



HAL
open science

Commande prédictive et estimation des paramètres d'environnement pour un rover rapide

Mohamed Fnadi

► **To cite this version:**

Mohamed Fnadi. Commande prédictive et estimation des paramètres d'environnement pour un rover rapide. Automatique. Sorbonne Université, 2019. Français. NNT : . tel-03196900v1

HAL Id: tel-03196900

<https://hal.science/tel-03196900v1>

Submitted on 13 Apr 2021 (v1), last revised 16 Sep 2021 (v2)

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

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

THÈSE DE DOCTORAT

présentée à

Sorbonne Université

par

Mohamed FNADI

pour obtenir le diplôme de

Doctorat de Sorbonne Université

Spécialité : Robotique

Commande prédictive et estimation des paramètres d'environnement pour un rover rapide

soutenance prévue le 13 Décembre 2019

JURY

M. Luc JAULIN	Professeur des Universités à ENSTA-Bretagne	Rapporteur
M. Roland LENAIN	Directeur de recherche à Irstea, Clermont-Ferrand	Rapporteur
M. Nazih MECBAL	Professeur des Universités à Arts et Métiers ParisTech	Examineur
M. Pascal MORIN	Professeur des Universités à Sorbonne Université	Examineur
M. Eric LUCET	Chargé de recherche à CEA LIST	Examineur
M. Faïz BEN AMAR	Professeur des Universités à Sorbonne Université	Directeur de thèse
M. Frédéric PLUMET	Maître de conférences à Sorbonne Université	Co-directeur de thèse

Commande prédictive et estimation des paramètres d'environnement pour un rover rapide

Résumé

Les recherches menées dans ce mémoire concernent le contrôle-commande d'un rover rapide tout terrain à deux trains directeurs (4WS).

Un robot mobile se déplaçant à une vitesse élevée sur des terrains extérieurs généralement accidentés est soumis à des phénomènes de glissement et de dérapage qui peuvent dégrader les performances, voire déstabiliser la commande du système en suivi de trajectoire ou chemin. Il est donc nécessaire de doter ce type de robot de contrôleurs robustes et efficaces permettant à la fois de générer des chemins admissibles et stables ainsi que de garantir leur suivi précis. Ces contrôleurs doivent également réagir à la présence d'obstacles non-prévus initialement et modifier le chemin de consigne du robot.

En premier lieu, ce mémoire de thèse commence par la modélisation dynamique du véhicule décrivant à la fois sa dynamique fonctionnelle et la dynamique de ses appuis au sol. Il s'agit d'un modèle bicyclette dynamique couplé à un modèle cinématique prenant en compte les conditions d'adhérence roue/sol et la géométrie du terrain. Un premier contrôleur de suivi de chemin est synthétisé sur la base de ce modèle et qui est basé sur l'approche LQR. Ce contrôleur de suivi de chemin sera mis en place pour valider nos observateurs de paramètres géométriques et physiques du sol.

Afin de pouvoir exploiter le modèle dynamique avec glissement, il est nécessaire de procéder à l'estimation en ligne des variables liées au contact roue/sol et de la géométrie locale du terrain. Pour les conditions d'adhérence roue/sol, un nouvel observateur non linéaire est développé dans ce mémoire permettant l'estimation en ligne de la rigidité de dérive avant et arrière des pneumatiques. Pour la géométrie locale du terrain, un second estimateur basé sur la théorie de Luenberger est développé dans cette thèse afin d'estimer en temps réel l'angle de pente et l'angle de dévers du véhicule à partir de sa vitesse latérale et des mesures inertielles.

Afin de pouvoir anticiper les futures consignes et éliminer les retards de réponse de la chaîne d'actionnement du véhicule, nous optons pour une commande prédictive MPC sous contraintes. Cette commande MPC est privilégiée par rapport à la première commande LQR pour sa faculté à prédire la sortie sur un horizon de temps fini et à intégrer facilement toutes les contraintes intrinsèques du système et les contraintes liées aux limites de la zone d'adhérence du pneu. En s'appuyant sur la dynamique latérale du véhicule et le modèle linéaire des pneumatiques, cette commande est synthétisée via la minimisation d'un critère quadratique composé par l'erreur entre la sortie et la référence sur un horizon de temps fini. Elle est exprimée sous la forme d'un problème quadratique (QP) dont la fonction coût définit la tâche de suivi de chemin soumise aux contraintes du véhicule (e.g., dynamique, contact, glissement, braquage, etc).

Enfin, un contrôleur haut-niveau destiné à la planification locale de chemin admissible pour l'évitement réactif d'obstacles est développé dans cette thèse. Ce planificateur local génère des chemins opérationnels de contournement dès qu'un obstacle est détecté proche du chemin de référence global. Dans ce mémoire, ce chemin de contournement est constitué de deux courbes de Bézier cubiques qui sont à la fois admissibles et réalisables du point de vue de la cinématique et de la dynamique du véhicule.

L'ensemble des contributions proposées dans ce mémoire sont évaluées à la fois en simulation et en expérimentation sur le démonstrateur "*SPIDO*", rover capable d'atteindre la vitesse de $12m.s^{-1}$.

Mots clefs : Dynamique, rover, commande prédictive, contraintes, observateur, glissement, planification locale, suivi de chemin.

Model predictive control with environment parameters estimation for a fast rover

Abstract

The research works carried out in this thesis deal with the control of a fast double-steering off-road mobile robot.

Outdoor mobile robot has to explore and intervene efficiently and securely in large areas, where it is subjected to different phenomena such as slippage and friction conditions as well as the ground roughness, both can affect the feasibility and robustness of path or trajectory tracking tasks. Such autonomous vehicle requires highly accurate and stable control laws which should respect vehicle constraints, even if the terrain geometry and wheel-ground contact conditions are expected to change, mainly at high speed.

First and foremost, the vehicle dynamics should be incorporated in control blocks of off-road vehicles because their mobility is highly influenced by wheel-ground interactions. In this work, the dynamic model is developed relying on the slippage-friction conditions at wheel-ground contacts and the ground geometry. Then, a first controller is synthesized based on the LQR approach and a dynamic model to ensure the path tracking task. This controller is used to validate the ground parameters observers.

In order to use the vehicle dynamic model including sliding parameters, some factors related to wheel-ground contact condition have to be observed on-line. Thus, two observers have been designed in this thesis. The first one is a non linear observer that allows to estimate independently and in real time the front and rear tire cornering stiffness, these variables are related both on tire and soil properties. The second observer is based on the Luenberger theory and aims to estimate the local ground parameters geometry. This observer gives in real time the road bank and road grade angles using the lateral velocity and IMU measurements. Those estimated parameters are injected on-line in our path tracking controllers so as to enhance their efficiency.

The LQR controller does not take into account any physical or intrinsic constraints of the system. Therefore, a new constrained model predictive control (MPC) is synthesized in this thesis for a dynamic path tracking of an off-road mobile robot with a double steering axle. The main advantage of the MPC is indeed the ability to anticipate future changes in setpoints and handle constraints that are critical and necessary for the safety and stability of the vehicle. This controller is based on a dynamic model that includes wheel-ground lateral slippage and terrain geometry parameters. It is formulated as an optimization problem that computes at each time-step the optimal front and rear steering angles required to perform a desired path, with respect to multiple constraints, essentially the steering joint limits and the tire adhesion area bounds (i.e., pseudo-sliding zone limits). The control problem is expressed as a Linearly Constrained Quadratic Programming (QP) to compute the optimal and dynamically-consistent front and rear steering angles required to achieve the desired path.

Finally, a new local path planning strategy is designed for obstacle-skirting in real time. This method computes rapidly and in real-time a smooth local path for obstacle avoidance that guarantees vehicle kinematic and dynamic constraints. It is automatically generated according to several waypoints based on the obstacle's coordinates and the current state of the robot (e.g., vehicle direction, current positions, steering angles, etc.). Two cubic Bézier curves are designed to connect these waypoints such that harsh curvatures and wide variation in steering angles are prevented.

All contributions proposed in this manuscript have been validated through several tests on both advanced simulations under ROS/GAZEBO and experiments on a real off-road mobile robot “*SPIDO*”, that can achieve the velocity of $12m.s^{-1}$. Finally, all the results obtained are quite satisfactory.

Keywords : Dynamic, rover, model predictive control, constraints, observer, sliding, local path planning, path tracking.

Table des matières

Table des matières	i
Table des figures	iii
Liste d'abréviations	ix
Liste des symboles	x
1 Introduction générale	1
1 La robotique mobile tout-terrain	2
2 Objectifs et contributions	11
2 Modélisation et stabilisation du robot en suivi de chemin	16
1 Introduction	17
2 Modélisation de robots mobiles à double braquage	17
3 Synthèse de loi de commande pour le suivi de chemin	28
4 Validation en simulation et expérimentale	34
5 Conclusion	40
3 Estimation des paramètres physiques et géométriques du sol	41
1 Introduction	42
2 Observateur non linéaire de paramètres de glissement	42
3 Estimation des paramètres de la géométrie locale du terrain	55
4 Conclusion	60
4 Commande prédictive sous contraintes en suivi de chemin	62
1 Introduction	63
2 La commande prédictive basée modèle	63
3 Stratégie de commande prédictive sous contraintes	67
4 Résultats de simulation avancée	75
5 Résultats expérimentaux	80

6	Conclusion	82
5	Planification locale du chemin de contournement pour l'évitement d'obstacles	84
1	Introduction	85
2	État de l'art sur la planification de chemin pour l'évitement d'obstacles	85
3	Aperçu sur les courbes de Bézier	88
4	Détection d'obstacles	89
5	Planification locale du chemin pour l'évitement d'obstacle	96
6	Conclusion	112
6	Conclusions et perspectives	113
1	Synthèse des travaux réalisés	113
2	Perspectives	115
A	Plate-formes numérique et expérimentale	I
1	Environnement de simulation	I
2	Plate-forme expérimentale	III
B	Détection et évitement d'obstacles	X
1	Algorithme de détection d'obstacles	XI
2	Évitement d'obstacles	XII
C	Localisation rapide du véhicule	XIV
1	Modèle cinématique des robots à 4 roues directrices	XIV
2	Mise en œuvre d'un EKF pour la localisation rapide	XV
	Bibliographie	XVII

Table des figures

1.1	Quelques exemples de robots mobiles tout-terrain. De gauche à droite : Husky robot UGV, SWINCAR e-Spider, véhicule intelligent APACHe (laboratoire Heudiasyc).	3
1.2	Exemples des méthodes de mesure directe des paramètres d'état du véhicule. [Gauche] Capteur pour la mesure de la vitesse latérale développés par l'EN-SEEIHT. [Droite] Capteur de force installé dans la roue Doumiati <i>et al.</i> (2010).	8
1.3	Schéma global du contrôle-commande d'un rover tout-terrain.	11
1.4	[Gauche] : La maquette numérique de simulation ; [Droite] : La plateforme expérimentale "SPIDO" et les différents capteurs embarqués.	12
2.1	Notations utilisées pour la modélisation décrivant les forces de contact dans le repère de véhicule, les forces de contact dans le repère lié à la roue i , ainsi que les vitesses de rotation et de translation de la roue i	18
2.2	Courbe représentative de l'effort latéral du pneumatique en fonction de l'angle de dérive, montrant la zone d'adhérence et la zone de dérapage où le robot devient incontrôlable.	21
2.3	Modèle dynamique bicyclette.	23
2.4	Paramètres de la géométrie locale de terrain.	23
2.5	Évolution du véhicule par rapport au chemin de référence.	25
2.6	Commande linéaire de suivi de chemin et observateur Kalman-Bucy.	33
2.7	Résultats de simulation à $V_x = 5m.s^{-1}$ et $V_x = 8m.s^{-1}$: (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Angles de braquage avant et arrière.	34
2.8	Résultats de simulation à $V_x = 5m.s^{-1}$ et à $V_x = 8m.s^{-1}$: (a) Écart latéral de suivi de chemin. (b) Écart angulaire.	35
2.9	Résultats de simulation à $V_x = 5m.s^{-1}$ (a) et à $V_x = 8m.s^{-1}$: (a) Vitesse latérale estimée ; (b) Vitesse de lacet mesurée (ligne continue) et observée (ligne pointillée) par le filtre de Kalman-Bucy.	35
2.10	Résultats expérimentaux de chemin en virage droit : (a) Chemin de référence et chemins réalisés à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$ dans le plan (x, y) ; (b) Angles de braquage à $V_x = 3m.s^{-1}$	37

2.11	Résultats expérimentaux à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$ de chemin en U : (a) Écart latéral. (b) Écart angulaire de suivi de chemin.	37
2.12	Résultats expérimentaux à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$ de chemin en U : (a) Vitesse latérale estimée; (b) Vitesse de lacet observé par le filtre de Kalman-Bucy.	38
2.13	Résultats expérimentaux de chemin à deux manœuvres : (a) Chemin de référence et chemins réalisés à $V_x = 2m.s^{-1}$ et à $V_x = 3m.s^{-1}$ dans le plan (x, y) ; (b) Angles de braquage à $V_x = 3m.s^{-1}$	39
2.14	Résultats expérimentaux de chemin à deux manœuvres : (a) Écart latéral. (b) Écart angulaire de suivi de chemin à $V_x = 2m.s^{-1}$ et à $V_x = 3m.s^{-1}$	39
2.15	Résultats expérimentaux de chemin à deux manœuvres à $V_x = 2m.s^{-1}$ et à $V_x = 3m.s^{-1}$: (a) Vitesse latérale estimée. (b) Vitesse de lacet observé par le filtre de Kalman-Bucy.	40
3.1	Schéma global de contrôleur et des observateurs développés. Le premier bloc présente l'observateur NLO destiné à estimer C_f et C_r . Ces variables seront injectées dans le deuxième bloc (contrôleur) qui calcule les angles de braquage appropriés, en tenant compte du chemin désiré et du vecteur d'état estimé par le troisième bloc (filtre de Kalman-Bucy).	46
3.2	Résultats de simulation à $V_x = 8m.s^{-1}$ pour les deux scénarios avec et sans NLO : (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Angles de braquage avant et arrière.	49
3.3	Résultats de simulation à $V_x = 5m.s^{-1}$ et à $V_x = 8m.s^{-1}$ pour les deux scénarios : (a) Écart latéral en suivi de chemin; (b) Écart angulaire en suivi de chemin.	50
3.4	Résultats de simulation à $V_x = 5m.s^{-1}$ et à $V_x = 8m.s^{-1}$: (a) Rigidités de dérive avant et arrière estimées à différentes conditions initiales; (b) Angles de glissement avant et arrière pour les deux scénarios.	50
3.5	Résultats de simulation à $V_x = 8m.s^{-1}$: (a) Effort latéral estimé et calculé de l'essieu avant; (b) Effort latéral estimé et calculé de l'essieu arrière.	51
3.6	Résultats expérimentaux à $V_x = 11km.s^{-1}$ pour les deux scénarios : (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Angles de braquage avant et arrière.	52
3.7	Résultats expérimentaux à $V_x = 11km.s^{-1}$: (a) Rigidités de dérive avant et arrière estimées à différentes condition initiales; (b) Angles de glissement avant et arrière avec et sans l'NLO	52
3.8	Résultats expérimentaux à $V_x = 11km.s^{-1}$: (a) Écart latéral en suivi de chemin pour les deux situations avec et sans l'NLO; (b) Efforts latéraux de l'essieu avant et arrière estimés.	53
3.9	Environnement Gazebo avec un terrain accidenté d'une pente de $\pm 15^\circ$	58
3.10	Résultats de simulation avec et sans OAT à $V_x = 8m.s^{-1}$: (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Les angles de braquage avant et arrière.	58
3.11	Résultats de simulation à $V_x = 4m.s^{-1}$ et à $V_x = 8m.s^{-1}$ pour les deux situations avec et sans le OAT : (a) Écart latéral en suivi de chemin; (b) Écart angulaire en suivi de chemin.	59
3.12	Paramètres estimés à $V_x = 4m.s^{-1}$ et à $V_x = 8m.s^{-1}$: (a) Angles estimés de la géométrie locale; (b) Rigidités de dérive avant et arrière.	59
3.13	Résultats de simulation à $V_x = 8m.s^{-1}$: (a) Angles de glissement avant et arrière; (b) Efforts latéraux avant et arrière.	60

4.1	Principe de la commande prédictive : à chaque instant t_k , l'algorithme de résolution calcule la séquence de commande optimale U à appliquer au système jusqu'à l'instant t_{k+N_c} (représenté en vert). Pour cela, il se base sur le modèle du système qui permet de prédire la sortie y jusqu'à l'instant t_{k+N_p} (représenté en bleu) et sur la consigne y_d à atteindre (représenté en rouge).	65
4.2	Comportement non linéaire du pneu, réduit à une zone de pseudo-glisement limitée par les bornes de glissement.	72
4.3	Résultats de simulation pour le chemin en "Z" à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Chemin de référence en "Z" et chemins réalisés dans le plan (x, y) ; (b) Accélération latérales.	75
4.4	Résultats de simulation pour le chemin en "Z" à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Déviations latérales; (b) Déviations angulaires.	76
4.5	Résultats de simulation pour le chemin en "Z" : (a) Angles de braquage à $V_x = 5m.s^{-1}$; (b) Angles de braquage à $V_x = 10m.s^{-1}$	76
4.6	Résultats de simulation pour le chemin en "Z" : (a) Angles de glissement à $V_x = 5m.s^{-1}$; (b) Angles de glissement à $V_x = 10m.s^{-1}$	77
4.7	Résultats de simulation pour le chemin en "O" à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Chemin de référence en "O" et chemins réalisés dans le plan (x, y) ; (b) Accélération latérales.	77
4.8	Résultats de simulation pour le chemin en "O" à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Déviations latérales; (b) Déviations angulaires.	78
4.9	Résultats de simulation pour le chemin en "O" : (a) Angles de braquage à $V_x = 5m.s^{-1}$; (b) Angles de braquage à $V_x = 10m.s^{-1}$	78
4.10	Résultats de simulation pour le chemin en "O" : (a) Angles de glissement à $V_x = 5m.s^{-1}$; (b) Angles de glissement à $V_x = 10m.s^{-1}$	79
4.11	Résultats expérimentaux à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$: (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Accélération latérale.	80
4.12	Résultats expérimentaux à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$: (a) Déviations latérales; (b) Déviations angulaires.	81
4.13	Résultats expérimentaux : (a) Angles de braquage à $V_x = 2m.s^{-1}$; (b) Angles de braquage à $V_x = 3m.s^{-1}$	81
4.14	Résultats expérimentaux : (a) Angles de glissement à $V_x = 2m.s^{-1}$; (b) Angles de glissement à $V_x = 3m.s^{-1}$	82
5.1	Courbes de Bézier cubiques avec différents positionnements des points de contrôle.	88
5.2	Chaîne d'information pour la détection d'obstacles.	90
5.3	Élimination du nuage de points appartenant au robot et au sol.	91
5.4	Velodyne height map.	92
5.5	Restriction de la zone à traiter.	92
5.6	Validation en simulation de l'algorithme de détection.	94
5.7	Approche de détection d'obstacles sous GAZEBO. [Gauche] : Nuage de points initial fourni par le LiDAR 3D virtuel. [Droite] : Résultats de traitement du nuage de points pour la détection d'obstacles.	94
5.8	Validation expérimentale de l'algorithme de détection d'obstacles.	95

5.9	Approche de détection d'obstacles. [Gauche] : Nuage de points brut fournie par le LiDAR 3D. [Droite] : Résultats de traitement du nuage de points pour la détection d'obstacles, la distance entre chaque obstacle et le centre de gravité du véhicule est calculée. Notez que deux types d'obstacles sont distingués dans ce travail, les plus sûrs (nuages verts) et les dangereux (nuages rouges) proches du chemin de référence global.	95
5.10	Distance de sécurité	96
5.11	Stratégie d'évitement d'obstacle.	97
5.12	Stratégie haut-niveau pour l'évitement d'obstacles.	98
5.13	Principales étapes de la planification locale du chemin de contournement.	99
5.14	Centre instantanée de rotation	100
5.15	Schéma global du contrôle. Le contrôleur de haut-niveau permet la perception de l'environnement en utilisant le capteur Velodyne pour la détection d'obstacles, puis la génération d'un chemin de référence local pour la tâche d'évitement. Ce chemin planifié est utilisé comme consigne à exécuter par le contrôleur de bas-niveau qui calcule les bons angles de direction.	102
5.16	Résultats sur le " Chemin rectiligne " à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé; (b) Écart latéral et angulaire de suivi de chemin.	104
5.17	Résultats de simulation sur le " Chemin rectiligne " à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière; (b) Angles de glissement avant et arrière.	104
5.18	Résultats de simulation sur le " Chemin rectiligne " à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière; (b) Accélération latérale.	104
5.19	Résultats sur le " Chemin en O " à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé; (b) Écart latéral et angulaire de suivi de chemin.	105
5.20	Résultats de simulation sur le " Chemin en O " à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière; (b) Angles de glissement avant et arrière.	105
5.21	Résultats de simulation sur le " Chemin en O " à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière; (b) Accélération latérale.	105
5.22	Résultats sur le " Chemin en S " à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé; (b) Écart latéral et angulaire de suivi de chemin.	106
5.23	Résultats de simulation sur le " Chemin en S " à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière; (b) Angles de glissement avant et arrière.	106
5.24	Résultats de simulation sur le " Chemin en S " à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière; (b) Accélération latérale.	106
5.25	Résultats sur le " Chemin en Z " à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé; (b) Écart latéral et angulaire de suivi de chemin.	107
5.26	Résultats de simulation sur le " Chemin en Z " à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière; (b) Angles de glissement avant et arrière.	107
5.27	Résultats de simulation sur le " Chemin en Z " à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière; (b) Accélération latérale.	107
5.28	(a) Chemin de référence global et local et chemin réalisé à $V_x = 2.5m.s^{-1}$; (b) Écart latéral et angulaire de suivi de chemin à $V_x = 2.5m.s^{-1}$	110
5.29	(a) Angles de braquage avant et arrière à $V_x = 2.5m.s^{-1}$; (b) Angles de glissement avant et arrière à $V_x = 2.5m.s^{-1}$	110
5.30	(a) Efforts latéraux avant et arrière à $V_x = 4m.s^{-1}$; (b) Accélération latérale à $V_x = 4m.s^{-1}$	110
5.31	(a) Chemin de référence global et local et chemin réalisé à $V_x = 4m.s^{-1}$; (b) Écart latérale et angulaire de suivi de chemin à $V_x = 4m.s^{-1}$	111

5.32	(a) Angles de braquage avant et arrière à $V_x = 4m.s^{-1}$; (b) Angles de glissement avant et arrière à $V_x = 4m.s^{-1}$	111
5.33	(a) Efforts latéraux avant et arrière à $V_x = 4m.s^{-1}$; (b) Accélération latérale à $V_x = 4m.s^{-1}$	111
A.1	Le modèle CAO de la maquette virtuelle de la simulation.	II
A.2	Modèle cinématique au niveau de la roue i	II
A.3	[Gauche] : Vue principale de l'interface GAZEBO de la plateforme "SPIDO"; [Droite] : Architecture des articulations sur Rviz.	III
A.4	"SPIDO", véhicule électrique 4WS utilisé pour évaluer les algorithmes proposés.	IV
A.5	L'architecture mécanique de la plateforme expérimentale "SPIDO".	IV
A.6	La barre anti-roulis intégrée sur le robot "SPIDO" pour la stabilisation du roulis.	IV
A.7	Capteurs extéroceptifs embarqués sur le robot "SPIDO".	V
A.8	Architecture matérielle embarquée du "SPIDO".	VII
A.9	Architecture logicielle embarquée du "SPIDO".	VIII
A.10	Le champs utilisé pour les expérimentations [Google Maps].	IX
C.1	Vitesses de rotation des roues.	XV

Liste des abréviations

Abréviations	Description
MPC	M odel P redictive C ontrol
OAT	O bservateur A ngles du T errain
LQR	L inear Q uadratic R egulator
NLO	O bservateur N on L inéaire
LiDAR	L ight D etection A nd R anging
s.t.	subject to
4WS	4 W heel S teering
QP	Q uadratic P rogramming
ROS	R obot O perating S ystem
PCL	P oint C loud L ibrary
RSG	R oulement S ans G lisement
GPS	G lobal P ositioning S ystem
RTK	R eal T ime K inematic
CIR	C entre I ntantanée de R otation

Liste des symboles

- Modélisation et stabilisation du robot en suivi de chemin

Symboles	Description
$\mathcal{R}_i = \{O, X, Y, Z\}$	Repère global (ou le repère inertiel)
p_G	Position du centre de masse du robot
p_F	Position du point le plus proche de p_G sur le chemin de référence
$\mathcal{R}_G = \{x_G, y_G, z_G\}$	Repère lié au corps du robot, centré en p_G
$\mathcal{R}_{wi} = \{x_{wi}, y_{wi}, z_{wi}\}$	Repère lié à la roue i , centré en p_i
$\mathcal{R}_F = \{x_F, y_F, z_F\}$	Repère de Serret-Frénet le long du chemin de référence, centré en p_F
Γ	Chemin de référence global
s	Abscisse curviligne du chemin de référence
ρ	Courbure du chemin de référence
δ_i	Angle de braquage de la roue i
δ_f, δ_r	Angle de braquage avant et arrière
β_i	Angle de glissement de la roue i
β	Angle de glissement global
β_f, β_r	Angle de glissement avant et arrière
C_i	Coefficient de rigidité transversale au niveau de la roue i
$C_{\sigma i}$	Coefficient de rigidité longitudinale au niveau de la roue i
C_f, C_r	Coefficient de rigidité de dérive du train avant et arrière
f_{xi}	Effort longitudinal de la roue $i \in \{f, r\}$ exprimé dans \mathcal{R}_{wi}
f_{yi}	Effort latéral de la roue $i \in \{f, r\}$ exprimé dans \mathcal{R}_{wi}
f_{zi}	Effort vertical de la roue $i \in \{f, r\}$ exprimé dans \mathcal{R}_{wi}
F_{xi}	Effort longitudinal de la roue $i \in \{f, r\}$ exprimé dans \mathcal{R}_G
F_{yi}	Effort latéral de la roue $i \in \{f, r\}$ exprimé dans \mathcal{R}_G
F_{zi}	Effort vertical de la roue $i \in \{f, r\}$ exprimé dans \mathcal{R}_G
a, b, d	Empattement avant et arrière et la demie-voie avant
ψ, ψ_d	Angle de lacet et l'angle de lacet de référence
V_x, V_y, V_ψ	Vitesse longitudinale, latérale et de lacet
v_{xi}, v_{yi}	Vitesse longitudinale et latérale du centre de rotation de roue i
σ_i	Taux de glissement longitudinal
e_y, e_ψ	Déviations latérale et angulaire de suivi de chemin
g	Intensité de la pesanteur
μ_i	Coefficient de frottement de la roue i
μ	Coefficient de frottement global
m, I_z	Masse de robot mobile et l'inertie suivant l'axe z
I_{ω_i}	Moment d'inertie de la roue i
C_{mi}, C_{fi}	Couple moteur et de freinage au niveau de la roue i
A, B, C	Matrices de la représentation d'état du modèle véhicule
S	Vecteur de dérive du modèle dynamique
K	Gain de retour d'état de la commande LQR
K_b, N	Gain de filtre Kalman-Bucy et la matrice de pré-filtre
P_r^*	Solution algébrique de l'équation de Riccati
$\xi, \hat{\xi}$	Vecteur d'état et son estimé de modèle dynamique du véhicule
y	Vecteur mesure

u	Vecteur d'entrée regroupant les angles de braquage
ξ_{ss}, u_{ss}	Vecteurs d'état stationnaires
ζ, γ	Bruits blancs sur le modèle et les mesures
$\Gamma_\zeta, \Gamma_\gamma$	Matrice de covariance des bruits blancs ζ et γ
Γ_t	Matrice de covariance de l'état
Q, R	Matrices de pondération sur l'état et la commande
Q_B, Q_S	Matrice d'observabilité et commandabilité

- Estimation des paramètres physiques et géométriques de sol

Symboles	Description
a_x, a_y, a_z	Accélérations linéaires longitudinale, latérale et verticale
$\omega_x, \omega_y, \omega_z$	Vitesses angulaires de véhicule
θ, ψ, ϕ	Angles d'Euler (tangage, lacet et roulis)
\hat{C}_f, \hat{C}_r	Rigidité de dérive estimées avant et arrière
ω_c	Vecteur des rigidités de dérive avant et arrière
$\hat{\omega}_c$	Estimée des rigidités de dérive avant et arrière
$\tilde{\omega}_c$	Erreur d'estimation des rigidités de dérive avant et arrière
G	Matrice gain de l'observateur NLO
A_c, B_c	Matrices d'état de l'observateur NLO
T	Matrice définie positive de la fonction de Lyapounov
S_l	Vecteur gravité de l'observateur NLO
x_c	Vecteur d'état de l'observateur NLO
$c(\cdot)$	Vecteur intermédiaire de l'observateur NLO
Λ	Vecteur auxiliaire de l'observateur NLO
$V(\cdot)$	Fonction de Lyapounov quadratique candidate
θ_s	Angle de déformation généré par le système de suspension
x_r, u_r	Vecteur d'état et commande de l'OAT
\hat{x}_r	État estimé de l'OAT
\tilde{x}_r	Erreur d'estimation de l'OAT
θ_r, ϕ_r	Angle de pente et de dévers du véhicule
$\hat{\theta}_r, \hat{\phi}_r$	Variable d'estimation de θ_r et ϕ_r
L	Matrice gain de l'OAT
A_r, B_r	Matrices d'état de l'OAT
T_r	Période d'échantillonnage de la centrale inertielle

- Commande prédictive sous contraintes

Symboles	Description
N_p	Horizon de prédiction
N_c	Horizon de prédiction de la commande
Q_n, R_n	Matrice de pondération sur l'état et la commande
T_d	Période de la discrétisation exacte
Φ	Matrice d'état du système échantillonné
Γ	Matrice commande du système échantillonné
X_k	Vecteur d'état du système échantillonné à l'instant k
y_k	Vecteur sortie échantillonné à l'instant k

U_k	Commande à l'instant k
U	Séquence des commandes sur l'horizon de prédiction
U^*	Séquence des commandes optimale sur l'horizon de prédiction
ΔU	Variation de la commande U
Y	Sortie estimée sur l'horizon de prédiction
Y_d	Référence projetée sur l'horizon de prédiction
P_x, P_u	Matrices augmentées du modèle
J	Critère quadratique à minimiser
H, f	Matrice quadratique et le vecteur linéaire du critère J
$(\cdot)^{\max}, (\cdot)^{\min}$	Extremums relatifs aux différentes contraintes
$\bar{\beta}$	Séquence des angles de glissement sur l'horizon de prédiction
G, h	Matrice globale des contraintes et leurs extremums associés

- Planification locale du chemin de contournement

Symboles	Description
R_o, R_g, R_e	Rayon de cercle entourant l'obstacle, cercle grossi et le cercle englobant
α_g, α_e	Coefficients d'augmentation d'obstacle
S_k	Courbe de Bézier k
d_s, d_{min}	Distance de sécurité et sa distance minimale
k_1, k_2	Deux gains d'ajustement de la distance de sécurité
$a_{x_{max}}$	Décélération longitudinale maximale du véhicule
$a_{y_{max\ k}}$	Accélération latérale maximale le long de la courbe de Bézier S_k
d_b, d_m, d'_m, d_f	Distances entre deux points de contrôle consécutifs
d_1, d_2	Deux distances pour le choix de point d'évitement
$P_i(x_i, y_i)$	Point de contrôle de la courbe de Bézier $\forall i$
C_j	Cluster j
p_i^j	Point i appartenant au cluster de nuage de point C_j
$O_c(x_c, y_c)$	Barycentre de l'obstacle de coordonnées x_c et y_c
x_d, y_d, ψ_d	Données du chemin désiré (chemin de référence)
L	Empattement du véhicule
v_1, v_2	Deux paramètres de temps pour les deux courbes de Bézier
ρ_{max}, δ_{max}	Courbure et le braquage maximaux
$\rho_{j,k}$	Courbure de la courbe de Bézier S_k au point j
ρ_k	Courbure maximale de la courbe de Bézier S_k
ρ_0	Courbure initiale
F_{frott}	Force de frottement

Introduction générale

Sommaire

1	La robotique mobile tout-terrain	2
1.1	Architectures cinématiques des véhicules tout-terrain	2
1.2	Modélisation et modèles d'interaction roue-sol	5
1.3	Commande et estimation pour les véhicules tout-terrain	6
1.4	Planification locale des chemins pour l'évitement réactif d'obstacle	9
2	Objectifs et contributions	11
2.1	Contexte et objectifs de la thèse	11
2.2	Approche proposée	11
2.3	Description de la plateforme numérique et expérimentale	12
2.4	Contributions de la thèse	12
2.5	Organisation du mémoire	14

L'ensemble des travaux effectués au cours de cette thèse s'inscrit dans le domaine de la robotique mobile à roues en milieu extérieur. Nous allons au cours de ce chapitre présenter dans un premier temps certains axes de recherche de la robotique mobile tout-terrain. Ensuite, le contexte et les motivations sous-jacentes de ce projet seront abordés, ainsi que les différentes contributions de ce travail.

1 La robotique mobile tout-terrain

Durant ces dernières années, les avancées technologiques ont favorisé l'émergence des véhicules autonomes en milieu naturel dans plusieurs domaines d'application (agriculture, exploration minière, sécurité civile). La plupart des thématiques de recherche abordées sur ces robots mobiles en milieu naturel visent à élargir leur champ d'action et à améliorer leur temps d'intervention. De tels objectifs nécessitent l'utilisation de véhicules ayant des capacités d'évolution sûres et rapides.

Les robots mobiles destinés aux activités d'intérieur se déplacent généralement à basses vitesses (e.g., surveillance [Michelsoni et al. \(2007\)](#), logistique [Stoyanov et al. \(2013\)](#), etc.) et leurs commandes sont définies sur la base d'un modèle cinématique avec l'hypothèse de roulement sans glissement (RSG). En revanche, quand la vitesse est élevée et/ou le robot se déplace en environnement extérieur, cette hypothèse n'est plus valide, et le risque d'instabilité et la dégradation des performances de la commande augmentent. Le glissement peut apparaître durant l'accélération ou la décélération du robot et même dans un virage à grande vitesse. Si le glissement est négligé dans la stratégie de commande, la stabilité du système peut être affectée et le robot mobile peut ne pas accomplir ses tâches. Ainsi, la prise en compte des variables de glissement dans ces conditions paraît opportune car elle permet de synthétiser des lois de commande plus performantes par rapport à des lois purement cinématiques. Ceci est valable par exemple pour les engins mobiles de types Quad qui peuvent atteindre des vitesses relativement élevées de l'ordre de $18m.s^{-1}$. Ces engins sont utilisés pour réaliser des tâches d'extérieur, notamment dans un cadre agricole. Ces engins nécessitent donc des lois de commande fiables afin de les maintenir dans un domaine de stabilité quelles que soient les conditions d'adhérence et de géométrie du terrain [Bouton \(2009\)](#). Dans le cadre de cette thèse, nous considérerons donc que les effets de glissement ne sont pas négligeables vu le caractère tout terrain et la dynamique rapide exigée.

Dans le reste de cette section, nous donnerons un bref aperçu de l'état de l'art concernant les modèles de véhicules utilisables pour les robots mobiles à roues tout-terrain aussi que des modèles roue/sol existants. Puis, nous présenterons quelques algorithmes de commande développés spécifiquement pour le contrôle de robots mobiles rapides tout-terrain. Ces véhicules étant destinés à se déplacer de manière sûre dans des environnements ouverts, nous donnerons également un aperçu des principales méthodes utilisées pour mettre en place des fonctions d'évitement d'obstacles.

