu de facteurs d'échelle estimés localement le long de la trajectoire de référence. Dans l'optique de mettre en œuvre l'approche de commande globale du convoi à partir des informations de localisation par vision monoculaire, plusieurs stratégies ont été élaborées pour obtenir les facteurs d'échelle et recouvrer les informations d'état des véhicules dans un référentiel métrique. Une stratégie hors-ligne a tout d'abord été proposée. Elle exploite, selon un processus d'optimisation, des distances obtenues par vision monoculaire et les mêmes quantités mesurées dans le monde réel à l'aide d'un récepteur RTK-GPS. Dans la perspective de réaliser un guidage manuel du convoi avec une localisation par vision monoculaire, deux autres méthodes ont été proposées pour évaluer les facteurs d'échelle en-ligne cette fois, à l'aide de capteurs adaptés au milieu urbain. Pour la première approche, les données métriques du récepteur RTK-GPS sont remplacées par l'utilisation d'un unique télé-

mètre laser monté sur le deuxième véhicule de la flotte et opérant le suivi du véhicule de tête dans l'image télémétrique. Les distorions sont alors évaluées à partir d'une approximation polynômiale du facteur d'échelle sur une portion de la trajectoire. Une approche alternative, s'appuyant uniquement sur des données odométriques, est considérée pour la deuxième approche. Elle est basée sur un observateur et le modèle d'évolution du véhicule. Pour ces deux approches, des batteries de simulations ont été réalisées pour évaluer l'influence des paramètres et mettre en évidence les meilleurs configurations à utiliser en vue d'applications expérimentales.

Au cours de cette thèse, un travail conséquent a été fourni pour amener à maturité l'implémentation des algorithmes de commande en convoi. La principale difficulté réside dans la gestion des données qui sont récupérées de façon asynchrone, d'abord au niveau du véhicule lui-même, et bien sûr au niveau des communications inter-véhicule. Afin de gérer correctement le recueil des données, des requêtes NTP sont envoyées à intervalles réguliers pour accorder l'horloge des véhicules à celle d'un serveur et ainsi partager une référence de temps commune au sein du convoi. Le middleware AROCCAM permet alors de collecter et estampiller les données, ainsi que de gérer le cadencement de la tâche qui consiste au calcul et à l'envoi des commandes. A cet instant, les données les plus récentes sont alors utilisées pour alimenter le modèle des véhicules, qui est intégré par la méthode classique de Runge-Kutta d'ordre quatre, afin de pouvoir disposer d'une estimation précise du convoi lors du calcul de la commande. Les contributions théoriques concernant les algorithmes de commande pour la conduite urbaine ont ainsi pu être mises en œuvre dans un cadre expérimental avec des véhicules de type CyCab et RobuCab. Les expérimentations dans un mode de navigation manuel ont été testées avec quatre véhicules. Les résultats attestent que les membres du convoi ont suivi la trajectoire générée en ligne par le leader de manière aussi satisfaisante que si cette dernière avait été établie au préalable. Des tests de guidage ont aussi été réalisés avec trois véhicules en considérant la modalité capteur par vision monoculaire. Les trois approches proposées ont été comparées et des performances de commande très satisfaisantes, avec le maintien des erreurs d'inter-distance sous les $10cm$, montrent ainsi la pertinence des corrections et l'efficacité des méthodes proposées.

Limites et perspectives

De nombreuses perspectives sont envisageables avec un tel système. Quelques-unes sont repertoriées ici.

Au cours de nos expérimentations, la vitesse des membres du convoi s'aligne, en phase nominale, sur celle du leader qui est maintenue constante. Cette modalité est naturelle en mode tout automatique lorsque les véhicules circulent sur voie dédiée, pratiquant de rares arrêts. Cependant, si le convoi est dirigé manuellement par un opérateur, alors l'hypothèse n'est plus réaliste puisque la vitesse du véhicule de tête fluctue en fonction du pilotage. Il serait alors intéressant que les autres véhicules puissent anticiper ce comportement et réagir dans une telle situation. De meilleures performances pourraient être obtenues si des informations concernant l'accélération de leader étaient

transmises à toute la flotte. Il faudrait également utiliser un modèle des actionneurs pour exploiter avec précision cette information.

Dans le cas du guidage par vision monoculaire, on a vu dans l'état de l'art présenté au chapitre 3, que la connaissance du déplacement particulier de la caméra, montée sur un véhicule, peut être exploitée lors de la reconstruction 3D. Dans [Fraundorfer 10], les auteurs supposent que la caméra effectue localement un mouvement circulaire. De cette façon, une estimation de la pose relative entre deux positions successives peut s'obtenir très facilement et avec précision à l'aide d'une unique correspondance d'un amer entre les deux images. Je pense que cette idée peut être approfondie et que d'autres solutions, qui s'appuient sur des modèles d'évolution plus réalistes, peuvent être explorées. L'utilisation de courbes paramétrées permettrait notamment de représenter l'allure de la trajectoire caméra de manière plus variée. Des contraintes pourraient alors être intégrée sur le mouvement de la caméra pour traduire la non holonomie du système ainsi que ses limites mécaniques. La reconstruction 3D serait aussi facilitée puisque les poses de la caméra ne dépendraient plus que des quelques paramètres qui caractérisent la courbe considérée.

Après avoir lu cette thèse, une perspective des plus intéressantes se dégage et concerne la mise en œuvre d'un convoi en mode manuel avec une localisation par vision monoculaire. Comme première étape, on peut dans un premier temps considérer que la reconstruction 3D de l'environnement est disponible, obtenue au préalable via une phase d'apprentissage. La stratégie de création de trajectoire, décrite au chapitre 2 peut ensuite être mise en place pour générer en ligne la trajectoire vision d'un véhicule qui évolue et se localise dans l'environnement reconstruit. Grâce à l'une des méthodes de correction en ligne des distorsions, décrites au chapitre 3, il devrait être possible d'obtenir de bonnes performances longitudinales. Dans un deuxième temps, on pourrait considérer d'utiliser l'algorithme proposé dans [Mouragnon 06], qui réalise une reconstruction en temps réel de l'environnement. La difficulté serait alors accentuée par l'utilisation de données de localisation plus bruitées et la présence de distorsions plus fortes dans la reconstruction.

Annexes

Annexe A

Eléments de géométrie projective et épipolaire pour la reconstruction 3D à partir d'une séquence d'images

Introduction

L'objectif de cette annexe est de présenter plus en détail les éléments de vision géométrique à la base de l'algorithme de localisation 3D par vision monoculaire utilisé dans cette thèse pour commander un convoi de véhicules. L'algorithme de localisation s'appuie sur une reconstruction 3D de la scène. Les briques élémentaires permettant cette reconstruction sont présentées ici, puis l'algorithme complet avec les raffinements apportés est décrit à l'annexe B. Plus précisément, dans cette annexe, la modélisation d'une caméra est tout d'abord rappelée, puis les différents algorithmes utilisés pour la reconstruction 3D sont présentés. Enfin, la stratégie de reconstruction 3D incrémentale, basée sur les principes de la géométrie épipolaire, est détaillée.

Formulation du problème Le processus de formation de l'image peut être formalisé assez simplement. Le modèle le plus courant pour une caméra est le modèle sténopé représenté Fig. A.1. A partir de ce modèle, il est facile de déterminer la position dans l'image de la projection d'un point dont la position 3D est connue. Le principe est rappelé à la section A.1. Le problème inverse ne peut pas être résolu si on ne dispose que d'une seule image. Il faut utiliser la projection du point dans deux images au minimum pour obtenir une solution. Dans ce cas, à condition que la position et l'orientation dans le repère monde des deux caméras soient connues, la solution peut être obtenue par triangulation avec la construction géométrique simple représentée sur la Fig. A.2. Dans la pratique, il est fréquent qu'on ne dispose pas de ces informations sur les caméras, en particulier lorsque la caméra est en mouvement (par exemple montée sur un véhicule). Des algorithmes permettant d'évaluer les positions et orientations des caméras (appelées poses), ainsi que les positions des objets de la scène dans le repère monde à partir d'une séquence vidéo sont présentés à la section A.2.

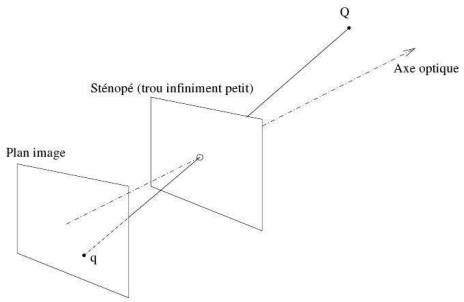


FIG. A.1 – Projection d'un point grâce au modèle sténopé d'une caméra

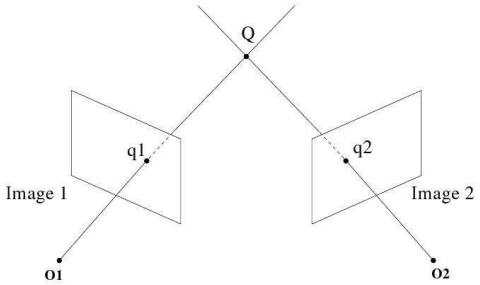


FIG. A.2 – Triangulation d'un point

A.1 Modèle sténopé d'une caméra

Il s'agit du modèle le plus simple pour une caméra : l'objectif est décrit par un trou infiniment petit situé à une distance f du plan où se forme l'image. On appelle distance focale la longueur f. A partir de ce modèle, la projection d'un point de coordonnées (X_C, Y_C, Z_C) dans le repère caméra est représentée par un point de coordonnées (x_i, y_i) dans le plan de l'image, avec $x_i = X_C/Z_C$ et $y_i = Y_C/Z_C$. Ces relations se traduisent de façon équivalente en introduisant les coordonnées homogènes :

$$\begin{bmatrix} sx_i \\ sy_i \\ s \end{bmatrix} \equiv \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} \tag{A.1}$$

où le signe $\equiv$ désigne une égalité à un facteur multiplicatif près. Enfin, la position exprimée en pixels du point image sur le capteur, notée (x_{pu}, y_{pu}), est donnée par :

$$\begin{bmatrix} x_{pu} \\ y_{pu} \\ 1 \end{bmatrix} \equiv \begin{bmatrix} f/dx & 0 & u_0 \\ 0 & f/dy & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \equiv K \cdot \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \tag{A.2}$$

où le couple (u_0, v_0) est la position en pixels du centre de l'image, dx et dy sont les dimensions des pixels de l'image et K représente la matrice des paramètres intrinsèques.

Les points observés par la caméra sont exprimés dans le repère du monde R_W. Il faut au minimum six paramètres, appelés paramètres extrinsèques, pour définir la position et l'orientation de la caméra dans R_W. Ils correspondent aux coordonnées du centre optique de la caméra dans le repère du monde $T = [T_x, T_y, T_z]^T$ et à la matrice de rotation R entre R_W et le repère caméra R_C. Soit un point $Q(X_W, Y_W, Z_W)$ dont les coordonnées sont données dans le repère du monde. Ses coordonnées (X_C, Y_C, Z_C) dans le repère caméra sont données par la relation :

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} \equiv \begin{bmatrix} R & -RT \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} \tag{A.3}$$

En regroupant (A.1), (A.2) et (A.3), on obtient que les coordonnées d'un point dans le repère monde sont reliées aux coordonnées en pixels de son point image via un simple produit de matrices :

$$\begin{bmatrix} x_{pu} \\ y_{pu} \\ 1 \end{bmatrix} \equiv P \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} \tag{A.4}$$

Dans cette équation P est de taille 3×4, on l'appelle matrice de projection associée à la caméra, ou parfois matrice de la caméra.

A.2 Géométrie épipolaire

Dès lors qu'on dispose de plusieurs images d'une même scène, il devient intéressant d'établir des relations liant le déplacement de la caméra à celui de la projection des points de la scène dans les images. Les paragraphes suivants ont pour but de présenter les principales méthodes.

A.2.1 Matrice essentielle

On s'intéresse au problème suivant : on connaît la projection de n points de l'espace dans deux images et on souhaite déterminer la pose relative des caméras à partir de cette information. Lorsque la caméra est calibrée, ce problème peut être adressé en utilisant la notion de matrice essentielle, introduite initialement dans [Longuet-Higgins 87].

A.2.1.1 Définition

Un couple de points pris dans chacune des deux images est dit couple de points homologues s'ils sont l'image d'un même point de la scène. Il a été établi que les coordonnées homogènes de tout couple de points homologues q et q' sont liées par une matrice E appelée matrice essentielle :

$$q'^T E q = 0 \tag{A.5}$$

avec $q = (x, y, 1)^T$ et $q' = (x', y', 1)^T$, où (x, y) et (x', y') sont les coordonnées des points homologues dans le plan de chacune des deux images (et non les coordonnées exprimées en pixels). La relation (A.5) étant indépendante de la scène, il est possible de calculer E à partir de la connaissance des coordonnées des points dans les images seulement. L'algorithme aujourd'hui le plus efficace utilise cinq couples de points homologues, il est décrit à la section A.2.1.2 ci-dessous. Lorsqu'on connaît E, il est possible de calculer la pose relative des caméras puisqu'on peut montrer que :

$$E = [T]_\times R \tag{A.6}$$

où R et T désignent respectivement la matrice de rotation et le vecteur de translation entre les deux poses caméra, et $[T]_\times$ désigne la matrice antisymétrique construite à partir

de T. La reconstruction qu'on peut obtenir à partir du calcul de la matrice essentielle est une reconstruction euclidienne, ce qui signifie qu'elle est définie à une similitude près : le choix de la position de l'origine, de l'orientation et de l'échelle est défini de façon arbitraire.

A.2.1.2 Algorithme des 5 points pour deux vues

Le calcul de la matrice essentielle E caractérisant deux poses caméra peut être réalisé en utilisant uniquement cinq couples de points homologues (q^i, q'^i) avec $i \in \{1, \ldots, 5\}$. Pour cela on écrit tout d'abord la relation (A.7) qui traduit que (q^i, q'^i) sont homologues, et on ajoute à ce système les contraintes (A.8) qui traduisent que toute matrice essentielle est nécessairement de rang deux et que ses deux valeurs singulières non nulles sont égales. Un algorithme permettant de résoudre (A.7) sous les contraintes (A.8) a été proposé dans Nistèr [Nistér 03c] et amélioré par la suite dans [Nistér 04].

$$q'^{i^T} E q^i = 0, \ \forall i \in \{1 \ldots 5\} \tag{A.7}$$

$$EE^T E - \frac{1}{2} trace(EE^T)E = 0 \tag{A.8}$$

A.2.1.3 Méthode de RANSAC

Dans la pratique, si l'appariement est fait de façon automatique, les points ne sont pas tous bien appariés. Il existe parmi les n couples de points homologues proposés, une proportion non négligeable de faux appariements. Pour que ceux-ci ne faussent pas les différents calculs à réaliser, la méthode de RANSAC introduite par [Fischler 81] qui propose de faire plusieurs fois les calculs en prenant à chaque fois aléatoirement un échantillon de taille minimale (cinq dans le cas d'une matrice essentielle) parmi l'ensemble des points homologues proposés. Chaque fois qu'un échantillon est considéré, on vérifie ensuite que la solution obtenue est cohérente avec chacun des appariements. Finalement, c'est la solution qui donne le plus grand nombre d'appariements cohérents qui est retenue.

A.2.2 Quelques algorithmes

Les quelques algorithmes décrits ci-dessous permettent d'adresser les problèmes classiques en géométrie épipolaire et sont notamment exploités pour reconstruire la trajectoire 3D d'une caméra à partir d'une séquence d'images.