1.1 Architectures cinématiques des véhicules tout-terrain

Il existe plusieurs architectures cinématiques des robots mobiles tout-terrain à roues.



Figure 1.1. Quelques exemples de robots mobiles tout-terrain. De gauche à droite : Husky robot UGV, SWINCAR e-Spider, véhicule intelligent APACHe (laboratoire Heudiasyc).

Roues non directionnelles (*skid-steering*)

Les robots mobiles *skid-steering*, autrement dit de type “char”, ne disposent pas de roues directionnelles. La modification dynamique des vitesses permet de changer la position du centre instantané de rotation (CIR) du robot, ce qui permet au robot d’exécuter tous types de trajectoire compatibles avec la structure de châssis.

La société Canadienne Clearpath Robotics¹ commercialise depuis 2009 des véhicules téléopérés rapides à roues non directionnelles (Unmanned Ground Vehicles, UGV), principalement utilisés pour des applications d’exploration et d’agriculture, à titre d’exemple le robot *skid-steering* “Husky UGV” illustré dans la Figure 1.1.

L’avantage de ce type de structure réside dans leur facilité de conception mécanique (4 roues fixes motorisées), qui apporte une bonne robustesse mécanique. En revanche, la manoeuvrabilité est faible (la vitesse de rotation dépend fortement des conditions d’adhérence au sol) avec un rendement énergétique faible également puisque les changements de direction ne peuvent se faire qu’à travers des glissements au niveau des interfaces roues/sol.

Robot type voiture

Le mécanisme de direction est assuré par l’épure de Jeantaud, appelé aussi épure d’Ackermann, qui consiste à utiliser une barre de direction qui relie rigidement des biellettes de direction, de manière à satisfaire les contraintes cinématiques du véhicule. En virage, et dans l’hypothèse de non-glissement, les axes des roues directrices (essieu d’avant) se coupent au centre instantané de rotation (CIR) situé sur le prolongement de l’essieu arrière. Cette cinématique est couramment utilisée sur les véhicules routiers ainsi que sur des robots mobiles. Un exemple d’un véhicule intelligent autonome utilisant ce type de cinématique est illustré par la Figure 1.1. Il s’agit du véhicule intelligent APACHe² (une Renault Fluence ZE électrique) du laboratoire Heudiasyc.

Robot à roues directionnelles indépendantes

Ces dernières années, les constructeurs automobiles s’intéressent de plus en plus au développement et l’implémentation de la technique des quatre roues directrices et motrices (en anglais :

1. [La société Clearpath Robotics](#)

2. [Véhicules intelligents du laboratoire Heudiasyc](#)

four wheel-steering, soit 4WS). Ce type de véhicule 4WS s'avère intéressant et prometteur pour les différentes applications robotiques, essentiellement pour le contrôle de la dynamique latérale à grande vitesse et pour la réalisation des chemins de fortes courbures et des manœuvres agressives [Cariou et al. \(2009\)](#). Ainsi, la cinématique 4WS permet au véhicule de se déplacer facilement dans des environnements complexes et de se faufiler dans des passages étroits et sinueux [Micaelli et Samson \(1993\)](#). La particularité de ce type de système est que les roues avant et arrière sont généralement orientées dans le sens opposé afin d'offrir une meilleure flexibilité et maniabilité, notamment à grande vitesse [Whitehead \(1988\)](#). Cependant, les effets de glissement ne peuvent pas être négligés vu le caractère tout terrain et la dynamique rapide de ces véhicules. C'est le cas de la plateforme RobuFAST conçue pour atteindre la vitesse de $12m.s^{-1}$ qui est l'objectif du projet FAST [Krid \(2012\)](#), [Lucet et al. \(2015\)](#), ainsi que d'autres robots tout-terrain rapides à roues directrices et motrices, développés pour la recherche, qu'on cite à titre d'exemple dans le tableau 1.1.

Tableau 1.1. Quelques exemples des robots mobiles à roues rapides

Rover	Braquage max ($^{\circ}$)	Vitesse max ($m.s^{-1}$)	Source
RobuFAST	20	12	Lucet et al. (2015)
Quad	40	18	Bouton (2009)
EZ10	15	8	Nizard et al. (2016)
Ares 620	45	12	Lenain et al. (2006)
Arroco	25	5	Hugo et Couble (2013)

Un autre exemple de robot mobile tout-terrain utilisant ce type de cinématique est donné dans la Figure 1.1. La société SWINCAR³, spécialisée dans le secteur automobile (loisirs, agriculture, etc) en France, a conçu un robot mobile SWINCAR e-Spider. Ce robot est un véhicule électrique tout terrain à quatre roues motrices et directrices, capable d'évoluer sur des terrains très variés et accidentés. Ce véhicule possède un centre de gravité très bas et il peut s'incliner en virage et rester d'aplomb dans les dévers tout en maintenant le contact roues/sol dans les passages les plus escarpés grâce au concept pendulaire et aux systèmes de suspension.

Le système de suspension

Le système de suspension sur une plate-forme tout-terrain permet d'augmenter sa stabilité et ses capacités de franchissement tout en garantissant le maintien du contact des roues avec le sol pour garder la meilleure traction et propulsion possible. Ces suspensions peuvent être passives ou actives. Dans le cas des suspensions actives, il est possible d'élaborer des lois de commande agissant sur ces suspensions afin d'améliorer la locomotion du système et d'adapter le comportement du véhicule en fonction des irrégularités rencontrées, voire d'anticiper des chocs importants.

Dans cette thèse, nous choisissons une plate-forme à quatre roues directrices et motrices équipée de quatre suspensions passives afin d'améliorer ses capacités d'évolution et d'accroître sa stabilité et permettre d'adapter son comportement selon les irrégularités du terrain et les chocs imprévus.

3. [La société SWINCAR](#)

1.2 Modélisation et modèles d'interaction roue-sol

Dans la littérature, il existe plusieurs modèles de véhicule dont les principaux sont les modèles de type 4 roues et les modèles bicyclettes. Les modèles de type bicyclette cinématique sont généralement utilisés dans des cas d'application où l'hypothèse de RSG est valide. Lors des grandes sollicitations dynamiques, ces modèles cinématiques ne sont plus suffisants pour élaborer les lois de commande de stabilisation du véhicule. Dans certains travaux, le modèle cinématique est étendu en intégrant les paramètres de glissement [Cariou et al. \(2009\)](#), et reste suffisant jusqu'à une vitesse de déplacement donnée. Lorsque le phénomène de glissement est important et ne peut pas être négligé, les modèles dynamiques restent les plus représentatifs pour mieux contrôler le comportement du véhicule, notamment à haute vitesse. Néanmoins, ces modèles dépendent largement de la modélisation des forces de contact roue/sol.

Tableau 1.2. Les modèles d'interaction roue-chaussée dans la littérature.

Nom de modèle	Type	Description	Source
Modèle Pacejka	Semi-empirique	Le modèle le plus utilisé et le plus important. Il peut décrire l'interaction avec justesse mais il est considéré sur-paramétré.	Pacejka (2005)
Modèle Dahl	Analytique	Un ancien modèle basé sur la généralisation de frottement de Coulomb.	Dahl (1968)
Modèle Bliman-Sorine	Analytique	Il peut représenter plusieurs phénomènes de frottement, amélioration du modèle de Dahl.	Bliman et Sorine (1993)
Modèle LuGre	Analytique	Amélioration du modèle de Bliman-Sorine. Il combine les avantages du modèle de Dahl et les caractéristiques de régime permanent.	De Wit et Tsiotras (1999)
Modèle Fiala	Empirique	Modèle simple basé sur les caractéristique du régime permanent couple/force de contact roue/sol.	Blundell et Harty (2004)
Modèle Dugoff	Analytique	Modèle analytique moins réaliste que le modèle de Pacejka car il suppose une répartition des forces uniformes sur la zone de contact pneu/sol.	Dugoff et al. (1969)

La traction est influencée par la déformation, l'usure et l'adhérence des pneus ainsi que par la matière du sol. Ces différents facteurs augmentent la difficulté de décrire analytiquement les forces de traction [Sidek et Sarkar \(2008\)](#). Pour des raisons de simplification on s'intéresse dans ce travail, uniquement aux forces générées par la déformation tangentielle du pneu, qui peuvent être décomposées en deux : la force longitudinale et la force latérale.

Dans la littérature, plusieurs chercheurs ont essayé de donner des modèles mathématiques d'interaction roue-chaussée qui peuvent être classifiés en deux branches majeures : les modèles empiriques et les modèles analytiques. Les modèles empiriques sont basés sur les techniques d'ajustement de courbe, ils peuvent capter avec justesse les caractéristiques non-linéaires des forces de traction, mais la plupart de ces modèles manquent d'interprétation physique. En outre, ils ne peuvent pas refléter directement l'effet de certains facteurs dynamiques tel que l'hystérésis, la pression du pneu et l'humidité de la surface. Par contre les modèles analytiques qui sont composés d'équations différentielles peuvent modéliser ces facteurs dynamiques. Le tableau 1.2 présente quelques modèles d'interaction existants dans la littérature pour décrire le comportement du contact roue/sol.

En général, les modèles d'interaction roue/sol nécessitent l'identification de nombreux coefficients qui sont parfois difficiles à estimer en temps réel du fait de leurs non-linéarités. Par exemple, la formule de Pacejka permet de tracer les réponses des efforts latéraux et longitudinaux en fonction des paramètres de glissement, mais elle nécessite l'identification expérimentale de 12 paramètres pour l'effort latéral et 8 paramètres pour l'effort longitudinal. Afin de pouvoir réduire le nombre des paramètres à estimer, une forme linéaire des efforts de contact très simple est largement appliquée sous l'hypothèse dite de pseudo-glissement (les glissements sont considérés faibles). Cette hypothèse est prise en compte dans [Krid et Benamar \(2011\)](#) pour décrire la dynamique de roulis, ainsi que [Bouton et al. \(2007\)](#) sur la dynamique dans le plan du lacet. Cependant, les variables de glissement sont difficilement mesurables à travers des moyens directs. Ce qui a incité les chercheurs à développer des algorithmes capables d'estimer et de reconstruire l'état du robot et de son environnement à partir des capteurs embarqués, tant extéroceptifs que proprioceptifs (e.g., GPS, centrale inertielle, etc.) [Sentouh et al. \(2007\)](#), [Lenain et al. \(2010\)](#). Cette forme linéaire des efforts de contact sous l'hypothèse de pseudo-glissement simplifie grandement le modèle du véhicule, facilitant ainsi la synthèse de la loi de commande.

1.3 Commande et estimation pour les véhicules tout-terrain

Trois tâches peuvent être distinguées dans la commande des robots mobiles à roues rapides non-holonomes : (1) la stabilisation de configurations fixes dont l'objectif est de stabiliser asymptotiquement le robot dans une position d'équilibre donnée, (2) la stabilisation de trajectoires dont le but est de permettre au robot de suivre une courbe de référence avec une contrainte horaire dite trajectoire et (3) le suivi de chemin.

De nombreux types de loi de commande pour la stabilisation de trajectoire ou le suivi de chemin ont été développés dans la littérature. Ces lois de commande pilotent les actionneurs de direction et de vitesse de rotation des roues. La plupart des stratégies de suivi de chemin sont basées sur un modèle cinématique. Par exemple, [Egerstedt et al. \(1997\)](#) et [Ailon et al. \(2005\)](#) ont développé deux techniques de suivi de trajectoire en utilisant un modèle cinématique du véhicule. Ce type de commande reste efficace à faible vitesse et pour des glissements dans la liaison roue/sol négligeables. En effet, les phénomènes liés à l'évolution à haute vitesse sur des terrains naturels et non préparés (glissements, irrégularités du terrain, etc.) doivent être considérés en tenant compte de leurs variabilités (modification des conditions du terrain). Dans ce contexte, un algorithme de contrôle adaptatif de suivi de trajectoire a été développé dans [Lenain et al. \(2003\)](#) basé sur un modèle cinématique étendu (i.e., modèle cinématique bicyclette intégrant les angles de glissement). Cet algorithme permet au véhicule de suivre un chemin

de référence dont la précision dépend de l'estimation correcte des angles de dérive (l'angle qui relie le vecteur vitesse réel du centre de la roue et son vecteur vitesse longitudinale). Une autre commande adaptative non linéaire de suivi de chemin basée sur la théorie du contrôle en mode glissant a été développée dans Wang *et al.* (2016). Ce contrôleur présente l'avantage de pouvoir découpler la direction des roues avant et arrière en calculant indépendamment les commandes adéquates de braquage avant et arrière. De telles approches restent fiables pour certains types de chemins en régime permanent (e.g., ligne droite, grand cercle à courbure constante, etc.), mais cette fiabilité peut être compromise en régime transitoire, aux endroits de variation de courbure et particulièrement en présence de latence dans la chaîne d'actionnement. Dans ce cas, des stratégies de commande prédictives ou adaptatives peuvent être mises en place pour anticiper les variations de consigne et les retards et pallier aux variations d'environnement. Une commande prédictive généralisée non linéaire à temps continu (NCGPC) a été développée dans Krid *et al.* (2017) basée sur la dynamique latérale du véhicule. Aussi, deux commandes prédictive et adaptative ont été utilisées dans Lenain *et al.* (2005) pour assurer la tâche du suivi de chemin. L'avantage de cette approche réside dans l'anticipation de la variation de la courbure pour prédire à l'avance les commandes à envoyer aux actionneurs qui ont leurs propres temps de réponse. Cependant, la plupart de ces commandes ne prend pas en compte les contraintes physiques et/ou intrinsèques du véhicule qui sont inévitables et doivent être respectées (par exemple, les limites sur l'angle et la vitesse du braquage, contraintes sur les conditions d'adhérence).

Au cours d'un déplacement à haute vitesse dans un environnement naturel (terrains peu structurés, variabilité forte des paramètres d'environnement), le véhicule subit une excitation dynamique importante (avec notamment des vibrations et des chocs) qui entraîne une dégradation de la qualité de la transmission des efforts dans les liaisons roue/sol, ce qui peut conduire à une dégradation de la qualité de suivi de chemin voire une perte de stabilité du véhicule. Les performances dynamiques en terme de manœuvrabilité et de stabilité du véhicule se trouvent alors être particulièrement affectées. Dans ce contexte, il est donc nécessaire de doter ces robots de contrôleurs efficaces et robustes afin de pouvoir maîtriser les glissements aux interfaces roue/sol, ainsi que de réagir convenablement à la présence d'obstacles en modifiant en temps réel la consigne du robot.

Afin de pouvoir disposer d'une loi de commande prenant en compte les conditions d'adhérence des roues sur le sol, il est nécessaire d'avoir accès à une estimation en ligne des paramètres intervenant dans la dynamique du véhicule au sein de son environnement. Pour ce faire, certaines stratégies utilisent des moyens de mesure directe tout en développant des capteurs spécifiques à intégrer dans les pneumatiques. Par exemple, L'ENSEEIH⁴ de Toulouse a développé un capteur de vitesse qui permet de mesurer la vitesse latérale du véhicule basée sur le principe effet doppler (cf. Figure 1.2). Des capteurs ont été utilisés dans Doumiati *et al.* (2010) pour mesurer les forces de contact roue/sol et l'angles de glissement de la roue (cf. Figure 1.2). Néanmoins, les résultats donnés par ces mesures directes ne sont généralement pas encourageants. En effet, compte tenu de la rugosité routière, irrégularité de terrain et la qualité de montage de ces capteurs, une certaine imprécision est induite sur les mesures qui sont généralement très bruitées (en dehors de fait que ce type de capteurs est généralement coûteux et encombrant).

4. École Nationale Supérieure d'Électrotechnique, d'Électronique, d'Informatique, d'Hydraulique et des Télécommunications



Figure 1.2. Exemples des méthodes de mesure directe des paramètres d'état du véhicule. [Gauche] Capteur pour la mesure de la vitesse latérale développés par l'ENSEEIH. [Droite] Capteur de force installé dans la roue [Doumiati et al. \(2010\)](#).

D'autres approches utilisent la dynamique latérale, longitudinale ou de roulis pour reconstruire en temps réel les variables difficilement mesurables par les méthodes directes telles que la vitesse latérale du véhicule. Ces approches sont généralement basées sur des modèles linéaires. Par exemple, un observateur multi-modèles fusionnant un modèle cinématique étendu et un modèle dynamique a été utilisé dans [Lenain et al. \(2010\)](#) pour reconstruire en temps réel les angles de glissements et les rigidités de dérive des pneumatiques afin d'assurer un suivi précis du chemin de référence. Une autre approche par backstepping a été mise en place dans [Bouton et al. \(2007\)](#) pour observer la rigidité de dérive en utilisant un modèle dynamique et en supposant que les rigidités de dérive avant et arrière sont égales. [Wang et Junmin \(2013\)](#) ont proposés une méthode d'estimation de la rigidité de dérive et du coefficient de frottement en utilisant la dynamique longitudinale du véhicule et une synthèse basée sur une fonction de Lyapunov. Un estimateur temps réel de coefficient de frottement au niveau de contact roue/sol a été développé dans [Kaldenbach et al. \(2017\)](#). Cet estimateur combine le modèle d'un quart du véhicule et la formule empirique de Pacejka pour estimer en ligne le coefficient de frottement. Néanmoins, la plupart de ces estimateurs sont basés sur des approches linéaires conçues en s'appuyant sur des hypothèse de simplification et de linéarisation, ce qui peut dégrader la précision de l'estimation de ces paramètres du contact roue/sol, notamment dans le cas de terrains naturels et de nature physique variée. Ainsi, des approches d'estimation non linéaire seront plus appropriées et requises pour reconstruire avec précision et rapidité les paramètres du sol que ce soient physiques (conditions d'adhérence au niveau du contact roue/sol) ou géométriques (irrégularité du terrain).

De même, l'estimation de la géométrie du terrain (e.g., angle de pente, etc.) est tout aussi primordiale pour compenser les effets de la gravité et assurer la stabilité du véhicule sur des terrains irréguliers. Dans [Chapuis et al. \(2002\)](#), les auteurs présentent une méthode de reconstruction de la pente et du dévers basée sur un système de vision. Cette méthodologie consiste à identifier les limites de la route ainsi que les tracés délimitants les voies à l'aide d'un modèle d'image de la chaussée. Une fois que la délimitation de la chaussée est reconstruite, une identification des paramètres d'un modèle en trois dimensions de la chaussée est réalisée à l'aide d'un modèle de connaissance. Certaines méthodes combinent les mesures GPS et inertielles pour observer en temps réel les angles de dévers et de pente du sol [Jo et al. \(2013\)](#). Une méthode d'estimation du roulis et du dévers est présentée dans [Ryu et Gerdes \(2004a\)](#). Cette estimation est basée sur un modèle bicyclette dynamique de roulis et un modèle linéaire d'interaction roue/chaussée.

L'algorithme d'observation utilisé ici est un observateur de Luenberger, dont le gain a été calculé à l'aide d'un filtre de Kalman. Une des mesures mise à la disposition de cet algorithme est celle du roulis absolu obtenue à l'aide de deux GPS différentiels, posé à chaque extrémités du véhicule expérimental (avant et arrière). Ce qui rend cette approche coûteuse. [Tseng \(2001\)](#) a proposé deux méthodes d'estimation de l'angle de dévers de la chaussée. La première est basée sur une hypothèse statique et la seconde est considérée dans le cadre dynamique. L'algorithme est basé sur un modèle de bicyclette linéaire et sur l'identification du dévers issus de l'accéléromètre latéral ainsi que sur la vitesse de lacet. [Hahn et al. \(2002\)](#) a proposé une méthode de reconstruction du dévers à l'aide d'un observateur de perturbation. Le modèle du véhicule est un modèle bicyclette dynamique étendu à un vecteur de perturbations contenant les variations de la matrice d'évolution dues aux incertitudes sur les rigidités de dérive ainsi qu'un terme lié au dévers de la chaussée

1.4 Planification locale des chemins pour l'évitement réactif d'obstacle

Il est possible de distinguer deux niveaux de planification. Le premier niveau calcule l'itinéraire global du véhicule qui rallie les deux points initial et final. On parle dans ce cas de la planification globale. Dans cette thèse ce type de planification n'est pas considéré. Le deuxième niveau de planification, dit local, permet au véhicule de modifier localement son chemin ou trajectoire de référence global, notamment pour l'élaboration d'un comportement réactif d'évitement d'obstacles.

La navigation d'un rover autonome en milieu ouvert et non maîtrisé nécessite deux actions primordiales : la détection en temps réel des obstacles et la génération automatique des chemins d'évitement qui doivent être réalisables et admissibles par le véhicule d'un point de vue cinématique et dynamique. Bien entendu, plus la vitesse du robot est grande plus le temps de traitement de ces deux actions doit être petit. Pour cette raison, les données capteurs doivent subir très peu de traitement pour calculer rapidement les bonnes entrées de commande à envoyer au véhicule.

Le chemin de référence global à suivre est défini lors de la préparation de la mission et peut être modifié dans le cas de la détection d'un obstacle afin d'assurer un évitement réactif de celui-ci, les contraintes cinématiques et dynamiques imposées. Dans ce cadre, plusieurs techniques ont été développées pour assurer la tâche réactive d'évitement d'obstacles. Celles-ci portent généralement sur la définition d'autres chemins à suivre en cas de présence d'un obstacle. Par exemple, la méthode classique des champs de potentiels peut être appliquée pour planifier localement des chemins admissibles d'évitement d'obstacles [Khatib \(1985\)](#). Le principal inconvénient de cette méthode est la présence de minima locaux dans le potentiel qui peut faire en sorte que le robot soit bloqué dans une "impasse". Il est également difficile de prédire la courbure du chemin obtenu, qui peut être très forte pour des obstacles polygonaux [Ulrich et Borenstein \(2000\)](#), et donc de définir des chemins d'évitement qui sont compatibles avec les contraintes du véhicule (cinématique, dynamique, angle de braquage, etc.). Par ailleurs, elle n'est pas compatible avec une tâche de suivi de chemin puisqu'elle nécessite un point cible et qu'il est difficile de l'extrapoler à des systèmes dynamiques non-holonomes.

Une autre approche similaire à celle des champs de potentiel a été proposée dans [Borenstein et al. \(1991\)](#) avec les VFH (Vector Field Histogram). Cette approche souffre aussi de la présence de minima locaux. Plusieurs extensions ont été proposées, notamment pour éviter les blocages,

telles que, VFH* dans Ulrich et Borenstein (2000) et VFH⁺ dans Ulrich et Borenstein (1998). Une autre méthode basée sur un VFH adapté aux véhicules à roues non-holonomes en fonction de la plage autorisée d'angle de braquage Feiten *et al.* (1994). Cette méthode donne des chemins d'évitement en respectant les contraintes cinématiques du véhicule. Aussi, Simmons (1996) présente une méthode basée aussi sur le concept VFH pour assurer l'évitement tout en respectant les contraintes dynamiques du véhicule. Concrètement, l'algorithme génère des arcs de cercle qui permettent au véhicule l'évitement d'obstacles. Cependant, ces chemins d'évitement (arcs de cercle) restent fiables à faible vitesse et induisent des variations brutales de courbure qui peuvent générer des instabilités du véhicule à haute vitesse.

D'autres méthodes de planification locale des chemins d'évitement sont basées sur la commande prédictive (MPC). Une commande MPC est utilisée dans Du Toit et Burdick (2010) pour assurer la tâche d'évitement d'obstacle. Cet algorithme est présenté sous la forme d'un problème d'optimisation où les incertitudes de mesure sont prises en compte. La méthode calcule une trajectoire d'évitement par la résolution temps réel de ce problème d'optimisation. Une autre commande MPC est également utilisée dans Morette (2009) pour la planification locale des chemins d'évitement d'obstacle à partir d'un modèle dynamique du véhicule. Cependant, ces approches sont généralement trop complexes pour être résolues en temps-réel en raison de leur charge calculatoire importante.

Une autre méthode, appelée "méthode des tentacules", se base sur la génération temps réel d'un ensemble de chemins circulaires et/ou rectilignes (parallèles à la route) pour assurer l'évitement d'obstacles Himmelsbach *et al.* (2011). Cependant ces formes de chemins ne sont pas convenables à haute vitesse à cause du changement important de la courbure (entre une ligne et un arc de cercle). Pour améliorer cette approche, la forme clothoïde a été adoptée dans Alia *et al.* (2015) et Fokam (2014) permettant de changer la courbure d'une façon proportionnelle à l'abscisse curviligne. Néanmoins, la difficulté de ce type de chemin est d'assurer de la continuité aux points de passage.

Différentes méthodologies de planification locale pour l'évitement d'obstacles basées sur l'utilisation des chemins polynomiaux existent dans la littérature. Par exemple, un chemin d'évitement est générée dans Shiller *et al.* (1991). il est basée sur des B-splines, et il est le résultat d'une optimisation multicritère très complète : temps de parcours, cinématique et dynamique du véhicule, risque de renversement, etc. Guechi *et al.* (2008) et Simba *et al.* (2016) ont présentés une stratégie d'évitement d'obstacles dynamiques par l'intermédiaire de deux courbes de Bézier conçues en temps-réel selon la vitesse et la direction de l'obstacle mobile. Néanmoins, cette méthode est valide seulement si la vitesse et l'accélération du robot de type unicycle sont faibles. Un bilan détaillé des techniques de planification les plus utilisées sur véhicule est donné par González *et al.* (2015).

À l'issue de cette brève présentation, il est notable que les méthodes de planification et commande destinées aux véhicules rapides à deux trains directeurs sont peu nombreuses et en pratique souvent inefficaces. Le problème de contrôle-commande d'un véhicule autonome tout-terrain dépend bien entendu de son architecture mécanique, de sa cinématique, de sa dynamique d'évolution, du type de terrain, etc. Cette multiplicité de facteurs explique la diversité des solutions qu'on trouve dans la littérature. Dans notre cas, nous nous positionnons dans le cadre de la navigation autonome d'un robot mobile tout-terrain, à deux trains directeurs, capable d'évoluer rapidement dans des milieux naturels.

2 Objectifs et contributions

2.1 Contexte et objectifs de la thèse

Dans le cadre d'une application robotique telle que l'agriculture, le sol peut présenter localement des irrégularités géométriques fortes (pente, dévers, bosse, etc) ainsi que des variations physiques soudaines (herbe, sable, argile, boue). Aussi, les champs agricoles sont des grands espaces qui requièrent des vitesses de déplacement sans commune mesure avec celles des robots d'intérieurs. Les contraintes de déplacement rapide dans de tels environnements impliquent de fortes variations des efforts inertiels et gravitationnels couplées à une variabilité des paramètres d'environnement, qu'il convient de prendre en compte dans la génération de trajectoires et les commandes des actionneurs. Ainsi, l'objectif de ce travail de thèse est de développer des lois de commandes capables de générer à la fois des chemins admissibles et réalisables pour l'évitement réactif des obstacles, ainsi que les actions nécessaires qui assurent le suivi de chemin planifié tout en respectant les différentes contraintes intrinsèques et physiques du robot.

2.2 Approche proposée

D'un point de vue fonctionnel, notre travail peut se décomposer en 3 fonctions principales : la perception de l'environnement et l'estimation d'état, la planification des chemins et le contrôle du véhicule (voir Figure 1.3). La première fonction permet d'une part la perception de l'environnement et l'identification d'obstacles. D'autre part, elle permet l'estimation des paramètres liés à l'état du robot et de son environnement. La seconde fonction, quant à elle, permet la planification locale du chemin admissible d'un point de vue cinématique et dynamique pour l'évitement réactif des collisions. Enfin, pour le bas-niveau, on privilégie une commande MPC sous contraintes basée sur un modèle dynamique du véhicule et un modèle linéaire des pneumatiques pour assurer la tâche de suivi du chemin local ou global et respecter les limites des actionneurs du véhicule.

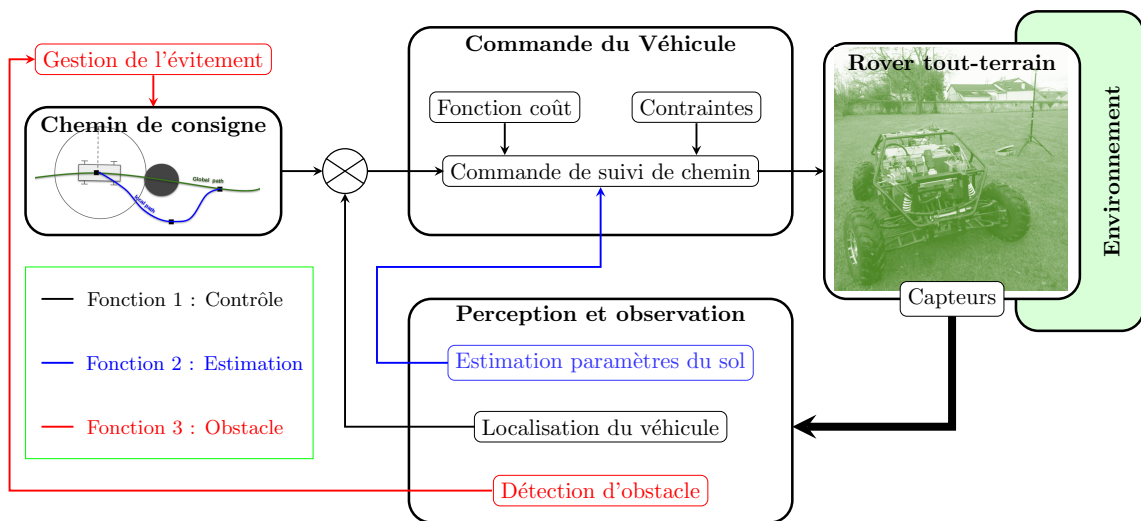


Figure 1.3. Schéma global du contrôle-commande d'un rover tout-terrain.

2.3 Description de la plateforme numérique et expérimentale

Tous les algorithmes développés dans le cadre de cette thèse seront validés dans un premier temps à haute vitesse sur la plateforme numérique sous ROS-GAZEBO, puis par des tests réels sur la plateforme expérimentale, appelé “*SPIDO*”, qui a été acquise dans le cadre des équipements d’excellence RoboTex⁵ (cf. Figure 1.4). La plateforme possède 4 roues motrices, 2 essieux directeurs indépendants, 4 suspensions à double-triangle et deux barres anti-roulis actives. Elle est équipée d’une odométrie, de capteurs de position pour les angles de braquage, d’un GPS-RTK, d’une centrale inertielle et d’un LiDAR 3D pour la détection d’obstacles.

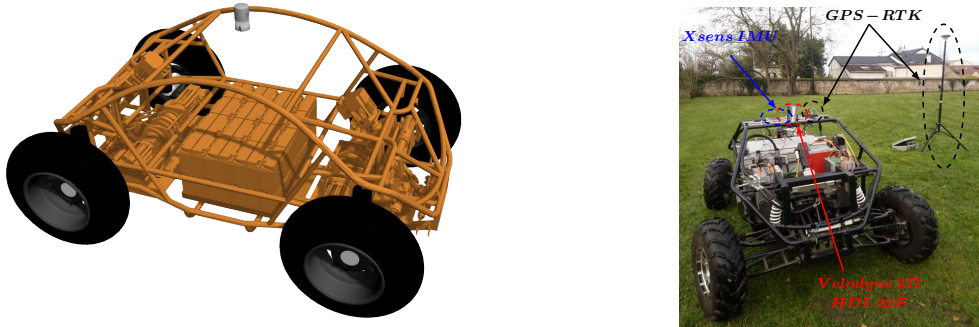


Figure 1.4. [Gauche] : La maquette numérique de simulation ; [Droite] : La plateforme expérimentale “*SPIDO*” et les différents capteurs embarqués.

Les principales caractéristiques de la maquette numérique utilisée sous ROS-GAZEBO, ainsi qu’une description détaillée de l’architecture matérielle et logicielle de la plateforme expérimentale sont exposées dans l’annexe A.

2.4 Contributions de la thèse

Commande de véhicules autonomes en présence de glissement

Les robots terrestres du futur auront à aller plus vite et plus loin et à intervenir d’une façon autonome dans des milieux naturels et incertains. Dans ce contexte, il est indispensable de prendre en compte la dynamique du véhicule ainsi que ses interactions avec son environnement. L’interaction roue/sol est une liaison complexe qui dépend de la rhéologie du pneu, celle du sol, et des propriétés tribologiques entre ces deux matériaux. Maîtriser la dynamique du véhicule passe nécessairement par la maîtrise de ses interactions avec le sol. Il est alors nécessaire de développer des lois de commande qui permettent à la fois le suivi du chemin planifié et le respect des différentes contraintes liées principalement au robot et à l’adhérence roue/sol (braquage, contact, glissement, etc). Les lois de commande synthétisées se basent sur un modèle dynamique du véhicule et un modèle linéaire du pneumatique. Ce modèle a été développé en prenant en compte les glissements roue/sol et la géométrie locale du terrain (pente, bosses, etc.)

Dans un premier temps, une commande LQR mixée avec un observateur d’état de Kalman-Bucy a été développée pour le suivi d’un chemin de référence défini hors ligne. Cette commande a

⁵. ROBOTEX est un réseau national de plateformes expérimentales de robotique qui constitue un «équipement d’excellence» - (ANR-10-EQPX-44-01) | [Equipex-Robotex - Plateformes](#)

pour objectif de minimiser à la fois les erreurs de suivi du chemin désiré mais aussi les glissements latéraux dans le but de renforcer davantage la stabilité latérale du véhicule. Cette commande est utilisée par la suite pour la validation de nos algorithmes d'observation de paramètres du sol. La validation a été réalisée en simulation à grande vitesse sous ROS-GAZEBO puis par des expérimentations sur le robot réel “*SPIDO*” (voir Figure (1.4)) [Fnadi et al. \(2018\)](#).

Une autre commande prédictive sous contraintes (MPC) a été synthétisée pour assurer la tâche de suivi de chemin (commande bas-niveau). Cette commande MPC est privilégiée par rapport à la commande LQR du fait de sa faculté à prendre en compte à la fois la dynamique courante, future (l'état prédit sur un horizon de temps fini) et surtout les différentes contraintes (limites de braquage, limites de la zone d'adhérence du pneu, etc). Cette approche est développée sous forme d'un problème d'optimisation quadratique (QP) qui minimise un critère quadratique tout en respectant les différentes contraintes du véhicule (e.g., limites de direction, limites de la zone de pseudo-glissement, etc.). Ce travail a fait l'objet de l'article [Fnadi et al. \(2019a\)](#).

Estimation en ligne de paramètres physiques et géométriques du sol

Les contrôleurs de suivi de chemin pour un véhicule autonome sont souvent conçus en utilisant un modèle dynamique ou cinématique dont certains utilisent des paramètres du contact roue/sol. Cela rend l'efficacité de ces contrôleurs fortement dépendante de l'estimation des paramètres géométriques et physiques du sol, en particulier pour les robots mobiles destinés à se déplacer dans des milieux naturels et accidentés.

Globalement, les modèles de contact roue/sol existants dans la littérature décrivent les réponses des efforts latéraux et longitudinaux en fonction de paramètres de glissement. Ces paramètres d'adhérence dépendent des conditions instantanées du contact roue/sol (par exemple, type de sol, pneu, conditions météorologiques, etc.) et sont difficilement accessibles par des mesures directes. Un nouvel observateur non linéaire a été conçu pour estimer en temps réel et indépendamment les rigidités de dérive avant et arrière, qui sont liées à la fois aux propriétés du pneu et du sol. Les performances de cet observateur non linéaire mixé avec le contrôleur LQR sont évaluées à la fois par des simulations numériques et expérimentalement sur le rover “*SPIDO*” à différentes vitesses. Cette nouvelle contribution a fait l'objet de l'article [Fnadi et al. \(2019b\)](#).