A.2.2.1 Triangulation : méthode du point milieu

Soient (q_1, q_2) un couple de points homologues, et Q le point de la scène dont ils sont l'image. S'il n'y avait aucun bruit de mesure, alors les coordonnées 3D du point Q pourraient s'obtenir facilement en écrivant que Q est à l'intersection des droites $(O_1 q_1)$ et $(O_2 q_2)$, où O_1 et O_2 sont les centres optiques des deux caméras, voir Fig. A.3. Malheureusement, dans la pratique, ces deux droites se coupent rarement. Pour obtenir

une estimation pertinente des coordonnées 3D de Q, la méthode du point milieu propose de procéder comme suit :

- On calcule tout d'abord le vecteur $\overrightarrow{v} = \overrightarrow{O_1q_1} \wedge \overrightarrow{O_2q_2}$ qui définit la direction de la perpendiculaire commune aux droites (O_1q_1) et (O_2q_2) ($\wedge$ désigne le produit vectoriel).
- On calcule ensuite l'intersection, notée Q_1, entre la droite (O_1q_1) et le plan Π_2 contenant la droite (O_2q_2) et le vecteur $\overrightarrow{v}$ (si les mesures étaient parfaites, Q_1 serait bien sûr sur (O_2q_2)). On montre que :

$$\overrightarrow{O_1Q_1} = \alpha\overrightarrow{O_1q_1} \tag{A.9}$$

avec

$$\alpha = \frac{\left(\overrightarrow{O_1O_2} \cdot \overrightarrow{O_2q_2}\right)\left(\overrightarrow{O_1q_1} \cdot \overrightarrow{O_2q_2}\right) - \overrightarrow{O_1O_2} \cdot \overrightarrow{O_1q_1}\ \|\overrightarrow{O_2q_2}\|^2}{\left(\overrightarrow{O_1q_1} \cdot \overrightarrow{O_2q_2}\right)^2 - \|\overrightarrow{O_1q_1}\|^2\|\overrightarrow{O_2q_2}\|^2} \tag{A.10}$$

- De la même façon, on calcule l'intersection, notée Q_2, entre la droite (O_2q_2) et le plan Π_1 contenant (O_1q_1) et le vecteur $\overrightarrow{v}$. La droite (Q_1Q_2) ainsi obtenue est la perpendiculaire à (O_1q_1) et (O_2q_2).
- Enfin, pour répartir l'erreur de reprojection entre les deux images, on définit Q comme étant le milieu du segment $[Q_1Q_2]$, voir Fig. A.3.

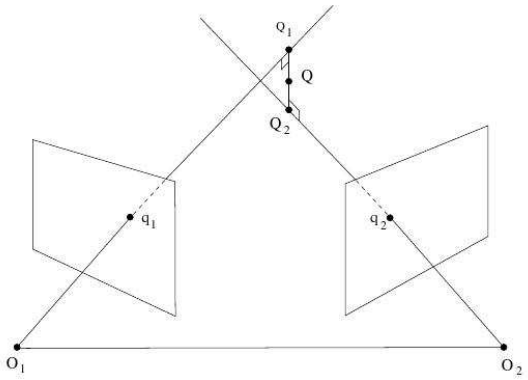


FIG. A.3 – Triangulation de points vus dans deux images

A.2.2.2 Calcul de pose à partir de points 3D : méthode de Grunert

Pour calculer la pose de la caméra, il faut connaître au minimum la projection de trois points 3D sur le plan image. Plusieurs méthodes de calcul peuvent être utilisées. Les principales sont détaillées et comparées par Haralick et al. [Haralick 94]. Parmi celles présentées, celle de Grunert [Grunert 41] est l'une des plus stables numériquement. Cette méthode est basée sur des calculs trigonométriques (formule d'Al-Kashi) dans le tétrahèdre formé par le centre optique de la caméra et les trois points 3D considérés.

Le calcul de la pose de la caméra se ramène à la résolution d'une équation polynômiale de degré quatre. Il peut donc y avoir quatre solutions même si le plus souvent il n'y en a que deux. L'utilisation de la méthode RANSAC permet de faire le tri parmi les solutions.

A.2.2.3 Algorithme des 5 points pour trois vues

L'algorithme des 5 points présenté à la section A.2.1.2 permet de calculer la pose relative entre deux images. Cependant, dans quelques situations, si la scène est plane et tous les points sont plus proches d'une des deux caméras, alors cet algorithme peut renvoyer plusieurs solutions. Aussi, pour éviter de telles singularités, on préfère généralement calculer les poses relatives en utilisant trois images. L'algorithme est toujours basé sur l'utilisation de la méthode RANSAC :

- A partir d'un échantillon de cinq points en correspondance dans les images 1, 2 et 3, on commence par calculer la matrice essentielle entre les caméras 1 et 3 avec l'algorithme des 5 points présenté dans A.2.1.2. On en déduit le déplacement de la caméra entre ces deux vues.
- Ensuite, on triangule les 5 points (par la méthode du point milieu, section A.2.2.1) pour obtenir leur position 3D. Ces points 3D sont alors utilisés pour calculer la pose de la caméra 2 (par la méthode de Grunert, section A.2.2.2).
- Finalement chaque tirage aléatoire de 5 points donne une hypothèse pour la pose des trois caméras. Pour chaque hypothèse, on reconstruit l'ensemble des points appariés et on compte ceux qui sont compatibles avec la géométrie qui vient d'être calculée. L'hypothèse pour laquelle on obtient le plus grand nombre de points cohérents est retenue.

A.2.2.4 Calcul de pose incrémental

A partir de données issues de trois vues, l'algorithme des 5 points permet de reconstruire un nuage de points 3D qui sert à initialiser le processus de reconstruction incrémental dont un exemple, décrit ci dessous, se trouve dans la thèse d'Eric Royer [Royer 06]. Supposons qu'on connaisse la pose des caméras C_1 à C_N ($N \geq 3$). On cherche à déterminer la pose de la caméra C_{N+1}. Voici les étapes :

- On commence par détecter des points d'intérêt dans l'image $N+1$ et on les apparie avec les points qui ont été détectés dans l'image N. On obtient une liste d'appariements (q_N^i, q_{N+1}^i) (q_j^i désigne le point d'intérêt numéro i dans l'image j) de deux types différents. Dans la première catégorie on trouve les appariements pour lesquels q_N^i est aussi apparié à un point q_{N-1}^i dans l'image $N-1$. Dans la seconde, ce sont les points qui n'étaient pas visibles dans l'image $N-1$. Les points de la première catégorie ont déjà été triangulés (méthode du point milieu A.2.2.1) : leur position 3D est connue.
- Avec l'ensemble des points de la première catégorie, on calcule la pose de la caméra C_{N+1} (méthode de Grunert A.2.2.2). A ce stade, on pourrait trianguler les points de la seconde catégorie, mais pour obtenir un plus grand nombre de points, on

recommence une étape d'appariement entre l'image $N+1$ et l'image N. Cette fois-ci, la zone de recherche des points dans les images tient compte de la géométrie épipolaire : chaque point appartient au plan formé par lui-même et les centres optiques des deux caméras ; ainsi, connaissant la pose des caméras, chaque point détecté dans l'image $N+1$ est recherché dans l'image N sur une ligne, appelée ligne épipolaire, qui est l'intersection du plan image et du plan contenant le point et les deux centres optiques. Cette deuxième phase, qui réduit en pratique la zone de recherche à une bande de quelques pixels de large le long de la ligne épipolaire, produit environ 20% d'appariements corrects supplémentaires.

– Le calcul de la pose de la caméra C_{N+1} est répété avec ces nouveaux appariements. Ensuite, les points de la deuxième catégorie sont triangulés en utilisant les vues N et $N+1$. Ainsi la reconstruction partielle a été enrichie avec une caméra supplémentaire et de nouveaux points 3D reconstruits.
– Le processus se répète jusqu'à la fin de la séquence.

Annexe B

Localisation par vision monoculaire pour la navigation autonome

Introduction

L'objectif de cette annexe est de présenter l'algorithme de localisation 3D par vision monoculaire utilisé dans cette thèse pour commander un convoi de véhicules. Plus précisément, la stratégie développée par Eric Royer au cours de ses travaux de thèse [Royer 06], se déroule en deux étapes :

- Dans un premier temps le véhicule est guidé manuellement et il enregistre une séquence de référence avec un système vidéo monoculaire. Un modèle tridimensionnel de l'environnement est construit pendant cette phase d'apprentissage à partir des données enregistrées par la caméra.
- Ensuite, lors de la phase de rejeu, le véhicule est à proximité de la trajectoire apprise et il utilise l'image courante fournie par la caméra pour se localiser en temps réel dans le monde 3D reconstruit à l'étape précédente.

Dans cette annexe, ces deux étapes font l'objet des deux premières sections. La dernière section est dédiée à l'analyse des performances de la localisation.

B.1 Reconstruction 3D de la séquence de référence

Le principe de la reconstruction 3D a été introduit à l'annexe A : il s'agit d'obtenir la pose d'un sous-ensemble de caméras de la séquence de référence ainsi qu'un ensemble d'amers visuels avec leurs positions 3D, tout ceci exprimé dans un repère global. La méthode de reconstruction 3D incrémentale, détaillée à la section A.2.2.4, est complétée par d'autres stratégies pour obtenir une reconstruction de meilleure qualité. Dans cette section, certains prérequis déterminants pour la qualité de la reconstruction sont tout d'abord évoqués. Ensuite les différentes étapes de la stratégie de reconstruction sont présentées.

B.1.1 Mise en place de l'algorithme et prérequis

Pour la reconstruction, une séquence obtenue avec une seule caméra calibrée est utilisée. Dans les expériences, la caméra a été calibrée avec une mire plane selon la méthode proposée dans [Lavest 98]. Calibrer la caméra est important parce que l'objectif grand angle utilisé présente une forte distorsion radiale. Si on connaît les paramètres intrinsèques et les coefficients de distorsion alors le calcul du mouvement de la caméra est à la fois plus robuste et plus précis. De plus les algorithmes utilisant une caméra calibrée fonctionnent même si la scène est plane dans un certain nombre d'images, ce qui n'est pas le cas si les données de calibrage sont inconnues (par exemple [Beardsley 96], [Pollefeys 98]).

Chaque étape de la reconstruction, ainsi que la localisation, reposent sur la mise en correspondance d'images. Ceci est fait en détectant des points d'intérêt de Harris [Harris 88] dans chaque image. Pour avoir des points répartis sur toute l'image, l'image est divisée en 64 cases et les 500 meilleurs points sur l'ensemble de l'image sont conservés ainsi qu'un minimum de 20 points pour chaque case. Pour chaque point d'intérêt dans l'image 1, des points correspondants possibles dans une région de recherche de l'image 2 sont sélectionnés. Pour chaque appariement possible, un score de corrélation centré et normé est calculé. Les couples avec le meilleur score sont retenus. Faire la mise en correspondance de cette façon peut paraître coûteux en temps de calcul, mais un détecteur de coins de Harris très efficace peut être programmé en utilisant les extensions SIMD des processeurs récents.

B.1.2 Etapes de la reconstruction

Les étapes de la reconstruction sont les suivantes :

- On extrait tout d'abord un ensemble d'images clef à partir de la séquence de référence (section B.1.2.1).
- Ensuite on calcule la géométrie épipolaire et le mouvement de la caméra entre les images clef à l'aide de la méthode de reconstruction incrémentale déjà présentée à la section A.2.2.4.
- Afin d'améliorer sa précision, la reconstruction fournie à l'étape précédente est raffinée par un ajustement de faisceaux. Le principe est présenté à la section B.1.2.3. Les points 3D ainsi reconstruits serviront d'amers pour la localisation. Ils sont stockés avec une imagette qui représente le voisinage du point dans l'image clef où il a été détecté. Ainsi il est possible de les mettre en correspondance avec les points qui seront détectés dans de nouvelles images.

B.1.2.1 Sélection des images clef

Si le mouvement entre deux images clef est trop faible, alors le calcul de la géométrie épipolaire est mal conditionné. Il faut donc faire en sorte que le déplacement de la caméra entre deux images clef soit le plus grand possible tout en étant toujours capable de faire une mise en correspondance entre les images. La première image de la séquence est toujours choisie comme image clef. Elle est notée I_1. La deuxième image clef I_2 est

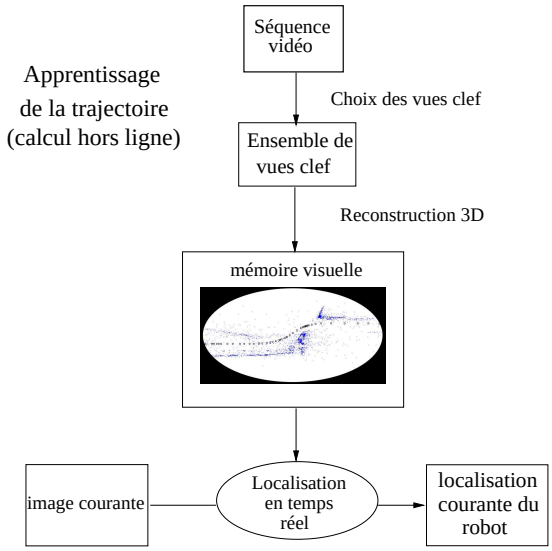


FIG. B.1 – Un aperçu du système de localisation

choisie la plus éloignée possible de I_1 dans le flux vidéo en respectant la contrainte qu'il y ait au moins M points d'intérêt communs entre I_1 et I_2. Une fois que les images clef I_1 à I_n sont choisies ($n > 1$), I_{n+1} est choisie pour qu'il y ait au moins M points d'intérêt communs entre I_{n+1} et I_n et au moins N points d'intérêt communs entre I_{n+1} et I_{n-1}. Cela assure qu'il y a suffisamment de points en correspondance entre deux images clef pour calculer le mouvement de la caméra. Dans les expériences environ 1500 points d'intérêt sont détectés par image et les seuils sont fixés à $M = 400$ et $N = 300$.

B.1.2.2 Calcul du déplacement de la caméra

A ce stade nous disposons d'un sous-ensemble des images de la séquence vidéo enregistrée pendant la phase d'apprentissage. L'algorithme utilisé pour calculer la reconstruction 3D de la scène et la trajectoire de la caméra à partir de cet ensemble d'images a été présenté à la section A.2.2.4. On rappelle ici brièvement son principe. L'algorithme peut être divisé en trois parties. Avec les trois premières images clef, on calcule le déplacement de la caméra en utilisant un calcul de matrice essentielle. Ensuite pour chaque triplet d'images, le déplacement est obtenu par un calcul de pose. Ces calculs produisent une solution initiale de la reconstruction qui est optimisée dans un ajustement de faisceaux hiérarchique présenté ci-après.

B.1.2.3 Ajustement de faisceaux hiérarchique

Le calcul du mouvement de la caméra décrit précédemment ne fournit pas une solution satisfaisante. En effet, le calcul de la pose de la caméra C_N dépend des résultats obtenus pour les caméras précédentes, et les erreurs de calcul peuvent s'accumuler tout au long de la séquence. Pour limiter grandement ce problème, on utilise un ajustement de faisceaux pour affiner la solution obtenue. Il s'agit d'un processus de minimisation basé sur l'algorithme de Levenberg-Marquardt. Toutes les variables de la reconstruction 3D sont optimisées, c.a.d. à la fois les paramètres extrinsèques C_E^i décrivant la $i^{ème}$ pose de la caméra (avec $1 \leq i \leq N$) et les coordonnées 3D Q^j des points d'intérêt (avec $1 \leq j \leq M$). La fonction de coût est généralement la somme sur toutes les images des carrés des erreurs de reprojection de tous les points visibles et correctement identifiés sur chaque image :

$$f(C_E^1, \cdots, C_E^N, Q^1, \cdots, Q^M) = \sum_{1 \leq i \leq N} \sum_{j \in A_i \subset \{1,\ldots,M\}} \|q_i^j - \pi(P_i Q^j)\|^2 \tag{B.1}$$

où $\|q_i^j - \pi(P_i Q^j)\|^2$ est le carré de la distance euclidienne entre $\pi(P_i Q^j)$, projection du point Q^j sur l'image de la $i^{ème}$ pose de la caméra, et q_i^j le point d'intérêt correspondant détecté sur cette image ; P_i est la matrice de projection 3×4 construite d'après les valeurs contenues dans C_E^i et les paramètres intrinsèques connus de la caméra ; A_i est l'ensemble des points visibles et correctement identifiés dans l'image i.