Les commandes synthétisées dépendent aussi de la géométrie locale du terrain qui doit être estimée en temps réel pour garantir la stabilité latérale du véhicule et la convergence rapide de la commande. Un autre estimateur de Luenberger a été développé dans ce travail de thèse qui utilise à la fois la vitesse latérale et les données des capteurs extéroceptifs (GPS-RTK, IMU) afin d'estimer en temps réel et avec précision l'angles de pente et l'angle de dévers [Fnadi et al. \(2019c\)](#).

Les résultats obtenus montrent que l'estimation en ligne des paramètres physiques et géométriques liés au sol (rigidités de dérive du pneu, géométrie du terrain) améliore grandement l'efficacité et la fiabilité de la commande du suivi de chemin (la convergence rapide et précise ainsi que la stabilité latérale du véhicule). Un article de revue détaillant tous ces travaux est en cours de préparation.

Planification du chemin de contournement pour l'évitement d'obstacle

En robotique mobile, on distingue deux niveaux de planification. Le premier niveau calcule l'itinéraire global du véhicule qui rallie les deux points initial et final. On parle dans ce cas de planification globale. Dans cette thèse ce type de planification n'est pas considéré, le chemin global du véhicule est défini hors ligne. Le deuxième niveau de planification, dit local, permet au véhicule de modifier localement ce chemin de référence, notamment pour l'élaboration d'un comportement réactif d'évitement d'obstacles.

Dans ce travail de thèse, nous traitons une commande haut niveau liée à la planification locale du chemin pour assurer la tâche d'évitement des collisions. Elle porte principalement sur la détection et l'évitement réactif des obstacles. Dans un premiers temps, nous avons développé un algorithme de détection d'obstacles basé sur un traitement du nuage de points délivré par un LiDAR 3D de perception de l'environnement. L'évitement d'obstacle, quant à lui, est basé sur la génération et l'évaluation d'un ensemble de courbes de Bézier cubiques qui sont générées automatiquement et en temps réel dès qu'un obstacle est détecté autour du chemin de référence global.

Notre approche de planification locale calcule un nouveau chemin de contournement des obstacles détectés qui respecte les contraintes cinématiques et dynamiques du véhicule. Ce chemin de contournement est généré dès qu'un obstacle est détecté dans un certain rayon d'incidence autour du robot et le long de son chemin nominal. Ce nouveau chemin est déterminé selon deux étapes successives. Premièrement, on détermine le point de débranchement et le point de raccordement sur le chemin nominal, ainsi qu'un point intermédiaire selon la taille de l'obstacle détecté et la configuration actuelle du véhicule. Ce chemin doit remplir, d'une part, les restrictions de la continuité et la dérivabilité, notamment au niveau des points de connexion avec le chemin de référence global. D'autre part, la sécurité et la stabilité latérale du véhicule doivent être aussi assurées, en évitant les chemins à grandes courbures qui nécessitent des variations brusques ou violentes des angles de direction. Dans un second temps, on calcule les deux courbes de Bézier cubiques à courbure minimale permettant de relier le point de débranchement et le point de raccordement en passant par le point intermédiaire. Cette contribution à la planification locale du chemin de contournement a fait l'objet d'un article soumis [Fnadi *et al.* \(2020\)](#).

2.5 Organisation du mémoire

Les travaux réalisés dans cette thèse sont développés et présentés dans 4 chapitres après la présente introduction.

- Tout d'abord, le chapitre 2 est composé de deux parties. Dans un premier temps, pour décrire le comportement du véhicule, plusieurs modèles cinématique et dynamique sont proposés dans la littérature, ceux-ci sont développés avec des simplifications plus ou moins importantes selon le besoin de l'application à concevoir. Nous utilisons un modèle bicyclette dynamique couplé au modèle cinématique pour concevoir nos différents algorithmes. Cette modélisation dynamique prend en compte le phénomène de glissement au niveau de la liaison roue/sol (voir section 2). Pour procéder à la validation de nos observateurs des paramètres géométriques et physiques relatifs au sol qui font l'objet du

chapitre 3, il est indispensable de disposer d'une commande de suivi de chemin désiré quelconque. C'est dans cet esprit que dans la section 3, nous synthétisons une loi de commande pour le suivi de chemin. Celle-ci est basée sur la dynamique latérale du véhicule et d'un modèle de contact roue/sol linéaire, ainsi elle utilise l'approche LQR basée sur une minimisation d'un critère quadratique exprimé en fonction des erreurs de suivi et des entrées de commande.

- Le chapitre 3 présente deux observateurs dédiés à l'estimation des paramètres pneumatiques et de la géométrie locale du terrain. Le premier est un observateur non linéaire (NLO) destiné à estimer indépendamment et en ligne les rigidités de dérive avant et arrière des pneumatiques. Cet observateur est développé à partir d'un modèle dynamique bicyclette et le modèle linéaire du pneumatique. Il est capable de tenir en compte les variations des conditions d'adhérence au niveau du contact pneumatique/chaussée. Un deuxième observateur des paramètres de la géométrie locale du terrain (OAT) est développé afin d'adapter en temps réel les angles d'inclinaison de la plateforme par rapport à son environnement. Ces deux estimateurs améliorent la précision de la commande de suivi de chemin (LQR) ainsi que la stabilité latérale du véhicule.
- Dans le chapitre 4, nous présentons une commande prédictive (MPC) sous contraintes pour la tâche de suivi de chemin. Cette commande est exprimée sous la forme d'un problème d'optimisation quadratique (QP) prenant en compte les contraintes intrinsèques et physiques du véhicule (e.g, braquage et glissement) et minimisant une fonction coût basée sur la prédiction de l'erreur entre la sortie choisie et la référence sur un horizon de temps fini.
- Dans le chapitre 5, une méthode de planification locale d'un chemin de contournement pour l'évitement réactif d'obstacles est introduite. Après l'étape de perception de l'environnement et la détection d'obstacles, nous détaillons l'algorithme d'évitement qui utilise deux courbes de Bézier cubiques admissibles et réalisables afin de contourner automatiquement et rapidement tous les obstacles existants et non-prévus dans l'environnement de navigation et d'assurer la convergence rapide vers le chemin de référence global.

Chacune des méthodes théoriques développées est validée par le biais de simulation numériques avancées avec une maquette virtuelle sous ROS-GAZEBO ainsi que par des expérimentations réelles sur la plateforme "*SPIDO*" (robot mobile à quatre roues directrices et motrices). Nous finissons ce manuscrit en présentant les principales conclusions et perspectives de nos travaux.

Modélisation et stabilisation du robot en suivi de chemin

Sommaire

1	Introduction	17
2	Modélisation de robots mobiles à double braquage	17
2.1	Aperçu de l'état de l'art	17
2.2	Modèle à quatre roues non linéaire	18
2.3	Modélisation du contact roue-sol	19
2.3.1	Glissement transversal	19
2.3.2	Glissement longitudinal	20
2.3.3	Modèle linéaire du pneumatique	20
2.4	Modèle bicyclette	22
2.5	Modélisation cinématique opérationnelle du robot	25
2.6	Modèle final du véhicule	26
3	Synthèse de loi de commande pour le suivi de chemin	28
3.1	État de l'art sur la commande des véhicules autonomes	28
3.2	Synthèse de la loi de commande de suivi de chemin	29
3.3	Mise en place d'un observateur d'état	31
4	Validation en simulation et expérimentale	34
4.1	Filtrage des mesures	34
4.2	Validation en simulation	34
4.3	Résultats expérimentaux	36
4.3.1	Chemin en virage à droite	36
4.3.2	Chemin à deux manœuvres gauche-droite	38
5	Conclusion	40

1 Introduction

Le suivi automatique d'une trajectoire de référence est un problème d'automatique qui consiste à minimiser une ou plusieurs erreurs de suivi. En robotique, il s'avère qu'il est peu utile de suivre temporellement une courbe géométrique donnée mais il est préférable de garantir le suivi de la courbe géométrique opérationnelle et contrôler indépendamment la vitesse d'avance. Le problème de suivi de chemin nécessite un modèle du véhicule reliant les paramètres articulaires au paramètres opérationnels. Ce modèle peut être de nature purement cinématique et corrige les vitesses opérationnelles du véhicule, ou de nature dynamique en prenant en compte l'inertie du robot et ses accélérations. Compte tenu du contexte applicatif de navigation en milieu ouvert, le contrôleur de suivi de chemin doit être performant malgré les incertitudes paramétriques, le changement d'adhérence du sol, l'irrégularité de l'environnement, etc.

Dans ce chapitre, nous commençons par une brève description de la modélisation dynamique d'un rover tout-terrain à quatre roues directrices et motrices (4WS). Ce modèle décrit au mieux le comportement du véhicule dans le plan de lacet en présence de glissement ainsi que la distribution des efforts de gravité due aux déclivités du sol. Ce modèle sera couplé à un modèle cinématique caractérisant la tâche de suivi de chemin. Ensuite, une commande de suivi de chemin en présence de glissement est abordée et synthétisée en utilisant ce modèle dynamique dans le plan de lacet. Finalement, nous validons cette commande par des simulations numériques sous ROS-GAZEBO et par des expérimentations sur le robot réel. Cette commande sera à la base de la suite de nos prochains travaux pour la validation de nos différents algorithmes d'observation.

2 Modélisation de robots mobiles à double braquage

2.1 Aperçu de l'état de l'art

Une bonne modélisation du système qui prend en compte les phénomènes physiques importants est un pré-requis primordial à la synthèse d'une loi de commande qui répond aux besoins opérationnels. On peut distinguer principalement deux types de modèles qui sont largement utilisés dans la littérature : les modèles bicyclettes [Krid \(2012\)](#), [Rajamani \(2011\)](#) et les modèles de type 4 roues [Gillespie \(1992\)](#).

Les modèles bicyclettes sont les plus utilisés pour la planification de chemin et l'élaboration des lois de commande pour le contrôle du véhicule. Dans ces modèles, les dynamiques du roulis et du tangage sont généralement négligées [Bouton *et al.* \(2007\)](#), [Li et Wang \(2007\)](#). On distingue principalement trois modèles bicyclettes : le modèle géométrique, le modèle bicyclette cinématique et le modèle bicyclette dynamique [Snider *et al.* \(2009\)](#). Les deux premiers approximent bien le déplacement du véhicule pour des applications à faibles vitesses [Thrun *et al.* \(2006\)](#), [Ailon *et al.* \(2005\)](#). De nos jours, les modèles bicyclettes dynamiques, plus appropriés aux vitesses élevées, sont les plus utilisés dans le contrôle du véhicule [Lenain *et al.* \(2010\)](#), [Ackermann *et al.* \(1995\)](#). Néanmoins, les modèles bicyclettes ne sont plus suffisants pour modéliser le comportement dynamique du véhicule, notamment lors de la prise en compte des efforts de traction/freinage dans la stabilisation de lacet [Lucet \(2010\)](#), tels que les systèmes de freinage antiblocage (ABS), la régulation antipatinage (ASR) et les programmes de stabilité électronique (ESP) [Bera *et al.* \(2011\)](#), [Kim *et al.* \(2003\)](#).

La difficulté principale de la modélisation dynamique du véhicule réside dans la complexité de la liaison pneumatique/chaussée. Le choix de ce modèle est une étape importante dans la modélisation de la dynamique d'un véhicule. La modélisation du contact pneu/sol revient à déterminer la relation liant les forces de contact aux glissements longitudinaux et latéraux. De très nombreux modèles existent dans la littérature, nous pouvons citer : les modèles physiques généralement non-linéaire [Mancosu *et al.* \(2000\)](#), [Bernard et Clover \(1995\)](#), le modèle de Pacejka connu par la formule magique [Pacejka \(2005\)](#), le modèle de Dugoff [Dugoff *et al.* \(1969\)](#), le modèle de Hahn [Hahn *et al.* \(2002\)](#) et le modèle de Fiala [Hindiye et Gerdes \(2009\)](#).

Les modèles développés dans cette section seront la base de la synthèse de nos différents algorithmes. Pour avoir plus de détails sur la modélisation dynamique du véhicule, on peut se référer à [Gillespie \(1992\)](#), [Krid \(2012\)](#) et [Thuilot \(1995\)](#). Pour que la modélisation soit abordable, on ne prend pas en considération la dynamique de rotation des roues, ainsi que la dynamique de roulis et de tangage.

2.2 Modèle à quatre roues non linéaire

Le modèle de type 4 roues est donné par la Figure 2.1. En appliquant le principe fondamental de la dynamique, les équations différentielles des différentes dynamiques (longitudinale V_x , latérale V_y , de lacet V_ψ) du modèle à 4 roues sont données par les équations (2.1). Ces équations sont données en supposant dans un premier temps que le véhicule évolue dans un plan de lacet horizontal.

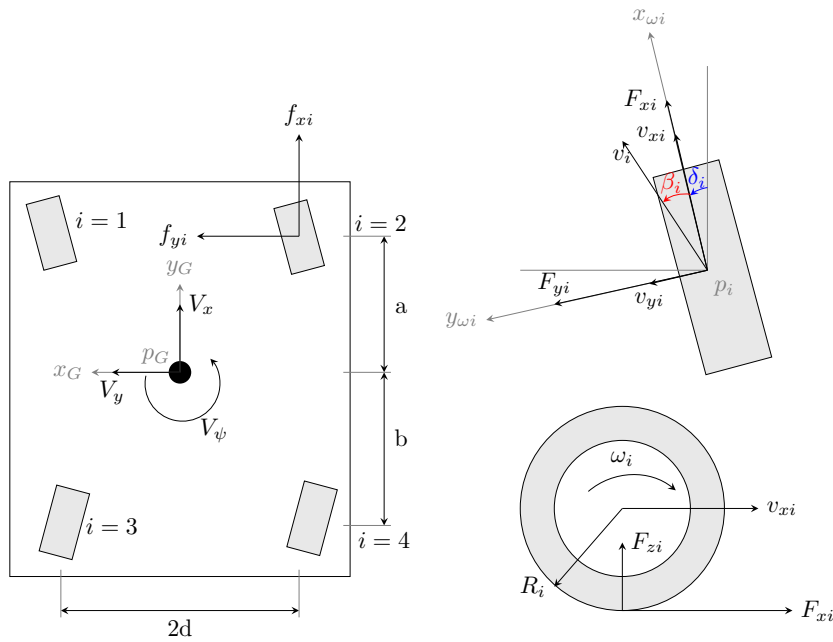


Figure 2.1. Notations utilisées pour la modélisation décrivant les forces de contact dans le repère de véhicule, les forces de contact dans le repère lié à la roue i , ainsi que les vitesses de rotation et de translation de la roue i .

$$m\dot{V}_x = mV_yV_\psi + \sum_{i=1}^4 f_{xi}, \quad (2.1a)$$

$$m\dot{V}_y = -mV_xV_\psi + \sum_{i=1}^4 f_{yi}, \quad (2.1b)$$

$$I_z\dot{V}_\psi = a(f_{y1} + f_{y2}) - b(f_{y3} + f_{y4}) + d(-f_{x1} + f_{x2} - f_{x3} + f_{x4}), \quad (2.1c)$$

où f_{xi} et f_{yi} sont respectivement les forces de contact roue/sol le long des axes longitudinal et latéral du véhicule (i.e., exprimées dans la base liée au corps du robot noté $\mathcal{R}_G = \{x_G, y_G, z_G\}$), et F_{xi} , F_{yi} sont respectivement les forces de contact roue/sol le long des axes longitudinal et latéral de la roue i (i.e., exprimées dans repère lié à la roue i noté $\mathcal{R}_{wi} = \{x_{wi}, y_{wi}, z_{wi}\}$). Les paramètres a , b et d sont les paramètres géométriques du véhicule (respectivement, l'empattement avant et arrière et la distance de la demi-voie), m et I_z sont respectivement la masse et le moment d'inertie du véhicule au centre de gravité.

Les composantes de la force longitudinale f_{xi} et latérale f_{yi} du pneu dans le repère véhicule sont modélisées comme suit :

$$f_{xi} = F_{xi} \cos \delta_i - F_{yi} \sin \delta_i, \quad (2.2a)$$

$$f_{yi} = F_{xi} \sin \delta_i + F_{yi} \cos \delta_i, \quad i \in \{1, 2, 3, 4\}, \quad (2.2b)$$

avec δ_i l'angle de braquage à la roue i . Nous introduisons l'hypothèse suivante sur les angles de braquage.

Hypothèse 1. *Les angles de braquage des roues droite et gauche de chaque essieu sont supposés égaux, i.e., $\delta_1 = \delta_2 = \delta_f$ et $\delta_3 = \delta_4 = \delta_r$.*

Les efforts longitudinaux et latéraux sont, en général, des fonctions complexes de plusieurs paramètres (e.g., coefficient de frottement, charge verticale, taux de glissement, etc.). Dans cette thèse, nous admettons un comportement linéaire et réversible de la roue pneumatique, mais auparavant nous présenterons un modèle élasto-frictionnel général.

2.3 Modélisation du contact roue-sol

Un pneumatique est un composant viscoélastique décrivant la seule interface entre le véhicule et le sol. Dès que le véhicule se met en mouvement, l'élasticité du pneumatique génère des déformations sur la surface de contact avec le sol. Cette déformation engendre des glissements dans les deux directions longitudinale et latérale.

2.3.1 Glissement transversal

Le glissement transversal permet de décrire la différence entre le comportement réel et théorique de la roue i . On peut le caractériser par l'angle de dérive. Celui-ci relie le vecteur vitesse

réel du centre de la roue v_i et son vecteur vitesse longitudinal v_{xi} (voir Figure 2.1). Il s'exprime à travers la relation suivante :

$$\beta_i = \arctan\left(\frac{v_{yi}}{|v_{xi}|}\right), \quad (2.3)$$

avec v_{xi} la vitesse longitudinale de la roue i et v_{yi} sa vitesse latérale. Ces vitesses peuvent être déduites à partir des vitesses du véhicule :

$$v_y = V_y + aV_\psi, \quad v_x = V_x - dV_\psi, \quad i = 1, \quad (2.4a)$$

$$v_y = V_y + aV_\psi, \quad v_x = V_x + dV_\psi, \quad i = 2, \quad (2.4b)$$

$$v_y = V_y - bV_\psi, \quad v_x = V_x - dV_\psi, \quad i = 3, \quad (2.4c)$$

$$v_y = V_y - bV_\psi, \quad v_x = V_x + dV_\psi, \quad i = 4, \quad (2.4d)$$

2.3.2 Glissement longitudinal

Le glissement longitudinal σ_i est un paramètre qui décrit la différence entre la vitesse longitudinale du centre de rotation de la roue i notée v_{xi} et la vitesse théorique sous la condition de RSG. Le taux de glissement longitudinal s'exprime de la façon suivante :

$$\sigma_i = \begin{cases} \frac{R_i\omega_i - v_{xi}}{|R_i\omega_i|}, & \text{Phase d'accélération} \\ \frac{R_i\omega_i - v_{xi}}{|v_{xi}|}, & \text{Phase de freinage} \end{cases} \quad (2.5)$$

avec R_i est le rayon effectif de la roue i et ω_i sa vitesse angulaire (voir Figure 2.1), les vitesses dans le repère roue sont données par les équations (2.4). Tel qu'il est défini, le glissement longitudinal est compris dans l'intervalle $\sigma_i \in [-1, 1]$. La dynamique de rotation des roues s'exprime quant à elle par l'équation du moment,

$$I_{wi}\dot{\omega}_i = -R_iF_{xi} - C_{fi} + C_{mi}, \quad i \in \{1, 2, 3, 4\}, \quad (2.6)$$

où I_{wi} est le moment d'inertie de la roue i , et C_{mi} et C_{fi} sont respectivement le couple moteur et le couple de freinage au niveau de la roue i .

2.3.3 Modèle linéaire du pneumatique

Quand la force de traction/freinage est plus faible que la force limite de frottement, on peut supposer que le glissement longitudinal σ_i reste faible (i.e. ne dépasse pas 10%). Dans ces cas, la relation liant la force longitudinale du pneumatique F_{xi} et σ_i est linéaire. Dans les conditions plus agressives, cette relation devient non-linéaire, le pneumatique tend dans ce cas à se bloquer (lors d'un freinage) ou à patiner (lors d'une traction). De la même façon, le glissement latéral β_i des pneumatiques reste faible durant une conduite normale (en général inférieur à 10°). La force latérale F_{yi} varie proportionnellement aux faibles valeurs de l'angle de dérive β_i (cf. Figure 2.2). Dans des conditions de conduite agressives ou en cas de mauvaise adhérence de la chaussée, la force latérale atteint une valeur maximale rendant la relation non-linéaire.

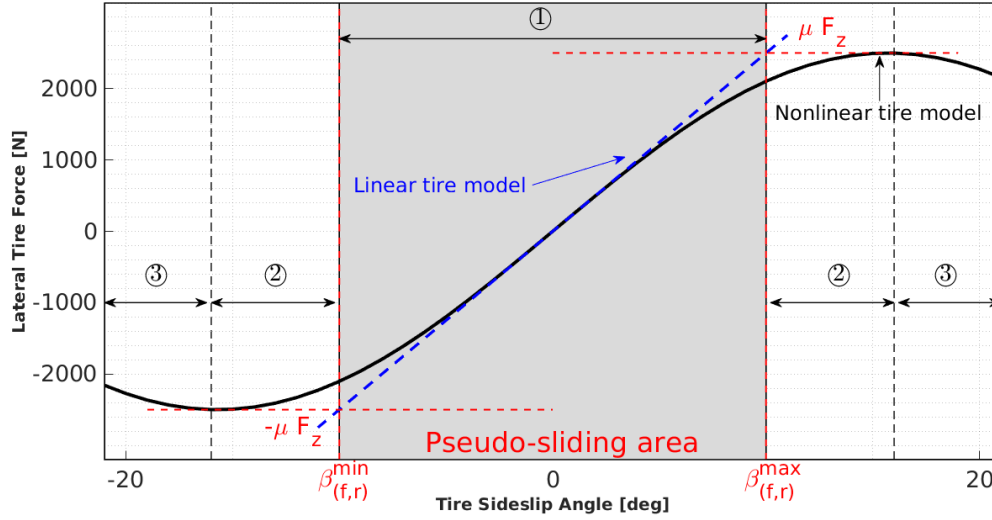


Figure 2.2. Courbe représentative de l'effort latéral du pneumatique en fonction de l'angle de dérive, montrant la zone d'adhérence et la zone de dérapage où le robot devient incontrôlable.

Dans les faits, les courbes décrivant les comportements de contact roue/sol ne sont en général pas uniques car elles dépendent de plusieurs paramètres comme la charge verticale, le coefficient de frottement, le revêtement de sol, les conditions météorologiques, etc. En revanche, l'analyse d'une façon générale de ces courbes conduit à distinguer principalement trois zones qui traduisent un comportement différent (cf. Figure 2.2) :

- Dans la première zone, la courbe est linéaire croissante ; c'est le domaine d'adhérence du pneumatique. Plus on augmente l'angle de dérive plus l'effort transversal croît linéairement par rapport à cet angle.
- Dans la deuxième zone, la courbe est non-linéaire et finit par atteindre un maximum. Plus on augmente l'angle de dérive moins l'effort transversal croît. Une partie de l'aire de contact roue/sol glisse. C'est une zone de transition.
- Dans la troisième zone, l'effort transversal diminue au fur et à mesure que l'angle de dérive croît (glissement du pneumatique). Supposons que ce phénomène survienne au niveau des pneumatiques avant d'un véhicule. Cela se traduit par une perte de directivité et le véhicule ne parvient pas à se remettre sur le chemin désiré par le conducteur. Si à ce moment le conducteur augmente le braquage des roues en accroissant l'angle au volant, il provoque une augmentation de l'angle de dérive dont la conséquence immédiate est une baisse de l'effort transversal généré par le pneumatique.

Sous l'hypothèse de pseudo-glissement (première zone), il y a une proportionnalité entre l'effort latéral et l'angle de dérive (cf. Figure 2.2). D'une façon analogue, il y a aussi une proportionnalité entre l'effort et le glissement longitudinal. Dans cet intervalle, le modèle linéaire du pneumatique se résume à :

$$F_{xi} = C_{\sigma_i} \sigma_i, \quad (2.7a)$$

$$F_{yi} = C_i \beta_i, \quad (2.7b)$$

où C_{σ_i} et C_i sont respectivement les rigidités longitudinales [N] et latérales [$N \cdot rad^{-1}$] de la roue i du véhicule. Elles dépendent principalement de l'adhérence de la chaussée, des propriétés rhéologiques du pneu et de la charge verticale sur le pneumatique.

Au-delà de la zone linéaire, la courbe représentant l'effort latéral F_{yi} en fonction de l'angle de dérive β_i atteint un maximum puis décroît (zones 2 & 3). Ce maximum est la force critique de la zone pseudo-glisement qui dépend principalement de coefficient de frottement μ_i et la force verticale F_{zi} . On met en place l'hypothèse suivante sur le coefficient de frottement.

Hypothèse 2. *Le coefficient de frottement est supposé connu et identique pour toutes les roues, i.e., $\mu_i = \mu$, $\forall i$ et constant sur un horizon de temps fini. Plusieurs techniques d'estimation en ligne de ce coefficient ont été développées dans la littérature, (e.g., [Hahn et al. \(2002\)](#), [Yamazaki et al. \(1997\)](#), [Wang et Junmin \(2013\)](#), [Liu et Peng \(1996\)](#)).*

2.4 Modèle bicyclette

Le modèle bicyclette est dérivé à partir du modèle non-linéaire de véhicule à 4 roues présenté dans la section 2.2. Il est basé sur une simplification dans laquelle les deux pneus avant sont regroupés ainsi que pour les deux pneus arrière, comme indiqué sur le schéma de la Figure 2.3. Le modèle à quatre roues (2.1) devient,

$$m\dot{V}_x = mV_yV_\psi + 2f_{xf} + 2f_{xr}, \quad (2.8a)$$

$$m\dot{V}_y = -mV_xV_\psi + 2f_{yf} + 2f_{yr}, \quad (2.8b)$$

$$I_z\dot{V}_\psi = 2af_{yf} - 2bf_{yr}, \quad (2.8c)$$

avec $f_{x(f,r)}$ et $f_{y(f,r)}$ sont respectivement les forces longitudinales et latérales avant et arrière dans le repère véhicule \mathcal{R}_G . Ces composantes sont exprimées de la façon suivante :

$$f_{x(f,r)} = F_{x(f,r)} \cos \delta_{(f,r)} - F_{y(f,r)} \sin \delta_{(f,r)}, \quad (2.9a)$$

$$f_{y(f,r)} = F_{x(f,r)} \sin \delta_{(f,r)} + F_{y(f,r)} \cos \delta_{(f,r)}, \quad (2.9b)$$

avec $\delta_{(f,r)}$ est l'angle de braquage avant et arrière, $F_{x(f,r)}$ et $F_{y(f,r)}$ sont respectivement les forces longitudinales et latérales avant et arrière dans le repère roue \mathcal{R}_ω .

Les équations de mouvement du véhicule dans le repère inertiel absolu \mathcal{R}_G sont,

$$\dot{X} = V_x \cos \psi - V_y \sin \psi, \quad (2.10a)$$

$$\dot{Y} = V_x \sin \psi + V_y \cos \psi, \quad (2.10b)$$

$$\dot{\psi} = V_\psi, \quad (2.10c)$$

où \dot{X} et \dot{Y} sont respectivement la vitesse longitudinale et latérale dans le repère inertiel \mathcal{R}_G , ψ est l'angle d'orientation de véhicule dans le plan de lacet.

Par ailleurs, en utilisant les équations cinématiques du véhicule et en introduisant les angles de dérive avant et arrière β_i avec $i \in \{f, r\}$ on peut écrire les relations :

$$\tan(\delta_f + \beta_f) = \frac{V_y + aV_\psi}{V_x}, \quad (2.11a)$$

$$\tan(\delta_r + \beta_r) = \frac{V_y - bV_\psi}{V_x}, \quad (2.11b)$$

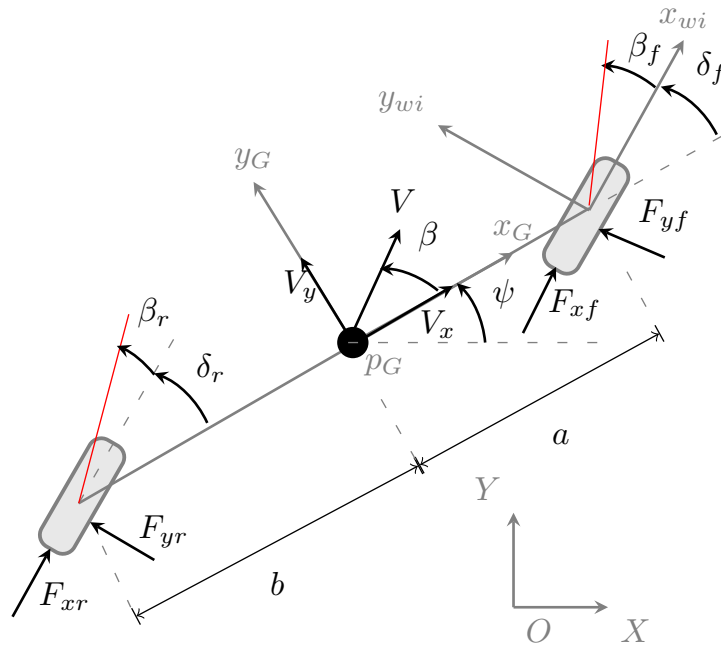


Figure 2.3. Modèle dynamique bicyclette.

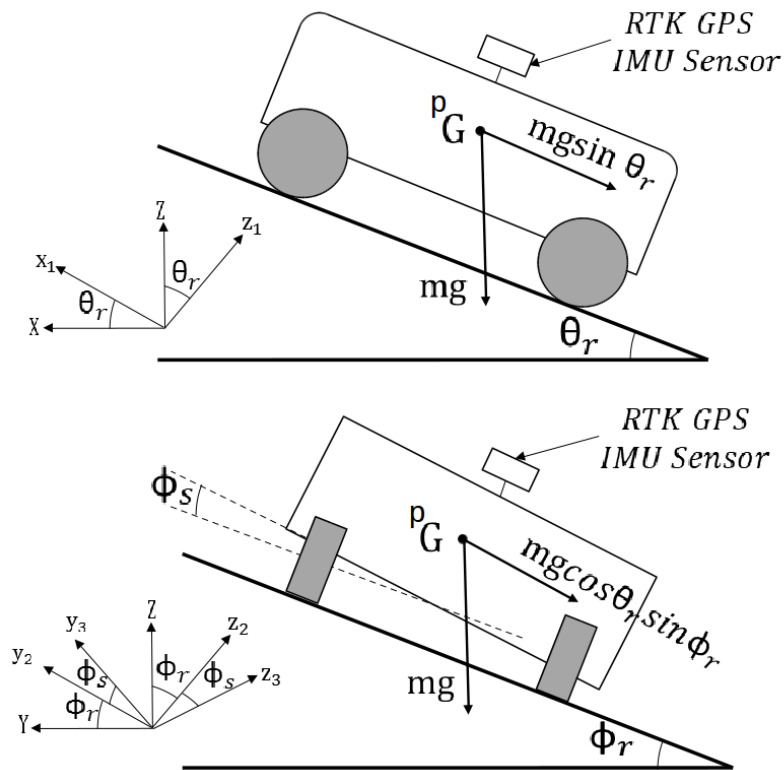


Figure 2.4. Paramètres de la géométrie locale de terrain.

En supposant que les angles de braquage et l'angle de dérive sont faibles, on peut réécrire ces deux équations comme suit,

$$\beta_f = \frac{V_y + aV_\psi}{V_x} - \delta_f, \quad (2.12a)$$

$$\beta_r = \frac{V_y - bV_\psi}{V_x} - \delta_r, \quad (2.12b)$$

En utilisant (2.12), le modèle linéaire de la force latérale en fonction du glissement donné par l'équation (2.7b) devient,

$$F_{yf} = C_f \left[\frac{V_y + aV_\psi}{V_x} - \delta_f \right], \quad (2.13a)$$

$$F_{yr} = C_r \left[\frac{V_y - bV_\psi}{V_x} - \delta_r \right], \quad (2.13b)$$

avec $F_{y(f,r)}$ sont les forces latérales des pneumatiques avant et arrière $[N]$, et $C_{(f,r)}$ sont les rigidités de dérive latérales avant et arrière $[N.rad^{-1}]$.

Le robot mobile peut parfois se déplacer sur un terrain irrégulier (e.g., pente, bosse, etc.), ce qui peut générer des efforts de gravité à prendre aussi en compte dans la dynamique latérale. Une représentation exhaustive de ces forces selon la géométrie locale du terrain est représentée sur la Figure 2.4.

Pour simplifier le modèle bicyclette (2.8) et le rendre linéaire, on met en place l'hypothèse suivante sur la vitesse longitudinale.

Hypothèse 3. *On suppose que la vitesse longitudinale V_x est constante ($\dot{V}_x = 0$).*

En prenant en compte la géométrie locale du terrain et en utilisant les équations (2.9) et le modèle linéaire du pneumatique latéral (2.13), les équations du modèle bicyclette (2.8) peuvent être réécrites dans le plan de lacet sous la forme suivante,

$$\dot{V}_y = -V_x V_\psi - 2 \frac{V_y}{mV_x} (C_f + C_r) - 2 \frac{V_\psi}{mV_x} (aC_f - bC_r) + \frac{2C_f}{m} \delta_f - \frac{2C_r}{m} \delta_r + g \cos \phi_r \sin \theta_r, \quad (2.14a)$$

$$\dot{V}_\psi = -2 \frac{V_y}{I_z V_x} (aC_f - bC_r) - 2 \frac{V_\psi}{I_z V_x} (a^2 C_f + b^2 C_r) + \frac{2aC_f}{I_z} \delta_f + \frac{2bC_r}{I_z} \delta_r, \quad (2.14b)$$

avec θ_r et ϕ_r sont les angles de la géométrie locale du terrain, angle de pente (road grade angle) et angle de dévers (road bank angle), respectivement. Une estimation précise de ces angles selon les irrégularités de l'environnement est requise pour compenser les effets de la gravité dans la dynamique latérale.

2.5 Modélisation cinématique opérationnelle du robot

Nous avons présenté dans la section précédente le modèle dynamique bicyclette développé dans le repère véhicule. Afin de paramétrer la tâche de suivi de chemin, il faut modéliser le mouvement du véhicule par un modèle cinématique. On considère le modèle 2D dans le plan d'évolution du véhicule décrit dans la Figure (2.5).