L'ajustement de faisceaux n'est pas utilisé une seule fois sur la totalité de la séquence : il est utilisé tout au long de la reconstruction afin d'éviter que l'accumulation des erreurs n'éloigne trop la solution initiale de la vraie solution, et empêche que l'ajustement de faisceaux converge. Cependant, faire un ajustement après chaque nouvelle image est très coûteux en temps de calcul. Aussi, une solution plus rapide, décrite dans [Hartley 00], consiste à faire un ajustement hiérarchique selon les étapes suivantes :

1. Une longue séquence est divisée en deux parties avec un recouvrement de deux images pour pouvoir les fusionner ensuite. Chaque sous-séquence est elle-même divisée en deux et ceci de façon récursive jusqu'à ce que chaque sous-séquence contienne seulement trois images.
2. Pour le premier triplet, on utilise un calcul de matrice essentielle (algorithme des 5 points, voir section A.2.2.4. Pour chaque triplet suivant, les deux premières images sont communes avec le précédent. Donc les poses de deux des caméras sont déduites directement du triplet précédent et la dernière est obtenue par un calcul de pose. Une fois qu'une solution initiale a été calculée sur un triplet, cette solution est optimisée dans un ajustement de faisceaux sur trois images.
3. Pour fusionner deux séquences S^1 et S^2, on utilise les deux dernières images $S^1_{N_1-1}$ et $S^1_{N_1}$ de S^1, avec N_1 le nombre d'images de S^1, et les deux premières images S^2_1 et S^2_2 de S^2. Comme les images sont les mêmes, les poses caméra associées après la fusion doivent être les mêmes. Pour cela on applique une rotation et une translation à S^2 pour que les poses associées $S^1_{N_1}$ et S^2_2 soient identiques. Ensuite

un facteur d'échelle est appliqué pour que $d(S^1_{N_1-1}, S^1_{N_1}) = d(S^2_1, S^2_2)$, où $d(S^i_n, S^j_m)$ est la distance euclidienne entre les centres optiques des caméras associées à S^i_n et S^j_m. Cela n'assure pas que les poses caméra $S^1_{N_1-1}$ et S^2_1 soient les mêmes, mais un ajustement de faisceaux est utilisé sur le résultat de la fusion et limite cette erreur.

4. On fusionne ainsi jusqu'à ce que la séquence complète soit reconstruite. La reconstruction se termine avec un ajustement de faisceaux global. Dans cette dernière étape plusieurs dizaines de milliers de points sont traités.

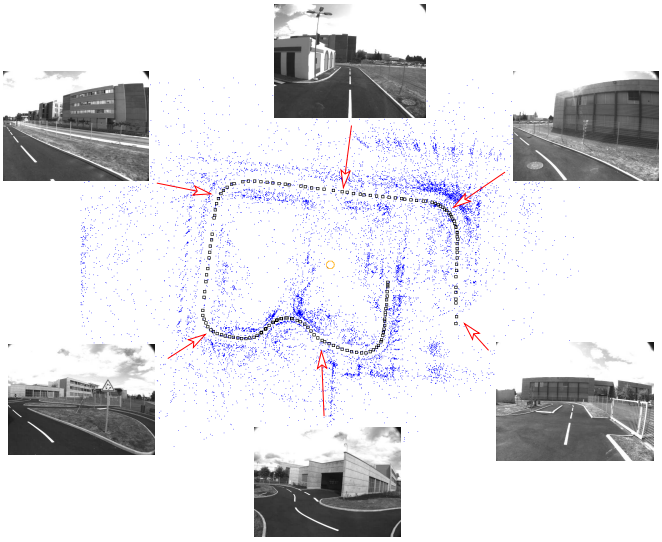


FIG. B.2 – Reconstruction 3D de la plateforme PAVIN sur le campus des Cézeaux (longueur $\simeq 350m$)

B.2 Localisation en temps réel

Le résultat de la phase d'apprentissage est une reconstruction 3D de la scène : on dispose de la pose de la caméra pour chaque image clef et d'un ensemble de points 3D associés ainsi que leur position 2D dans les images. Au début du processus de localisation, on ne sait pas où le véhicule se trouve. Il faut donc comparer l'image courante avec toutes les images clef pour trouver la plus proche. Ceci est fait en mettant en correspondance des points d'intérêt entre image courante et image clef et en calculant une pose par RANSAC pour ne garder que les appariements cohérents. La pose obtenue avec le plus grand nombre d'appariements cohérents est une bonne estimation de la position initiale de la caméra. Cette étape demande quelques secondes mais n'est réalisée qu'une fois tout au début.

Après cela, on dispose en permanence de la pose approximative de la caméra. Il suffit de mettre à jour cette pose, ce qui est beaucoup plus rapide. Voici la méthode

utilisée pour cela.

1. On commence par faire une prédiction de la pose. Dans notre cas, la pose calculée à l'image précédente est simplement reprise, mais une mesure d'odométrie pourrait par exemple être intégrée. Cette prédiction nous permet de choisir l'image clef la plus proche et donc de sélectionner dans la base de données un ensemble de points d'intérêt potentiellement visibles avec leurs coordonnées 3D.
2. On détecte ensuite les points d'intérêt dans l'image courante que l'on apparie avec l'image clef choisie. Cet appariement est rapide car les zones de recherche sont réduites (20×30 pixels) grâce à la prédiction de la pose courante qui nous donne également une prédiction sur la position où les points peuvent être retrouvés dans l'image. A ce stade, on dispose d'un ensemble d'appariements entre points 2D de l'image courante et points 3D de la base d'apprentissage.
3. Un calcul de pose robuste avec RANSAC permet alors de calculer la pose de l'image courante.
4. Enfin, cette dernière est affinée en utilisant la méthode itérative proposée dans [Araújo 98] avec quelques modifications pour la rendre robuste aux faux appariements. Cet algorithme est une minimisation de l'erreur de reprojection pour tous les points en utilisant la méthode de Newton. Très précisément :
   - si e désigne les erreurs de reprojection de chaque point en x et en y,
   - et si J désigne la matrice Jacobienne de l'erreur,
   - alors un vecteur δ de corrections à soustraire aux paramètres de la pose est déterminé en minimisant (via l'algorithme de Newton) l'écart entre les deux membres de l'égalité $J\delta = e$.

   Dans notre implémentation, à chaque itération de l'algorithme de minimisation, la projection des points est recalculée en ne gardant que ceux dont l'erreur de reprojection est inférieure à deux pixels. En général, moins de 5 itérations sont nécessaires.

La Fig. B.3 montre les appariements obtenus entre une image clef et des images prises pendant la localisation. Comme on le voit sur ces images, cette méthode de localisation est robuste à des occultations importantes grâce à l'utilisation de la structure 3D de la scène qui permet d'imposer des contraintes fortes sur la position des points lors de l'appariement.

B.3 Résultats

Plusieurs expériences ont été menées pour évaluer la précision des algorithmes de reconstruction et de localisation. En environnement extérieur, un récepteur GPS cinématique a été utilisé pour enregistrer la position du véhicule et représenter la vérité terrain. Cependant, l'enregistrement de données GPS avec une précision centimétrique n'est pas possible partout. Aussi ces expérimentations ont été conduites dans des environnements avec peu de d'immeubles pour ne pas masquer les satellites. Néanmoins il s'agit d'un cas défavorable pour les algorithmes de vision qui sont plutôt conçus pour

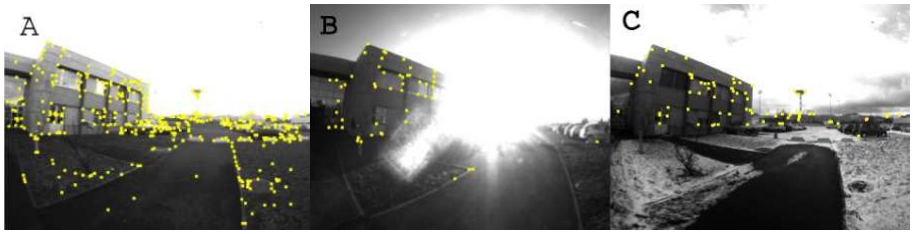


FIG. B.3 – Appariement de points entre une image du flux vidéo (B et C) et l'image clef la plus proche (A). En B, soleil dans le champ de la caméra, en C, sol enneigé. Seuls les appariements corrects sont affichés.

des zones urbaines denses où l'environnement est riche en informations visuelles. Pour disposer également de quelques résultats quantitatifs dans ce type d'environnement où le GPS n'est pas exploitable, des trajectoires en boucle ont été enregistrées, et la dérive totale au bout de la reconstruction a été mesurée. Enfin, la précision de l'orientation donnée par l'algorithme de localisation a été évaluée au moyen de mesures faites en intérieur sur une platine orientable.

Ces expérimentations ont également permis de tester la robustesse du processus de localisation à différentes conditions de luminosité et aux occultations. Sur ce volets les résultats obtenus ont été satisfaisants : la Fig. B.3 montre deux cas difficiles où la navigation autonome d'un robot a été possible en utilisant l'algorithme de localisation décrit ici. La robustesse de cette approche découle de deux facteurs : l'utilisation d'un objectif grand angle et l'exploitation d'un grand nombre de points répartis sur la totalité de l'image. Le grand angle de vue permet d'être moins sensible aux occultations provoquées par un piéton ou un véhicule par exemple. Le grand nombre de points permet d'assurer qu'il est possible d'en retrouver un nombre minimum dans la majorité des situations. Mais bien sûr chaque occultation, modification de l'environnement, ou zone de l'image où le capteur est saturé fait perdre un certain nombre de points, ce qui dégrade la qualité de la localisation, voire peut conduire à une impossibilité de calculer la pose.

B.3.1 Précision sur la position

Pour comparer les positions obtenues par l'algorithme de vision à celles fournies par le GPS, deux opérations sont nécessaires : tout d'abord, l'antenne du récepteur GPS est placée à l'aplomb du centre de l'essieu arrière, alors que la caméra est à l'avant. Les deux capteurs n'ont donc pas exactement la même trajectoire. Par conséquent la position de chaque capteur sur le véhicule est mesurée manuellement. Puis, à partir des positions GPS, on calcule les informations que donnerait un GPS virtuel qui serait placé à la même position que la caméra. D'autre part, la reconstruction 3D est faite dans un repère euclidien arbitraire lié à la caméra, alors que les positions GPS sont données dans un système de coordonnées lié au sol. La reconstruction 3D obtenue doit donc subir une

transformation rigide (rotation, translation et mise à l'échelle) pour être exprimée dans le même repère que le GPS. Une fois ces corrections effectuées, on peut calculer pour chaque caméra une erreur de position en mètres.

Quatre séquences vidéo ont été enregistrées le même jour, approximativement sur la même trajectoire (avec un écart latéral entre deux trajectoires de $1m$ au maximum), et une reconstruction 3D a été calculée pour chaque séquence. La longueur de la trajectoire est d'environ $80m$. On peut voir quelques images extraites des vidéos sur la Fig. B.4 a). La Fig. B.4 b) montre la reconstruction obtenue à partir de la vidéo $campus_1$ en vue de dessus. Les carrés noirs représentent la position des images clef, et les points 3D reconstruits apparaissent sous forme de points. Quelques piétons circulaient pendant l'enregistrement des images, mais cela n'a pas perturbé le fonctionnement des algorithmes.

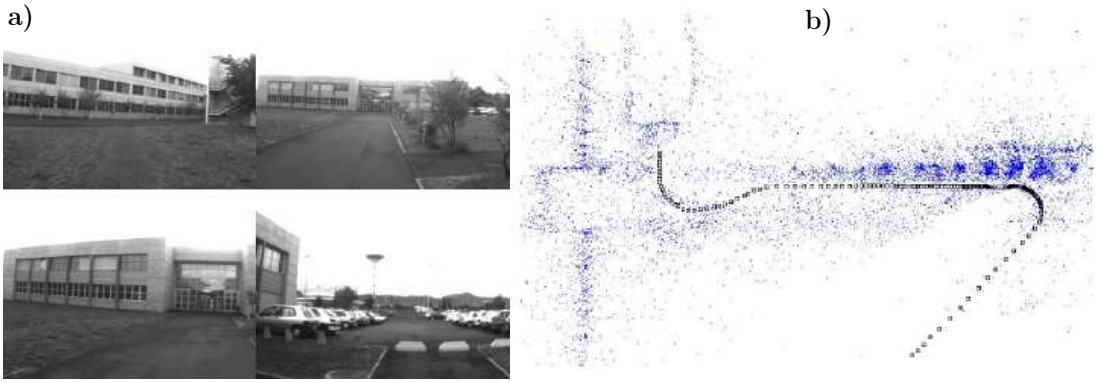


FIG. B.4 – a) Quelques images extraites de $campus_1$, b) vue de dessus de la reconstruction 3D de la séquence

Précision de la reconstruction Pour évaluer l'algorithme de localisation, chacune de ces séquences a été tour à tour utilisée comme séquence de référence et l'algorithme de localisation a été lancé sur les autres. On calcule alors l'erreur de position par rapport au GPS pour chaque pose caméra (pas seulement les images clef). La moyenne de ces

séquence	erreur de position moyenne	nombre d'images clef	nombre de points 3D
$campus_1$	25 cm	113	14847
$campus_2$	40 cm	121	15689
$campus_3$	34 cm	119	15780
$campus_4$	24 cm	114	14802

FIG. B.5 – Erreur de reconstruction moyenne pour une séquence de $80m$

erreurs sur toute la séquence figure dans le tableau de la Fig. B.5, de même que le nombre d'images clef et le nombre de points utilisés dans la séquence. Enfin, la Fig. B.6 montre les positions calculées pour chaque image clef (petits cercles) comparées à la trajectoire enregistrée par le GPS (ligne continue). On observe que l'erreur de reconstruction est en majeure partie due à une lente dérive du processus de reconstruction. Elle augmente avec la longueur et la complexité (nombre de virages) de la trajectoire.

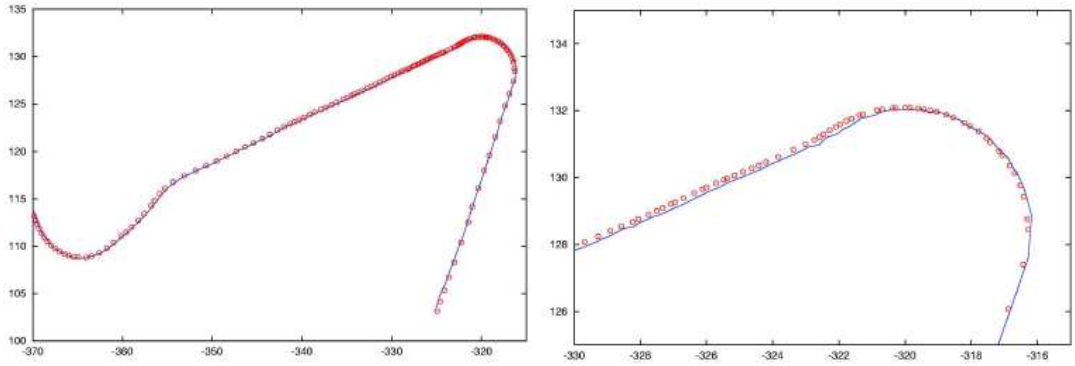


FIG. B.6 – Positions des images clef (cercles rouges) superposées à la trajectoire enregistrée par le GPS (ligne continue bleue), les unités sont en mètres. La trajectoire complète est à gauche, un agrandissement apparaît à droite.