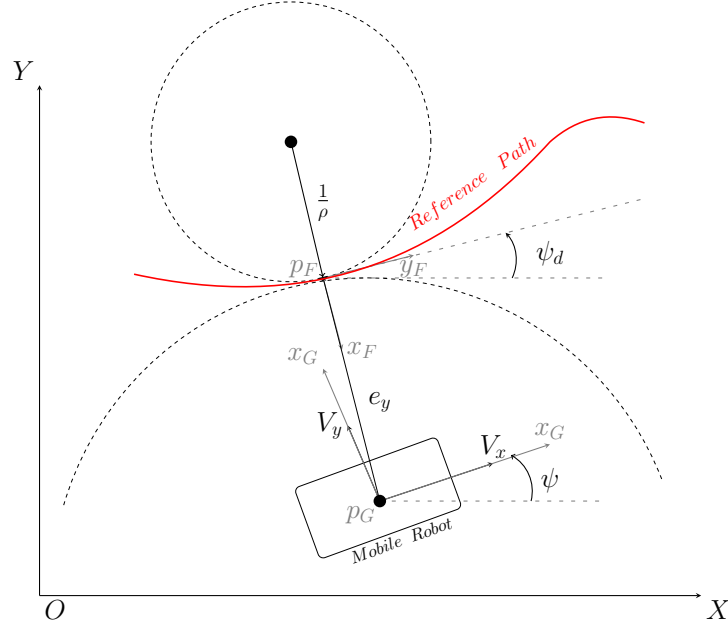


Figure 2.5. Évolution du véhicule par rapport au chemin de référence.

les déviations latérale et angulaire sont présentées dans la Figure 2.5 par rapport au chemin désiré. Tout d'abord, on définit le point le plus proche du centre de gravité du véhicule p_G au chemin de référence. Ce point définira l'origine du repère de Serret-Frénet \mathcal{R}_F (noté p_F). Le cap de référence ψ_d est défini par l'angle de rotation entre le repère de Serret-Frénet \mathcal{R}_F et celui lié au terrain \mathcal{R}_i (voir Figure 2.5). Ainsi la déviation angulaire représente la différence entre la cap de référence ψ_d et la direction du véhicule.

$$e_\psi = \psi - \psi_d, \quad (2.15)$$

Nous exprimons la dérivée de la déviation angulaire \dot{e}_ψ qui dépendra de la dynamique du véhicule V_ψ et la variation de la direction de référence $\dot{\psi}_d$.

$$\dot{e}_\psi = V_\psi - \dot{\psi}_d, \quad (2.16)$$

Par ailleurs, on définit la déviation latérale par la distance algébrique entre le centre de gravité du véhicule p_G et le point le plus proche au chemin désiré p_F . La dynamique de cette déviation est écrite en fonction de la dynamique du véhicule et la déviation angulaire donnée par l'équation (2.17a). Pour simplifier l'expression (2.17a) de la variation de la déviation latérale, on

suppose que l'écart angulaire est faible, donc on pourrait l'approximer par l'équation (2.17b).

$$\dot{e}_y = V_x \sin(e_\psi) + V_y \cos(e_\psi), \quad (2.17a)$$

$$\approx V_x e_\psi + V_y, \quad (2.17b)$$

La variation de la direction de référence $\dot{\psi}_d$ peut être exprimé par l'équation (2.18a). Afin de simplifier cette expression, on suppose que la déviation latérale e_y est faible devant le rayon de courbure $\frac{1}{\rho}$ ($\rho e_y \ll 1$). On peut donc approximer $\dot{\psi}_d$ par l'équation (2.18b).

$$\dot{\psi}_d = \frac{V_x \cos(e_\psi) - V_y \sin(e_\psi)}{\frac{1}{\rho} + e_y}, \quad (2.18a)$$

$$\approx \rho V_x - \rho^2 V_x e_y, \quad (2.18b)$$

Ainsi on déduit l'expression de \dot{e}_ψ

$$\dot{e}_\psi = V_\psi - \rho V_x + \rho^2 V_x e_y, \quad (2.19)$$

On résume donc l'ensemble des équations du modèle cinématique par les équations suivantes :

$$\dot{e}_y = V_x e_\psi + V_y, \quad (2.20a)$$

$$\dot{e}_\psi = V_\psi - \rho V_x + \rho^2 V_x e_y, \quad (2.20b)$$

2.6 Modèle final du véhicule

Pour développer nos lois de commande et nos différents autres algorithmes, nous utiliserons le modèle dynamique bicyclette (2.14) combiné aux équations cinématiques (2.20). Ce modèle dans le plan de lacet est donné par les équations suivantes,

$$\dot{V}_y = -V_x V_\psi - 2 \frac{V_y}{m V_x} (C_f + C_r) - 2 \frac{V_\psi}{m V_x} (a C_f - b C_r) + \frac{2 C_f}{m} \delta_f - \frac{2 C_r}{m} \delta_r + g \cos \phi_r \sin \theta_r, \quad (2.21a)$$

$$\dot{V}_\psi = -2 \frac{V_y}{I_z V_x} (a C_f - b C_r) - 2 \frac{V_\psi}{I_z V_x} (a^2 C_f + b^2 C_r) + \frac{2 a C_f}{I_z} \delta_f + \frac{2 b C_r}{I_z} \delta_r, \quad (2.21b)$$

$$\dot{e}_y = V_x e_\psi + V_y, \quad (2.21c)$$

$$\dot{e}_\psi = V_\psi - \rho V_x + \rho^2 V_x e_y, \quad (2.21d)$$

Ainsi, on peut mettre les équations (2.21) sous la forme de la représentation d'état à temps continu donnée comme suit :

$$\begin{cases} \dot{\xi} = A\xi + Bu + S \\ y = C\xi \end{cases} \quad (2.22)$$

avec $\xi = [V_y, V_\psi, e_y, e_\psi]^T$ est le vecteur d'état, $u = [\delta_f, \delta_r]^T$ vecteur commande et $y = [V_\psi, e_y, e_\psi]^T$ vecteur sortie (e.g. les capteurs extéroceptifs utilisés permettent de mesurer V_ψ , e_y et e_ψ). Le vecteur S est une composante de dérive qui ne dépend ni de l'état ni de la commande.

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 \\ 1 & 0 & 0 & V_x \\ 0 & 1 & \rho^2 V_x & 0 \end{bmatrix} = \begin{bmatrix} -2\frac{C_f + C_r}{mV_x} & -2\frac{aC_f - aC_r}{mV_x} - V_x & 0 & 0 \\ -2\frac{aC_f - aC_r}{I_z V_x} & -2\frac{a^2 C_f + a^2 C_r}{I_z V_x} & 0 & 0 \\ 1 & 0 & 0 & V_x \\ 0 & 1 & \rho^2 V_x & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{2C_f}{m} & \frac{-2C_r}{m} \\ \frac{2aC_f}{I_z} & \frac{2aC_r}{I_z} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad S = \begin{bmatrix} g \cos \phi_r \sin \theta_r \\ 0 \\ -\rho V_x \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

On calcule les états stationnaires ξ_{ss} et u_{ss} (i.e., points d'équilibre) du système linéaire (2.22) afin de compenser la dérive S . On obtient le modèle suivant :

$$\begin{cases} \dot{\xi} = A(\xi - \xi_{ss}) + B(u - u_{ss}) \\ y = C(\xi - \xi_{ss}) \end{cases} \quad (2.23)$$

Les composantes stationnaires ξ_{ss} et u_{ss} dépendent principalement de paramètres décrivant les caractéristiques géométriques et physiques du terrain sur lequel le robot se déplace. Ces composantes sont exprimées comme suit,

$$u_{ss} = \begin{bmatrix} \frac{\rho V_x (a_{11} a_{22} - a_{21} a_{12}) - a_{21} g \cos \phi_r \sin \theta_r}{a_{21} (b_{11} - b_{12}) - a_{11} (b_{21} - b_{22})} \\ \frac{-\rho V_x (a_{11} a_{22} - a_{21} a_{12}) + a_{21} g \cos \phi_r \sin \theta_r}{a_{21} (b_{11} - b_{12}) - a_{11} (b_{21} - b_{22})} \end{bmatrix}$$

$$\xi_{ss} = \begin{bmatrix} \frac{\rho V_x [a_{22} (b_{11} - b_{12}) - a_{12} (b_{21} - b_{22})] - g \cos \phi_r \sin \theta_r (b_{21} - b_{22})}{a_{11} (b_{21} - b_{22}) - a_{21} (b_{11} - b_{12})} \\ \rho V_x \\ 0 \\ \frac{-\rho V_x [a_{22} (b_{11} - b_{12}) - a_{12} (b_{21} - b_{22})] + g \cos \phi_r \sin \theta_r (b_{21} - b_{22})}{V_x (a_{11} (b_{21} - b_{22}) - a_{21} (b_{11} - b_{12}))} \end{bmatrix}$$

3 Synthèse de loi de commande pour le suivi de chemin

3.1 État de l'art sur la commande des véhicules autonomes

L'architecture de contrôle d'un robot autonome peut être décomposée en trois modules : la perception de l'environnement, la planification de chemins et le contrôle du véhicule. Le contrôle du véhicule consiste à manœuvrer le véhicule en utilisant sa partie opérative via une commande pour réaliser la tâche du suivi de trajectoire ou chemin désiré.

Plusieurs stratégies de commande sont utilisées par la communauté robotique concernant le problème du suivi de chemin ou trajectoire, allant de commandes simples (P, PI, PID) aux commandes avancées (e.g., adaptative, optimale, prédictive, mode glissants, etc.). L'objectif de ces lois de commande est de positionner le robot le plus précisément possible par rapport à un chemin ou trajectoire nominal à une vitesse demandée en fonction des informations issues de la localisation. Ces lois de commande calculent les bonnes entrées afin de piloter correctement les actionneurs de direction et de vitesse de rotation des roues. La plupart des stratégies de la tâche de suivi de chemin ou trajectoire sont basées sur un modèle cinématique. Par exemple, des techniques de la commande par retour d'état basées sur des modèles cinématiques sont développées dans [De Luca et al. \(1998\)](#) pour les robots mobiles non-holonomes qui se déplacent dans un terrain plat et horizontal. Afin de pouvoir faire face aux incertitudes du modèle ainsi que les perturbations venant du sol, des approches utilisant la commande par mode glissant sont abordées dans [Yang et Kim \(1999\)](#) et [Guldner et Utkin \(1994\)](#). Une autre commande adaptative non linéaire de suivi de chemin basée sur la théorie du contrôle en mode glissant a été développée dans [Wang et al. \(2016\)](#). Ce contrôleur calcule en temps réel et indépendamment les angles de direction des roues d'avant et d'arrière. D'autres techniques de suivi de chemin ou trajectoire proposent une linéarisation exacte du modèle cinématique et une conversion sous une forme dite chaînée pour la partie qui n'est pas linéarisable du modèle (e.g., [Morin et Samson \(2006\)](#), [Sordalen \(1993\)](#)). Ces contrôleurs à base cinématique développés pour des robots type unicycle ou de type voiture restent les plus utilisés dans le cas où les glissements et la dynamique latérale du véhicule sont négligeables. Dans ce cadre, une commande adaptative de suivi de chemin basée sur un modèle cinématique étendu élaborée dans un repère de Frénet (i.e., modèle cinématique classique intégrant les angles de glissement) a été synthétisée dans [Lenain et al. \(2003\)](#). Cette commande permet au véhicule de suivre un chemin de référence en contrôlant l'angle de direction, en revanche, son efficacité est soumise à la bonne estimation des angles de dérive. Cependant, ces approches restent robustes et fiables pour certains types de chemin nominal en régime stationnaire (e.g., chemin rectiligne, chemin à grand rayon de courbure, etc.), mais cette robustesse peut être considérablement amoindrie quand le glissement et la dynamique du véhicule deviennent extrêmement importants, notamment à grandes vitesses et/ou sur des sols non préparés. Dans ce contexte, des commandes basées sur des approches prédictives et des modèles dynamiques doivent être mises en place pour commander les véhicules tout terrain en milieu ouvert. Par exemple, une loi de commande adaptative et prédictive de suivi de chemin en prenant en compte le glissement a été développée dans [Cariou et al. \(2009\)](#) et [Lenain et al. \(2007\)](#), où la partie prédictive est principalement liée à l'anticipation du changement de la courbure du chemin nominal. Toutefois, afin de prendre en compte l'inertie du robot et les conditions d'adhérence au sol, plusieurs approches de suivi de chemin intègrent la dynamique latérale (e.g., [Mammar et al. \(2010\)](#), [Eaton et al. \(2009\)](#)) et longitudinale (e.g., [Shakouri et al. \(2011\)](#), [Nouveliere et al. \(2007\)](#)). Citons, à titre d'exemple, une stratégie de commande prédictive

généralisée non linéaire à temps continu (NCGPC) qui a été développée dans [Krid *et al.* \(2017\)](#). Cette commande est basée sur la dynamique latérale du véhicule destinée à assurer la tâche de suivi de trajectoire dans le plan de lacet. Aussi, des commandes de stabilisation du véhicule en suivi de chemin ou trajectoire basées sur sa dynamique ont été abordées dans [Falcone *et al.* \(2007\)](#), [Marino *et al.* \(2011\)](#) et [Zhao *et al.* \(2011\)](#). Dans ces cas la robustesse de la commande et sa capacité de rejet des perturbations résident dans la bonne estimation des paramètres liés au contact roue/sol. Pour résoudre ce problème, plusieurs travaux ont traité les problèmes d'estimation des paramètres de glissement qui seront injectés par la suite dans le bloc de la commande (e.g., [Boada *et al.* \(2016\)](#), [Bouton *et al.* \(2007\)](#)).

Dans la suite, nous allons synthétiser une commande LQR (Linear Quadratic Regulator) pour le suivi de chemin. Cette commande est synthétisée en combinant les modèles dynamiques (modèle bicyclette (2.23) et le modèle linéaire des pneumatiques (2.13)) et cinématiques du véhicule (2.20). Elle sera mise en place par la suite pour valider nos observateurs des variables physiques et géométriques du sol qui feront l'objet du chapitre 3.

3.2 Synthèse de la loi de commande de suivi de chemin

Dans cette section, nous allons mettre en place une commande linéaire quadratique (LQR "Linear Quadratic Regulator") qui a comme objectif le suivi d'un chemin de référence. Pour synthétiser cette commande, on se base sur le modèle donné par la représentation d'état (2.23).

Généralement pour un système linéaire commandable, il existe une infinité de contrôleurs possibles et il est intéressant de disposer de critères pour le choix de ceux-ci. L'approche par placement de pôles est déjà plus intuitive puisque le choix des pôles est directement lié au taux de convergence des solutions vers zéro et à la forme du transitoire. L'approche de type Lyapunov, très générale mais assez indirecte, ne donne pas beaucoup d'indices pour choisir la commande. Nous allons utiliser la commande optimale LQR qui minimise à la fois l'état et l'entrée du modèle (2.23). Le principe consiste à déterminer la commande qui va minimiser un critère quadratique le long des solutions du système. Plus précisément, on se donne le critère suivant :

$$J = \lim_{t \rightarrow +\infty} \int_0^t \left[(\xi - \xi_{ss})^T Q (\xi - \xi_{ss}) + (u - u_{ss})^T R (u - u_{ss}) \right] dt, \quad (2.24)$$

où $Q \in \mathbb{R}^{4 \times 4}$ et $R \in \mathbb{R}^{2 \times 2}$ sont les matrices de pondération symétriques et réelles de l'état et de la commande, elles sont respectivement semi-définie positive¹ et définie-positive².

Nous cherchons alors la commande permettant de minimiser le critère (2.24). La solution à ce problème est donnée sous la forme d'une commande par retour d'état $u = -K(\xi - \xi_{ss})$, avec K directement liée aux matrices Q et R . La matrice gain K joue un rôle primordiale dans le maintien de la stabilité et la convergence de la commande vers la solution optimale. Elle est choisie de telle façon à imposer à la matrice dynamique du système asservi $A - BK$ des valeurs propres à parties réelles strictement négatives, i.e. matrice de Hurwitz. Afin de pouvoir envisager la synthèse d'une commande par retour d'état, il est nécessaire que le système (2.23) soit commandable. On rappelle le critère de Kalman utilisé pour vérifier la commandabilité d'un système linéaire.

1. Une matrice $M \in \mathcal{M}_n(\mathbb{R})$ (symétrique) est dite semi-définie positive si $\mathbf{x}^T M \mathbf{x} \geq 0$ pour tout \mathbf{x} .

2. Une matrice $M \in \mathcal{M}_n(\mathbb{R})$ (symétrique) est dite définie positive si $\mathbf{x}^T M \mathbf{x} > 0$ pour tout \mathbf{x} non nul.

Considérons un système linéaire à temps continu de la forme :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (\star)$$

avec $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ et $y(t) \in \mathbb{R}^r$. Les matrices d'état sont $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{r \times n}$ et $D \in \mathbb{R}^{r \times m}$. La commandabilité d'un tel système peut être étudiée à l'aide du critère de Kalman donné ci-dessous.

Le système (\star) est commandable si et seulement si sa matrice de commandabilité Q_S est de rang plein. La matrice de commandabilité est la matrice $Q_S \in \mathbb{R}^{n \times (n \times m)}$ définie de la façon suivante :

$$Q_S = \begin{bmatrix} B & AB & \dots & A^{n-1}B \end{bmatrix}$$

De plus, si cette propriété est satisfaite, pour tout couple (x_0, x_f) et tout $T > 0$, on peut trouver une commande u amenant x_0 à x_f sur l'intervalle $[0, T]$ (i.e., T peut être choisi arbitrairement petit).

Hypothèse 4. Les matrices A et B dépendent de la vitesse longitudinale V_x et de la courbure ρ . On suppose qu'elles sont invariantes sur un horizon de temps fini (i.e., le système (2.23) est linéaire à paramètres invariants (LPI) sur un horizon de temps fini).

Hypothèse 5. La paire (A, B) est commandable. Cette propriété est vérifiée numériquement en ligne avant le calcul de la loi de commande.

Les hypothèses ci-dessus étant admises, l'objectif de ce type d'asservissement LQR est d'amener, en régime permanent, le vecteur de sortie, $y(t)$, à une valeur de consigne $y_d(t)$ tout en garantissant la stabilité interne du système asservi. Pour cela, on pourra envisager une loi de commande optimale qui minimise le critère quadratique (2.24) de la forme suivante :

$$u = Ny_d(t) - K(\hat{\xi} - \xi_{ss}) + u_{ss}, \quad (2.25)$$

avec N est la matrice gain du pré-filtre, $\hat{\xi} = [\hat{V}_y, \hat{V}_\psi, \hat{e}_y, \hat{e}_\psi]^T$ est l'état estimé par le filtre de Kalman-Bucy présenté dans la section 3.3. ξ_{ss} and u_{ss} sont l'état et la commande stationnaires (i.e., points d'équilibre du système linéaire avec dérive (2.22)).

Précisons que les matrices N et K sont les paramètres du contrôleur. La synthèse du contrôleur consiste donc à déterminer judicieusement ces deux matrices. La matrice gain K est calculée comme suit :

$$K = R^{-1}B^T P_r^*, \quad (2.26)$$

où P_r^* est la solution algébrique symétrique de l'équation de Riccati suivante :

$$P_r A + A^T P_r + Q - P_r B R^{-1} B^T P_r = 0, \quad (2.27)$$

La matrice gain K est directement liée aux matrices de pondération Q et R . Ces dernières doivent être réglées convenablement. Elles sont considérées invariantes et constantes tout au long du temps de calcul. Après plusieurs tests, les matrices diagonales retenues sont :

$$Q = \begin{bmatrix} q_{V_y} & 0 & 0 & 0 \\ 0 & q_{V_\psi} & 0 & 0 \\ 0 & 0 & q_{e_y} & 0 \\ 0 & 0 & 0 & q_{e_\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}$$

$$R = \begin{bmatrix} r_{\delta_f} & 0 \\ 0 & r_{\delta_r} \end{bmatrix} = \begin{bmatrix} 2 \cdot 10^4 & 0 \\ 0 & 2 \cdot 10^4 \end{bmatrix}$$

Après avoir déterminé K , on détermine la matrice pré-filtre N de façon à régler la précision du système asservi. Or celle-ci dépend du gain statique de la matrice de transfert en boucle fermée qui doit être égal à 1 (voir démonstration ci-dessous). La matrice de transfert du système bouclé, dans le cas des systèmes MIMO³, s'écrit de la façon suivante :

$$F(s) = C [sI_{4 \times 4} - [A - BK]]^{-1} BN, \quad (2.28)$$

où I est la matrice identité, s est la variable complexe de Laplace, $F(s) \in \mathbb{C}^{3 \times 2}$ est la matrice de transfert du système bouclé.

Nous démontrons ci-dessous pourquoi nous choisissons de faire en sorte que $F(0) = 1$.

Preuve : Supposons que la consigne $y_d(t)$ soit un échelon d'amplitude α .

La transformée de Laplace de la consigne s'écrit $Y_d(s) = \mathcal{L}\{y_d(t)\} = \frac{\alpha}{s}$

On en déduit que $Y(s) = F(s) \frac{\alpha}{s}$. L'erreur est donc $E(s) = \mathcal{L}\{\varepsilon(t)\} = Y_d(s) - Y(s) = \frac{\alpha}{s}(1 - F(s))$

La matrice K ayant été choisie de telle façon que le système asservi soit stable, on peut appliquer le théorème de la valeur finale afin de déterminer la valeur de l'erreur, en régime permanent :

$$\lim_{t \rightarrow \infty} \varepsilon(t) = \lim_{s \rightarrow 0} sE(s) = \alpha(1 - F(0))$$

Par conséquent afin que l'erreur soit nulle en régime permanent, on choisira N tel que $F(0) = 1$.

D'après la matrice de transfert (2.28) et l'expression de $F(0) = 1$, on en déduit :

$$N = - [C [A - BK]^{-1} B]^{-1} \quad (2.29)$$

3.3 Mise en place d'un observateur d'état

Le problème d'optimisation LQR que nous avons présenté dans la section précédente nécessite, à chaque instant, la connaissance du vecteur d'état. Cependant, il n'est pas toujours possible de mesurer directement tout le vecteur d'état, soit parce que nous ne disposons pas d'un nombre de capteurs suffisant soit parce que nous ne disposons pas du type de capteur adéquat. Les capteurs embarqués permettent de mesurer V_ψ , e_y et e_ψ (cf. Figure 1.4(b)). Afin d'estimer la

vitesse latérale V_y et compléter le vecteur d'état nécessaire au calcul de la commande de suivi de chemin, un observateur de Kalman-Bucy linéaire est conçu dans cette section.

Précisons que pour qu'une telle approche soit possible, le système physique doit être observable. On rappelle, ci-dessous, le critère d'observabilité pour les systèmes linéaires à temps continu,

Le système (\star) est observable si et seulement si sa matrice d'observabilité est de rang n . La matrice d'observabilité est la matrice $Q_B \in \mathbb{R}^{n \times (n \times r)}$ définie de la façon suivante :

$$Q_B = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

De plus, si cette propriété est satisfaite, on pourrait reconstruire le vecteur d'état $x(t)$ à partir de la connaissance de la mesure y et de l'entrée de commande u sur un intervalle de temps fini $[t - \tau, t]$ avec $\tau > 0$.

Hypothèse 6. *La paire (A, C) est observable. Cette propriété est vérifiée numériquement en ligne avant la synthèse de l'observateur de l'état.*

Le modèle dynamique est donné par l'équation (2.23) auquel on ajoute du bruit sur la dynamique de l'état ainsi que sur la mesure :

$$\begin{cases} \dot{\xi} = A(\xi - \xi_{ss}) + B(u - u_{ss}) + \zeta \\ y = C(\xi - \xi_{ss}) + \gamma \end{cases} \quad (2.30)$$

Les variables aléatoires ζ et γ , représentant les bruits sur le modèle et sur les mesures respectivement, sont supposées des variables indépendantes et identiquement distribuées suivant une loi normale centrée (i.e., $\zeta \sim \mathcal{N}(0, \Gamma_\zeta)$ et $\gamma \sim \mathcal{N}(0, \Gamma_\gamma)$ où Γ_ζ et Γ_γ sont leurs matrices de covariance).

On est donc capable en l'absence de bruit (i.e., si ζ et γ sont nuls) de synthétiser un observateur stable de ξ , sous la forme :

$$\dot{\hat{\xi}} = A(\hat{\xi} - \xi_{ss}) + B(u - u_{ss}) + K_b(y - C(\hat{\xi} - \xi_{ss})), \quad (2.31)$$

Le terme $y - C(\hat{\xi} - \xi_{ss})$ est nommé l'innovation et K_b est le gain de Kalman-Bucy. Le filtre de Kalman-Bucy est une version du filtre de Kalman à temps continu, il alterne deux phases : la correction et la prédiction (voir par exemple [Kalman \(1960\)](#) et [Jaulin \(2015\)](#) pour plus de détails sur ce type de filtre). Les entrées de ce filtre sont y , u , A , B , C , Γ_ζ , Γ_γ , dt (période de discrétisation). Ses sorties sont le gain Kalman-Bucy K_b , l'état estimé $\hat{\xi}$ et la matrice de covariance associée Γ_t . Le filtre de Kalman-Bucy complet est donné par les équations suivantes :

$$\hat{\xi} = A(\hat{\xi} - \xi_{ss}) + K_b(y - C(\hat{\xi} - \xi_{ss})) + B(u - u_{ss}), \quad (2.32a)$$

$$\dot{\Gamma}_t = A\Gamma_t + \Gamma_t A^T - K_b\Gamma_\gamma K_b^T + \frac{1}{dt^2}\Gamma_\zeta, \quad (2.32b)$$

$$K_b = \Gamma_t C^T \Gamma_\gamma^{-1}, \quad (2.32c)$$

Lors d'une implémentation du filtre de Kalman-Bucy, le réglage consiste principalement à déterminer les matrices de covariance Γ_ζ et Γ_γ . Elles sont considérées invariantes et constantes tout au long du temps de calcul. Durant ce réglage, nous avons mis plus de confiance sur le modèle et moins sur les mesures. Après plusieurs tests empiriques, les deux matrices diagonales Γ_ζ et Γ_γ retenues sont les suivantes :

$$\Gamma_\zeta = \begin{bmatrix} \sigma_{V_y}^2 & 0 & 0 & 0 \\ 0 & \sigma_{V_\psi}^2 & 0 & 0 \\ 0 & 0 & \sigma_{e_y}^2 & 0 \\ 0 & 0 & 0 & \sigma_{e_\psi}^2 \end{bmatrix} = \begin{bmatrix} 10^{-4} & 0 & 0 & 0 \\ 0 & 10^{-1} & 0 & 0 \\ 0 & 0 & 10^{-2} & 0 \\ 0 & 0 & 0 & 10^{-1} \end{bmatrix}$$

$$\Gamma_\gamma = \begin{bmatrix} \sigma_{V_\psi}^2 & 0 & 0 \\ 0 & \sigma_{e_y}^2 & 0 \\ 0 & 0 & \sigma_{e_\psi}^2 \end{bmatrix} = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10^2 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

La commande de suivi de chemin mixée avec l'observateur d'état de Kalman-Bucy peuvent être représentés par le schéma fonctionnel ci-dessous donné dans la Figure 2.6.

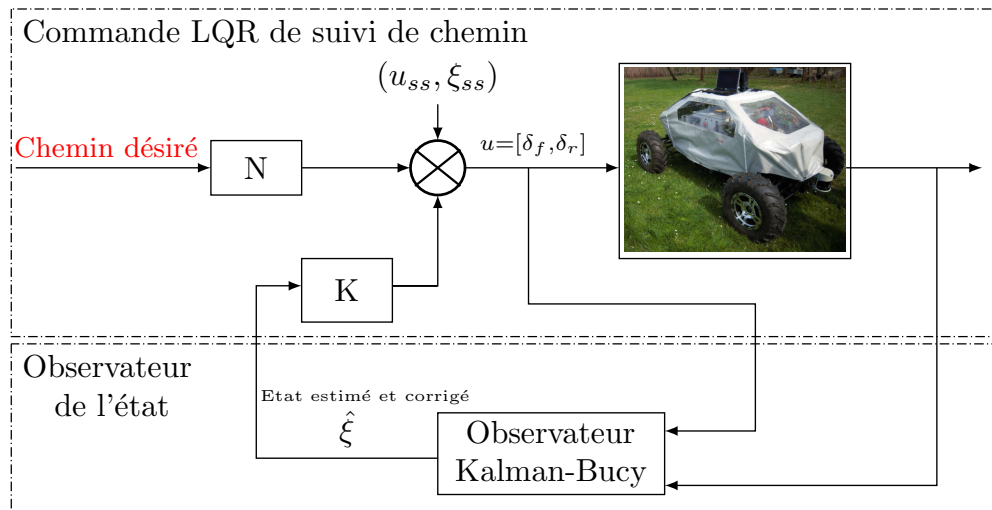


Figure 2.6. Commande linéaire de suivi de chemin et observateur Kalman-Bucy.

Dans la suite, nous passons à la validation de cette loi de commande, tout d'abord numériquement avec la maquette numérique sous ROS-GAZEBO puis expérimentalement avec le robot mobile "SPIDO".

4 Validation en simulation et expérimentale

4.1 Filtrage des mesures

Les capteurs proprioceptifs et extéroceptifs utilisés dans ce travail de thèse sont présentés dans l'annexe A. Les mesures expérimentales sont généralement bruitées, d'où l'importance de filtrer ces données avant de les utiliser. Nous avons mis en place un filtre passe-bas de Butterworth d'ordre 8 et de fréquence de coupure f_s . La fonction *butter* de la librairie python de traitement de signal (SciPy.signal) permet de fournir les coefficients du numérateur et du dénominateur de ce filtre. Afin aussi d'éliminer la distorsion de phase, nous effectuons un filtrage causal aller et retour. La fonction *filtfilt* de la même bibliothèque python, permet d'appliquer ce filtre de Butterworth en aller et retour.

4.2 Validation en simulation

Dans cette section, les performances de la commande LQR et l'observateur d'état de Kalman-Bucy sont étudiées, en premier lieu, sur la base de simulations avancées (ROS-GAZEBO) réalisées avec la maquette numérique de la Figure 1.4, puis au travers d'expérimentations dans les conditions réelles avec la plateforme "SPIDO" (cf. Figure 1.4).

Les coefficients de rigidité de dérive avant et arrière sont supposés constants dans toutes les simulations ($C_f = C_r = 15\,000\text{N.rad}^{-1}$). Ces constantes correspondent aux valeurs approximatives de contact roue/sol sur un terrain couvert d'herbe. Ainsi, le terrain est supposé plat et horizontal ($\theta_r = \phi_r = 0^\circ$).

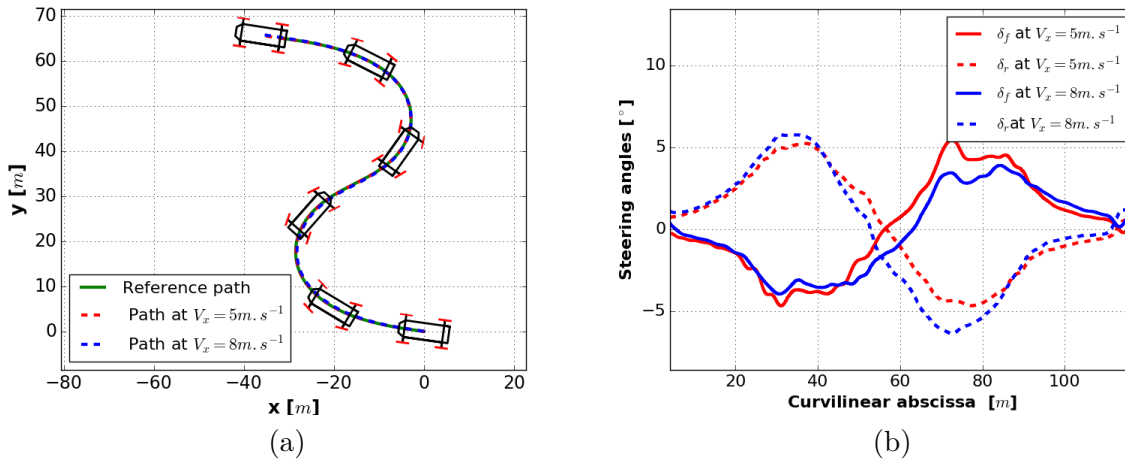


Figure 2.7. Résultats de simulation à $V_x = 5\text{ m.s}^{-1}$ et $V_x = 8\text{ m.s}^{-1}$: (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Angles de braquage avant et arrière.

Nous choisissons un chemin de référence qui comporte deux virages gauche-droite à faible rayon de courbure afin de montrer l'efficacité de la commande de suivi de chemin et d'exciter la dynamique du véhicule (cf. Figure 2.7(a)). Ce chemin de référence est connu à l'avance et donné par ses coordonnées GPS et centrale inertielle $(x, y, \psi)_d$. En simulation, les gains K et

N de la commande LQR sont calculés en ligne car les matrices A et B dépendent de la vitesse longitudinale V_x et la courbure ρ .

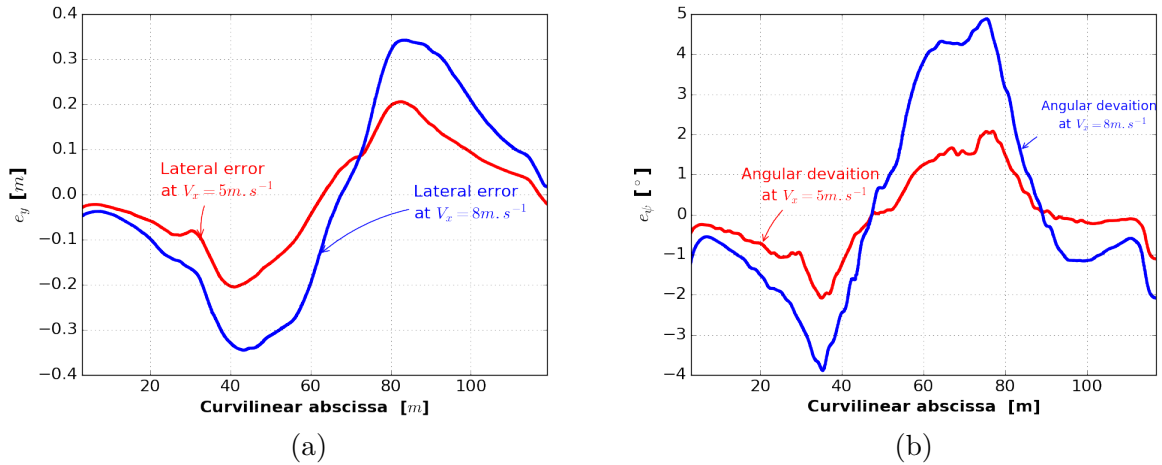


Figure 2.8. Résultats de simulation à $V_x = 5 \text{ m.s}^{-1}$ et à $V_x = 8 \text{ m.s}^{-1}$: (a) Écart latéral de suivi de chemin. (b) Écart angulaire.

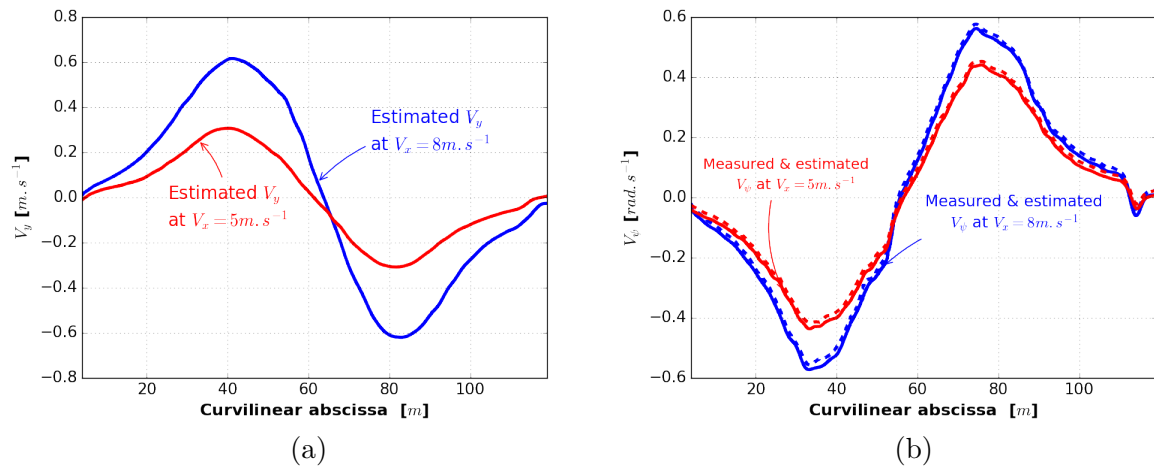


Figure 2.9. Résultats de simulation à $V_x = 5 \text{ m.s}^{-1}$ (a) et à $V_x = 8 \text{ m.s}^{-1}$: (a) Vitesse latérale estimée; (b) Vitesse de lacet mesurée (ligne continue) et observée (ligne pointillée) par le filtre de Kalman-Bucy.

Nous pouvons constater sur la Figure 2.7 que les chemins réalisés par le robot aux vitesses longitudinales $V_x = 5 \text{ m.s}^{-1}$ et $V_x = 8 \text{ m.s}^{-1}$ suivent bien le chemin désiré. Nous remarquons par ailleurs que la qualité de suivi de chemin se dégrade généralement au niveau des virages où la courbure est importante et quand la vitesse longitudinale est plus grande. Pour confirmer ce point, nous donnons les tracés des déviations latérale e_y et angulaire e_ψ , respectivement sur les Figures 2.8(a) et (b). On note que ces erreurs de suivi de chemin restent importantes même pour des faibles rayons de courbure (par exemple à l'abscisse curviligne 80m, l'erreur latérale vaut 20cm à $V_x = 5 \text{ m.s}^{-1}$ et 35cm à $V_x = 8 \text{ m.s}^{-1}$). En effet, la dégradation de la qualité de suivi de chemin à haute vitesse est due à la grande accélération latérale qui varie proportionnellement au rayon de courbure ($a_y = \rho V_x^2$). En plus, sur les même Figures 2.8(a) et (b), on peut distinguer

le virage à droite du virage à gauche en fonction du signe de la déviation.

Au niveau de la commande, les angles de direction avant et arrière sont donnés par la Figure 2.7(b) à la vitesse longitudinale $V_x = 5m.s^{-1}$ et $V_x = 8m.s^{-1}$. Nous constatons que ces deux angles sont quasi-égaux en valeur absolue à basse vitesse ($V_x = 5m.s^{-1}$) sont faibles ($\pm 6^\circ$) et ont des évolutions douces. Par ailleurs à haute vitesse ($V_x = 8m.s^{-1}$), nous remarquons que le principe de signe opposé n'est plus conservé et l'angle de braquage des roues avant devient plus important que celui des roues arrière. Par conséquent, la commande LQR permet d'assurer la tâche de suivi de chemin en calculant en ligne des commandes qui sont à la fois petites et très douces, donc moins agressives et moins perturbantes.

Afin d'évaluer le comportement de l'observateur d'état synthétisé dans la section 3.3, nous traçons dans la Figure 2.9(b) la vitesse de lacet du véhicule mesurée par la centrale inertielle (IMU Xsens) et celle estimée par le filtre de Kalman-Bucy. On constate que l'état estimé et l'état mesuré se comportent de la même façon avec une erreur d'estimation faible (de l'ordre de $10^{-2} rad.s^{-1}$). Par conséquent, le comportement de l'observateur est satisfaisant. En outre, la vitesse latérale observée par le filtre de Kalman-Bucy est donnée par la Figure 2.9(a) à $V_x = 5m.s^{-1}$ et à $V_x = 8m.s^{-1}$. Cette vitesse latérale augmente légèrement le long des grandes courbures du fait de l'accélération latérale qui augmente avec le carré de la vitesse du véhicule.

4.3 Résultats expérimentaux

Les expérimentations présentées dans cette partie sont issues d'essais réels réalisés avec le robot "SPIDO" décrit par la Figure 1.4 sur un terrain en pelouse humide. Nous décrivons en détail dans l'annexe A une présentation détaillée de la plateforme expérimentale ainsi que son informatique embarquée.

Afin de valider expérimentalement le comportement de la loi de commande et l'observateur de Kalman-Bucy, deux scénarios de suivi de chemin sont proposés dans cette section. Le premier chemin tracé sur la Figure 2.10(a) présente un virage à grande courbure et le deuxième tracé sur la Figure (2.13)(a) comporte une double manœuvre rapide gauche-droite (i.e, avec un point d'inflexion où le changement de la concavité de chemin désiré est rapide). Dans les deux cas, le chemin de référence est connu à l'avance et donné par ses coordonnées désirées $(x, y, \psi)_d$ qui ont été enregistrés manuellement et hors ligne avec la plateforme expérimentale. Les coefficients de rigidité de dérive avant et arrière sont supposées constantes dans toute la partie expérimentale ($C_f = C_r = 15\ 000N.rad^{-1}$). Ces constantes correspondent aux valeurs approximatives de contact roue-sol sur un terrain en herbe humide. Par ailleurs, le terrain est supposé plat dans ces expérimentations ($\theta_r = \phi_r = 0^\circ$).

4.3.1 Chemin en virage à droite

Pour montrer l'apport de la commande LQR et l'observateur d'état, nous choisissons dans un premier temps un chemin de référence sous la forme U avec un virage vers la droite de petit rayon de courbure (voir Figure 2.10). Toutes les expérimentations sont réalisées aux vitesses longitudinales $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$. Le chemin de référence et les chemins réalisés sont

tracés sur la Figure 2.10(a). Le résultat de suivi de chemin est conforme aux attentes puisque le robot, avec notre algorithme de suivi, suit correctement le chemin désiré⁴.

Les erreurs de suivi latérale et angulaire enregistrées sont tracées sur les Figures 2.11(a) et (b). Premièrement, il peut être remarqué que ces erreurs sont larges pendant le virage où la courbure est importante et quand la vitesse longitudinale augmente. Malgré un terrain d'essai petit et encombré (données GPS affectées par les bâtiments et les arbres autour), le suivi est toujours satisfaisant : le décalage par rapport au chemin de référence n'excède pas $1.1m$.

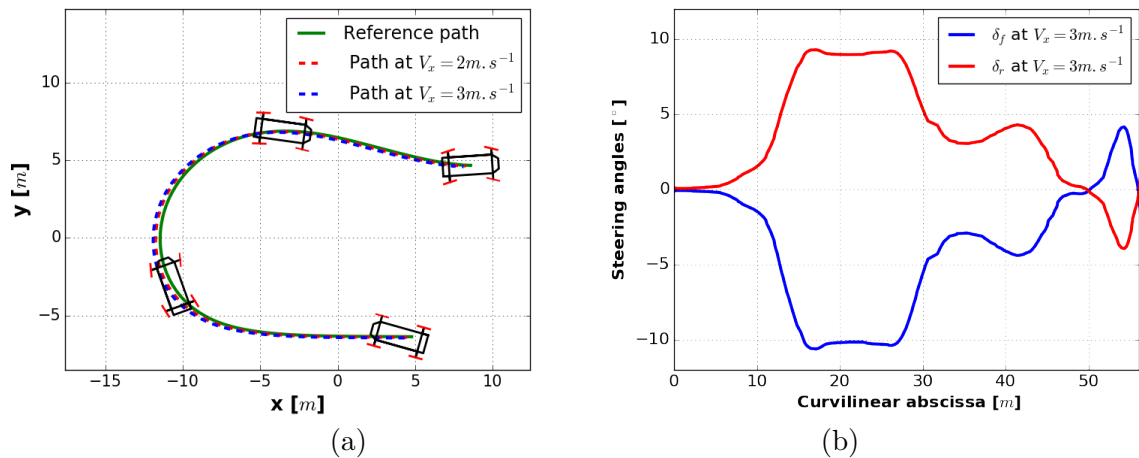


Figure 2.10. Résultats expérimentaux de chemin en virage droit : (a) Chemin de référence et chemins réalisés à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$ dans le plan (x, y) ; (b) Angles de braquage à $V_x = 3m.s^{-1}$.

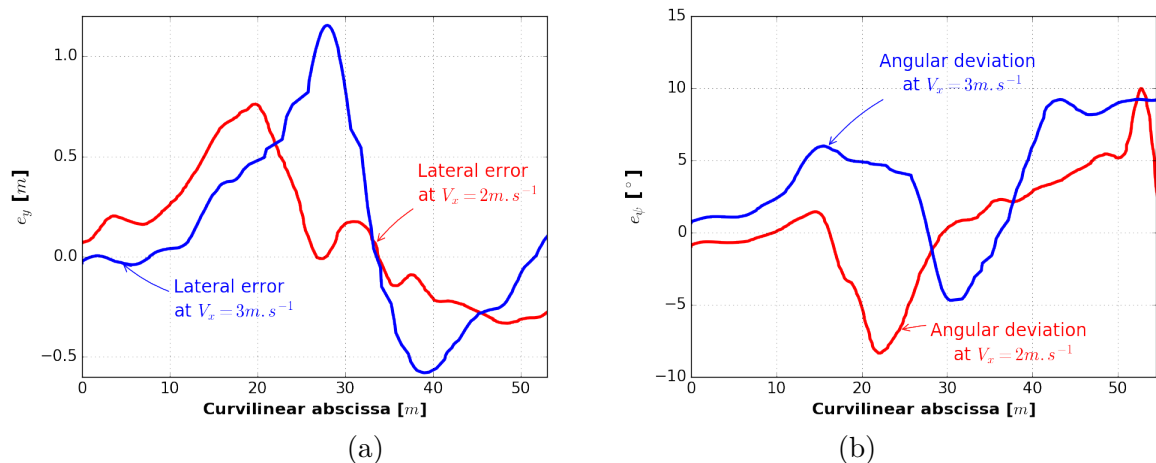


Figure 2.11. Résultats expérimentaux à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$ de chemin en U : (a) Écart latéral. (b) Écart angulaire de suivi de chemin.

4. Voir la vidéo suivante sur la validation expérimentale (Chemin en virage à droite) : [▶▶▶▶▶](#)

rapide gauche-droite sont plus importantes que celui à virage simple en U (cf. Figure 2.11). En effet, celui-ci présente un point d'inflexion où la concavité du chemin change rapidement ce qui nécessite un changement rapide de signe des angles de braquage. Ainsi, le suivi est toujours satisfaisant : le décalage par rapport au chemin de référence n'excède pas $1.1m$.

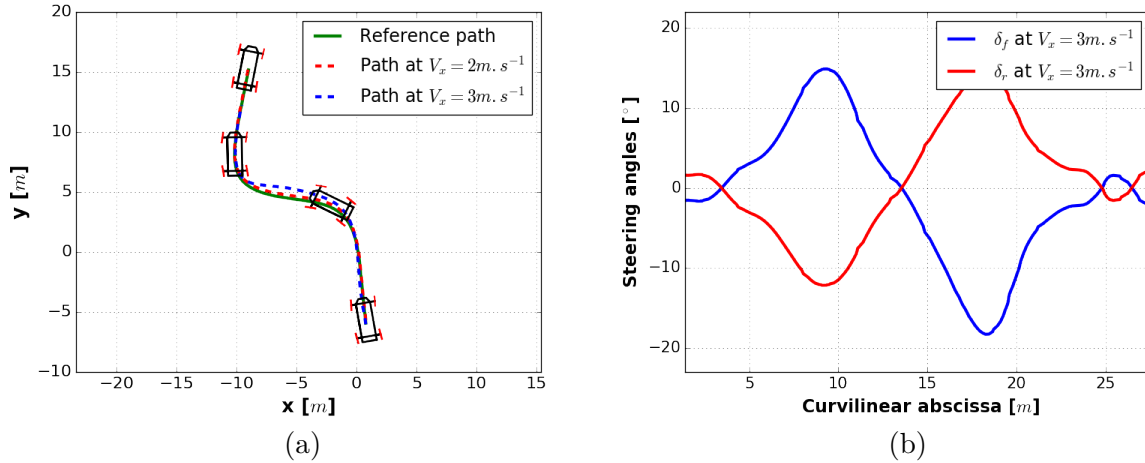


Figure 2.13. Résultats expérimentaux de chemin à deux manœuvres : (a) Chemin de référence et chemins réalisés à $V_x = 2m.s^{-1}$ et à $V_x = 3m.s^{-1}$ dans le plan (x, y) ; (b) Angles de braquage à $V_x = 3m.s^{-1}$.

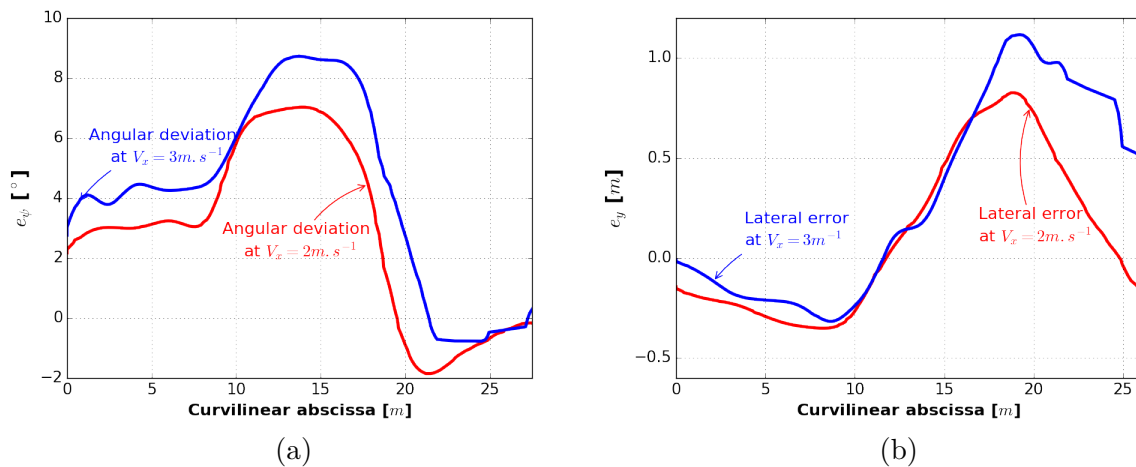


Figure 2.14. Résultats expérimentaux de chemin à deux manœuvres : (a) Écart latéral. (b) Écart angulaire de suivi de chemin à $V_x = 2m.s^{-1}$ et à $V_x = 3m.s^{-1}$.

Les vitesses latérale et de lacet observées par le filtre de Kalman-Bucy sont respectivement tracées dans les Figures 2.15(a) et (b). On peut remarquer que ces deux vitesses changent de signe au point d'inflexion et croissent légèrement quand la vitesse longitudinale augmente et lors du déplacement dans des virages à grandes courbures.

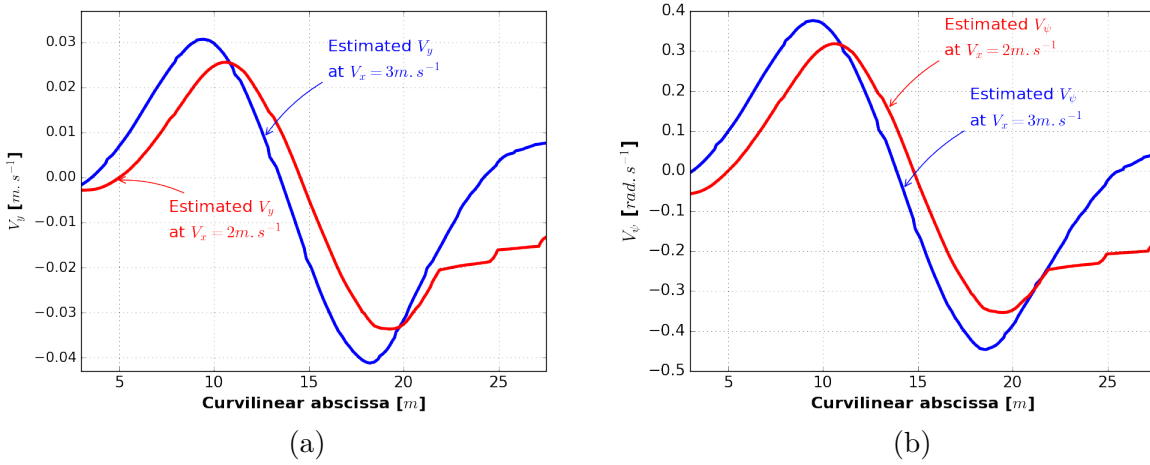


Figure 2.15. Résultats expérimentaux de chemin à deux manœuvres à $V_x = 2m \cdot s^{-1}$ et à $V_x = 3m \cdot s^{-1}$: (a) Vitesse latérale estimée. (b) Vitesse de lacet observé par le filtre de Kalman-Bucy.

5 Conclusion

Nous avons développé dans ce chapitre une modélisation analytique de la dynamique du robot prenant en compte les glissements des roues, dans le but de synthétiser une première loi de commande permettant le suivi d'un chemin de référence donné. Ce modèle analytique fusionne la dynamique latérale du véhicule avec le modèle cinématique en relation avec la tâche de suivi de chemin. Ensuite dans la deuxième partie, une loi de commande du suivi de chemin et un observateur d'état ont été synthétisés sur la base de ce modèle. Les résultats de simulation et expérimentaux montrent que les algorithmes développés gèrent efficacement la tâche de suivi de chemin.

Généralement, la commande LQR permet de faire tendre l'état vers zéro sans prendre en compte ni l'état future du système ni ses contraintes physiques. Par conséquent, une autre commande prédictive avec une fonction coût différente soumise aux différentes contraintes du système sera développée dans le chapitre 4.

Toutefois, pour être cohérent lorsque le véhicule se déplace en milieu naturel, l'estimation en ligne des coefficients de rigidités de dérive doit être prise en compte surtout quand les conditions d'adhérence sont variables et dépendent principalement du type de sol, de la charge supportée par les roues, des pneumatiques, etc. C'est dans ce contexte que nous avons synthétisé préalablement la commande LQR qui sera utilisée par la suite pour la validation de tous nos algorithmes d'estimation de paramètres du sol. Cela fera l'objet du chapitre suivant qui présentera un observateur non linéaire des rigidités de dérive pneumatiques et un observateur des paramètres de la géométrie locale du terrain.

Estimation des paramètres physiques et géométriques du sol

Sommaire

1	Introduction	42
2	Observateur non linéaire de paramètres de glissement	42
2.1	Observation dans la robotique mobile à roue	42
2.2	Modèle de l'observateur non linéaire	44
2.3	Synthèse de l'observateur non linéaire	44
2.4	Stabilité de l'observateur non linéaire	47
2.5	Résultats de simulations avancées	49
2.6	Résultats expérimentaux	52
3	Estimation des paramètres de la géométrie locale du terrain	55
3.1	Introduction	55
3.2	Synthèse de l'observateur de la géométrie locale du terrain	56
3.3	Stabilité de l'observateur	57
3.4	Validation par des simulations avancées	58
4	Conclusion	60

1 Introduction

Comme nous l'avons vu au chapitre précédent, le contrôleur LQR et l'observateur de Kalman-Bucy utilisent à la fois les variables de glissement et les paramètres de la géométrie locale du terrain (rigidités de dérive, coefficient du frottement, angles de la pente, etc.). Cette loi de commande mixée avec l'observateur d'état a pour objectif de calculer en temps-réel les angles de braquage avant et arrière les plus adéquats pour la tâche de suivi de chemin. Naturellement, la qualité de suivi ne sera pertinente que si ces paramètres roue/sol sont estimés de telle façon qu'ils reflètent la réalité du terrain et les conditions réelles du contact.

Notre motivation dans ce chapitre est d'améliorer les performances de nos algorithmes de suivi de chemin basés sur le modèle dynamique bicyclette et le modèle linéaire du pneumatique. À cet effet, l'estimation des paramètres d'adhérence et de la géométrie locale du terrain revêt alors d'une importance capitale pour améliorer ces performances, essentiellement dans des environnements naturels où la surface du sol est quelconque et où les conditions d'adhérence sont fortement variables. Dans un premier temps, un nouvel observateur non linéaire (noté par la suite **NLO**) sera succinctement développé dans la section 2, il permet d'estimer en ligne et indépendamment les rigidités de dérive des pneumatiques. Ensuite dans la section 3, un deuxième observateur sera synthétisé dans le but d'estimer les paramètres de la géométrie locale du terrain (noté **OAT**). Les variables estimées seront injectées à chaque instant dans le bloc commande dans le but d'améliorer son efficacité et la qualité de suivi de chemin. L'apport des observateurs proposés sera illustré par des simulations numériques sous le simulateur physique ROS-GAZEBO et des expériences réelles sur le robot mobile rapide.

2 Observateur non linéaire de paramètres de glissement

2.1 Observation dans la robotique mobile à roue

Une bonne conduite autonome d'un robot mobile passe en général par la maîtrise de son comportement dynamique. Cette maîtrise nécessite souvent la connaissance complète et à tout instant des différents paramètres et variables constituant ce modèle (e.g., vecteur d'état, paramètres pneumatiques, paramètres de la géométrie de terrain, etc.). Globalement, il est souvent difficile d'accéder à toutes ces variables par des mesures directes compte tenu des contraintes économiques et technologiques (i.e., manque et/ou cherté des capteurs adéquats). Par conséquent, il s'avère opportun d'estimer ces paramètres via des techniques d'observation. Dans ce contexte, la communauté robotique s'est intéressée à la conception des observateurs et des estimateurs. La majorité de ceux-ci est destinées principalement aux systèmes linéaires à paramètres invariants (LPI) au cours de temps et en utilisant des approches standards telles que le filtrage de Kalman ou de Luenberger [Bouton et al. \(2007\)](#), [Lenain et al. \(2010\)](#). Cependant, la plupart des comportements sont non linéaires ce qui a incité les chercheurs à synthétiser des observateurs mathématiques non linéaires [Grip et al. \(2008\)](#), [Lenain et al. \(2017\)](#). Contrairement aux systèmes linéaires, il n'existe pas dans la littérature à l'heure actuelle une méthode globale concernant la synthèse des observateurs non linéaires.

Nous nous intéressons dans cette section à l'amélioration des performances de nos algorithmes de suivi de chemin par le biais d'estimation des conditions d'adhérence à l'interface pneumatique/sol. Comme première approximation dans la plupart des travaux de recherche, les paramètres pneumatique/chaussée sont considérés constants voire définis d'une manière arbitraire ou estimés expérimentalement hors ligne (comme le cas de la commande LQR développée et validée dans le chapitre précédent). Par exemple, [Krid et al. \(2017\)](#) et [Borrelli et al. \(2005\)](#) présentent deux commandes prédictives non linéaires (NMPC) basées sur un modèle dynamique. Celles-ci sont destinées aux véhicules à deux trains directeurs (4WS) pour le suivi de chemin, en revanche, elles considèrent des coefficients de rigidité de dérive constants. Toutefois, ces contrôleurs ne peuvent pas garantir la stabilité du système si les conditions d'adhérence ne sont pas bien définies.

La rigidité de dérive transversale du pneu $C_{(f,r)}$ est le paramètre principal qui caractérise le comportement latéral du contact roue/sol. Pour des roues souples (pneu, caoutchouc, etc) et des terrains mous (sable, argile, pelouse, etc.), ce paramètre permet de caractériser plusieurs phénomènes physiques (e.g., déformation élastique du pneu, paramètres de sol en cisaillement, paramètres de frottement sec et/ou visqueux, etc.) [Badji \(2009\)](#). Il est défini comme le rapport entre la force latérale appliquée à la roue et son angle de glissement (cf. Figure 2.2).

En outre, les contrôleurs dynamiques des systèmes de guidage autonome pour robots mobiles ou de conduite assistée par ordinateur en automobile doivent intégrer des techniques d'observation avancées pour estimer les variables d'état et les paramètres de l'environnement. De nombreuses stratégies d'observation ont été développées dans la littérature afin d'estimer avec une grande précision les variables de contact roue/sol, allant de la construction de dispositifs spécifiques de mesure à la conception de filtres et d'algorithmes combinant des variables d'état et d'entrée qui sont déjà disponibles. Par exemple, un capteur optique sans fil est utilisé dans [Tuononen \(2009\)](#) pour mesurer les forces pneumatiques qui sont exploitées pour observer la vitesse latérale du véhicule. Le coefficient de frottement et la rigidité de dérive sont estimés en fonction de la vitesse de la roue et d'un modèle de comportement anisotrope du pneu dans [Liu et Peng \(1996\)](#). Néanmoins, les principaux inconvénients de ces approches basées sur le comportement des pneus sont : d'une part, elles peuvent facilement être influencées par certains facteurs tels que la rugosité routière, la pression des pneumatiques et les systèmes de freinage, d'autre part, les conditions d'adhérence et la mauvaise installation de ces capteurs peuvent endommager grandement la précision de l'estimation. Par conséquent, au lieu de développer des capteurs spécifiques à installer et implémenter dans les pneumatiques (comme ce qu'est fréquemment utilisé dans l'industrie automobile), la communauté robotique s'intéresse à l'utilisation de la dynamique du véhicule pour l'estimation de ces paramètres roue/sol. En effet, l'utilisation de la dynamique du véhicule pour l'estimation des rigidités de dérive présente l'avantage de réutiliser certains signaux préalablement mesurés pour d'autres objectifs (tels que les angles de braquage, la vitesse de lacet, l'accélération latérale, etc.). Par conséquent, le coût du matériel embarqué sera réduit.

En général, les méthodes d'estimation des forces de contact et des paramètres pneumatiques sont basées sur la dynamique latérale du véhicule. À titre d'exemple, des capteurs GPS et inertiels sont combinés afin d'observer en ligne la rigidité de dérive du véhicule dans [Anderson et Bevly \(2005\)](#) et [Bevly et al. \(2006\)](#). Deux méthodes (temporelle et fréquentielle) ont été développées afin d'estimer les rigidités de dérive transversales en reposant sur la dynamique de véhicule dans [Sierra et al. \(2006\)](#). Un observateur basé sur une méthode de backstepping est proposé dans

[Bouton et al. \(2007\)](#) pour adapter la rigidité de dérive en temps réel (la rigidité de dérive avant et arrière sont supposées égales) en utilisant une vitesse de lacet obtenue par un observateur linéaire de Luenberger. De plus, [Baffet et al. \(2009\)](#) et [Doumiati et al. \(2011\)](#) présentent une méthode permettant d'estimer les forces d'interaction roue/sol, les angles de glissement et les rigidités de dérive en combinant deux filtres de Kalman étendus (EKF) et en utilisant certaines mesures (e.g., angles de braquage, débattements de suspension, etc.). Finalement, une estimation des paramètres d'adhérence est réalisée par l'intermédiaire d'un filtre de Kalman-Bucy étendu dans [Ray \(1997\)](#) et grâce à un observateur adaptatif dans [Gustafsson \(1997\)](#).

L'objectif de nos travaux dans cette section étant d'améliorer la précision et la stabilité de nos lois de commande de suivi de chemin via une estimation précise des conditions d'adhérence et des paramètres utilisés dans le modèle de commande. Nous utilisons la dynamique de lacet et latérale (2.14) pour reconstruire les coefficients de rigidité de dérive C_f et C_r , la dynamique longitudinale étant négligée.

2.2 Modèle de l'observateur non linéaire

Les robots mobiles tout-terrain sont conçus pour traverser différents types de terrains où les conditions d'adhérence roue/sol sont variables. Par conséquent, l'adaptation en temps-réel des rigidités de dérive avant et arrière C_f et C_r est primordiale pour contrôler efficacement l'interaction physique de ces robots avec l'environnement naturel. En isolant ces rigidités de dérive du modèle dynamique (2.14), nous obtenons le modèle non linéaire suivant :

$$\dot{x}_c = A_c x_c + B_c(x_c, u) \omega_c + S_l, \quad (3.1)$$

avec

$$A_c = \begin{bmatrix} 0 & -V_x \\ 0 & 0 \end{bmatrix}, \quad \omega_c = \begin{bmatrix} C_f \\ C_r \end{bmatrix}, \quad x_c = \begin{bmatrix} V_y \\ V_\psi \end{bmatrix}, \quad u = \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix}$$

Le vecteur S_l dépend des angles de la géométrie du terrain et la matrice $B_c(x_c, u)$ quant à elle dépend de la vitesse latérale V_y et angulaire V_ψ ainsi que les angles de braquage avant et arrière du robot $\delta_{(f,r)}$. Elles sont exprimées comme suit :

$$B_c(x_c, u) = \begin{bmatrix} \frac{-2V_y - 2aV_\psi + 2V_x\delta_f}{mV_x} & \frac{-2V_y + 2aV_\psi - 2V_x\delta_r}{mV_x} \\ \frac{-2aV_y - 2a^2V_\psi + 2aV_x\delta_f}{I_zV_x} & \frac{2aV_y - 2a^2V_\psi + 2aV_x\delta_r}{I_zV_x} \end{bmatrix}, \quad S_l = \begin{bmatrix} g \cos \phi_r \sin \theta_r \\ 0 \end{bmatrix},$$

2.3 Synthèse de l'observateur non linéaire

La matrice $B_c(x_c, u)$ décrite par l'équation (3.1) est non singulière si et seulement si $V_y \neq -aV_\psi + V_x\delta_f$ et $V_y \neq aV_\psi - V_x\delta_r$. Dans ce cas, le vecteur ω_c peut être représenté par l'équation ci-dessous :

$$\omega_c = B_c(x_c, u)^{-1}(\dot{x}_c - A_c x_c - S_l), \quad (3.2)$$

Rappelons que l'objectif de l'observateur non linéaire (NLO) est de fournir un estimé des rigidités de dérive avant et arrière $\hat{\omega}_c(t) = [\hat{C}_f, \hat{C}_r]^T$ qui doit être le plus proche possible de la valeur réelle de $\omega_c(t)$. Nous appelons l'erreur d'estimation le signal $\tilde{\omega}_c(t)$ définie par la formule suivante :

$$\tilde{\omega}_c(t) = \omega_c(t) - \hat{\omega}_c(t), \quad (3.3)$$

Nous souhaitons que l'erreur d'estimation soit nulle en régime permanent,

$$\lim_{t \rightarrow \infty} \tilde{\omega}_c(t) = \mathbf{0}_{2 \times 1}, \quad (3.4)$$

En s'inspirant des travaux de [Nikoobin et Haghghi \(2009\)](#) appliqués aux robots manipulateurs, la structure de l'observateur NLO peut être déduite comme suit :

$$\dot{\hat{\omega}}_c = G(x_c, u)(\omega_c - \hat{\omega}_c), \quad (3.5a)$$

$$= G(x_c, u)B_c^{-1}\dot{x}_c - G(x_c, u)B_c^{-1}A_c x_c - G(x_c, u)B_c^{-1}S_l - G(x_c, u)\hat{\omega}_c, \quad (3.5b)$$

avec $G(x_c, u)$ la matrice gain de l'observateur NLO qui dépend de la vitesse latérale V_y , la vitesse de lacet V_ψ ainsi que les angles de braquage avant et arrière $\delta_{(f,r)}$. Cette matrice gain doit être conçue de telle façon que l'erreur d'estimation soit nulle en régime permanent.

L'observateur NLO a pour but d'estimer en ligne ω_c en supposant que la zone non linéaire du pneumatique n'est pas atteinte dans laquelle le robot devient incontrôlable (voir Figure 2.2). Les variables estimées, représentant la vérité du terrain contact roue-sol, seront introduites en temps réel dans le contrôleur LQR pour ajuster ses matrices gain (voir Figure 3.1). En supposant que les paramètres d'adhérence changent plus lentement que ceux de l'état du robot (i.e. le taux de changement de C_f et C_r est négligeable par rapport à la dynamique du véhicule), nous pouvons supposer que :

$$\dot{\omega}_c = 0, \quad (3.6)$$

En utilisant les équations (3.5a) et (3.6), la dérivée de l'erreur d'estimation $\dot{\tilde{\omega}}_c(t)$ peut être exprimée sous la forme suivante :

$$\dot{\tilde{\omega}}_c(t) = \dot{\omega}_c - \dot{\hat{\omega}}_c, \quad (3.7a)$$

$$= 0 - G(x_c, u)(\omega_c - \hat{\omega}_c), \quad (3.7b)$$

En prenant en compte (3.7), on peut en déduire,

$$\dot{\tilde{\omega}}_c(t) + G(x_c, u)\tilde{\omega}_c(t) = 0, \quad (3.8)$$

La matrice gain $G(x_c, u)$ doit être alors choisie judicieusement afin d'assurer la stabilité asymptotique du NLO. Nous définissons un vecteur auxiliaire de la forme suivante :

$$\Lambda = \hat{\omega}_c - c(x_c, u), \quad (3.9)$$

La dérivée temporelle de l'équation (3.9) est donnée comme suit,

$$\dot{\hat{\omega}}_c = \dot{\Lambda} + \frac{dc(x_c, u)}{dt}, \quad (3.10a)$$

$$= \dot{\Lambda} + \frac{\partial c(x_c, u)}{\partial x_c} \dot{x}_c + \frac{\partial c(x_c, u)}{\partial u} \dot{u}, \quad (3.10b)$$

D'un point de vue pratique, les angles de braquage varient d'une façon lisse et moins agressive sur un horizon de temps fini, i.e. $\dot{u} \approx 0$. Sous cette hypothèse, nous pouvons écrire en combinant les équations (3.5b) et (3.10b),

$$\dot{\Lambda} + \frac{\partial c(x_c, u)}{\partial x_c} \dot{x}_c = G(x_c, u)B_c^{-1} \dot{x}_c - G(x_c, u)B_c^{-1} A_c x_c - G(x_c, u)B_c^{-1} S_l - G(x_c, u) \hat{\omega}_c, \quad (3.11)$$

Nous choisissons :

$$\frac{\partial c(x_c, u)}{\partial x_c} = G(x_c, u)B_c^{-1}, \quad (3.12)$$

La dynamique de l'observateur NLO (cf. Figure 3.1) peut être alors exprimée sous la forme suivante :

$$\begin{cases} \dot{\Lambda} &= -G(x_c, u)\Lambda - G(x_c, u) [B_c^{-1}A_c x_c + B_c^{-1}S_l + c(x_c, u)], \\ \hat{\omega}_c &= \Lambda + c(x_c, u), \end{cases} \quad (3.13)$$

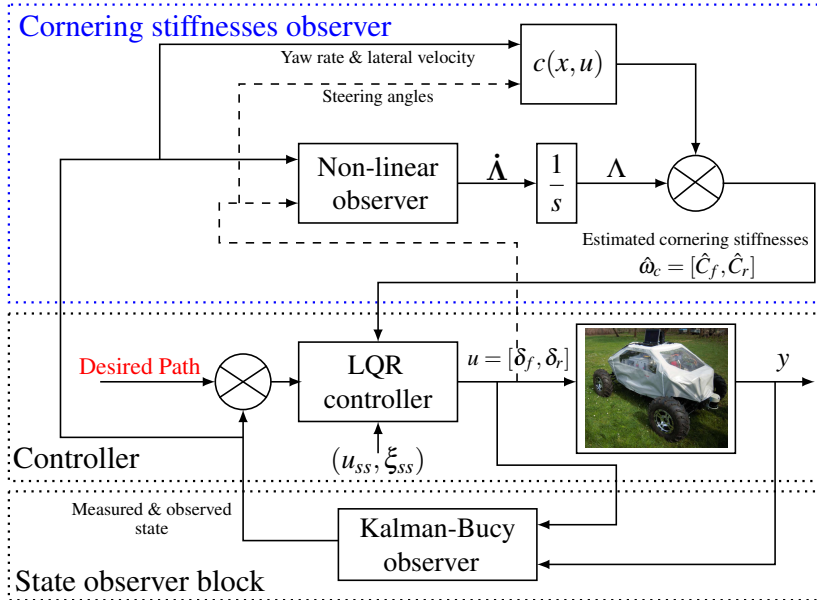


Figure 3.1. Schéma global de contrôleur et des observateurs développés. Le premier bloc présente l'observateur NLO destiné à estimer C_f et C_r . Ces variables seront injectées dans le deuxième bloc (contrôleur) qui calcule les angles de braquage appropriés, en tenant compte du chemin désiré et du vecteur d'état estimé par le troisième bloc (filtre de Kalman-Bucy).

2.4 Stabilité de l'observateur non linéaire

Comme on le voit sur le modèle de l'observateur NLO donné par l'équation (3.13), qu'il y a deux paramètres à concevoir $G(x_c, u)$ et $c(x_c, u)$. Cependant, ces deux paramètres ne sont pas indépendants et sont liés l'un à l'autre par l'équation (3.12). Nous devons choisir judicieusement l'un des deux paramètres de telle sorte que la stabilité asymptotique de l'observateur NLO soit garantie. Dans cette partie, nous proposons une expression pour $c(x_c, u)$ en fonction des angles de braquage et vitesses du véhicule afin d'assurer la stabilité de l'observateur NLO proposé dans (3.13).