Précision de la localisation Une localisation a été calculée pour chaque image de $campus_i$ en utilisant $campus_j$ comme séquence de référence, pour tout i et j tels que $i \neq j$ (soit 12 expériences de localisation au total). La mesure de l'erreur de localisation est plus complexe que celle de l'erreur de reconstruction. Si on se contente de calculer l'écart entre la position donnée par l'algorithme de vision et celle donnée par le GPS, on calcule une erreur de localisation globale qui intègre à la fois les erreurs commises à la localisation mais aussi à la reconstruction, et on retrouve finalement approximativement la même erreur que celle de la reconstruction.

Cependant, dans un grand nombre d'applications, une position globale n'est pas nécessaire. C'est le cas en particulier quand on veut faire naviguer un unique véhicule. Dans ce cas on a juste besoin de la distance entre la position courante du véhicule et sa position lors de l'apprentissage de la trajectoire, ainsi que de l'écart angulaire entre ces positions. Dit autrement, il suffit d'une position relative par rapport à la trajectoire de référence. Par conséquent on définit ci-dessous l'erreur de localisation comme l'erreur commise sur cette position relative.

On commence par calculer l'écart latéral y^G entre la position G_1 courante du centre de l'essieu arrière du véhicule donnée par le récepteur GPS et la position la plus proche sur la trajectoire de référence. Ensuite, à partir de la position C_1 (localisation par vision de la caméra courante, voir Fig.B.7 a)), on calcule G_1 (position correspondante du vé-

hicule estimée par vision). Pour cela on utilise la transformation décrite en introduction de B.3.1. Ensuite on cherche la position G_0 la plus proche de G_1 sur la trajectoire de référence, et en G_0, on calcule la tangente $\overrightarrow{T}$ et la normale unitaire $\overrightarrow{N}$ à la trajectoire en G_0. L'écart latéral calculé par vision est alors $y^v = \overrightarrow{G_0G_1} \cdot \overrightarrow{N}$. Finalement, l'erreur de localisation est définie par $\epsilon = y^v - y^G$. Cette erreur a été calculée pour chacune des 12 expériences et on appelle erreur de localisation moyenne, l'écart-type de ϵ sur la totalité de la trajectoire. Celle-ci varie de $1.4cm$ à $2.2cm$, avec une moyenne de $1.9cm$. La Fig. B.7 b) montre l'écart latéral et l'erreur de localisation calculés pour une des expériences où l'erreur de localisation moyenne était de $1.9cm$. Il faut noter que l'erreur obtenue ici est du même ordre de grandeur que la précision donnée par le récepteur GPS ($1cm$). Cette erreur intègre donc pour partie le bruit de mesure du récepteur GPS.

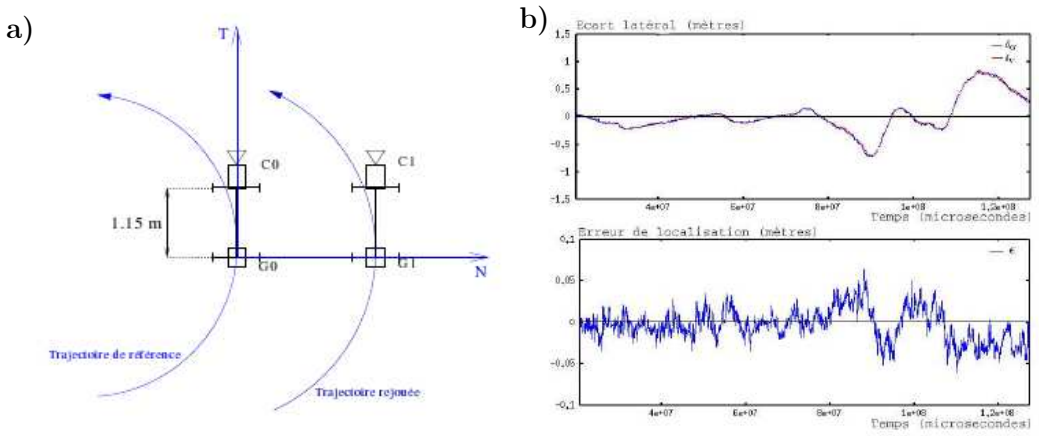


FIG. B.7 – Calcul de l'écart latéral par rapport a la trajectoire de référence a). Erreur latérale (partie b) en haut) mesurée avec le GPS y^G (bleu) ou par vision y^v (rouge) et erreur de localisation (partie b) en bas)

Trajectoires en boucle et cas des longues séquences La Fig. B.8 a) montre la reconstruction obtenue sur une trajectoire mesurant approximativement $400m$. La séquence comporte 333 images clef et 22468 points ont été reconstruits. Cette trajectoire constitue en un aller-retour et le point d'arrivée devrait être confondu avec le point de départ. La Fig. B.8 a) permet de visualiser la dérive du processus de reconstruction. L'erreur commise en fin de séquence peut paraître importante mais, comme mentionné prédemment, cela gêne peu la navigation autonome d'un véhicule tant que celui-ci est tout seul. La Fig. B.8 b) présente la reconstruction obtenue avec une trajectoire qui boucle sur elle-même. Là encore, la position de la dernière caméra devrait coïncider avec la première, mais à cause de l'erreur de reconstruction, la boucle ne se referme pas exactement.

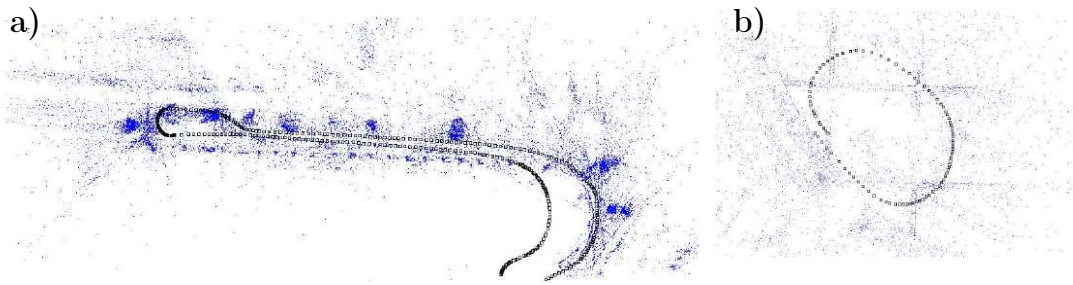


FIG. B.8 – Reconstruction 3D d'une séquence de $400m$ de long en aller-retour (partie a)) et d'une séquence en boucle (partie b))

B.3.2 Précision sur l'orientation

Pour mesurer la précision de l'orientation fournie par l'algorithme de localisation, la caméra a été placée sur une plate-forme rotative qui peut être orientée précisément (l'erreur sur la lecture est de l'ordre de $\pm 0.1°$). La séquence d'apprentissage était composée de 18 images clef enregistrées sur une trajectoire en ligne droite de $1m$, orientée dans le même sens que l'axe optique de la caméra (qui correspond à l'orientation 0°). La reconstruction de la trajectoire ainsi que le nuage de points associé sont visibles en vue de dessus sur la Fig. B.9 a). L'objectif utilisé donne un champ de 130° dans la diagonale de l'image.

Dans la phase de localisation, pour chaque orientation de la caméra entre $-94°$ et $+94°$ par incrément de 2°, l'angle α_0 mesuré sur la graduation de la plate-forme et l'angle α fourni par l'algorithme de localisation ont été notés. Au delà de 95°, la zone d'image commune entre l'image courante et l'image de référence devient très petite et on n'est plus certain de trouver un nombre de points en correspondance suffisant pour continuer à se localiser. L'erreur angulaire calculée à partir de ces données est dessinée sur la Fig. B.9 b). On observe que l'algorithme de localisation est capable de fournir une orientation avec une précision de l'ordre du dixième de degré même si on s'écarte de 90° d'un côté ou de l'autre par rapport à l'axe d'apprentissage. Pour quantifier plus précisément l'erreur, il faudrait disposer d'un moyen de mesure plus précis de l'orientation de la plate-forme.

B.3.3 Temps de calcul et espace mémoire

Les mesures sont faites sur un Pentium 4 à $3.4GHz$ avec des images de 640×480 pixels et 1500 points d'intérêt détectés par image. Le code utilise le jeu d'instructions SSE2 pour toutes les opérations de traitement d'image. Le temps de reconstruction pour la séquence $campus_1$ (113 images clef) est d'environ une heure. Le processus de localisation tourne en $60ms$. La détection des points d'intérêt prend $35ms$, l'appariement $15ms$, le calcul de pose avec RANSAC et l'optimisation itérative demandent $10ms$. La reconstruction de la séquence $campus_1$ occupe $38Mo$ sans compression de données.

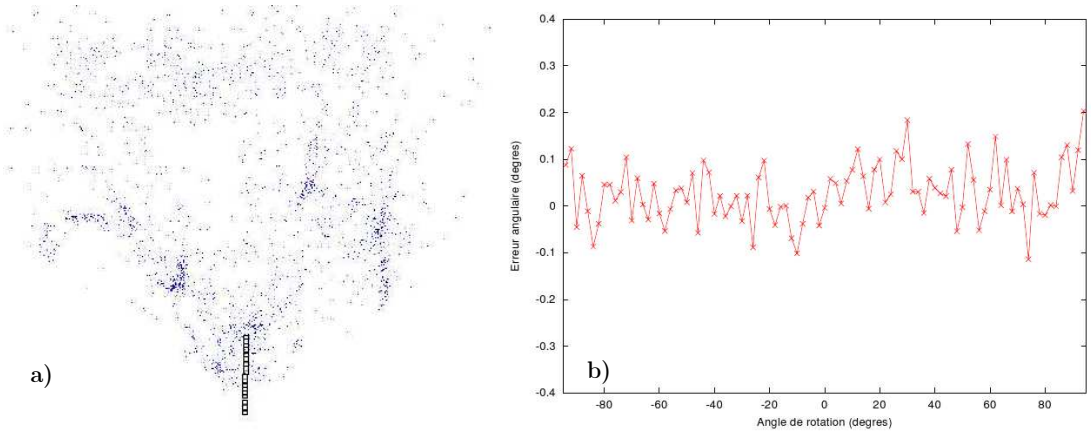


FIG. B.9 – Reconstruction 3D utilisée pour évaluer la précision de l'orientation (partie a)). Erreur angulaire $|\alpha_0| - |\alpha|$ en fonction de α_0 (partie b))

Annexe C

Suivi des véhicules expérimentaux grâce à des données télémétriques

Introduction

On souhaite utiliser des mesures de distance relative évaluées avec précision dans un référentiel métrique, afin de les comparer avec des mesures obtenues dans le monde vision et calculer alors le facteur d'échelle qui lie localement les deux référentiels. Une proposition consiste à s'appuyer sur des mesures télémétriques. En effet, pour des raisons de sécurité, les véhicules expérimentaux sont équipés de télémètres laser permettant de détecter d'éventuels obstacles présents dans le champ de vue. La précision de ces capteurs, de l'ordre du centimètre, justifie parfaitement leur utilisation dans ce contexte.

Cette annexe décrit les différentes étapes pour détecter un véhicule et extraire sa position dans une image télémétrique. Tout d'abord, une zone d'intérêt est isolée dans l'image télémétrique. Puis, dans cette zone, l'arrière du véhicule précédent, modélisé par un segment de droite de longueur connue (cas du RobuCab) ou par une ellipse de petit et grand axe connus (cas du CyCab), est recherché. La localisation du segment de droite ou de l'ellipse dans l'image télémétrique permet enfin d'obtenir l'interdistance euclidienne entre les deux véhicules, utilisée à la section 3.4 pour corriger les mesures de distances fournies par la vision monoculaire. Ces différentes étapes sont détaillées à la section C.1, puis les performances et les limitations de cette approche sont discutées à la section C.2.

C.1 Méthodologie

On considère une flotte de véhicules de type CyCab et/ou RobuCab (illustrés Fig. C.2 et C.1) évoluant en convoi le long d'une trajectoire de référence. L'objectif est d'estimer en temps réel la distance euclidienne entre le véhicule 1 et le véhicule 2, exprimée dans un repère métrique, par le biais du télémètre équipant le véhicule 2. Dans notre

cas, il s'agit d'un SICK LMS 221 dont le champ d'action est de 180° (360° en réalité, mais la moitié du champ est occulté à cause du montage sur le véhicule), la fréquence d'acquisition de $60Hz$ et la résolution angulaire de 1°.

C.1.1 Définition de la zone d'intérêt

La première étape, commune quel que soit le type du véhicule leader, consiste à isoler parmi les données télémétriques celles qui concernent le véhicule à suivre. Afin de limiter le temps de calcul, une zone de recherche est tout d'abord définie en fonction de la position obtenue lors du suivi à l'itération précédente.

Les données télémétriques de la zone d'intérêt sont ensuite segmentées afin de regrouper les impacts laser adjacents en fonction de la distance entre les points. Chaque élément obtenu est retenu selon un critère de longueur : la taille de la ligne brisée passant par tous les impacts doit se situer entre $0.70m$ et $1.30m$ (la largeur du véhicule, qui est de $1.2m$, peut en effet être perçue plus petite selon l'orientation relative des deux véhicules). A l'issue de cette étape, il arrive que plusieurs ensembles soient candidats. Chacun va être testé pour tenter d'établir une correspondance avec le modèle du véhicule.

C.1.2 Localisation d'un RobuCab

Dans le cas où le véhicule à suivre est de type RobuCab, on constate au vu de la Fig. C.1 que l'arrière du véhicule ressemble à une "accolade". Par conséquent, une manière naturelle de le détecter consiste à rechercher la droite passant par cette accolade. Compte-tenu de ses dimensions, le modèle d'un RobuCab sera donc un segment de droite de taille $1.2m$.

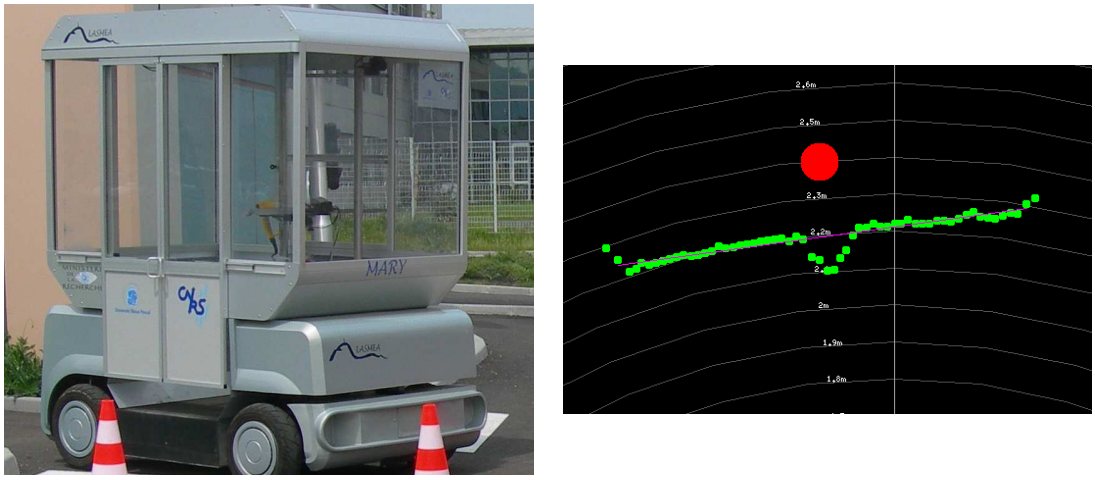


FIG. C.1 – Impacts télémétriques sur l'arrière d'un RobuCab

La position du véhicule est déterminée en quelques itérations en suivant les étapes

ci-dessous :

1. On considère tout d'abord un ensemble de données télémétriques sélectionné à l'étape précédente. On cherche alors l'équation de la droite $y = Cx + D$ qui passe au plus près des points par le critère des moindres carrés. Si le résidu est inférieur à un seuil, on valide la détection du RobuCab, sinon on reprend cette étape avec un autre ensemble candidat de données expérimentales.

2. Si un RobuCab est détecté, il faut ensuite éliminer les points qui n'appartiennent pas à la droite, c.a.d. les bords et le centre de l'accolade où les points sont plus éloignés de la droite $y = Cx + D$. Les points supprimés au centre, ainsi que les points extrêmes restant vont permettre d'éstimer le centre du segment. Ensuite la pente de la droite permet de remonter au centre de l'essieu, affiché par un point rouge sur la Fig.C.1.

Les performances et la robustesse de cette estimation sont étudiées en section C.2.

C.1.3 Localisation d'un CyCab

Au vu de la Fig. C.2, on remarque que la coque arrière du véhicule présente la forme d'une ellipse. Après avoir enregistré des données de tirs laser sur un CyCab alors que ce dernier était statique, on a pu vérifier que ce modèle était approprié. Les paramètres de l'ellipse, c.a.d. la longueur du grand et petit axe sont respectivement : $2R = 1.2m$ et $2r = 0.2m$ (voir la Fig.C.3).

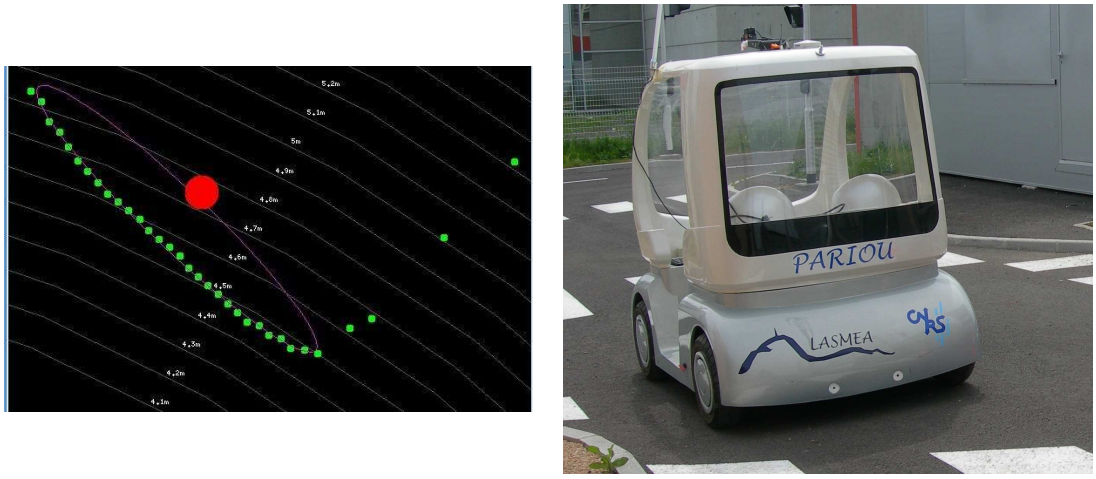


FIG. C.2 – Impacts télémétriques sur l'arrière d'un CyCab

La méthode qui a tout d'abord été testée pour identifier une ellipse dans une image télémétrique est celle proposée dans [Fitzgibbon 99]. Cependant, dans un second temps, nous avons développé une approche alternative qui tire partie du fait que certains paramètres de l'ellipse sont connus (le petit et le grand axe). Cette deuxième approche offre des résultats plus précis et plus robustes lorsqu'une partie de l'arrière du véhicule est occultée dans l'image télémétrique. Les deux stratégies sont décrites dans les deux sous-sections qui suivent.

C.1.3.1 Méthode de Fitzgibbon

Cette méthode a retenu notre attention car elle permet de retrouver les paramètres d'une conique de type ellipse de façon directe (par moindres carrés), alors que la plupart des autres approches proposent soit des méthodes itératives soit des solutions qui ne sont pas spécifiques aux ellipses mais à l'ensemble des coniques. Dans notre cas, le principal avantage de la méthode est sa robustesse au bruit sur les mesures. Le principe de l'approche proposée dans [Fitzgibbon 99] est décrit ci-dessous.

Une conique est complètement caractérisée par l'équation suivante :

$$F(\mathbf{a},\mathbf{x}) = \mathbf{a} \cdot \mathbf{x} = ax^2 + bxy + cy^2 + dx + ey + f = 0, \tag{C.1}$$

où $\mathbf{a} = [a, b, c, d, e, f]^T$ et $\mathbf{x} = [x^2, xy, y^2, x, y, 1]^T$. Soit $\mathbf{x}_i$ le vecteur de dimension six construit à partir des coordonnées d'un point du plan. Alors la fonction $F(\mathbf{a}, \mathbf{x}_i)$ représente la distance algébrique de ce point à la conique $F(\mathbf{a},\mathbf{x}) = 0$. En conséquence, si on connaît les coordonnées $(x_i,\ y_i)_{1 \leq i \leq n}$ d'un ensemble de points disposés selon une conique, alors les paramètres de cette conique peuvent être estimés en minimisant le critère :

$$\sum_{i=1}^{n} F(\mathbf{a}, \mathbf{x}_i)^2 \tag{C.2}$$

Afin d'éviter la solution triviale $\mathbf{a} = O_6$, on impose la contrainte $4ac - b^2 = 1$ qui impose que la conique soit une ellipse. La contrainte peut être ré-écrite sous une forme matricielle $\mathbf{a}^T C \mathbf{a} = 1$ avec :

$$C = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{C.3}$$

En posant $D = [\mathbf{x}_1\ \mathbf{x}_2 \ldots \mathbf{x}_n]^T$, le critère (C.2) peut se réécrire $E = \|D\mathbf{a}\|^2$. Sous la contrainte $\mathbf{a}^T C \mathbf{a} = 1$, la solution optimale peut être calculée en utilisant le formalisme des multiplicateurs de Lagrange [Bookstein 79] : en dérivant E et en introduisant le multiplicateur de Lagrange λ, on obtient le système :

$$2D^T D\mathbf{a} - 2\lambda C\mathbf{a} = 0 \tag{C.4}$$

$$\mathbf{a}^T C \mathbf{a} = 1 \tag{C.5}$$

qui peut être réécrit en introduisant $S = D^T D$:

$$S\mathbf{a} = \lambda C\mathbf{a} \tag{C.6}$$

$$\mathbf{a}^T C \mathbf{a} = 1 \tag{C.7}$$

Compte tenu de la dimension des matrices, il existe au plus six couples $(\lambda_i, \mathbf{a}_i)$ solutions de (C.6). Pour que le vecteur $\mathbf{a}_i$ minimise le critère (C.2), il faut que $\lambda_i > 0$. Il est

montré dans [Fitzgibbon 99] qu'il existe une seule solution $\lambda_i > 0$ et par conséquent la solution à considérer pour ce problème est le vecteur $\mathbf{a}_i$ associé à ce λ_i. Enfin, on choisit $\hat{\mathbf{a}}_i = \mu\mathbf{a}$ avec μ vérifiant $\mu^2\mathbf{a}_i^T C\mathbf{a}_i = 1$ pour satisfaire (C.7).

C.1.3.2 Méthode exploitant les paramètres connus

La méthode de Fitzgibbon, bien que très robuste au bruit comparativement à d'autres méthodes citées dans [Fitzgibbon 99], perd beaucoup de sa précision lorsque l'occlusion de l'ellipse est importante. En effet, lors des simulations on constate que si l'occlusion de l'ellipse est supérieure à 50%, le centre de l'ellipse estimée est souvent décalé et la largeur de l'ellipse souvent sous-estimée. Puisque l'image de l'arrière du véhicule ne forme qu'une demi ellipse, ce cas arrive malheureusement souvent (lorsque les véhicules ne sont pas alignés). Cependant cette stratégie n'exploite pas toutes les informations à disposition : le grand et petit axe de l'ellipse sont tous deux connus et pourraient permettre d'obtenir un résultat plus précis. Une approche alternative exploitant ces données est proposée ci-dessous.

Formulation du problème Une ellipse est complètement déterminée par cinq paramètres, représentés sur la Fig. C.3 :

- son centre $(x_0,\ y_0)$,
- le demi grand axe R et le demi petit axe r,
- l'orientation α du petit axe par rapport à l'horizontale du repère absolu

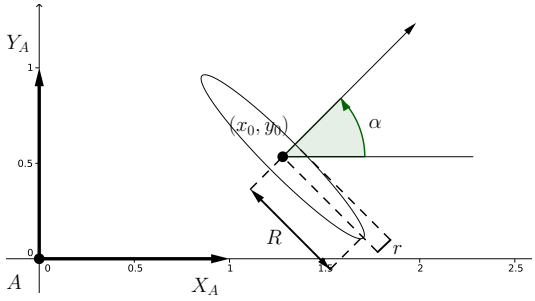


FIG. C.3 – paramétrisation d'une ellipse

En fonction de ces paramètres, l'équation d'une ellipse quelconque s'écrit de la façon suivante :

$$\mathcal{E}_{(x_0,y_0,R,r,\alpha)} : \frac{[(x-x_0)c_\alpha + (y-y_0)s_\alpha]^2}{R^2} + \frac{[(y-y_0)c_\alpha - (x-x_0)s_\alpha]^2}{r^2} = 1 \qquad \text{(C.8)}$$

où $c_\alpha = cos(\alpha)$, $s_\alpha = sin(\alpha)$.

Parmi les cinq paramètres, deux sont connus (R et r) et l'orientation α est bien estimée par la méthode de Fitzgibbon, même dans les situations difficiles quand l'occultation de l'ellipse est importante. L'idée directrice est alors d'identifier une première fois l'ellipse en utilisant la méthode de Fitzgibbon pour obtenir une estimation de l'orientation α, puis de faire une seconde identification de l'ellipse en supposant le triplet $(R,\ r,\ \alpha)$ connu pour avoir une meilleure estimée de $(x_0,\ y_0)$.

Calcul des paramètres $(x_0,\ y_0)$ Pour traiter la seconde identification par la méthode des moindres carrés, il faut réécrire l'équation (C.8) en fonction des paramètres liés à x_0 et y_0, puis présenter toutes les données expérimentales sous forme matricielle. Une possibilité est celle que nous proposons ci-dessous.

Soient $(x_i, y_i)_{1\leq i\leq n}$ les coordonnées des n impacts télémétriques appartenant à l'arrière du véhicule, modélisé par une ellipse de demi axes $R = 0.6m$ et $r = 0.1m$ et d'orientation α obtenue par la méthode de Fitzgibbon. Alors, en exploitant (C.8) on montre que ces coordonnées doivent vérifier l'équation matricielle $M \cdot \mathbf{a} = B$. Le vecteur de paramètres $\mathbf{a}$ est lié aux inconnues $(x_0,\ y_0)$ comme suit :

$$\begin{aligned}\mathbf{a} &= \mathbf{a}(x_0, y_0) \\ &= \left[\ -2x_0 \quad -2y_0 \quad (x_0c_\alpha + y_0s_\alpha)^2 + \tfrac{R^2}{r^2}\left(x_0s_\alpha - y_0c_\alpha\right)^2 - R^2\ \right]^T\end{aligned} \tag{C.9}$$

et les matrices M et B sont données par :

$$M = \begin{bmatrix} x_1(c_\alpha^2 + \frac{R^2}{r^2}s_\alpha^2) + y_1c_\alpha s_\alpha(1 - \frac{R^2}{r^2}) & y_1(s_\alpha^2 + \frac{R^2}{r^2}c_\alpha^2) + x_1c_\alpha s_\alpha(1 - \frac{R^2}{r^2}) & 1 \\ \vdots & \vdots & \vdots \\ x_i(c_\alpha^2 + \frac{R^2}{r^2}s_\alpha^2) + y_ic_\alpha s_\alpha(1 - \frac{R^2}{r^2}) & y_i(s_\alpha^2 + \frac{R^2}{r^2}c_\alpha^2) + x_ic_\alpha s_\alpha(1 - \frac{R^2}{r^2}) & 1 \\ \vdots & \vdots & \vdots \\ x_n(c_\alpha^2 + \frac{R^2}{r^2}s_\alpha^2) + y_nc_\alpha s_\alpha(1 - \frac{R^2}{r^2}) & y_n(s_\alpha^2 + \frac{R^2}{r^2}c_\alpha^2) + x_nc_\alpha s_\alpha(1 - \frac{R^2}{r^2}) & 1 \end{bmatrix} \tag{C.10}$$

$$B = \begin{bmatrix} -x_1^2\left(c_\alpha^2 + s_\alpha^2\frac{R^2}{r^2})\right) - y_1^2\left(s_\alpha^2 + c_\alpha^2\frac{R^2}{r^2}\right) + 2x_iy_ic_\alpha s_\alpha\left(\frac{R^2}{r^2} - 1\right) \\ \vdots \\ -x_i^2\left(c_\alpha^2 + s_\alpha^2\frac{R^2}{r^2})\right) - y_i^2\left(s_\alpha^2 + c_\alpha^2\frac{R^2}{r^2}\right) + 2x_iy_ic_\alpha s_\alpha\left(\frac{R^2}{r^2} - 1\right) \\ \vdots \\ -x_n^2\left(c_\alpha^2 + s_\alpha^2\frac{R^2}{r^2})\right) - y_n^2\left(s_\alpha^2 + c_\alpha^2\frac{R^2}{r^2}\right) + 2x_iy_ic_\alpha s_\alpha\left(\frac{R^2}{r^2} - 1\right) \end{bmatrix} \tag{C.11}$$

On peut alors obtenir la valeur optimale de $\mathbf{a}$ par la méthode des moindres carrés. On conserve les deux premières composantes de ce vecteur pour estimer le centre de l'ellipse $(x_0,\ y_0)$.

C.2 Performances et limitations

Pour évaluer de manière exhaustive la précision avec laquelle les algorithmes proposés peuvent mesurer la distance euclidienne entre deux véhicules, et la robustesse de ces

algorithmes en fonction des différentes configurations que les véhicules peuvent prendre lors des expériences, un simulateur émulant le capteur télémètre a été construit. Il est décrit à la section C.2.1 et les résultats de qualification des algorithmes sont fournis à la section C.2.2.