Pour prouver la stabilité asymptotique de l'observateur, nous définissons la fonction de Lyapunov quadratique candidate comme ci-dessous :

$$V(\tilde{\omega}_c) = \frac{1}{2} \tilde{\omega}_c^T T \tilde{\omega}_c, \quad (3.14)$$

avec T est une matrice définie positive.

La dérivée temporelle de $V(\tilde{\omega}_c)$ doit être négative, i.e., $\frac{dV(\tilde{\omega}_c)}{dt} < 0$, afin de garantir la condition de la stabilité asymptotique. En prenant en compte l'équation (3.8), nous obtenons l'expression suivante :

$$\frac{dV(\tilde{\omega}_c)}{dt} = \tilde{\omega}_c^T T \dot{\tilde{\omega}}_c = -\tilde{\omega}_c^T [T G(x_c, u)] \tilde{\omega}_c, \quad (3.15)$$

Par conséquent, le terme $[T G(x_c, u)]$ doit être définie positive pour assurer la stabilité asymptotique de l'observateur NLO. Nous choisissons T en tant qu'une matrice identité $I_{2 \times 2}$. Par la suite, $G(x_c, u) \in \mathbb{R}^{2 \times 2}$ doit être choisie définie positive.

On commence par le choix adéquat du vecteur $c(x_c, u)$ et on calcule par la suite le gain $G(x_c, u)$ en utilisant l'équation (3.12). En s'appuyant sur le modèle de l'observateur NLO (3.1), nous choisissons avec soin $c(x_c, u)$ qui rend la matrice $G(x_c, u)$ définie positive.

$$c(x_c, u) = \begin{bmatrix} (V_y + aV_\psi)(2V_x\delta_f - V_y - aV_\psi) \\ (-V_y + bV_\psi)(2V_x\delta_r + V_y - bV_\psi) \end{bmatrix}, \quad (3.16)$$

En prenant en compte la symétrie du véhicule, l'empattement avant et arrière sont égaux. Alors la Jacobienne de $c(x_c, u)$ par rapport à l'état x_c peut être dérivée sous la forme suivante :

$$\frac{\partial c(x_c, u)}{\partial x} = \begin{bmatrix} 2V_x\delta_f - 2V_y - 2aV_\psi & 2aV_x\delta_f - 2aV_y - 2a^2V_\psi \\ -2V_x\delta_r - 2V_y + 2aV_\psi & 2aV_x\delta_r + 2aV_y - 2a^2V_\psi \end{bmatrix}, \quad (3.17)$$

En utilisant l'équation (3.12), le gain $G(x_c, u)$ peut être exprimé comme suit :

$$G(x_c, u) = \frac{\partial c(x_c, u)}{\partial x_c} B_c(x_c, u) = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}, \quad (3.18)$$

avec,

$$\begin{aligned} G_{11} &= \frac{I_z + a^2 m}{m I_z V_x} (-2V_y - 2aV_\psi + 2V_x \delta_f)^2, \\ G_{22} &= \frac{I_z + a^2 m}{m I_z V_x} (-2V_y + 2aV_\psi - 2V_x \delta_r)^2, \\ G_{12} &= \frac{I_z - a^2 m}{m I_z V_x} (-2V_y + 2aV_\psi - 2V_x \delta_r)(-2V_y - 2aV_\psi + 2V_x \delta_f), \\ G_{21} &= \frac{I_z - a^2 m}{m I_z V_x} (-2V_y + 2aV_\psi - 2V_x \delta_r)(-2V_y - 2aV_\psi + 2V_x \delta_f), \end{aligned}$$

Maintenant, il nous reste à prouver que $\frac{dV(\tilde{\omega}_c)}{dt} < 0$. Pour cela, il faut montrer que $G(x_c, u)$ est une matrice définie positive. On met en place le principe algébrique de Sylvester énoncé ci-dessous.

Théorème : Une matrice symétrique $M \in \mathcal{M}_n(\mathbb{R})$ est définie positive si et seulement si tous ses déterminants mineurs principaux sont strictement positifs.

Les mineurs principaux de la matrice symétrique $G(x_c, u)$ peuvent être écrits comme ci-dessous :

-1^{er} mineur principal :

$$G_{11} = \frac{I_z + a^2 m}{m I_z V_x} (-2V_y - 2aV_\psi + 2V_x \delta_f)^2 > 0,$$

-2^{ème} mineur principal :

$$G_{11}G_{22} - G_{12}G_{21} = \left[\frac{I_z^2 - a^4 m^2}{(m I_z V_x)^2} \right]^2 (2aV_y + 2a^2V_\psi - 2aV_x \delta_f)^2 (2aV_y - 2a^2V_\psi + 2aV_x \delta_r)^2 > 0,$$

Comme on peut le constater, les déterminants de tous les mineurs de $G(x, u)$ sont strictement positifs, alors la matrice gain de l'observateur NLO est définie positive. Ainsi, la stabilité asymptotique de l'observateur NLO est garantie.

Dans les parties suivantes, on procède à la validation de cet observateur, dans un premier temps, sur la base de simulations sous ROS-GAZEBO à hautes vitesses, ensuite à travers des expérimentations dans les conditions réelles. Toutes ces validations sont faites sur un terrain supposé préalablement plat et horizontal (i.e., $\theta_r = \phi_r = 0^\circ$).

Pour montrer l'efficacité de l'observateur NLO proposé, les erreurs latérales et angulaires pour les deux scénarios sont tracées respectivement sur les Figures 3.3(a) et (b) aux vitesses longitudinales $V_x = 5m.s^{-1}$ et $V_x = 8m.s^{-1}$. Les erreurs obtenues via l'observateur NLO sont représentées par une ligne continue et celles acquises sans observateur ($C_{(f,r)} = 14\ 000N.rad^{-1}$) sont dessinées en traits pointillés. On peut remarquer que ces erreurs sont élevées quand le véhicule se déplace le long des grandes courbures où le phénomène de glissement prédomine. Néanmoins, les erreurs de suivi de chemin sont plus faibles quand l'observateur NLO est mis en place (**Scénario 2**) que celles fournies quand il n'est pas pris en compte (**Scénario 1**) ($\pm 10cm$ avec l'observateur NLO, contre $\pm 30cm$ sans observateur). Par conséquent, l'estimation précise des conditions d'adhérence à l'interface pneumatique/chaussé par l'intermédiaire de l'observateur NLO améliore la précision de l'algorithme de suivi de chemin, malgré les conditions glissantes et les fortes variations de courbure.

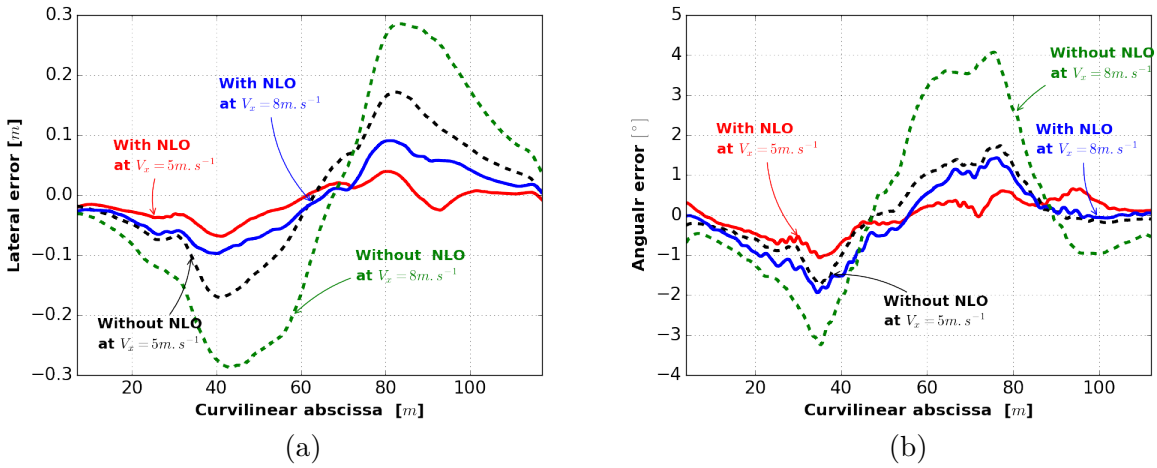


Figure 3.3. Résultats de simulation à $V_x = 5m.s^{-1}$ et à $V_x = 8m.s^{-1}$ pour les deux scénarios : (a) Écart latéral en suivi de chemin ; (b) Écart angulaire en suivi de chemin.

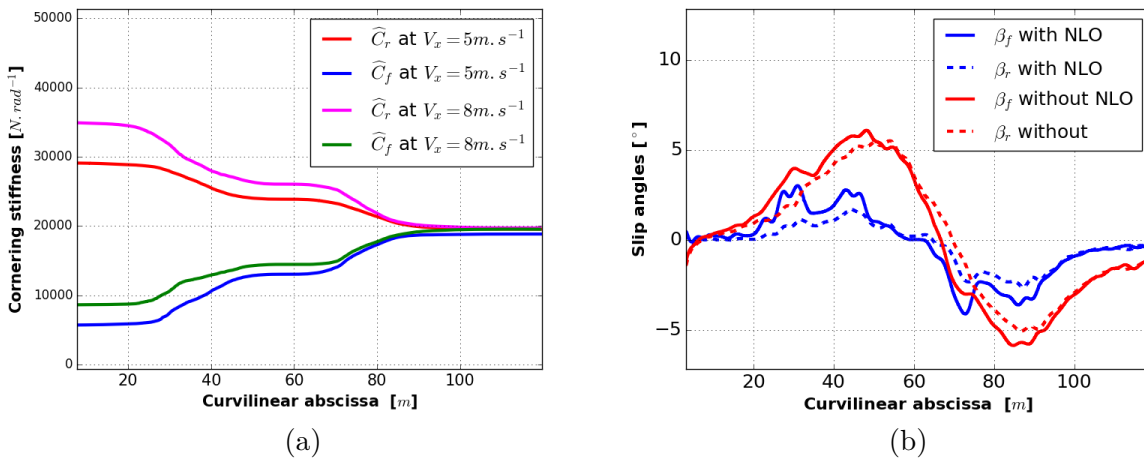


Figure 3.4. Résultats de simulation à $V_x = 5m.s^{-1}$ et à $V_x = 8m.s^{-1}$: (a) Rigidités de dérive avant et arrière estimées à différentes conditions initiales ; (b) Angles de glissement avant et arrière pour les deux scénarios.

Les rigidités de dérive avant et arrière $C_f(\cdot)$ et $C_r(\cdot)$ ont alors été estimées à partir des valeurs initiales différentes qui ont été choisies arbitrairement, comme décrit sur la Figure 3.4(a). Ces quatre courbes se superposent rapidement et convergent vers les mêmes valeurs, ce qui permet de valider la robustesse et la convergence de l'algorithme d'observation vis-à-vis des conditions initiales. Ensuite, en régime permanent, ces rigidités de dérive convergent vers une valeur de l'ordre de $19\,000N.rad^{-1}$, qui est vraisemblablement la rigidité réelle du contact roue/sol.

Sur les Figures (3.5)(a) et (b), les efforts latéraux avant et arrière mesurés sous Gazebo (trait pointillé en noir) sont comparés aux efforts reconstruits via l'observateur NLO (traits continu en bleu) à la vitesse longitudinale $V_x = 8m.s^{-1}$. Ceux qui sont fournis par le simulateur Gazebo sont très bruités en raison des surfaces de contact discrétisées entre le robot et l'environnement Gazebo. Nous avons adopté un filtre passe-bas de Butterworth² à faible fréquence de coupure pour éliminer les bruits du simulateur. Les efforts calculés et estimés restent proches entre eux et sont plus importants à faible rayon de courbure où le phénomène de glissement prédomine.

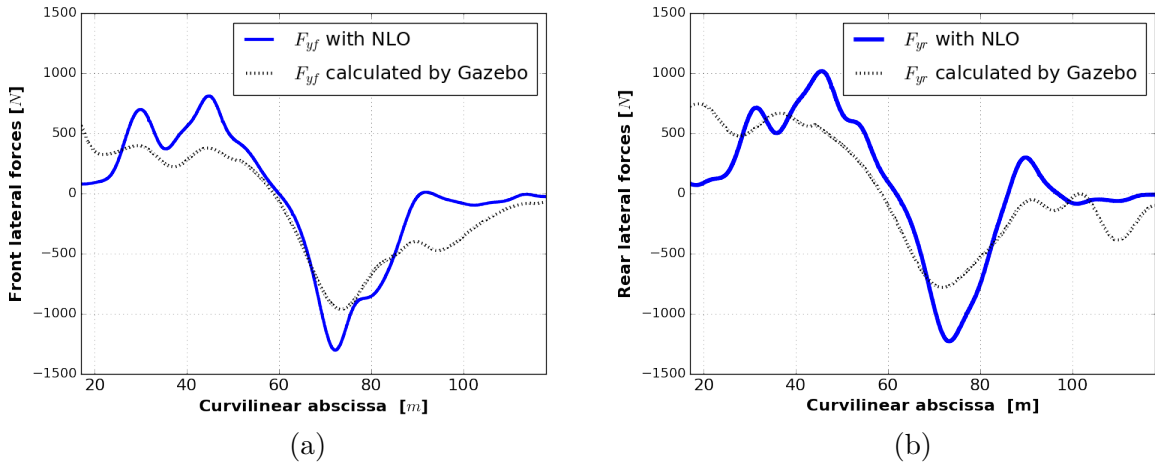


Figure 3.5. Résultats de simulation à $V_x = 8m.s^{-1}$: (a) Effort latéral estimé et calculé de l'essieu avant ; (b) Effort latéral estimé et calculé de l'essieu arrière.

En outre, les angles de glissement avant et arrière pourraient être calculés à partir de la vitesse latérale et de lacet, ainsi que les angles de braquage ($\beta_f = \frac{V_y + aV_\psi}{V_x} - \delta_f$ et $\beta_r = \frac{V_y - bV_\psi}{V_x} - \delta_r$). Ces angles sont représentés sur la Figure 3.4(b) pour chaque scénario à $V_x = 8m.s^{-1}$. Dans les faits, nous constatons que les angles calculés dans le **scénario 1** sont plus importants que ceux fournis par le **scénario 2** (6° à l'abscisse curviligne $50m$ sans NLO, contre 3° avec NLO). Ainsi, l'utilisation de cet observateur NLO joue un rôle primordial pour maîtriser les conditions du contact roue/sol et minimiser le glissement dans ce contact. Ceci est intéressant pour la stabilité latérale du véhicule, notamment dans les virages, en préservant le contact dans la zone de pseudo-glissement des pneumatiques (cf. Figure 2.2) où le robot reste contrôlable.

2. La synthèse de ce filtre est faite à partir d'un gabarit comprenant toutes les spécifications (e.g., fréquences de coupure, bande passante, bande de transition, etc) en utilisant les fonctions `buttord` et `butter` de la librairie Python `scipy.signal`

2.6 Résultats expérimentaux

Dans cette section, l'objectif est d'illustrer l'intérêt de l'observateur NLO qui permet une meilleure estimation des rigidités de dérive selon le type de pneumatique et de sol sur lequel évolue le véhicule. Un chemin de référence représenté sur la Figure 3.6(a) a été choisi avec un rayon de courbure plus petit afin d'obtenir la même accélération latérale des simulations numériques à la vitesse 11km.h^{-1} . En effet, cette accélération est proportionnelle au carré de la vitesse longitudinale et la courbure de chemin ($a_y = \rho V_x^2$). La fréquence des différents capteurs dans le champ d'expérimentation est de 10Hz^3 .

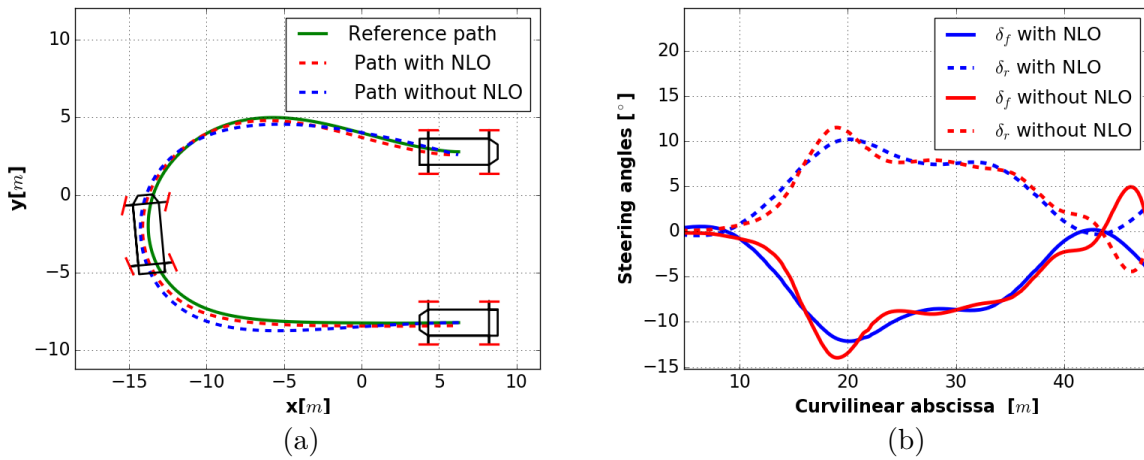


Figure 3.6. Résultats expérimentaux à $V_x = 11\text{km.s}^{-1}$ pour les deux scénarios : (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Angles de braquage avant et arrière.

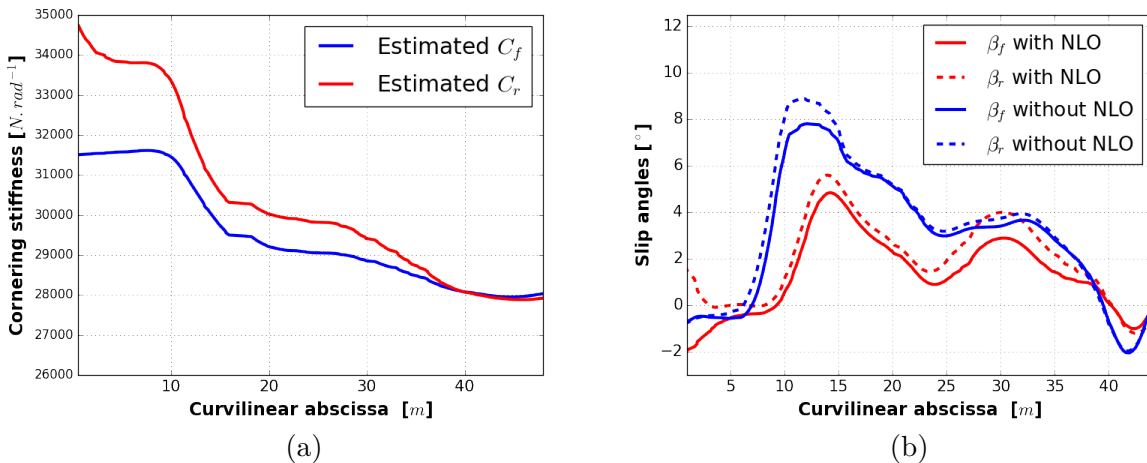


Figure 3.7. Résultats expérimentaux à $V_x = 11\text{km.s}^{-1}$: (a) Rigidités de dérive avant et arrière estimées à différentes condition initiales; (b) Angles de glissement avant et arrière avec et sans l’NLO

3. Voir la vidéo suivante sur la validation expérimentale de l'observateur NLO : [▶▶▶▶▶](#)

La surface sur laquelle le véhicule évolue est recouverte d'une pelouse sèche. Ce terrain est plat mais entouré de bâtiments et d'arbres, ce qui affecte les mesures de GPS-RTK (e.g., apparition des points aberrants) (cf. Figure C.1). Cependant, nous envisageons par la suite de fusionner les données odométriques et GPS-RTK afin de surmonter ce problème de la localisation par le biais d'un Filtre de Kalman Étendu (EKF).

Comme prévu, le chemin enregistré dans le **scénario 2** est plus proche au chemin de référence que celui calculé par le **scénario 1** (cf. Figure 3.6(a)). Les résultats expérimentaux ont été fortement influencés par le signal GPS-RTK qui nécessite un espace très ouvert pour obtenir une bonne localisation. Malgré ce problème, l'observateur NLO permet d'assurer une grande précision de suivi de chemin comme montré par la Figure 3.8(a) où l'écart latéral est tracé (à l'abscisse curviligne 10m, l'écart latéral reste autour de 0.8m avec NLO, contre 1.1m sans NLO).

En outre, les véhicules à deux trains directeurs ont l'avantage de fournir des angles de direction lisses, faibles et moins agressifs comme illustré par la Figure 3.6(b). Ces angles restent dans l'intervalle $\pm 13^\circ$ pour les deux **scénarios**.

A partir des mesures de la vitesse latérale, de la vitesse de lacet et les angles de braquage, les rigidités de dérive avant et arrière sont adaptées (Figure 3.7(a)). On peut observer qu'elles convergent vers la même valeur en régime établi (aux alentours de $28\,000\text{N}\cdot\text{rad}^{-1}$) qui caractérise la condition du contact roue-sol dans un sol couvert d'herbe sèche. Nous remarquons bien comme attendu, que la rigidité de dérive estimée est importante, ce qui traduit que le véhicule se déplace sur un sol très adhérent (type herbe sèche, surfaces dures). D'ailleurs, une faible rigidité reflète le fait que le véhicule bouge sur un terrain peu adhérent (neige, herbe humide, etc.). Nous constatons par ailleurs que entre 0 et 10m les estimés restent quasi constants car la trajectoire est une ligne droite qui n'est pas excitante et que dès le début du virage entre 10 et 15m l'observateur converge assez rapidement mais stagne entre 15m et 20m probablement c'est le régime stationnaire sur la partie circulaire du chemin.

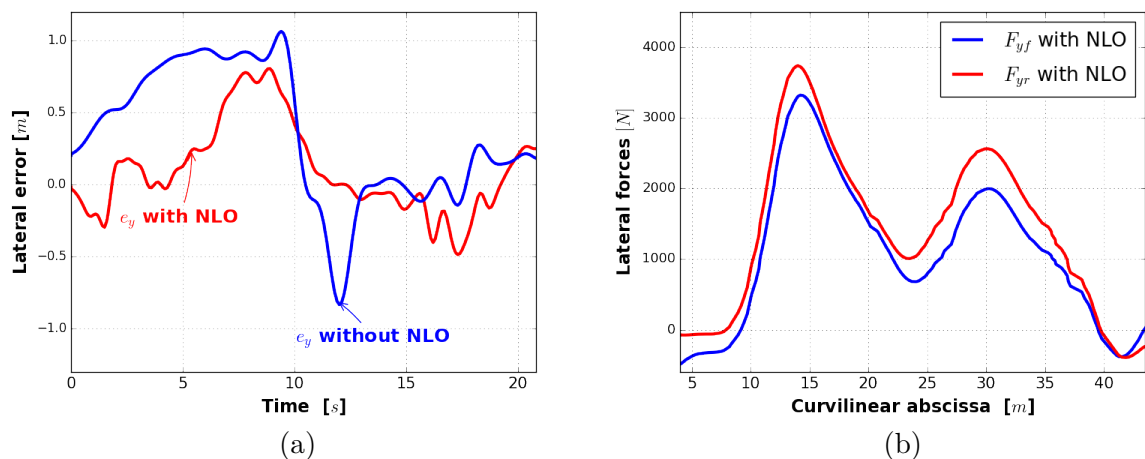


Figure 3.8. Résultats expérimentaux à $V_x = 11\text{km}\cdot\text{s}^{-1}$: (a) Écart latéral en suivi de chemin pour les deux situations avec et sans l'NLO ; (b) Efforts latéraux de l'essieu avant et arrière estimés.

Les forces latérales sont quant à elles assez importantes principalement le long du virage en conséquence de la forte accélération latérale (cf. Figure 3.8(b)). On rappelle que le robot pèse dans les $800kg$ et que la charge sur l'essieu avant est autour donc de $400kg$ (modèle bicyclette) qui est comparable à la force maximale estimée. Les angles de glissement latéral restent petits dans le cas où le contrôleur LQR est combiné avec l'observateur NLO (**Scénario 2**), comme le montre la Figure 3.7(b) (6° avec NLO à l'abscisse curviligne $15m$, contre 9° sans NLO). Ainsi, la stabilité latérale des robots mobiles tout-terrain peut être améliorée grâce à une bonne estimation de paramètres du sol.

En conclusion, les résultats de simulation et expérimentaux montrent que la loi de commande LQR couplée avec l'observateur NLO (**Scénario 2**) sont meilleurs que ceux obtenus sans observateur (**Scénario 1**). Ainsi, l'utilisation de cet observateur NLO réduit considérablement les angles de glissement et les erreurs de suivi de chemin, en particulier durant les grandes courbures et quand les conditions d'adhérence sont variables. Ceci montre l'intérêt d'estimation précise des conditions d'adhérence au niveau du contact roue/sol dans l'amélioration des performances de suivi de chemin. Cet observateur NLO sera pris en compte dans tous les travaux présentés par la suite

3 Estimation des paramètres de la géométrie locale du terrain

3.1 Introduction

Les systèmes de contrôle de la stabilité du véhicule et les estimateurs d'état utilisent couramment les mesures des accélérations provenant des accéléromètres pour calculer les angles de dérives du véhicule et les angles d'inclinaison du terrain [Jo et al. \(2008\)](#). Toutefois, ces mesures d'accélération sont facilement affectées par des perturbations telles que l'angle de dévers du terrain et l'angle de roulis du véhicule générés par la déformation du système de suspension. Par ailleurs, les effets liés à la gravité influencent fortement la dynamique latérale du véhicule en raison des irrégularités du terrain (bosse, pente, etc.) [Pastor et Tierney \(1995\)](#). Ainsi, la connaissance des angles d'inclinaison du véhicule selon la géométrie locale du terrain est extrêmement importante pour ces systèmes pour atteindre une précision et une convergence élevées de suivi de chemin, notamment à hautes vitesses.

La dynamique latérale peut être fortement influencée par la gravité dans le cas d'évolution sur des sols en pente et en dévers. Ces forces additionnelles créent des glissements latéraux des roues et qui doivent être compensées par les braquages des roues et/ou les forces de traction. Le double braquage peut être mis à profit pour compenser ces forces et suivre un chemin de référence, y compris en présence de forte dynamique. Plusieurs travaux de recherche portent spécifiquement sur l'estimation de l'angle de dévers du véhicule (road bank angle). Par exemple, de nombreux chercheurs ont montré que l'estimation de l'angle de dévers est cruciale pour assurer la stabilité latérale du véhicule [Tseng \(2001\)](#), [Ungoren et al. \(2002\)](#) et [Nishio et al. \(2001\)](#), mais le roulis du véhicule induit par la déformation de la suspension a été négligé ou a été associé à l'angle d'inclinaison du terrain. Ungoren et al. ont développé dans [Ungoren et al. \(2004\)](#) une méthode d'estimation de la vitesse latérale d'un véhicule à deux trains directeurs en s'appuyant sur trois approches différentes : les deux premières approches (par la fonction de transfert et la représentation d'état) sont basées sur un modèle dynamique bicyclette et la dernière approche est basée sur un modèle cinématique. En revanche, la formulation de base des trois méthodes suppose que l'angle de dévers du terrain est négligeable, l'estimation de la vitesse latérale dépend de l'estimation de l'angle de dévers du terrain, ce qui, à son tour, dépend de l'estimation de la vitesse latérale. [Sasaki et Nishimaki \(2000\)](#) ont proposé une méthode basée sur un réseau de neurones artificiels pour l'estimation de l'angle de glissement latéral. Cette méthode prend en entrées/sorties (I/O) des relations reliant l'accélération latéral, l'angle de glissement et la vitesse de lacet. Néanmoins, en cas de modification des caractéristiques de frottement pneus/chaussée ou en cas de dévers du terrain, cette technique de réseau de neurones ne fonctionne pas de manière satisfaisante. Aussi, un filtre de Kalman classique a été utilisé dans [Lenain et Thuilot \(2012\)](#) et [Richier et al. \(2011\)](#) pour estimer l'angle de dévers, en s'appuyant sur les mesures de l'accélération et de la vitesse du roulis. Un filtre de Kalman étendu (EKF) a été utilisé dans [Sebsadji et al. \(2008\)](#) et [Sentouh et al. \(2007\)](#) pour estimer l'angle de glissement transversal, qui est utilisé par un observateur linéaire proportionnel intégral (PI) afin d'estimer l'angle de dévers. En plus de cela, un autre observateur a été développé dans [Ryu et Gerdes \(2004a\)](#) pour estimer l'angle de pente par le biais de la vitesse de lacet et de roulis, les angles de dérapage et l'angle d'inclinaison du véhicule (i.e., somme de l'angle de roulis et de dévers). D'autres méthodes utilisent des mesures GPS pour estimer l'angle de la pente et l'angle de dévers du véhicule (comme par exemple [Ryu et Gerdes \(2004b\)](#) et [Jo et al. \(2013\)](#)).

Dans cette section, on envisage d'améliorer les performances de nos contrôleurs de suivi de chemin. En plus des rigidités de dérive, ceux-ci dépendent aussi de la géométrie locale du terrain. Cette géométrie nécessite l'estimation de deux angles (voir Figure 2.4) : l'angle de pente noté θ_r et l'angle de dévers noté ϕ_r . Dans cette section, nous proposons un observateur destiné à estimer ces angles avec précision et en temps-réel, en tenant compte de la déformation générée par le système de suspension. Cet observateur est basé sur la vitesse latérale et angulaire ainsi que les mesures de la centrale inertielle (accélérations linéaires et vitesses angulaires).

3.2 Synthèse de l'observateur de la géométrie locale du terrain

Les angles de la géométrie locale du terrain influencent fortement la dynamique latérale du véhicule (2.23), notamment quand il se déplace sur une pente. Ces angles d'inclinaison nécessitent alors une estimation en temps-réel afin de compenser les effets dus à la gravité. Dans ce contexte, un observateur de Luneberger linéaire est synthétisé afin d'estimer ces angles de la géométrie locale du terrain (Road grade angle θ_r & road bank angle ϕ_r) (noté par la suite OAT) . Cet estimateur OAT est basé sur la vitesse de lacet V_ψ , la vitesse latérale V_y et les mesures de la centrale inertielle (e.g., accélérations linéaires et vitesses angulaires).

En général, les mesures de la centrale inertielle sont composées à la fois de l'accélération du robot et de celle liée à la gravité le long du terrain accidenté (cf. Figure 2.4). Ces accélérations peuvent être exprimées sous la forme suivante :

$$a_x = \dot{V}_x - V_y V_\psi - g \sin \theta_r, \quad (3.19a)$$

$$a_y = \dot{V}_y + V_x V_\psi + g \cos \theta_r \sin \phi_r, \quad (3.19b)$$

$$a_z = g \cos \theta_r \cos \phi_r, \quad (3.19c)$$

avec a_x et a_y sont l'accélération longitudinale et latérale du véhicule et g est l'intensité de pesanteur, θ_r et ϕ_r sont respectivement l'angle de pente et l'angle de dévers du véhicule selon la géométrie locale du terrain.

En utilisant l'équation (3.19) et en supposant que les variations de vitesse \dot{V}_x et \dot{V}_y sont négligeables, nous pouvons approximer ces angles par des mesures directes des accéléromètres.

$$\theta_r = \arcsin \frac{-a_x - V_y V_\psi}{g}, \quad (3.20a)$$

$$\phi_r = \arctan \frac{a_y - V_x V_\psi}{a_z}, \quad (3.20b)$$

Néanmoins, l'équation (3.20) peut conduire à un signal très bruité; d'où la nécessité de synthétiser un observateur pour mieux estimer ces angles d'inclinaison du véhicule selon la géométrie locale de l'environnement. Leurs taux de variation ($\dot{\theta}_r$ et $\dot{\phi}_r$) peuvent être exprimés en fonction des angles d'Euler et des vitesses angulaires par les équations suivantes (voir [Ryu et Gerdes \(2004a\)](#) pour plus de détails) :

$$\dot{\theta}_r = \cos \phi \omega_y - \sin \phi \omega_z = u_{r1}, \quad (3.21a)$$

$$\dot{\phi}_r = \cos \theta \omega_x + \sin \phi \sin \theta \omega_y + \cos \phi \sin \theta \omega_z = u_{r2}, \quad (3.21b)$$

Selon la théorie de Luenberger et après la discrétisation exacte du modèle (3.21), l'équation de l'observateur OAT peut être écrite ainsi,

$$\hat{\theta}_{r[k+1]} = \hat{\theta}_{r[k]} + T_r u_{r1} + L_1 \tilde{\theta}_{r[k]} = \hat{\theta}_{r[k]} + T_r u_{r1} + L_1(\theta_{r[k]} - \hat{\theta}_{r[k]}), \quad (3.22a)$$

$$\hat{\phi}_{r[k+1]} = \hat{\phi}_{r[k]} + T_r u_{r2} + L_2 \tilde{\phi}_r = \hat{\phi}_{r[k]} + T_r u_{r2} + L_2(\phi_{r[k]} - \hat{\phi}_{r[k]}), \quad (3.22b)$$

avec $\hat{\theta}_r$ et $\hat{\phi}_r$ sont les angles estimés de la géométrie locale du terrain et $\tilde{\theta}_r$ et $\tilde{\phi}_r$ sont les erreurs d'estimations de ces angles. T_r est la période d'échantillonnage de la centrale inertielle.

L'équation (3.22) peut être mise sous la forme matricielle suivante :

$$\hat{x}_{r[k+1]} = A_r \hat{x}_{r[k]} + B_r u_{r[k]} + L \tilde{x}_{r[k]}, \quad (3.23)$$

avec,

$$A_r = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B_r = \begin{bmatrix} T_r & 0 \\ 0 & T_r \end{bmatrix} \quad \text{et} \quad L = \begin{bmatrix} L_1 & 0 \\ 0 & L_2 \end{bmatrix},$$

$\hat{x}_{r[k]} = [\hat{\theta}_{r[k]} \hat{\phi}_{r[k]}]^T$ est l'état estimé, $u_{r[k]} = [u_{r1[k]} u_{r2[k]}]^T$ est la variable de contrôle, $\tilde{x}_{r[k]} = x_{r[k]} - \hat{x}_{r[k]}$ est l'erreur d'estimation, $x_{r[k]} = [\theta_{r[k]} \phi_{r[k]}]^T$ est l'état mesuré et L est la matrice de gain de l'observateur OAT.

3.3 Stabilité de l'observateur

On souhaite que, en régime permanent, l'erreur d'estimation soit nulle i.e.,

$$\lim_{k \rightarrow \infty} \tilde{x}_{r[k]} = \mathbb{O}_{2 \times 1}, \quad (3.24)$$

En utilisant (3.23), l'expression de $\tilde{x}_{r[k+1]}$ devient alors,

$$\tilde{x}_{r[k+1]} = (A_r - L) \tilde{x}_{r[k]}, \quad (3.25)$$

Pour que l'erreur d'estimation converge rapidement vers zéro en régime permanent, L doit être sélectionné de telle façon que toutes les valeurs propres de $A_r - L$ soient à l'intérieur du cercle unité et que la dynamique de l'observateur soit plus rapide que la commande de suivi de chemin (i.e, le régime transitoire de l'observateur sera plus rapide que celui de la commande LQR). Cet observateur est initialisé par la première mesure fournie par l'équation (3.20). Le temps de réponse de cet observateur OAT doit être également inférieur à celui de l'observateur d'état (observateur de Kalman-Bucy) et l'observateur non-linéaire (NLO) des rigidités de dérive (C_f et C_r). Par conséquent, les valeurs estimées $\hat{\theta}_r$ et $\hat{\phi}_r$ peuvent être également disponibles pour ces observateurs.

Nous procédons dans la section suivante à la validation de l'observateur OAT par des simulations sous ROS-GAZEBO à hautes vitesses. Les angles estimés seront injectés en ligne dans le bloc commande et dans l'observateur NLO afin d'améliorer les performances de nos algorithmes.

3.4 Validation par des simulations avancées

Des simulations numériques sont établies à l'aide de ROS-GAZEBO à différentes vitesses sur un terrain en pente de $\pm 15^\circ$ (cf. Figure 3.9). Nous comparons deux scénarios : le contrôleur LQR combiné avec l'observateur OAT (l'estimé de θ_r et ϕ_r seront introduits dans le contrôleur LQR) par rapport au contrôleur LQR sans l'observateur OAT ($\theta_r = \phi_r = 0^\circ$). L'observateur NLO des rigidités de dérive est utilisé dans toutes les simulations.

Le chemin désiré et les chemins enregistrés dans les deux scénarios (avec et sans OAT) à la vitesse longitudinale $V_x = 8m.s^{-1}$ sont illustrés par la Figure 3.10(a). Aussi, les déviations latérales et angulaires pour les deux scénarios sont représentées par les Figures 3.11(a) et (b). On peut remarquer que ces erreurs sont faibles dans le cas où les angles de la géométrie locale du terrain sont bien identifiés par l'OAT ($\pm 10cm$ avec OAT, contre $\pm 20cm$ sans OAT). Ainsi, la précision de suivi de chemin peut être améliorée lorsque les angles de la géométrie locale du terrain sont bien estimés et injectés en ligne dans le contrôleur.

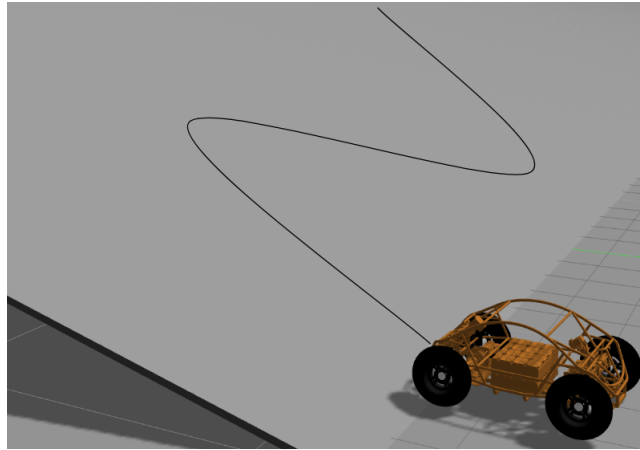


Figure 3.9. Environnement Gazebo avec un terrain accidenté d'une pente de $\pm 15^\circ$.

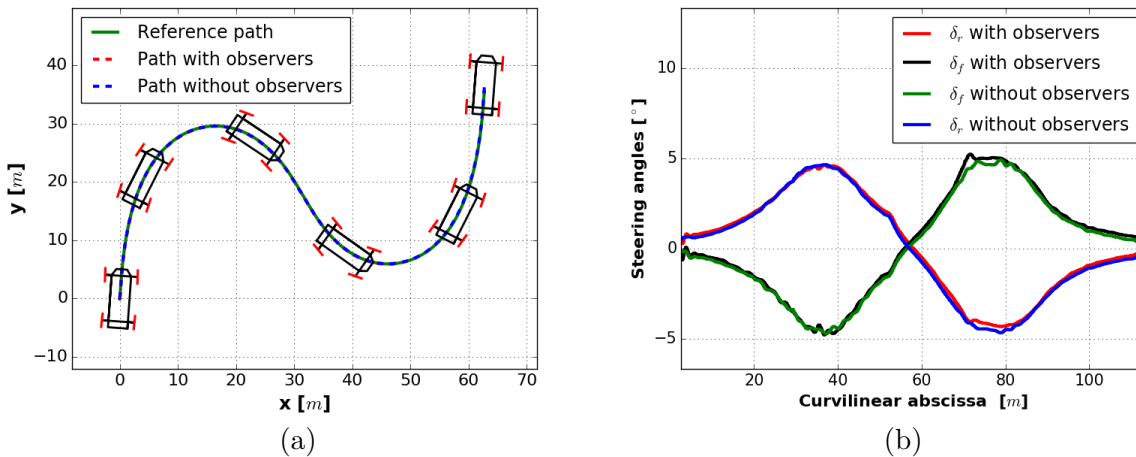


Figure 3.10. Résultats de simulation avec et sans OAT à $V_x = 8m.s^{-1}$: (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Les angles de braquage avant et arrière.

Les angles de braquage avant et arrière calculés par la commande LQR dans les deux scénarios (avec et sans OAT) à $V_x = 8m.s^{-1}$ sont tracés sur la Figure (3.10)(b). Ces angles restent toujours petits et varient doucement ($\pm 6^\circ$); ce qui montre l'efficacité de la commande mixée avec les estimateurs de paramètres du sol et sa capacité à tenir compte à la fois des exigences de la tâche et des contraintes liées à l'environnement.

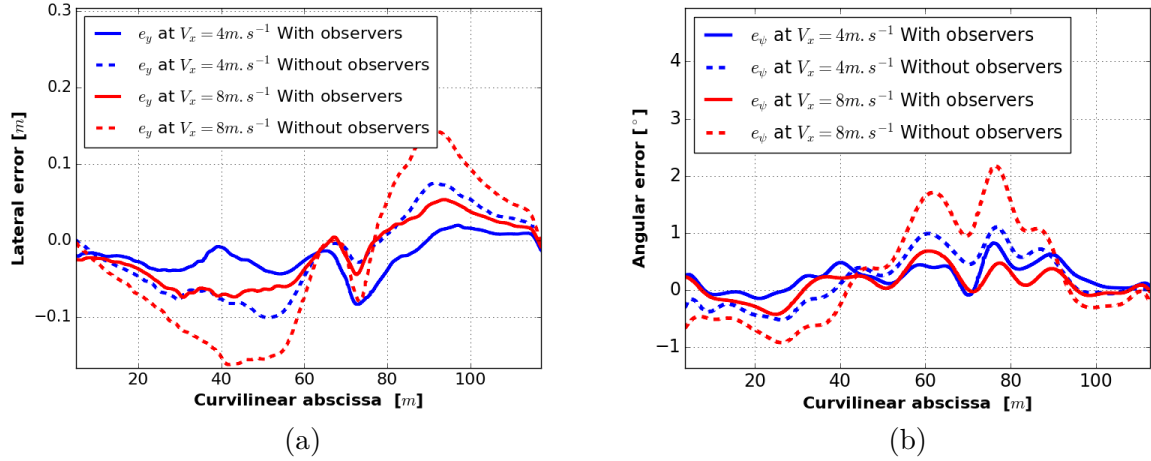


Figure 3.11. Résultats de simulation à $V_x = 4m.s^{-1}$ et à $V_x = 8m.s^{-1}$ pour les deux situations avec et sans le OAT : (a) Écart latéral en suivi de chemin; (b) Écart angulaire en suivi de chemin.

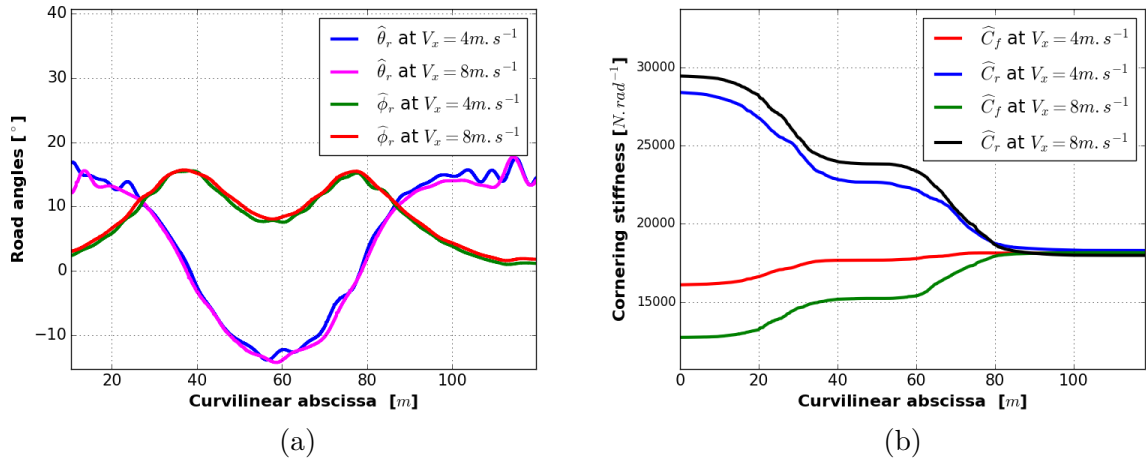


Figure 3.12. Paramètres estimés à $V_x = 4m.s^{-1}$ et à $V_x = 8m.s^{-1}$: (a) Angles estimés de la géométrie locale; (b) Rigidités de dérive avant et arrière.

Les angles de la géométrie locale de l'environnement (θ_r et ϕ_r) et les rigidités de dérive estimés aux vitesses longitudinales $V_x = 4m.s^{-1}$ et à $V_x = 8m.s^{-1}$ sont respectivement tracés dans les Figures 3.12(a) et (b). Comme prévu, ces angles estimés correspondent à la pente simulée dans Gazebo ($\pm 15^\circ$). En effet, les angles estimés prennent en compte le cap du robot sur la pente et la déformation générée par le système de suspension. Aussi, les coefficients de rigidités de dérive avant et arrière C_f et C_r sont estimées indépendamment et convergent vers la même valeur en régime permanent ($C_f \approx C_r \approx 18\,000 N.rad^{-1}$) qui semble refléter les conditions réelles du contact roue/sol.

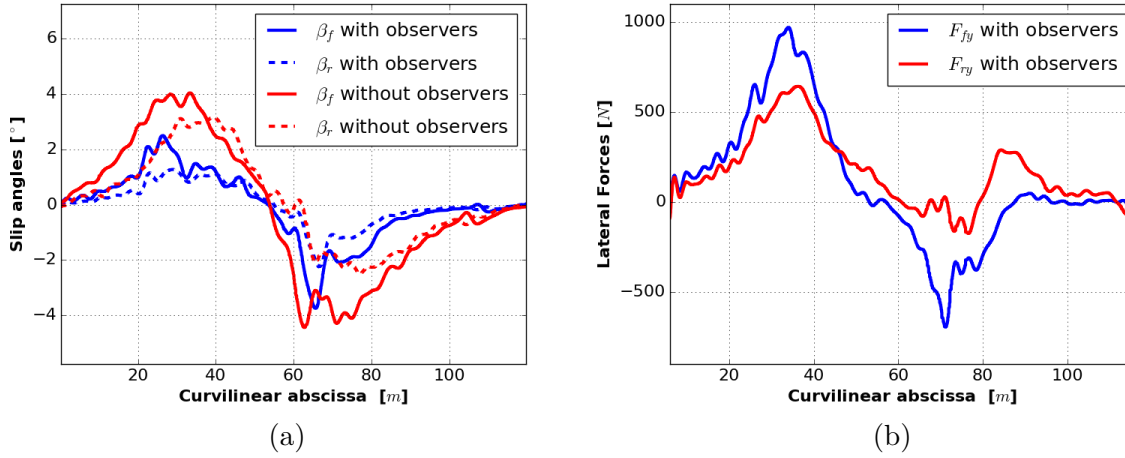


Figure 3.13. Résultats de simulation à $V_x = 8m.s^{-1}$: (a) Angles de glissement avant et arrière ; (b) Efforts latéraux avant et arrière.

Les forces latérales estimées à la vitesse longitudinale $V_x = 8m.s^{-1}$ sont montrées par la Figure 3.13(b). Elles sont importantes le long des grandes courbures où l'accélération latérale est élevée. Les angles de glissement latéraux avant et arrière à $V_x = 8m.s^{-1}$ sont illustrés par la Figure 3.13(a). En effet, ces angles de glissement sont faibles dans le cas où l'observateur OAT estime en temps réel les angles de la géométrie locale du terrain (4° à abscisse curviligne 40m sans OAT, contre 2° avec OAT). Par conséquent, la stabilité latérale du véhicule à double braquage peut être améliorée grâce à la bonne estimation de paramètres roue/sol et la géométrie locale du terrain.

4 Conclusion

Dans ce chapitre, nous avons présenté deux techniques d'observation des paramètres du contact roue/sol et de la géométrie locale du terrain. Dans un premier temps, un nouvel observateur non linéaire (noté NLO), basé sur un modèle dynamique, a été développé pour reconstruire indépendamment les rigidités de dérives avant et arrière d'une façon précise et quelles que soient les conditions d'adhérence. Ensuite, un observateur de Luenberger linéaire (noté OAT) a été mis en place pour estimer en temps réel les angles de la géométrie locale du terrain. Une fois ces paramètres estimés, ceux-ci sont injectés en ligne dans nos algorithmes de suivi de chemin, ce qui améliore la précision du suivi de chemin et la stabilité latérale du véhicule. Les résultats obtenus en simulations avancées et en vraie grandeur expérimentale ont permis de mettre en évidence la robustesse, la précision et la stabilité de ces méthodes d'observation. Ainsi, les performances de la commande LQR sont améliorées grâce à la bonne estimation des variables de l'environnement par l'intermédiaire de ces observateurs (NLO & OAT).

L'extension des travaux proposés dans ce chapitre concerne principalement la validation expérimentale de l'observateur OAT à l'aide du robot mobile tout-terrain rapide sur un sol composé de nombreuses pentes. En outre, pour montrer la robustesse et l'efficacité de ces observateurs (NLO et OAT), les prochains travaux vont porter également de les tester sur un terrain acci-

denté ayant différentes nature physique (e.g., herbe, sable, asphalte, etc.), en vue d'estimer les différentes rigidités de dérive des pneus et les angles de la géométrie locale du terrain associées à chaque condition du contact roue/sol en temps réel.

La commande LQR de suivi de chemin, qui a été synthétisée dans le chapitre 2 et utilisée pour valider nos observateurs développés dans ce chapitre, permet de faire tendre l'erreur vers zéro sans savoir ni l'état futur du système ni la consigne à suivre. En plus de cela, elle ne prend pas en compte aucune contrainte physique ou intrinsèque sur l'état et la commande pour assurer la tâche de suivi de chemin. Ces contraintes sont généralement inévitables et doivent être satisfaites à chaque instant. Pour cela, nous avons pensé à une autre technique de commande de la même famille mais avec un objectif différent. Cette technique est la commande prédictive sous contraintes qui fera l'objet du chapitre suivant et s'appuiera sur un autre critère d'optimisation.

Commande prédictive sous contraintes en suivi de chemin

Sommaire

1	Introduction	63
2	La commande prédictive basée modèle	63
2.1	Origines de la commande prédictive	63
2.2	Principe de la commande prédictive sous contraintes	64
2.3	Commande prédictive dans la robotique mobile	65
3	Stratégie de commande prédictive sous contraintes	67
3.1	Critère quadratique	67
3.2	Formulation des contraintes	70
3.2.1	Saturation des angles de braquage	70
3.2.2	Contraintes sur le taux de variation des angles de braquage	70
3.2.3	Adhérence pneumatique et contraintes de glissement	71
3.3	Problème final d'optimisation sous contraintes	74
4	Résultats de simulation avancée	75
4.1	Simulation sur le chemin en Z	75
4.2	Simulation sur le chemin en O	77
4.3	Interprétation des résultats	79
5	Résultats expérimentaux	80
6	Conclusion	82

1 Introduction

Comme on l'a vu dans le chapitre 2, la commande linéaire quadratique LQR permet de faire tendre l'erreur vers zéro sans savoir ni l'état futur du système ni la consigne future à suivre. Aussi, sur un plan expérimental, la loi de commande LQR s'est avéré assez oscillante. Cela est principalement dû aux retards dans la chaîne d'actionnement (communication ROS/contrôleur bas-niveau, variateurs) et aux temps de réponse des moteurs de direction. En outre, aucune contrainte intrinsèque sur l'état ou l'entrée n'est garantie par cette commande. Dans ce contexte, nous proposons dans ce chapitre une autre stratégie de commande de la même famille mais avec un objectif différent afin de résoudre ces impératifs. Cette stratégie est basée sur la commande prédictive (MPC : "Model Predictive Control") qui nous permet de pallier aux problèmes de retard en anticipant sur les consignes futures et de prendre en considération les contraintes importantes à satisfaire, la position de référence future du véhicule ainsi que celle prédite par le modèle et qui dépend de l'état présent.

2 La commande prédictive basée modèle

Depuis une vingtaine d'année, la stratégie de la commande prédictive est devenue une approche de plus en plus utilisée dans plusieurs secteurs d'applications. L'intérêt est que des spécifications de fonctionnement ainsi que des contraintes, du moins celles qui sont primordiales, puissent être conjointement traitées dans l'élaboration de la commande optimale.

2.1 Origines de la commande prédictive

La technique de la commande prédictive (MPC) est apparue au début des années 1960 avec les travaux de Propoi [Propoi \(1963\)](#). Dans ce travail, la commande MPC est basée sur un modèle en utilisant une méthode de programmation linéaire pour calculer une séquence de commandes sur un horizon de prédiction.

La première génération de la commande MPC a été connue dans le monde industriel par les travaux de Richalet et Testud [Richalet et al. \(1978\)](#) et [Testud et al. \(1978\)](#) dans le secteur de la pétrochimie sous le nom Identification et Commande (IDCOM) et par les travaux de Cutler et Ramaker sous le nom Dynamic Matrix Control (DMC) [Cutler et Ramaker \(1980\)](#). Dans ces approches, le principal objectif est d'atteindre une consigne sur un horizon de temps fini, les contraintes intrinsèques du système dynamique n'étant pas encore prises en compte. La deuxième génération de la commande MPC est apparue dans les années 1980 qui permet en plus d'intégrer des contraintes d'amplitude et de vitesse du procédé et de fournir des commandes compatibles avec les contraintes. Ce type de commande est généralement formulé sous la forme d'un problème quadratique (Quadratic Dynamic Matrix Control (QDMC)) [Garcia et Morshedi \(1986\)](#). Enfin, la génération actuelle des commandes MPC permet de distinguer divers degrés de contraintes (e.g., sous forme des inégalités, contraintes non linéaire, etc) et d'appliquer cette approche pour les systèmes dont la dynamique est relativement rapide [Künhe et al. \(2005\)](#), [Wieber \(2006\)](#), ainsi pour la robotique mobile [Krid et Benamar \(2011\)](#) et [Falcone et al. \(2007\)](#).

2.2 Principe de la commande prédictive sous contraintes

Les systèmes autonomes nécessitent de se projeter dans le futur pour mieux agir sur soi et anticiper les futurs événements. La notion de la commande MPC peut être assimilée à un être humain qui marche en regardant devant soi au lieu de regarder entre ses pieds pour pouvoir anticiper les genres d'actions pouvant améliorer son avancement et empêcher tout type de collisions. D'où la commande MPC dépend à la fois de l'état présent du système et aussi de son état futur prédit. Globalement, ce type de commande repose sur le principe d'un problème d'optimisation résolu à chaque itération de calcul. Ce problème est composé d'une fonction coût à optimiser tout en satisfaisant toutes les contraintes physiques et intrinsèques du système.

Le principe de la commande MPC est illustré dans la Figure 4.1. Celui-ci revient à optimiser la fonction coût à chaque temps discret t_k lorsque une nouvelle mesure est disponible, en prenant soin de satisfaire les contraintes de fonctionnement. Une séquence de N_c commandes est alors déterminée sur l'horizon de prédiction N_p . Finalement, seule la première composante de la séquence de commande est appliquée au système. Puis, lorsque une nouvelle mesure est disponible à t_{k+1} , l'état du système est mis à jour, et l'algorithme de résolution est à nouveau utilisé pour calculer une nouvelle séquence de N_c commandes à appliquer au système. Pour une introduction brève de la commande prédictive sous contraintes à temps continu ou discret, on peut se référer au [Morari \(1993\)](#), ainsi aux chapitre 2 & 7 de [Wang \(2009\)](#).

La mise en œuvre de ce type de stratégie requiert :

- Un modèle du comportement du procédé à commander qui permet d'anticiper les futurs états du système sur un horizon de prédiction fini.
- Une fonction coût qui définit mathématiquement les objectifs de la commande et qui peut être destinée à :
 - optimiser un critère de performance lié à des besoins économiques telles que l'augmentation du rendement énergétique, etc.
 - assurer la poursuite d'une trajectoire de référence en sortie du procédé qui permet de garantir des performances fonctionnelles telles que la stabilité, rapidité et précision.
- Des contraintes de fonctionnement à satisfaire sur les états et les entrées du système :
 - les actionneurs sont généralement limités en amplitude, mais peuvent l'être aussi en vitesse, en accélération ou en jerk.
 - certaines grandeurs du procédé, mesurées ou estimées, doivent rester dans une zone de fonctionnement, par exemple celles qui sont liées à :
 - + la sécurité du procédé (e.g., distance de sécurité entre le véhicule et l'obstacle).
 - + aux domaines de validité des hypothèses du modèle de procédé.
 - + le terrain et le milieu d'expérimentation.

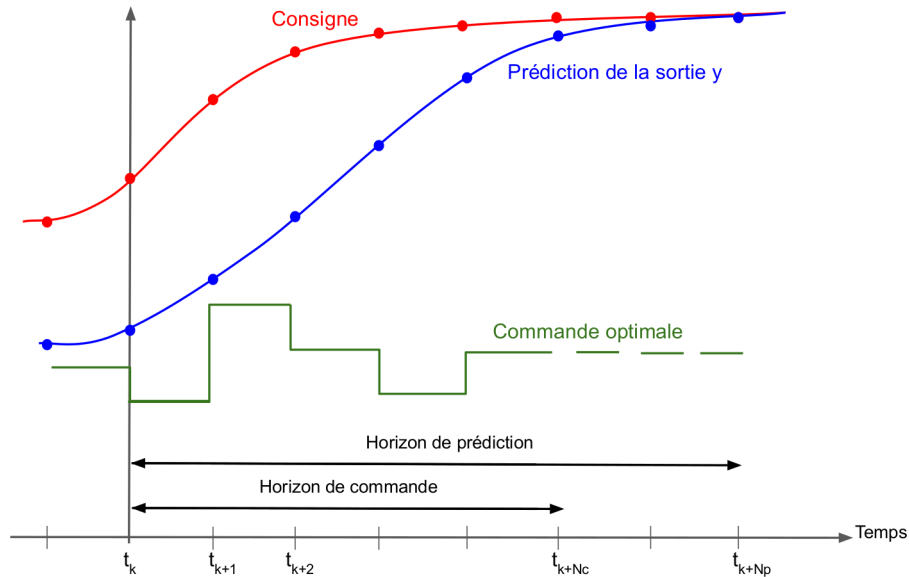


Figure 4.1. Principe de la commande prédictive : à chaque instant t_k , l'algorithme de résolution calcule la séquence de commande optimale U à appliquer au système jusqu'à l'instant t_{k+N_c} (représenté en vert). Pour cela, il se base sur le modèle du système qui permet de prédire la sortie y jusqu'à l'instant t_{k+N_p} (représenté en bleu) et sur la consigne y_d à atteindre (représenté en rouge).

2.3 Commande prédictive dans la robotique mobile

La commande prédictive (MPC) peut être mise en place sur des systèmes à une seule ou plusieurs variables, et peut être de type linéaire (MPC) ou non linéaire (NMPC). En général, ces contrôleurs MPC sont destinés à réaliser une tâche de suivi de chemin et sont basés soit sur un modèle cinématique soit un modèle cinématique étendu (i.e., les effets de glissement sont pris en compte) dans le repère de Frénet [Ollero et Amidi \(2004\)](#). Par exemple, [Lenain et al. \(2005\)](#) et [Lenain et al. \(2007\)](#) ont développé une loi de commande de suivi de chemin adaptative et prédictive en prenant en compte le glissement roue/sol, où la partie prédictive est principalement liée à l'anticipation du changement de la courbure du chemin de référence pour calculer des commandes qui sont à la fois lisses et moins violentes. Aussi, une commande MPC basée sur un modèle cinématique a été synthétisée dans [Lages et da Silva Jr \(2004\)](#). Celle-ci prend en compte certaines contraintes physiques du système (telles que les saturations de l'état et l'entrée). Par ailleurs, les commandes MPC basées sur la cinématique du véhicule sont plus pertinentes principalement pour les activités d'intérieure (telles que la logistique, l'industrie, etc.) et/ou lorsque le véhicule se déplace à faible vitesse. Il est à noter comme on l'a précédemment dit que l'efficacité des contrôleurs MPC cinématiques peut être considérablement endommagée si le glissement et la dynamique du véhicule deviennent extrêmement importants, notamment à grande vitesse et/ou pour des applications d'extérieur (agriculture, exploitation minière, etc.). Ainsi, des approches prédictives basées sur des modèles dynamiques non linéaires ont été abordées dans quelques travaux de recherche afin de surmonter les limites des contrôleurs cinématiques. Par exemple, une commande prédictive généralisée non linéaire à temps continu (NCGPC) est développée dans [Krid et al. \(2017\)](#), elle se base sur un modèle dynamique et a pour objectif d'assurer la tâche de suivi de trajectoire. Cependant, ce contrôleur ne prend en compte aucune contrainte

du système. Le principal avantage de la commande MPC réside dans sa capacité à prendre en compte à la fois la dynamique actuelle, future (l'état prédit) et éventuellement les contraintes qui sont à la fois critiques, inévitables et nécessaires pour la sécurité et la stabilité du véhicule. Par exemple, une commande prédictive non linéaire NMPC sous contraintes est développée dans [Van Essen et Nijmeijer \(2001\)](#) pour la stabilisation d'un robot mobile, celle-ci est soumise à des contraintes liées aux saturations de différents actionneurs du robot. Aussi, des commandes NMPC et MPC (linéarisation exacte en ligne de modèle du véhicule) sont conçues et comparées dans [Keviczky et al. \(2006\)](#), [Borrelli et al. \(2005\)](#) et [Falcone et al. \(2007\)](#) destinées à contrôler l'angle de direction des roues d'avant du véhicule sur des chaussées glissantes. Elles sont développées essentiellement pour stabiliser le véhicule le long d'une trajectoire de référence tout en respectant ses contraintes physiques (telles que les bornes de l'angle de braquage avant et sa vitesse de variation). Cependant, les commandes non linéaires utilisent des solveurs d'optimisation non linéaires tel que SQP (Sequential quadratic programming) qui ne garantissent pas la convergence de la résolution en temps fini, et restent en général contenues à des calculs en simulation. Deux approches MPC ont été présentées dans [Gao et al. \(2010\)](#), [Gray et al. \(2012\)](#) et [Qian et al. \(2016\)](#) pour les véhicules terrestres autonomes : la commande haut-niveau est utilisée pour la perception de l'environnement qui entoure le véhicule puis la détection et l'évitement d'obstacles en prenant en compte diverses contraintes, tandis que la commande de bas-niveau est utilisée pour permettre au robot de suivre le chemin planifié en contrôlant l'angle de braquage des roues avant.

Dans ce chapitre, nous présentons une méthode de commande MPC sous contraintes qui minimise un critère quadratique basé sur la prédiction de l'erreur entre la sortie du système et la référence. Nous allons appliquer cette commande sur des chemins aux virages rapides qui peuvent présenter des phénomènes de dérapage et à d'autres situations telles que le déplacement sur une pente. Le problème de contrôle est exprimé sous la forme d'un problème quadratique sous contraintes (QP) [Boyd et Vandenberghe \(2004\)](#) permettant de calculer les angles de braquage optimaux et respecter à chaque instant les contraintes de direction et d'adhérence roue/sol. Enfin, on valide cette étude par des simulations réalisées sous ROS-GAZEBO, ainsi que par des tests réels sur le robot mobile "*SPIDO*".

3 Stratégie de commande prédictive sous contraintes

Dans cette partie, nous allons concevoir une commande MPC pour assurer la tâche de suivi de chemin tout en respectant, à chaque pas de calcul, plusieurs contraintes physiques et intrinsèques du véhicule qu'on résume ci-dessous :

- Respecter les limites des actionneurs de direction (i.e., les angles de braquage avant et arrière doivent être limités pour respecter les butées articulaires) ;
- Éviter les grandes variations de braquage pour assurer la stabilité du véhicule, pour cela les vitesses des actionneurs de direction doivent également être limitées.
- Rester conforme à la zone d'adhérence du pneu en maintenant les angles de glissement entre deux valeurs données (cf. Figure 4.2) ;

Nous décrivons dans cette section une stratégie de commande prédictive sous contraintes sous forme d'un problème d'optimisation. Le but principal est de permettre au robot mobile d'assurer la tâche de suivi de chemin avec le respect de toutes les contraintes de braquage et de glissement sur l'horizon de prédiction.

3.1 Critère quadratique

La dynamique latérale du véhicule est utilisée pour prédire le comportement du système sur un horizon de temps fini et pour calculer la commande qui minimise une certaine fonction coût. Pour élaborer le critère quadratique utilisé pour le calcul de la commande, on discrétise le modèle dynamique développé dans l'équation (2.23) à la période d'échantillonnage T_d ,

$$\begin{cases} X_{k+1} = \Phi X_k + \Gamma U_k, \\ y_k = C X_k, \end{cases} \quad (4.1)$$

avec $\Phi = I_{4 \times 4} + T_d A$ et $\Gamma = T_d B$ sont respectivement la matrice dynamique et la matrice commande du système échantillonné, I est la matrice identité, $X_k = \xi_k - \xi_{ss}$ et $U_k = u_k - u_{ss}$ sont les vecteurs d'état et commande, avec ξ_{ss} et u_{ss} sont les vecteurs stationnaires de la dynamique latérale du véhicule donnés par l'équation (2.23).

La commande prédictive s'appuie sur le principe de prédiction de la sortie y à l'instant k sur un horizon de temps $N_p \times T_d$. Ainsi, on note par y_d le chemin désiré défini sur l'horizon de prédiction N_p . Généralement, l'horizon de prédiction N_p est plus grand que l'horizon de la commande ou l'horizon de contrôle N_c ($N_c \leq N_p$). On définit $y_{k+i|k}$ et $U_{k+i|k}$ la sortie et la commande prédites à l'horizon $k+i$ et calculées à partir de l'instant k (voir Figure 4.1). L'objectif de la commande est la régulation de la sortie y autour du chemin de référence y_d sur l'horizon $[k, k + N_p]$. Une fois la commande calculée sur l'horizon N_c , la première valeur est appliquée au système.

En notant U l'argument d'optimisation, en un instant fixé, le critère quadratique $J(U)$ à minimiser se ramène au suivant :

$$J(U) = \sum_{i=0}^{N_p} \left\| y_{k+i|k} - y_{d, k+i|k} \right\|_{Q_n}^2 + \sum_{i=0}^{N_c} \left\| U_{k+i|k} \right\|_{R_n}^2, \quad (4.2)$$

avec $\| \cdot \|$ est la norme Euclidienne, N_p est l'horizon de prédiction et N_c l'horizon de prédiction de contrôle. Pour simplifier la notation, on note $q = 3 \times N_p$ and $r = 2 \times N_p$. Ainsi, $Q_n \in \mathbb{R}^{q \times q}$ et $R_n \in \mathbb{R}^{r \times r}$ sont les matrices de pondération.

Pour simplifier l'expression du critère quadratique J , on établit l'expression de la sortie prédite $y_{k+i|k}$ en s'appuyant sur la dynamique du véhicule.

Ainsi, on peut écrire la sortie estimée $y_{k+1|k}$ sous la forme :

$$\begin{aligned} y_{k+1|k} &= CX_{k+1}, \\ &= C(\Phi X_k + \Gamma U_k), \\ &= C\Phi X_k + C\Gamma U_k, \end{aligned} \quad (4.3)$$

De même on déduit $y_{k+2|k}$ qu'on écrit sous la même forme de $y_{k+1|k}$.

$$\begin{aligned} y_{k+2|k} &= CX_{k+2}, \\ &= C\Phi X_{k+1} + C\Gamma U_{k+1}, \\ &= C\Phi(\Phi X_k + \Gamma U_k) + C\Gamma U_{k+1}, \\ &= C\Phi^2 X_k + C\Phi\Gamma U_k + C\Gamma U_{k+1}, \end{aligned} \quad (4.4)$$

De même jusqu'à l'ordre N_c ,

$$\begin{aligned} y_{k+N_c|k} &= CX_{k+N_c}, \\ &= C\Phi^{N_c} X_k + \sum_{i=0}^{N_c-1} C\Phi^i \Gamma U_{k+N_c-1-i}, \end{aligned} \quad (4.5)$$

On note que $\forall i \geq N_c$, on a $U_{k+i} = U_{k+N_c}$.

On continue la prédiction de y jusqu'à l'instant $k + N_p$,

$$\begin{aligned} y_{k+N_p|k} &= CX_{k+N_p}, \\ &= C\Phi^{N_p} X_k + \sum_{i=0}^{N_p-1} C\Phi^i \Gamma U_{k+N_p-1-i}, \end{aligned} \quad (4.6)$$

Enfin, on peut mettre les équations des sorties estimées y sur l'horizon de prédiction N_p sous la forme matricielle suivante :

$$Y = P_x X_k + P_u U, \quad (4.7)$$

avec :

$$Y = \begin{bmatrix} y_{k+1|k} \\ y_{k+2|k} \\ \vdots \\ y_{k+N_p|k} \end{bmatrix}, \quad P_x = \begin{bmatrix} C\Phi \\ C\Phi^2 \\ \vdots \\ C\Phi^{N_p} \end{bmatrix}, \quad U = \begin{bmatrix} U_k \\ U_{k+1} \\ \vdots \\ U_{k+N_p-1} \end{bmatrix},$$

$$P_u = \begin{bmatrix} C\Gamma & 0 & \dots & 0 \\ C\Phi\Gamma & C\Gamma & \dots & 0 \\ \vdots & & \ddots & \vdots \\ C\Phi^{N_p-1}\Gamma & \dots & C\Phi\Gamma & C\Gamma \end{bmatrix},$$

Afin de simplifier par la suite le calcul de la commande, reprenons le critère quadratique J et mettons le sous une forme matricielle,

$$J(U) = \|Y - Y_d\|_{Q_n}^2 + \|U\|_{R_n}^2, \quad (4.8)$$

avec Y_d est le chemin désiré défini sur l'horizon de prédiction N_p qu'on peut exprimer par le vecteur suivant,

$$Y_d = \begin{bmatrix} y_{d \ k+1|k} \\ y_{d \ k+2|k} \\ \vdots \\ y_{d \ k+N_p|k} \end{bmatrix},$$

En prenant en compte l'équation (4.7), le critère quadratique (4.8) peut s'écrire sous la forme,

$$J(U) = \|P_x X_k + P_u U - Y_d\|_{Q_n}^2 + \|U\|_{R_n}^2, \quad (4.9a)$$

$$= \frac{1}{2} U^T H U + f^T U + E^T Q_n E, \quad (4.9b)$$

avec H est une matrice Hessienne qui décrit la partie quadratique de la fonction coût à minimiser, le vecteur f décrit sa partie linéaire et $E^T Q_n E$ est indépendante de U et qui n'affecte pas le calcul de la solution optimale U^* . Elles sont exprimées comme suit,

$$H = 2(P_u^T Q_n P_u + R_n),$$

$$E = P_x X_k - Y_d,$$

$$f = 2P_u^T Q_n E,$$

3.2 Formulation des contraintes

Un des points avantageux de la commande prédictive est la possibilité de prendre en compte toutes les contraintes à respecter. Deux types de contraintes sont à distinguer ici, celles liées aux limites intrinsèques et physiques du véhicule (butées des angles de braquage et la limite de leurs taux de variation), et celle relatives aux conditions d'adhérences au niveau du contact roue/sol pour maintenir la contrôlabilité du véhicule (limites des angles de glissement).