C.2.1 Protocole de simulation

Simulation du capteur télémètre laser Pour la simulation, le véhicule suiveur, doté du télémètre laser, est orienté suivant l'axe des abscisses du repère global $[A, X_A, Y_A]$. Conformément à la documentation technique, les tirs sont simulés avec une résolution angulaire de 1° sur une plage de 180°. Si l'on note α_j l'angle du $j^{ème}$ tir par rapport à l'axe des abscisses, l'équation de la droite décrivant le tir s'écrit :

$$y = tan(\alpha_j)x \tag{C.12}$$

$$\theta_j = \frac{\pi(i-90)}{180}, \quad \text{pour } i \in \{1, \ldots, 180\} \tag{C.13}$$

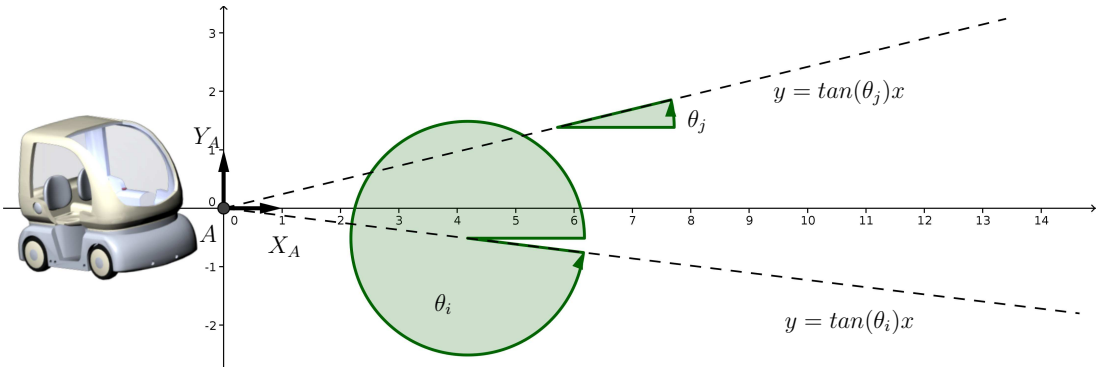


FIG. C.4 – modélisation des tirs télémétriques

Le bruit simulé sur les données télémétriques a été spécifié au vu des performances indiquées dans la documentation technique du matériel. Celles-ci sont explicitées en fonction de la portée, en considérant deux seuils caractéristiques : si le facteur de réflexion et la luminosité ambiante présentent des valeurs classiques, alors pour des portées inférieures à $8m$, les mesures de distances sont entachées d'une erreur systématique de $15mm$ et d'une erreur statistique de $5mm$, et pour des portées inférieures à $20m$, ces mêmes erreurs s'élèvent respectivement à $35mm$ et $15mm$. Cependant, avant d'entamer des expérimentations, le télémètre laser est systématiquement étalonné en comparant ses données avec les positions GPS des deux véhicules recueillies sur un parcours standard. Il est donc possible, pour les expérimentations suivantes, de corriger l'erreur systématique. Dans notre cas les données télémétriques réelles ne sont donc entachées que de la seule erreur statistique, qui par conséquent a été la seule prise en compte dans le simulateur. La précision sur les mesures de distance a été spécifiée de telle sorte que les écarts-type σ_x et σ_y sur les composantes cartésiennes soient fonction de la distance ρ

entre le télémètre et le véhicule à suivre :

$$\sigma_x(\rho) = \sigma_y(\rho) = \frac{0.01}{12}\rho + 0.005 - 8\frac{0.01}{12} \tag{C.14}$$

L'équation (C.14) assure que pour les distances $\rho = 8m$ et $\rho = 20m$, les écarts-types correspondent aux valeurs respectives de $5mm$ et $15mm$, en concordance avec la documentation constructeur.

Arrière du véhicule Les deux modèles considérés pour l'arrière d'un véhicule, c.a.d. un segment de droite (RobuCab) ou une ellipse (CyCab), ont été simulés. Pour être représentatif de toutes les configurations susceptibles d'être rencontrées lors des expérimentations, le centre du modèle $(x_0,\ y_0)$ (milieu du segment ou centre de l'ellipse) a été simulé de manière homogène à l'intérieur de la surface définie par :

$$\begin{array}{rcccl} 2m & \leq & \rho = \sqrt{x_0^2 + y_0^2} & \leq & 20m \\ -\frac{\pi}{2} & \leq & \beta = \arctan\left(\frac{y_0}{x_0}\right) & \leq & \frac{\pi}{2} \end{array} \tag{C.15}$$

et l'orientation α de la perpendiculaire au modèle (perpendiculaire au segment ou au grand axe de l'ellipse suivant le cas) définie par rapport à l'axe des abscisses a été simulée dans la plage (voir Fig. C.5) :

$$-\frac{\pi}{2} \leq \alpha \leq \frac{\pi}{2} \tag{C.16}$$

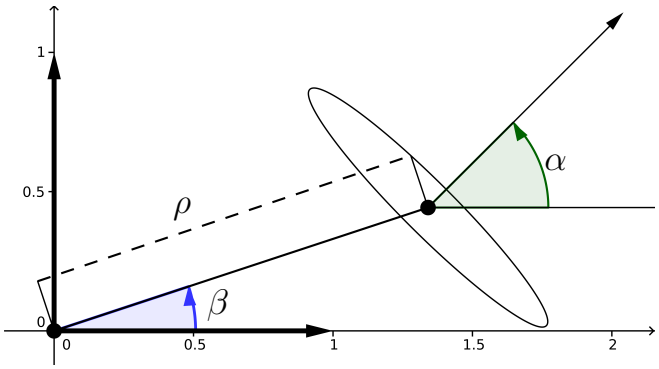


FIG. C.5 – variables ρ, α et β

Pour le RobuCab, les positions des tirs télémétriques sur le véhicule s'obtiennent en calculant l'intersection de la droite $y = tan(\alpha_j)x$ avec le segment issu de la droite :

$$y = Cx + D \text{ avec } C = -\frac{1}{\tan\alpha} \text{ et } D = y_0 + \frac{x_0}{\tan\alpha} \tag{C.17}$$

Pour le CyCab, ces mêmes positions s'obtiennent en calculant l'intersection avec l'ellipse, ce qui revient à résoudre l'équation du second degré (en x) suivante :

$$\begin{aligned} &\left[\frac{(c_\alpha + tan(\alpha_j)s_\alpha)^2}{R^2} + \frac{(-s_\alpha + tan(\alpha_j)c_\alpha)^2}{r^2}\right]x^2 \\ &+\left[\frac{2(c_\alpha + tan(\alpha_j)s_\alpha)(-x_0c_\alpha - y_0s_\alpha)}{R^2} + \frac{2(-s_\alpha + tan(\alpha_j)c_\alpha)(x_0s_\alpha - y_0c_\alpha)}{r^2}\right]x \\ &+\left[\frac{(-x_0c_\alpha - y_0s_\alpha)^2}{R^2} + \frac{(x_0s_\alpha - y_0c_\alpha)^2}{r^2}\right] = 0 \end{aligned} \tag{C.18}$$

Lorsque l'intersection entre un tir et l'ellipse est non nulle, alors la solution la plus proche du télémètre est évidemment celle qui est retenue.

C.2.2 Résultats obtenus

35557 configurations ($x_0,\ y_0,\ \alpha$) du véhicule précédent ont été testées lors des simulations, dans les limites données par (C.15) et (C.16). Pour chaque configuration, l'estimation de la position ($\hat{x}_0,\ \hat{y}_0$) a été répétée 300 fois de façon à pouvoir conduire une étude statistique. Pour simplifier l'analyse, ce sont les performances sur l'estimation de ($x_0,\ y_0$) qui ont été étudiées et non celles sur la position du centre de l'essieu arrière du véhicule précédent, étant entendu que cette dernière est complètement conditionnée par le résultat sur ($x_0,\ y_0$).

Concrètement, pour estimer ces performances, deux critères ont été considérés :

- premièrement, la distance entre la position estimée du véhicule et la position réelle : $\sqrt{(\hat{x}_0 - x_0)^2 + (\hat{y}_0 - y_0)^2}$. Ce premier critère permet d'évaluer les performances de suivi du véhicule précédent.
- deuxièmement, la différence entre l'estimation de la distance inter-véhicule et l'inter distance réelle $|\sqrt{\hat{x}_0^2 + \hat{y}_0^2} - \sqrt{x_0^2 + y_0^2}|$. Ce second critère permet d'évaluer les performances sur la variable directement utilisée pour l'estimation du facteur d'échelle sur les mesures obtenues par vision monoculaire.

Pour ces deux critères, les valeurs moyennes et les écart-types ont été tracés en fonction de la distance ρ et de la différence d'orientation $|\alpha - \beta|$. Ils sont représentés sur la Fig. C.8 dans le cas du CyCab et sur la Fig. C.9 dans le cas du RobuCab.

• **influence de la distance ρ :** Pour étudier l'influence de ce paramètre, on s'intéresse exclusivement aux simulations pour lesquelles $\alpha = \beta$. C'est le cas par exemple quand les deux véhicules sont parfaitement alignés l'un derrière l'autre, comme lors d'un suivi de ligne droite.

CyCab Dans le cas du suivi de CyCab, la Fig. C.6 montre les ellipses obtenues avec les deux algorithmes testés : la méthode de Fitzgibbon seule, en jaune, et la méthode

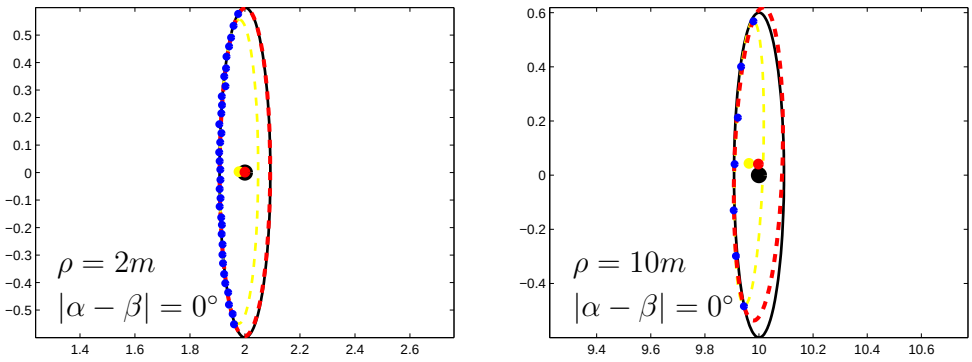


FIG. C.6 – Quelques résultats de simulation : $\alpha = \beta$

exploitant la connaissance de R et r, en rouge. On remarque que même pour des inter-distances faibles (partie gauche de la Fig. C.6), la méthode de Fitzgibbon seule a tendance à mal estimer la valeur de r et par suite la position du centre du modèle. Ce dernier connaît une erreur de $18mm$ pour la valeur $\rho = 2m$ alors qu'elle est de $7mm$ avec la seconde méthode. Dans le cas de l'inter-distance, l'erreur est de seulement $1.5mm$ à la même distance avec la seconde méthode alors qu'elle est de $16mm$ pour la méthode de Fitzgibbon.

Lorsque la distance est plus élevée (partie droite de la Fig. C.6), le nombre de tirs qui intersectent le modèle du véhicule diminue de même que leur précision. En conséquence, les extrémités du modèle (du grand axe) sont déterminées avec une précision moindre. Il en est de même pour la position du centre $(\hat{x}_0,\ \hat{y}_0)$ de ce modèle qui statistiquement connaît des fluctuations latérales (selon le grand axe de l'ellipse) et ce quel que soit l'algorithme utilisé. Les courbes de résultats sur la Fig. C.8 permettent d'évaluer la distance maximale à respecter lors des expérimentations : en effet les courbes noires indiquent que l'inter-distance doit rester inférieure à $10m$ pour que les erreurs sur la mesure d'inter-distance et sur son écart-type restent inférieures à $2cm$, et ce par les deux méthodes.

Pour l'estimation de la position du véhicule, la deuxième méthode apporte un gain constant de $2cm$ sur l'erreur moyenne quelle que soit la distance ρ (comparaison entre la courbe en trait plein et celle en tirets). La prise en compte des dimensions de l'ellipse permet de limiter l'erreur sur la position du centre de l'ellipse selon le petit axe, ce qui améliore significativement l'évaluation de l'inter-distance, permettant ainsi à l'erreur moyenne de rester inférieure à $2cm$ jusqu'à la distance $\rho = 11m$, alors que l'erreur sur la position s'élève à $6cm$.

RobuCab Globalement pour ce véhicule les performances sont meilleures que pour le CyCab. Comme le montre le cadre b) de la Fig. C.9, l'équation de la droite qui caractérise le modèle du RobuCab est très bien évaluée, puisque l'erreur sur l'estimation d'inter-

distance est quasi nulle quel que soit ρ (le cas $\alpha = \beta$ est représenté en trait noir). Le cadre a) de la Fig. C.9 montre à l'inverse une erreur croissante avec ρ sur l'estimation de la position du véhicule. Cette erreur découle encore une fois de la difficulté à déterminer les extrémités du segment (car le nombre de tirs intersectant la droite diminue quand ρ augmente), et donc à centrer le modèle du véhicule sur cette droite. Au vu des cadres c) et d) de la Fig. C.9, le bruit sur l'estimation des deux variables est linéairement corrélé à la distance ρ.

• **influence de l'angle α du véhicule précédent :** L'inclinaison du véhicule par rapport aux tirs est un facteur crucial puisque la surface du véhicule qui leur est offerte en dépend : la répartition des tirs sur le véhicule est uniforme si les deux véhicules sont alignés (cas $\alpha = \beta$). A l'inverse, plus $|\alpha - \beta|$ est grand, plus les tirs sur les extrémités du véhicule (extrémités du grand axe de l'ellipse ou du segment) se raréfient ce qui statistiquement décale latéralement l'estimée du centre du modèle. Ce phénomène est illustré Fig. C.7 dans le cas du CyCab. L'erreur d'estimation est bien sûr amplifiée si une augmentation de la distance ρ se cumule avec une différence d'orientation $|\alpha - \beta|$. Dans ce dernier cas, on remarque aussi que l'orientation fournie par les algorithmes est moins précise.

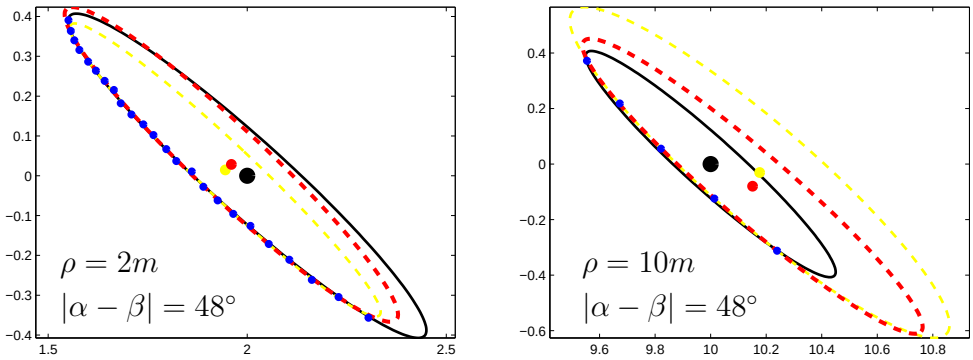


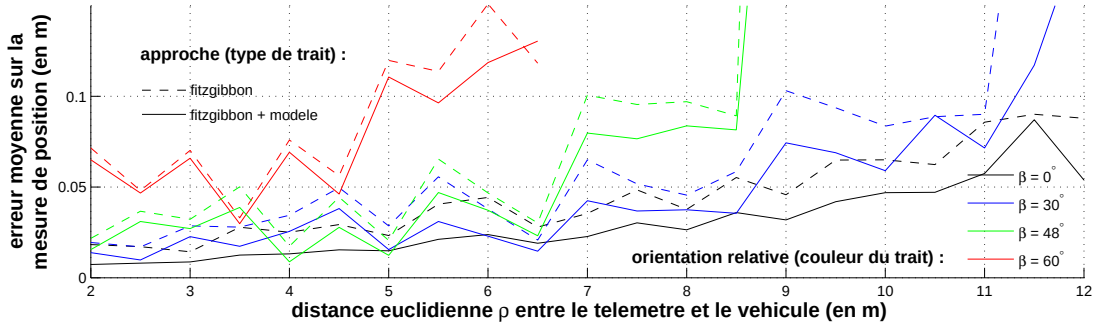
FIG. C.7 – Quelques résultats de simulation : $\alpha \neq \beta$

CyCab Au vu des courbes présentées sur la Fig. C.8, on constate que les erreurs moyennes et leur écart-type restent inférieurs à $2cm$ uniquement pour des distances ρ inférieures à $5m$ et des différences d'orientation $|\alpha - \beta|$ inférieures à 48°.

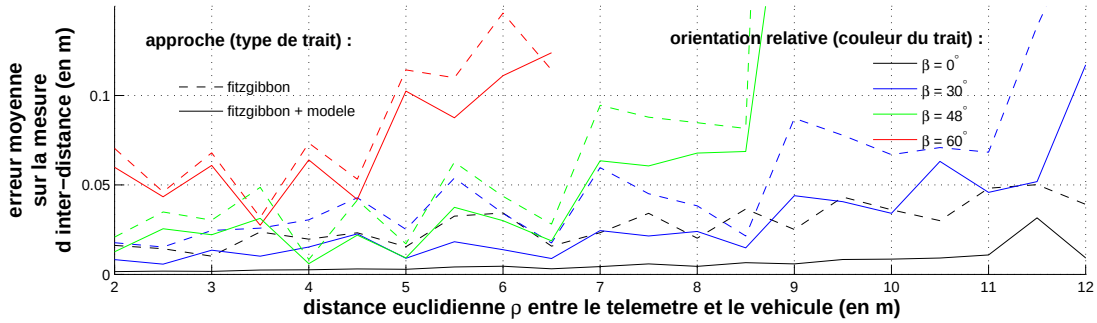
RobuCab Pour ce véhicule, les erreurs moyennes sur les inter-distances restent inférieures à $2cm$ jusqu'à $\rho = 5m$ pour $|\alpha - \beta| \leq 48°$. Quelques erreurs moyennes importantes sur la position du centre du véhicule sont visibles sur la Fig. C.9 courbes a) : elles sont toujours dues au problème du positionnement du véhicule une fois que l'équation de la droite du modèle a été déterminée. Par contre on peut noter que dans tous les cas, la dispersion des données est très faible.

Erreurs moyennes :

a)

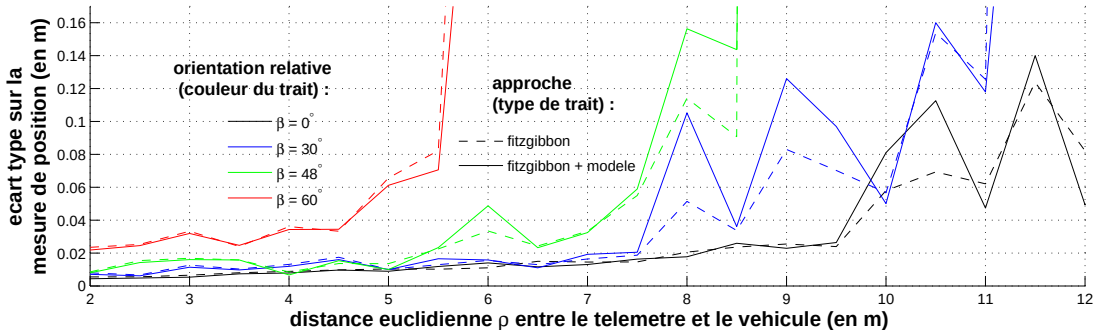


b)



Dispersion :

c)



d)

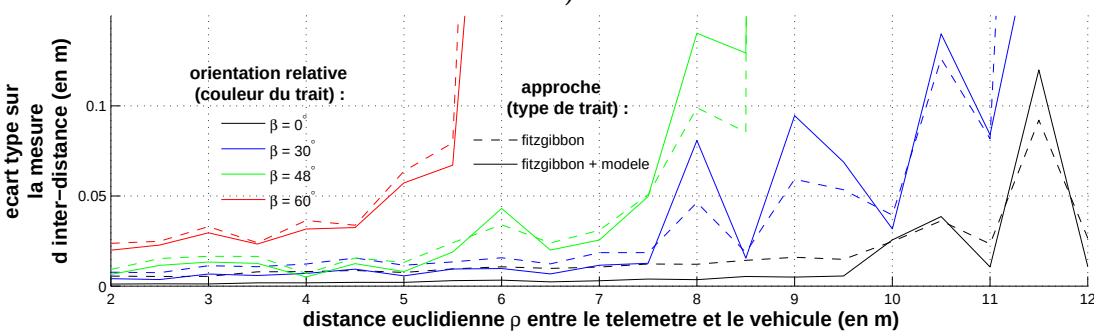
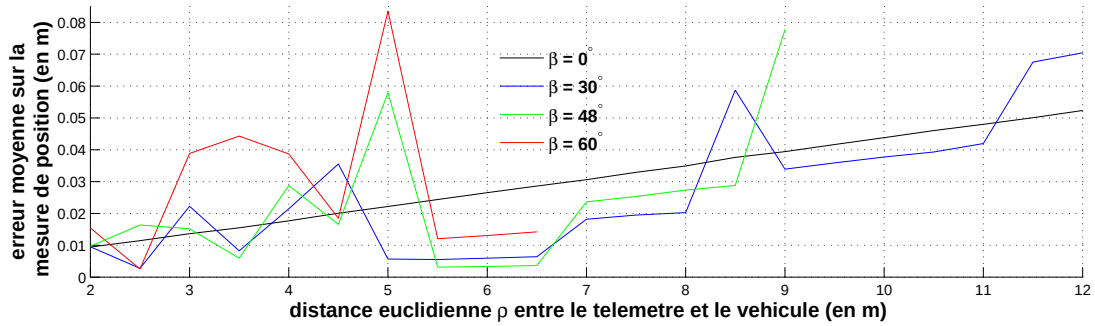


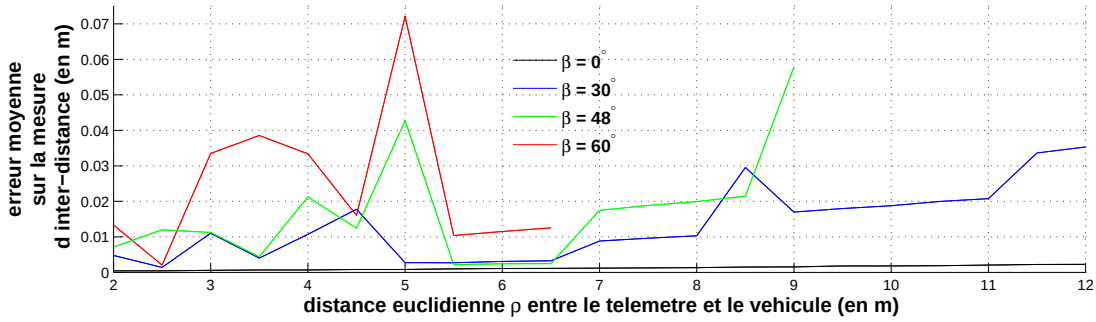
FIG. C.8 – Résultats de suivi du CyCab en fonction de ρ et $|\alpha - \beta|$

Erreurs moyennes :

a)

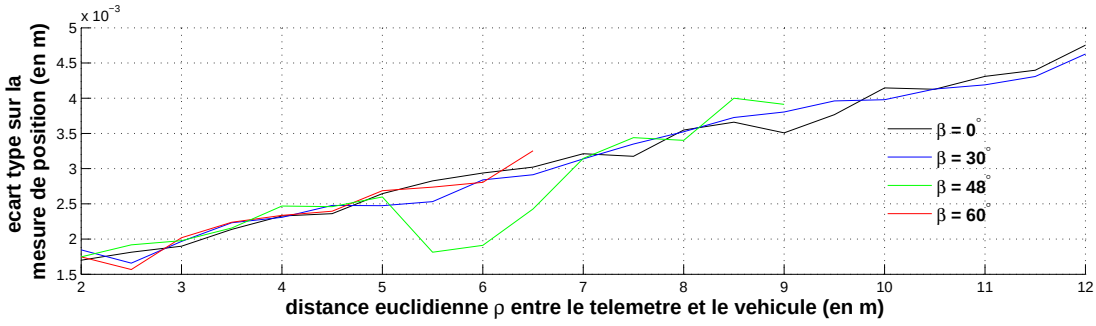


b)



Dispersion :

c)



d)

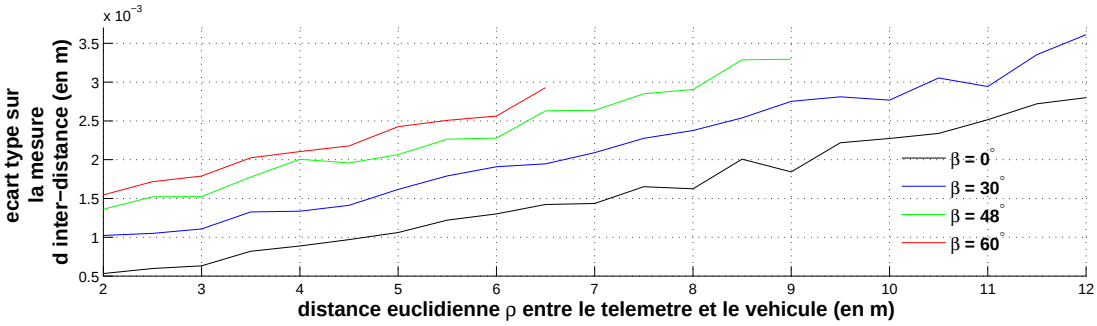


FIG. C.9 – Résultats de suivi du RobuCab en fonction de ρ et $|\alpha - \beta|$

Bibliographie

[Araújo 98] H. Araújo, R.J. Carceroni & C.M. Brown. *A fully projective formulation to improve the accuracy of Lowe's pose estimation algorithm.* Computer Vision and Image Understanding, vol. 70, no. 2, pp. 227–238, 1998.

[Argyros 05] A. Argyros, K. Bekris & S. Orphanoudakis. *Robot Homing by Exploiting Panoramic Vision.* Journal of Autonomous Robots, vol. 19, pp. 7–25, juillet 2005.

[Barfoot 04] T. D. Barfoot & C. M. Clark. *Motion planning for formations of mobile robots.* Robotics and Autonomous Systems, vol. 46, no. 2, pp. 65–78, 2004.

[Beardsley 96] P. Beardsley, P. Torr & A. Zisserman. *3D Model acquisition from extended image sequences.* European Conference on Computer Vision, pp. 683–695, 1996.

[Bom 05] J. Bom, B. Thuilot, F. Marmoiton & P. Martinet. *A global strategy fo urban vehicles platooning relying on nonlinear decoupling laws.* IEEE Intern. Conf. on Intelligent Robots and Systems (IROS), pp. 1995–2000, Edmonton (Canada), 2005.

[Bom 06] Jonathan Bom. *Étude et mise en oeuvre d'un convoi de véhicules urbains avec accrochage immatériel.* Thèse de doctorat, Université Blaise Pascal, Clermont-Ferrand (FR), Juillet 2006.

[Bookstein 79] Fred L. Bookstein. *Fitting conic sections to scattered data.* Computer Graphics and Image Processing, vol. 9, no. 1, pp. 56–71, 1979.

[Caicedo 03] R.E. Caicedo, J. Valasek & J.L. Junkins. *Preliminary Results of One-Dimensional Vehicle Formation Control Using Structural Analogy.* American Control Conference (ACC), Denver (USA), 2003.

[Chen 06] Z. Chen & S. Birchfield. *Qualitative Vision-based Mobile Robot Navigation.* IEEE International Conference on Robotics and Automation, ICRA'2006, pp. 2686–2692, Orlando, Floride (USA), 2006.

[Chu 74] Kai-ching Chu. *Decentralized Control of High-Speed Vehicular Strings*. Transportation Science, vol. 8, no. 4, pp. 361–384, 1974.

[Clipp 08] B. Clipp, J.H. Kim, J.M. Frahm, M. Pollefeys & R. Hartley. *Robust 6DOF Motion Estimation for Non-Overlapping, Multi-Camera Systems*. IEEE Workshop on Applications of Computer Vision, WACV '08, pp. 1–8, Washington, DC, USA, 2008.

[Contet 09] J.M. Contet, F. Gechter, P. Gruer & A. Koukam. *Bending Virtual Spring-Damper : A Solution to Improve Local Platoon Control*. Proceedings of the 9th International Conference on Computational Science (ICCS '09), pp. 601–610, Berlin, Heidelberg, 2009. Springer-Verlag.

[Courbon 09] Jonathan Courbon, Youcef Mezouar & Philippe Martinet. *Autonomous navigation of vehicles from a visual memory using a generic camera model*. IEEE Transactions on Intelligent Transportation Systems, vol. 10, no. 3, pp. 392–402, 2009.

[Das 02] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer & C. Taylor. *A Framework for Vision Based Formation Control*. IEEE Transactions on Robotics and Automation, vol. 18, no. 5, pp. 813–825, 2002.

[Daviet 96] P. Daviet & M. Parent. *Longitudinal and lateral servoing of vehicles in a platoon*. IEEE Symposium on Intelligent Vehicles, pp. 41–46, Tokyo, Septembre 1996.

[Davison 03] Andrew J. Davison. *Real-Time Simultaneous Localisation and Mapping with a Single Camera*. IEEE International Conference on Computer Vision, ICCV '03, pp. 1403–1410, Washington, DC, USA, 2003.

[Dierks 07] T. Dierks & S. Jagannathan. *Control of Nonholonomic Mobile Robot Formations : Backstepping Kinematics into Dynamics*. Control of Nonholonomic Mobile Robot Formations : Backstepping Kinematics into Dynamics, pp. 94–99, 2007.

[Diosi 07] A. Diosi, A. Remazeilles, S. Segvic & F. Chaumette. *Experimental evaluation of an urban visual path following framework*. 6th IFAC Symposium on Intelligent Autonomous Vehicles, IAV'07, Toulouse, France, 2007.

[Dubins 57] L. E. Dubins. *On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents*. American Journal of Mathematics, vol. 79, no. 3, pp. pp. 497–516, 1957.

[Eren 99] H. Eren, Fung C.C. & J. Evans. *Implementation of the Spline Method for Mobile Robot Path Control*. Instrumentation and Measurement Technology Conference, 1999.

[Eyre 98] J. Eyre, D. Yanakiev & I. Kanellakopoulos. *A Simplified Framework for String Stability Analysis of Automated Vehicles.* Vehicle System Dynamics : International Journal of Vehicle Mechanics and Mobility, vol. 30, no. 5, pp. 375–405, 1998.

[Fischler 81] M. A. Fischler & R. C. Bolles. *Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography.* Communications of the Association for Computing Machinery, vol. 24, no. 6, pp. 381–395, 1981.

[Fitzgibbon 99] A. W. Fitzgibbon, M. Pilu & R. B. Fisher. *Direct least-squares fitting of ellipses.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 5, pp. 476–480, mai 1999.

[Fraundorfer 10] F. Fraundorfer, D. Scaramuzza & M. Pollefeys. *A Constricted Bundle Adjustment Parameterization for Relative Scale Estimation in Visual Odometry.* IEEE International Conference on Robotics and Automation, ICRA'2010, pp. 1899–1904, Anchorage, Alaska, 2010.

[Gil-Pinto 07] A. Gil-Pinto, P. Fraisse & R. Zapata. *Decentralized strategy for car-like robot formations.* IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '2007), pp. 4176–4181, San Diego, CA, 2007.

[Gillet 99] D. Gillet & T. Chevroulet. *BURST : Bright Urban System for Transportation.* 2^{nd} Intern. Workshop on European Scientific and Industrial Collaboration (WESIC'99), Newport (Wales), Sept. 1999.

[Goedemé 05] T. Goedemé, T. Tuytelaars, G Vanacker, M. Nuttin & L. Van Gool. *Feature based omnidirectional sparse visual path following.* IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1806–1811, Edmonton (Canada), 2005.

[Gonzales-Banos 06] H.H. Gonzales-Banos, D. Hsu & J.C. Latombe. *Motion planning : Recent developments.* Autonomous Mobile Robots : Sensing, Control, Decision-Making and Applications, 2006.

[Grunert 41] J. A. Grunert. *Das pothenotische problem in erweiterter gestalt nebst über seine anwendungen in der geodäsie.* Grunerts archiv für mathematik und physik, no. 1, pp. 238-248, 1841.

[Guo 07] Y. Guo, Y. Long & W Sheng. *Global Trajectory Generation for Nonholonomic Robots in Dynamic Environments.* 2007 IEEE International Conference on Robotics and Automation, pp. 1324–1329, Rome, Italie, Avril 2007.

[Haralick 94] R. M. Haralick, C. N. Lee, K Ottenberg & M. Nölle. *Review and analysis of solutions of the three point perspective pose estimation*

problem. International Journal of Computer Vision, vol. 13, no. 3, pp. 331–356, 1994.

[Harris 88] C. Harris & M. Stephens. *A combined corner and edge detector*. Alvey Vision Conference, pp. 147–151, 1988.

[Hartley 00] R. Hartley & A. Zisserman. Multiple view geometry in computer vision. Cambridge University Press, 2000.

[Hol 07] J.D. Hol, T.B. Schon, H. Luinge, P.J. Slycke & F. Gustafsson. *Robust real-time tracking by fusing measurements from inertial and vision sensors*. Journal of Real-Time Image Processing, vol. 2, no. 2-3, pp. 149–160, Novembre 2007.

[Huppé 03] X. Huppé, J. De Lafontaine, M. Beauregard & F. Michaud. *Guidance and Control of a Platoon of Vehicles Adapted to Changing Environment Conditions*. IEEE International Conference on Systems, Man and Cybernetics, vol. 4, pp. 3091–3096, Hyatt Regency, Washington, D.C., (USA), Octobre 2003.

[Huppé 04] X. Huppé. Guidage et commande longitudinale d'un train de voitures adaptés aux conditions routières et climatiques Canadiennes. Mémoire de maîtrise es sciences appliquées, Université de Sherbrooke, Sherbrooke (Québec, Canada), Octobre 2004.

[Hwang 03] J.H. Hwang, R.C. Arkin & Kwoon D.S. *Mobile robots at your fingertip : Bezier curve on-line trajectory generation for supervisory control*. Intelligent Robotics and Systems, 2003.

[Ikeda 07] S. Ikeda, T. Sato, K. Yamaguchi & N. Yokoya. *Construction of Feature Landmark Database Using Omnidirectional Videos and GPS Positions*. Sixth International Conference on 3-D Digital Imaging and Modeling, 3DIM'07, pp. 249–256, Montreal, Quebec, 2007.

[Junkins 72] J.L. Junkins, J.R. Jancaitis & G.W. Miller. *Smooth irregular curves*. Photogrammetric Engineering, vol. 38, pp. 565–573, 1972.

[Kalman 60] R. E. Kalman. *A New Approach to Linear Filtering and Prediction Problems*. Transactions of the ASME - Journal of Basic Engineering, no. 82 (Series D), pp. 35–45, 1960.

[Kanayama 85] Y. J. Kanayama & N. Miyake. *Trajectory Generation for Mobile Robots*. Proceedings of the International Symposium on Robotics Research, pp. 16–23, 1985.

[Kanayama 97] Y. J. Kanayama & B. I. Hartman. *Smooth local-path planning for autonomous vehicles*. International Journal of Robotics Research, vol. 16, no. 3, pp. 263–284, 1997.

[Khatib 97] M. Khatib, H. Jaouni, R. Chatila & J.P. Laumond. *Dynamic Path Modification for Car-Like Nonholonomic Mobile Robots*. International Conference on Robotics and Automation, 1997.

[Khatir 04] M.E. Khatir & E.J. Davison. *Decentralized control of a large platoon of vehicles using non-identical controllers.* American Control Conference, vol. 3, pp. 2769–2776, 2004.

[Klinge 09] S. Klinge & R.H. Middleton. *Time headway requirements for string stability of homogeneous linear unidirectionally connected systems.* IEEE Conference on Decision and Control, CDC'2009, pp. 1992–1997, 2009.

[Komoriya 89] K. Komoriya & K. Tanie. *Trajectory Design and Control of a Wheel-type Mobile Robot Using B-Spline Curve.* IEEE/RSJ International Workshop on Intelligent Robots and Systems, 1989.

[Lafferriere 04] G. Lafferriere, J. Caughman & A. Williams. *Graph Theoretic Methods in the Stability of Vehicle Formations.* American Control Conference, vol. 4, pp. 3729–3734, 2004.

[Lamiraux 00] F. Lamiraux & J. P. Laumond. *Flatness and Small-Time Controllability of Multibody Mobile Robots : Application to Motion Planning.* IEEE Transactions on Automatic Control, 2000.

[Lavest 98] J. M. Lavest, M. Viala & M. Dhome. *Do we need an accurate calibration pattern to achieve a reliable camera calibration ?* European Conference on Computer Vision, pp. 158–174, 1998.

[Lenain 07] R. Lenain, B. Thuilot, C. Cariou & P. Martinet. *Adaptive and predictive path tracking control for off-road mobile robots.* European Journal of Control, vol. 13, no. 4, pp. 419–439, 2007.

[Levin 04] A. Levin & R. Szeliski. *Visual odometry and map correlation.* IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'2004, vol. 1, pp. 611–618, 2004.

[Levine 66] W. Levine & M. Athans. *On the optimal error regulation of a string of moving vehicles.* IEEE Transactions on Automatic Control, vol. 11, no. 3, pp. 355–361, 1966.

[Lindemann 05] S.R. Lindemann & S.M. LaValle. *Current issues in sampling-based motion planning.* Robotics Research Intern. Symposium, 2005.

[Longuet-Higgins 87] H. C. Longuet-Higgins. *A computer algorithm for reconstructing a scene from two projections.* pp. 61–62, 1987.

[Lothe 10] P. Lothe, S. Bourgeois, E Royer, M. Dhome & C. Naudet. *Real-Time Vehicle Global Localisation with a Single Camera in Dense Urban Areas : Exploitation of Coarse 3D City Models.* IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2010), San Francisco, Californie, Juin 2010.

[Massot 99] M. Massot & E. Lapierre. *Praxitele : the missing link.* The Association for European Transport Conference, Cambridge, 1999.

[Middleton 91] R. H. Middleton. *Trade-offs in linear control system design.* Automatica, vol. 27, no. 2, pp. 281–292, 1991.

[Montiel 06] J. M. M. Montiel, Javier Civera & Andrew J. Davison. *Unified Inverse Depth Parametrization for Monocular SLAM.* Robotics : Science and Systems, pp. 16–19, 2006.

[Mouragnon 06] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser & P. Sayd. *Real Time Localization and 3D Reconstruction.* IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'2006, vol. 1, pp. 363–370, 2006.

[Moutarlier 89] P. Moutarlier & R. Chatila. *Stochastic multisensory data fusion for mobile robot location and environment modeling.* 5th International Symposium on Robotics Research, 1989.

[Nelson 89] W. Nelson. *Continuous-curvature path for autonomous vehicles.* IEEE Proceeding on Robotics and Automation, vol. 3, pp. 1260–1264, Mai 1989.

[Newman 03] Paul M. Newman & John J. Leonard. *Consistent Convergent Constant Time SLAM.* International Joint Conference on Artificial Intelligence, Acapulco Mexico, 2003.

[Nistér 03a] D. Nistér. *An efficient solution to the five-point relative pose problem.* IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 195–202, 2003.

[Nistér 03b] D. Nistér. *Preemptive RANSAC for live structure and motion estimation.* IEEE International Conference on Computer Vision, pp. 199–206, 2003.

[Nistér 03c] David Nistér. *An efficient solution to the five-point relative pose problem.* Conference on Computer Vision and Pattern Recognition, pp. 147–151, 2003.

[Nistér 04] David Nistér. *An Efficient Solution to the Five-Point Relative Pose Problem.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 6, pp. 756–777, 2004.

[Nistér 06] D Nistér, O. Naroditsky & J. Bergen. *Visual odometry for ground vehicle applications.* Journal of Field Robotics, vol. 23, 2006.

[Nuetzi 10] G. Nuetzi, S. Weiss, D. Scaramuzza & R. Siegwart. *Fusion of IMU and Vision for Absolute Scale Estimation in Monocular SLAM.* International Conference & Exhibition on Unmanned Aerial Vehicles (UAV), Juin 2010.

[Peppard 74] L Peppard. *String stability of relative-motion vehicle control systems.* IEEE Transactions on Automatic Control, no. 19, pp. 579–581, 1974.

[Piazzi 02] A. Piazzi, C.G. Lo Bianco, M. Bertozzi, A. Fascioli & A. Broggi. *Quintic G^2-Splines for the Iterative Steering of Vision-Based Autonomous Vehicles.* IEEE Transactions on Intelligent Transportation Systems, 2002.

[Pollefeys 98] M. Pollefeys, R. Koch & L. Van Gool. *Self-Calibration and metric reconstruction in spite of varying and unknown internal camera parameters.* International Conference on Computer Vision, pp. 90–95, 1998.

[Reeds 90] J.A. Reeds & L.A. Shepp. *Optimal paths for a car that goes both forwards and backwards.* Pacific journal of mathematics, vol. 145, no. 2, pp. 367–393, 1990.

[Rivlin 03] E. Rivlin, I. Shimshoni & E. Smolyar. *Image-based robot navigation in unknown indoor environments.* IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'2003, vol. 3, pp. 2736–2742, 2003.

[Rogge 08] J.A. Rogge & D. Aeyels. *Vehicle Platoons Through Ring Coupling.* IEEE Transactions on Automatic Control, vol. 53, no. 6, pp. 1370–1377, 2008.

[Royer 04] E. Royer, M. Lhuillier, M. Dhome & T. Chateau. *Towards an alternative GPS sensor in dense urban environment from visual memory.* British Machine Vision Conference, vol. 1, pp. 197–206, Kingston (England), September 2004.

[Royer 05] E. Royer, J. Bom, M. Dhome, B. Thuilot, M. Lhuillier & F. Marmoiton. *Outdoor autonomous navigation using monocular vision.* IEEE Intern. Conf. on Intelligent Robots and Systems (IROS), pp. 3395–3400, Edmonton (Canada), 2005.

[Royer 06] Eric Royer. *Cartographie 3D et localisation par vision monoculaire pour la navigation autonome d'un robot mobile.* Thèse de doctorat, Université Blaise Pascal, Clermont-Ferrand (FR), Septembre 2006.

[Samson 95] C. Samson. *Control of chained systems : application to path following and time-varying point stabilization of mobile robots.* IEEE Transactions on Automatic Control, vol. 40, no. 1, pp. 64–77, Janvier 1995.

[Scheuer 09] A. Scheuer, O. Simonin & F. Charpillet. *Safe Longitudinal Platoons of Vehicles without Communication.* IEEE International Conference on Robotics and Automation (ICRA '09), pp. 70–75, Kobe, Japan, 2009.

[Schwartz 88] J.T. Schwartz & M. Sharir. *A survey of motion planning and related geometric algorithms.* Artificial Intelligence, 1988.

[Seiler 04] P. Seiler, A. Pant & K. Hedrick. *Disturbance propagation in vehicle strings.* IEEE transactions on automatic control, vol. 49, no. 10, pp. 1835–1841, 2004.

[Sheikholeslam 89a] S. Sheikholeslam & C. A. Desoer. *Longitudinal Control Of A Platoon Of Vehicles. I, Linear Model (ucb/erl M89/106).* Institute

of transportation studies, research reports, working papers, proceedings, Institute of Transportation Studies, UC Berkeley, 1989.

[Sheikholeslam 89b] S. Sheikholeslam & C. A. Desoer. *Longitudinal Control Of A Platoon Of Vehicles. II, First And Second Order Time Derivatives Of Distance Deviations.* Institute of transportation studies, research reports, working papers, proceedings, Institute of Transportation Studies, UC Berkeley, 1989.

[Sheikholeslam 90] S. Sheikholeslam & C. A. Desoer. *Longitudinal Control Of A Platoon Of Vehicles. III, Nonlinear Model.* Institute of transportation studies, research reports, working papers, proceedings, Institute of Transportation Studies, UC Berkeley, 1990.

[Shladover 79] S.E. Shladover. *Operation of automated guideway transit vehicles in dynamically reconfigured trains and platoons.* Rapport technique, U.S. Department of Transportation, Urban Mass Transportation Administration, Washington, D.C., Juillet 1979.

[Smith 87] R. Smith & P. Cheesman. *On the representation of spatial uncertainty.* International Journal of Robotics Research, vol. 5, no. 4, pp. 56–68, 1987.

[Swaroop 94] D. Swaroop. *String Stability of Interconnected Systems : An application to platooning in AHS.* Thèse de doctorat, University of California at Berkeley, Decembre 1994.

[Swaroop 96] D. Swaroop & J.K. Hedrick. *String stability of interconnected systems.* IEEE Transactions on Automatic Control, vol. 41, no. 3, pp. 349–357, 1996.

[Tanner 04] H.G. Tanner. *On the controllability of nearest neighbor interconnections.* IEEE Conference on Decision and Control, vol. 3, pp. 2467–2472, 2004.

[Tessier 06] C. Tessier, C. Cariou, C. Debain, F. Chausse, R. Chapuis & C. Rousset. *A real-time, multi-sensor architecture for fusion of delayed observations : application to vehicle localization.* IEEE Intelligent Transportation Systems Conference, ITSC '06, pp. 1316–1321, Toronto (Canada), Septembre 2006.

[Thuilot 04] B. Thuilot, J. Bom, F. Marmoiton & P. Martinet. *Accurate automatic guidance of an urban electric vehicle relying on a kinematic GPS sensor.* 5^{th} IFAC Symposium on Intelligent Autonomous Vehicles (IAV'04), Lisbonne (Portugal), Juillet 2004.

[Tilbury 93] D. Tilbury, R.M. Murray & Sastry S. *Trajectory Generation for the N-trailer problem using Goursat Normal Form.* Conference on Decision and Control, 1993.

[Üstünda 05] E. Üstünda & ¸Celebi. *A B-Spline Curve Fitting Approach by Implementing the Parameter Correction Terms.* International Computational Science and Engineering Conference, Istanbul, 2005.

[Vazquez 94] G.B. Vazquez, A. H. Sossa & S.J. Diaz-deLeon. *Auto Guided Vehicle Control using Expanded Time B-Splines.* Int. Conf. on Systems, Man, and Cybernetics, 1994.

[Xiang 10] J. Xiang & T. Braunl. *String formations of multiple vehicles via pursuit strategy.* Control Theory & Applications, IET, vol. 4, no. 6, pp. 1027–1038, 2010.

[Yanakiev 96] D. Yanakiev & I. Kanellakopoulos. *A Simplified Framework For String Stability Analysis In AHS.* 13th IFAC World Congress, pp. 177–182, 1996.

[Yi 04] S.Y. Yi & K.T. Chong. *Impedance control for a vehicle platoon system.* Mechatronics, 2004.