3.2.1 Saturation des angles de braquage

D'un point de vue pratique, les angles de braquage avant et arrière doivent être limités entre deux bornes qui correspondent aux butées mécaniques de fin de course. Ces contraintes s'écrivent sous la forme suivante :

$$-\delta_f^{\max} \leq \delta_{f,k} \leq \delta_f^{\max}, \quad (4.10)$$

$$-\delta_r^{\max} \leq \delta_{r,k} \leq \delta_r^{\max}, \quad (4.11)$$

Une expression globale de ces contraintes sur l'horizon de prédiction N_p et en fonction de l'argument d'optimisation U est donnée par l'inégalité suivante :

$$\begin{bmatrix} I_{r \times r} \\ -I_{r \times r} \end{bmatrix} U \leq \begin{bmatrix} U^{\max} - \Lambda_{ss} \\ -U^{\min} + \Lambda_{ss} \end{bmatrix}, \quad (4.12)$$

avec $U^{\max} = -U^{\min} = [\delta_f^{\max}, \delta_r^{\max}, \dots, \delta_f^{\max}, \delta_r^{\max}]^T \in \mathbb{R}^{r \times 1}$ sont les bornes des angles de braquage et $\Lambda_{ss} = [u_{ss}, u_{ss}, \dots, u_{ss}]^T \in \mathbb{R}^{r \times 1}$ vecteur contenant les commandes stationnaires.

3.2.2 Contraintes sur le taux de variation des angles de braquage

L'actionneur de direction a une certaine vitesse maximale que la commande doit respecter à chaque pas de calcul. Par conséquent, le taux de changement des angles de braquage U doit être limité pour respecter les capacités des actionneurs. Par ailleurs, ceci permet également la génération de commandes lisses sans changement abrupts qui garantit le maintien de la stabilité latérale du véhicule. Par conséquent, le taux de changement des angles de braquage ΔU doit être limité pour avoir aussi des commandes plus lisses.

La variation de la commande U sur l'horizon de prédiction N_p se ramène au suivant :

$$\underbrace{\begin{bmatrix} \Delta U_k \\ \Delta U_{k+1} \\ \vdots \\ \Delta U_{k+N_p-1} \end{bmatrix}}_{\Delta U} = \underbrace{\begin{bmatrix} U_k \\ U_{k+1} \\ \vdots \\ U_{k+N_p-1} \end{bmatrix}}_U - \underbrace{\begin{bmatrix} U_{k-1} \\ U_k \\ \vdots \\ U_{k+N_p-2} \end{bmatrix}}_{\Omega}, \quad (4.13)$$

On peut récrire Ω sous la forme :

$$\Omega = \underbrace{\begin{bmatrix} U_{k-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_M + \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 \\ I_{2 \times 2} & 0 & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & I_{2 \times 2} & 0 \end{bmatrix}}_F \underbrace{\begin{bmatrix} U_k \\ U_{k+1} \\ \vdots \\ U_{k+N_p-1} \end{bmatrix}}_U, \quad (4.14)$$

En injectant l'équation (4.14) dans (4.13), on peut exprimer la variation des angles de braquage sur l'horizon de prédiction N_p comme suit,

$$\underbrace{\begin{bmatrix} \Delta U_k \\ \Delta U_{k+1} \\ \vdots \\ \Delta U_{k+N_p-1} \end{bmatrix}}_{\Delta U} = \underbrace{\begin{bmatrix} U_k \\ U_{k+1} \\ \vdots \\ U_{k+N_p-1} \end{bmatrix}}_U - \underbrace{\begin{bmatrix} U_{k-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_M - \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 \\ I_{2 \times 2} & 0 & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & I_{2 \times 2} & 0 \end{bmatrix}}_F \underbrace{\begin{bmatrix} U_k \\ U_{k+1} \\ \vdots \\ U_{k+N_p-1} \end{bmatrix}}_U, \quad (4.15)$$

Ainsi, la variation des angles de braquage en fonction de variable d'optimisation U peut se mettre sous la forme matricielle suivante :

$$\Delta U = (I_{r \times r} - F)U - M, \quad (4.16)$$

En prenant en compte (4.16), les contraintes de taux de changement des angles de braquage avant et arrière s'expriment sous la forme suivante :

$$\begin{bmatrix} I_{r \times r} - F \\ -I_{r \times r} + F \end{bmatrix} U \leq \begin{bmatrix} \Delta U^{\max} + M \\ -\Delta U^{\min} - M \end{bmatrix}, \quad (4.17)$$

avec $F \in \mathbb{R}^{r \times r}$ et $M \in \mathbb{R}^{r \times 1}$. ΔU^{\max} et ΔU^{\min} sont les bornes des vitesses autorisées sur les angles de direction définies le long de N_p .

3.2.3 Adhérence pneumatique et contraintes de glissement

La commande prédictive est basée sur un modèle linéaire du pneumatique où la force latérale $F_{y(f,r)}$ varie linéairement par rapport à l'angle de glissement $\beta_{(f,r)}$ (voir la figure 4.2). Cette hypothèse qui caractérise la zone de pseudo-glissement permet d'assurer de bonne condition d'adhérence entre le sol et le pneu. Lorsque la force sur le pneu dépasse une certaine valeur

en raison d'une grande force centrifuge (lors d'un virage par exemple), la zone de dérapage est atteinte, dans laquelle la force latérale atteint le maximum de frottement autorisé ($\pm\mu F_z$), ce qui provoque l'incontrôlabilité du véhicule. Pour éviter cet état incontrôlable, des contraintes caractérisant certaines limites des angles de glissement latéraux doivent être appliquées.

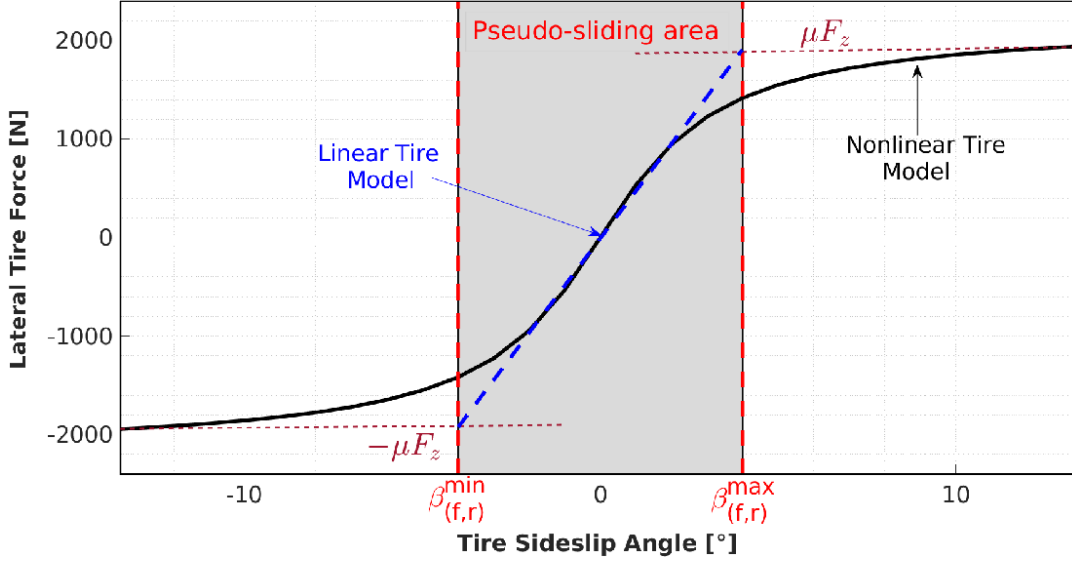


Figure 4.2. Comportement non linéaire du pneu, réduit à une zone de pseudo-glissement limitée par les bornes de glissement.

Pour maintenir la zone de pseudo-glissement, les angles de dérive $\beta_{(f,r)}$ doivent être limités entre certaines bornes inférieures et supérieures. Par conséquent, ces contraintes peuvent être déduites ainsi :

$$\beta_f^{min} \leq \beta_{f,k} \leq \beta_f^{max}, \quad (4.18)$$

$$\beta_r^{min} \leq \beta_{r,k} \leq \beta_r^{max}, \quad (4.19)$$

Nous rappelons que l'angle de dérapage $\beta_{(f,r)}$ est défini comme l'angle entre l'axe longitudinal de la roue et le vecteur vitesse linéaire du centre de la roue v_i (voir Figure (2.1)). Lorsque les angles de glissement sont faibles, ils peuvent être exprimés linéairement en fonction de la vitesse latérale et de lacet ainsi que les angles de braquage sous la forme suivante :

$$\beta_{f,k} = \frac{V_{y,k} + aV_{\psi,k}}{V_x} - \delta_{f,k}, \quad (4.20)$$

$$\beta_{r,k} = \frac{V_{y,k} - aV_{\psi,k}}{V_x} - \delta_{r,k}, \quad (4.21)$$

D'après le modèle discret (4.1), on réécrit (4.20) et (4.21) sous la forme matricielle,

$$\beta_k = \begin{bmatrix} \beta_{f,k} \\ \beta_{r,k} \end{bmatrix}, \quad (4.22a)$$

$$= T\xi_k + Ju_k, \quad (4.22b)$$

$$= T(X_k + \xi_{ss}) + J(U_k + u_{ss}), \quad (4.22c)$$

avec,

$$T = \begin{bmatrix} \frac{1}{V_x} & \frac{a}{V_x} & 0 & 0 \\ \frac{1}{V_x} & \frac{-b}{V_x} & 0 & 0 \end{bmatrix} \quad \text{et} \quad J = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix},$$

En utilisant l'équation (4.22c), les angles de glissement peuvent être alors exprimés sur l'horizon de prédiction N_p sous la forme suivante,

$$\underbrace{\begin{bmatrix} \beta_{k+1} \\ \beta_{k+2} \\ \vdots \\ \beta_{k+N_p} \end{bmatrix}}_{\bar{\beta} \in \mathbb{R}^{r \times 1}} = \underbrace{\begin{bmatrix} TX_{k+1} \\ TX_{k+2} \\ \vdots \\ TX_{k+N_p} \end{bmatrix}}_{\Xi} + \underbrace{\begin{bmatrix} JU_{k+1} \\ JU_{k+2} \\ \vdots \\ JU_{k+N_p} \end{bmatrix}}_{\Theta} + \underbrace{\begin{bmatrix} T\xi_{ss} + Ju_{ss} \\ T\xi_{ss} + Ju_{ss} \\ \vdots \\ T\xi_{ss} + Ju_{ss} \end{bmatrix}}_{\Pi}, \quad (4.23)$$

À partir de l'équation (4.1) et en supposant que les vecteurs stationnaires u_{ss} et ξ_{ss} sont constants le long de l'horizon de prédiction, nous pouvons étendre la récurrence Ξ comme suit :

$$\Xi = WX_k + ZU, \quad (4.24)$$

avec

$$W = \begin{bmatrix} T\Phi \\ T\Phi^2 \\ \vdots \\ T\Phi^{N_p} \end{bmatrix}, \quad Z = \begin{bmatrix} T\Gamma & 0 & \dots & 0 \\ T\Phi\Gamma & T\Gamma & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ T\Phi^{N_p-1}\Gamma & \dots & T\Phi\Gamma & T\Gamma \end{bmatrix},$$

En supposant que les commandes à l'horizon infini sont égales, i.e., $U_{k+N_p-1} \approx U_{k+N_p}$, nous pouvons également réécrire Θ sous la forme suivante :

$$\Theta = \Upsilon U, \quad (4.25)$$

avec,

$$\Upsilon = \begin{bmatrix} 0 & J & \dots & 0 & 0 \\ 0 & 0 & J & \dots & 0 \\ \vdots & \vdots & & \ddots & J \\ 0 & 0 & \dots & & J \end{bmatrix},$$

Par conséquent, les contraintes d'adhérence peuvent être exprimées en fonction de l'argument d'optimisation U en insérant les équations (4.24) et (4.25) dans (4.23).

$$\bar{\beta} = (Z + \Upsilon)U + WX_k + \Pi, \quad (4.26)$$

avec $W \in \mathbb{R}^{r \times 4}$, $Z \in \mathbb{R}^{r \times r}$, $\Upsilon \in \mathbb{R}^{r \times r}$ et $\Pi \in \mathbb{R}^{r \times 1}$.

A partir de l'équation (4.26), les contraintes sur les angles de glissement peuvent être reformulées sous la forme suivante :

$$\begin{bmatrix} Z + \Upsilon \\ -Z - \Upsilon \end{bmatrix} U \leq \begin{bmatrix} \bar{\beta}^{\max} - WX_k - \Pi \\ -\bar{\beta}^{\min} + WX_k + \Pi \end{bmatrix}, \quad (4.27)$$

avec $\bar{\beta}^{\max} = -\bar{\beta}^{\min} = [\beta_f^{\max}, \beta_r^{\max}, \dots, \beta_f^{\max}, \beta_r^{\max}]^T \in \mathbb{R}^{r \times 1}$ sont les bornes inférieure et supérieure de la zone de pseudo-glissement à définir judicieusement selon les conditions de frottement au contact roue/sol.

3.3 Problème final d'optimisation sous contraintes

Rappelons que dans le cadre de la stratégie de commande prédictive, l'algorithme doit être capable, à chaque sortie d'algorithme, de fournir des commandes situées dans le domaine des contraintes. Ainsi, les contraintes d'amplitude et de vitesse sur le braquage et le glissement doivent être vérifiées à chaque itération et sur tout l'horizon de prédiction.

Le contrôle du véhicule par notre approche nécessite de rechercher la commande optimale $U^* = [U_k^*, \dots, U_{k+N_p-1}^*]^T$ à chaque pas de calcul k qui satisfait à la fois toutes les contraintes et minimise la fonction objective. Ce problème peut être formulé sous forme d'un problème quadratique convexe (QP) qui est définie comme suit,

$$\begin{cases} U^* = \arg \min_U \frac{1}{2} U^T H U + f^T U + E^T Q_n E, \\ \text{s.t.} \quad G U \leq h, \end{cases} \quad (4.28)$$

avec G est la matrice globale regroupant toutes les contraintes préalablement définies dans (4.12), (4.17) et (4.27) et h est le vecteur regroupant toutes les limites associées à chacune de ces contraintes.

4 Résultats de simulation avancée

Les résultats présentés dans cette section sont issues des simulations numériques réalisées avec la maquette virtuelle sous ROS-GAZEBO de la Figure 1.4 (détaillé en annexe A). Le problème quadratique (QP) est implémenté en utilisant la librairie CVXOPT, un solveur open-source efficace et robuste pour résoudre le problème QP en temps-réel Andersen *et al.* (2013).

Pour valider la loi de commande prédictive sous contraintes, deux chemins de référence sont testés dans cette section aux vitesses longitudinales $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$. Le premier est le chemin en “Z” avec deux manœuvres agressives à droite et à gauche (cf. Figure 4.3(a)). Le deuxième est un chemin en “O” avec un virage à droite à faible rayon de courbure (cf. Figure 4.7(a)). Les deux chemins commencent par des lignes droites¹.

4.1 Simulation sur le chemin en Z

Dans un premier temps, nous testons notre algorithme d’optimisation sur un chemin en “Z” comprenant deux virages agressifs à gauche et à droite. Ce type de chemin permet d’exciter la dynamique latérale du véhicule et voir la capacité du problème QP à respecter les contraintes et assurer la tâche de suivi de chemin.

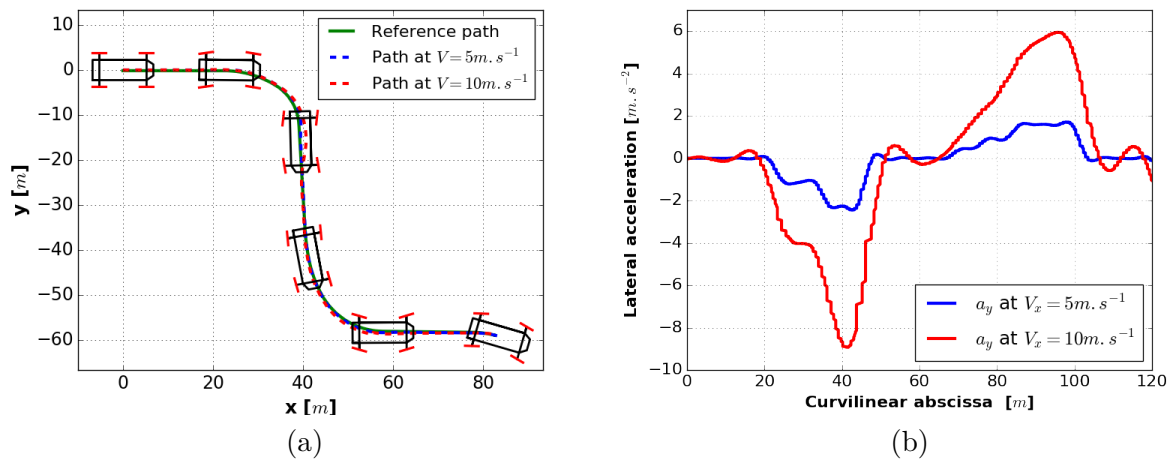


Figure 4.3. Résultats de simulation pour le chemin en “Z” à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Chemin de référence en “Z” et chemins réalisés dans le plan (x, y) ; (b) Accélérations latérales.

Pour ces simulations, nous définissons les paramètres suivants : période d’échantillonnage : $T_d = 0,2s$, contraintes relatives aux extrémums des angles de direction : $\pm 10^\circ$ et contraintes relatives aux extrémums de la variation de la commande entre deux pas de calcul consécutifs : $\pm 3^\circ$. Pour une vitesse longitudinale donnée V_x du véhicule, les paramètres à régler sont l’horizon de prédiction ($N_p = 20$ à $V_x = 5m.s^{-1}$ et $N_p = 40$ à $V_x = 10m.s^{-1}$) et les matrices de pondération Q_n et R_n .

1. Voir la vidéo suivante des simulations numériques de la commande MPC sous contraintes : [◇◇◇◇◇◇◇◇◇◇](#)

Le coefficient de frottement a été choisi constant ($\mu = 0,35$). Aussi, les rigidités de dérive avant et arrière C_f et C_r ont été supposées connues grâce à l'observateur non linéaire (NLO) proposé dans le chapitre 3. Ces deux variables convergent vers la même valeur constante ($C_{(f,r)} \approx 16\,000\text{ N.rad}^{-1}$). Par conséquent, les contraintes relatives aux extrémums sur les angles de glissement transversaux peuvent être approximées par l'intermédiaire de la force verticale F_z représentant le poids total du véhicule équitablement réparti entre les essieux avant et arrière, celles-ci peuvent être exprimées par la formule suivante :

$$\beta_{(f,r)}^{\max} = \frac{\mu F_z}{2C_{(f,r)}} = \frac{\mu mg}{2C_{(f,r)}} \approx 6^\circ, \quad (4.29)$$

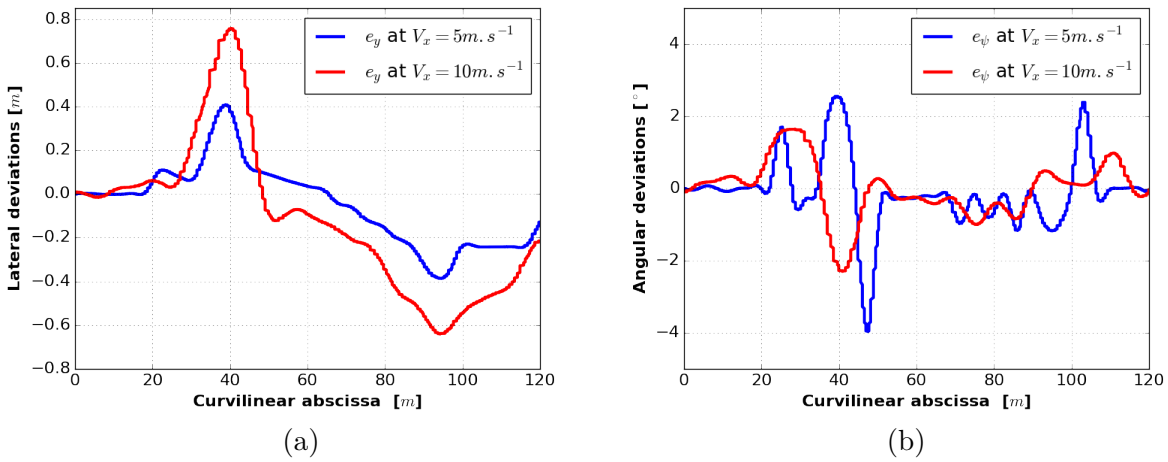


Figure 4.4. Résultats de simulation pour le chemin en “Z” à $V_x = 5\text{ m.s}^{-1}$ et $V_x = 10\text{ m.s}^{-1}$: (a) Déviations latérales; (b) Déviations angulaires.

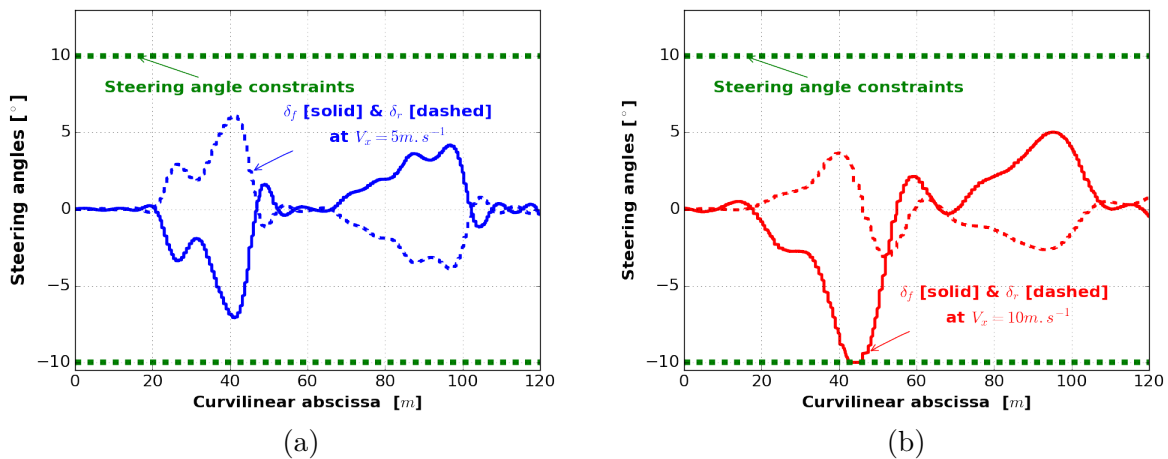


Figure 4.5. Résultats de simulation pour le chemin en “Z” : (a) Angles de braquage à $V_x = 5\text{ m.s}^{-1}$; (b) Angles de braquage à $V_x = 10\text{ m.s}^{-1}$.

Tous les tests de simulation ont été effectués sur un terrain en pente ($\pm 15^\circ$). Les angles de la géométrie locale du terrain (θ_r et ϕ_r) sont estimés en ligne par l'observateur linéaire de Luenberger (OAT) développé dans chapitre 3.

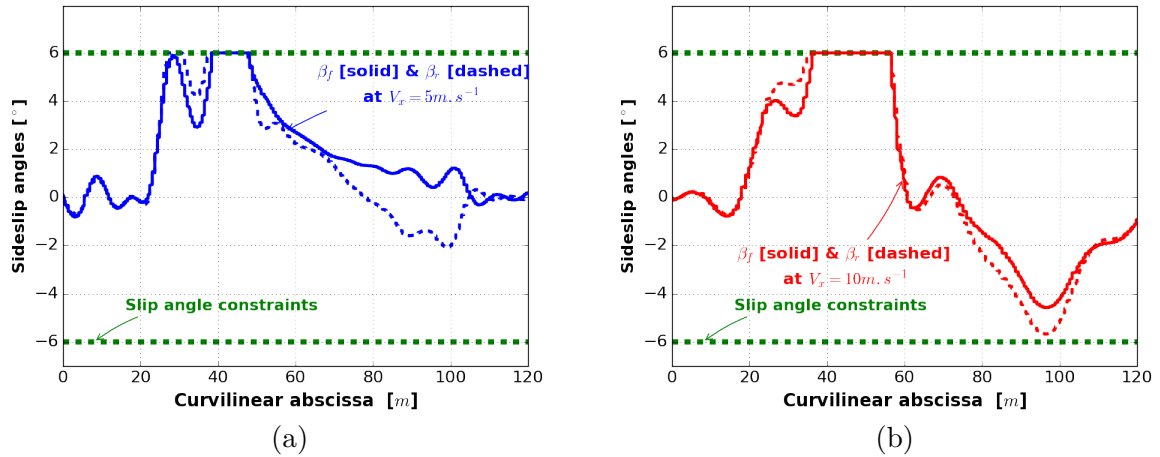


Figure 4.6. Résultats de simulation pour le chemin en “Z” : (a) Angles de glissement à $V_x = 5m.s^{-1}$; (b) Angles de glissement à $V_x = 10m.s^{-1}$.

4.2 Simulation sur le chemin en O

Pour montrer aussi l’apport de notre stratégie de suivi de chemin, nous la testons sur un deuxième chemin. Il s’agit d’un chemin en “O” comprenant un virage à droite à faible rayon de courbure difficilement atteignable par un véhicule ayant un seul essieu directeur.

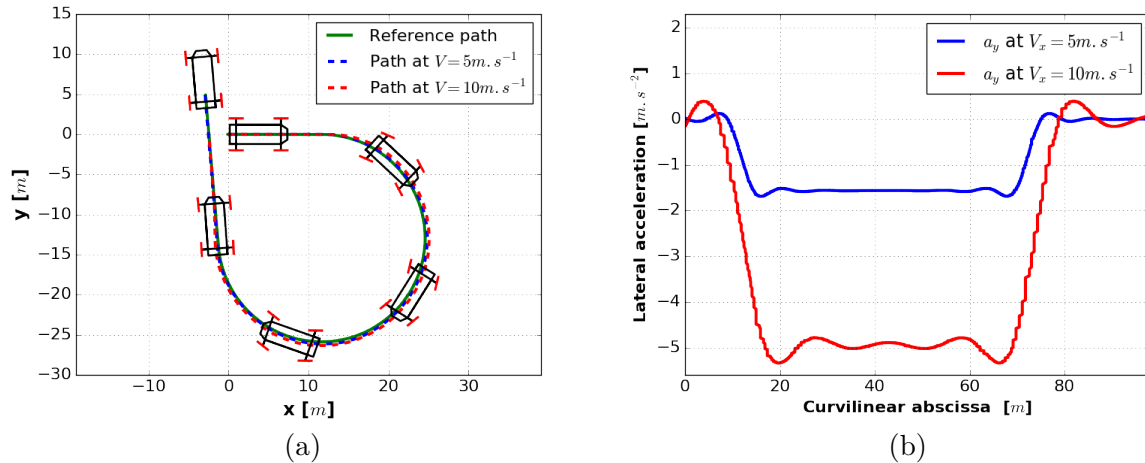


Figure 4.7. Résultats de simulation pour le chemin en “O” à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Chemin de référence en “O” et chemins réalisés dans le plan (x, y) ; (b) Accélérations latérales.

Pour ces simulations, nous définissons les paramètres suivants : période d’échantillonnage : $T_d = 0,2s$, contraintes relatives aux extrémums des angles de direction : $\pm 10^\circ$ et contraintes relatives aux extrémums de la variation de la commande entre deux pas de calcul consécutifs : $\pm 3^\circ$. Pour une vitesse longitudinale donnée V_x de véhicule, les paramètres à régler sont l’horizon de prédiction ($N_p = 20$ à $V_x = 5m.s^{-1}$ et $N_p = 40$ à $V_x = 10m.s^{-1}$) et les matrices de pondération Q_n et R_n .

Tous ces tests ont été réalisés sur un terrain en pente ($\pm 15^\circ$). Les angles de la géométrie

locale de terrain (θ_r et ϕ_r) sont estimés en temps-réel par l'observateur linéaire (OAT) proposé dans le chapitre 3.

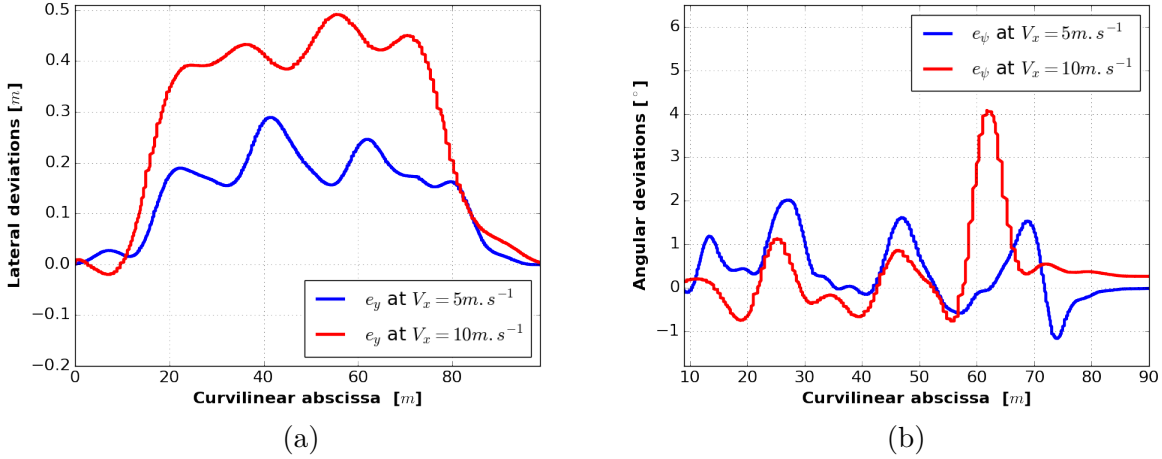


Figure 4.8. Résultats de simulation pour le chemin en “O” à $V_x = 5m.s^{-1}$ et $V_x = 10m.s^{-1}$: (a) Déviations latérales ; (b) Déviations angulaires.

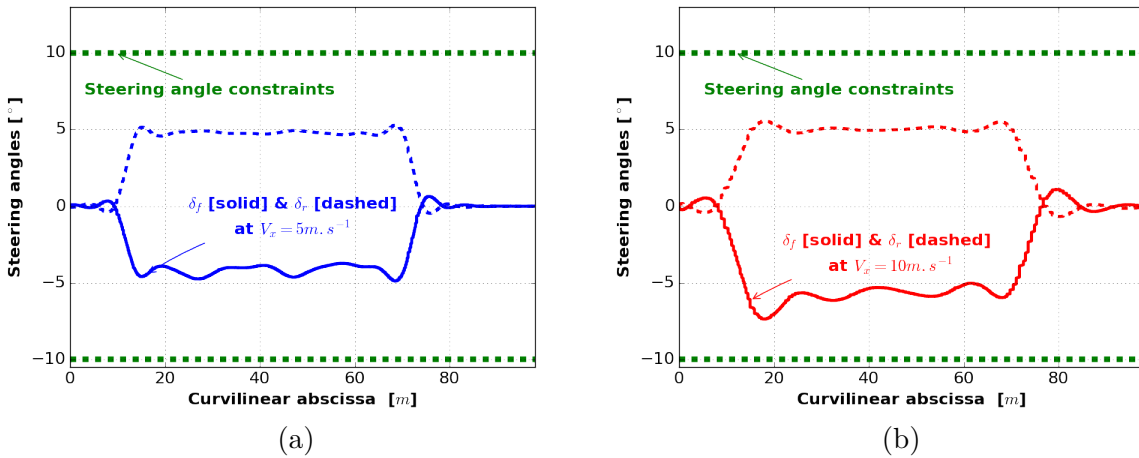


Figure 4.9. Résultats de simulation pour le chemin en “O” : (a) Angles de braquage à $V_x = 5m.s^{-1}$; (b) Angles de braquage à $V_x = 10m.s^{-1}$.

Le coefficient de frottement a été choisi constant ($\mu = 0,35$). Ainsi, les rigidités de dérive avant et arrière C_f et C_r ont été supposées connues grâce à l'observateur non linéaire (NLO) proposé dans le chapitre 3. Ces deux variables convergent vers la même valeur constante ($C_{(f,r)} \approx 16\ 000\ N.rad^{-1}$). Par conséquent, les contraintes relatives aux extrémums sur les angles de glissement transversaux peuvent être approximées par l'intermédiaire de la force verticale F_z représentant le poids total du véhicule équitablement réparti entre les essieux avant et arrière, celles-ci peuvent être exprimées par la formule suivante :

$$\beta_{(f,r)}^{\max} = \frac{\mu F_z}{2C_{(f,r)}} = \frac{\mu mg}{2C_{(f,r)}} \approx 6^\circ, \quad (4.30)$$

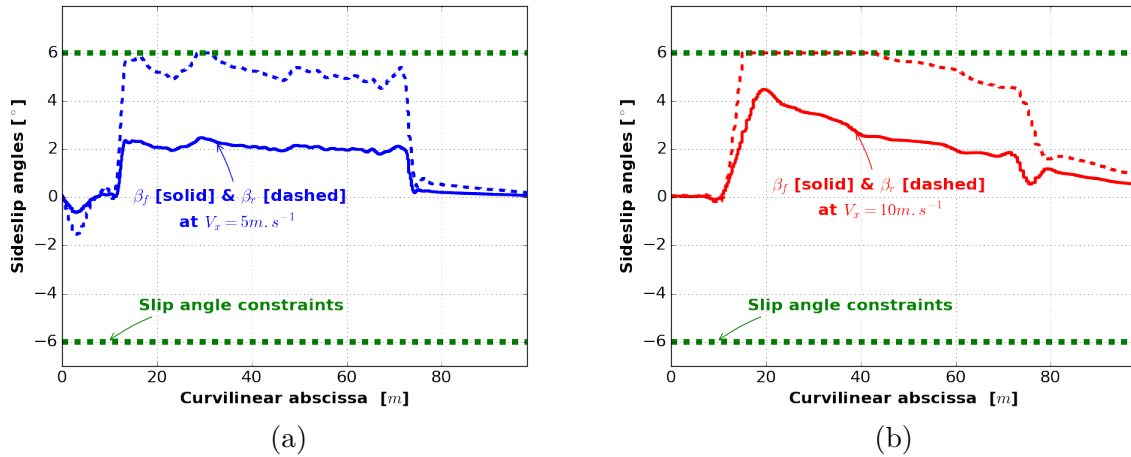


Figure 4.10. Résultats de simulation pour le chemin en “O” : (a) Angles de glissement à $V_x = 5 \text{ m.s}^{-1}$; (b) Angles de glissement à $V_x = 10 \text{ m.s}^{-1}$.

4.3 Interprétation des résultats

Les angles de braquage avant et arrière calculés par le contrôleur MPC aux vitesses longitudinales $V_x = 5 \text{ m.s}^{-1}$ et $V_x = 10 \text{ m.s}^{-1}$ sont illustrés respectivement sur les Figures 4.5 et 4.9 pour les chemins en “Z” & “O”. Comme on peut le constater pour les basses vitesses, ils sont lisses, assez petits et de signes opposés (cf. Figures 4.5(a) et 4.9(a)). Nous constatons aussi que la propriété des signes opposés des angles de braquage n’est plus conservée à hautes vitesses. Ceci est probablement dû aux forts glissements et l’existence d’un angle de glissement latéral au centre de gravité du véhicule qui est non négligeable qui casse la symétrie entre l’avant et l’arrière. Malgré ces fortes sollicitations, la commande garantit le maintien des angles de braquage dans l’intervalle fixé, ils sont toujours conservés entre les limites fixées ($\pm 10^\circ$) à chaque itération de calcul.

Rappelons que les angles de glissement latéraux avant et arrière sont calculés à l’aide de l’équation (4.20) et sont tracés sur les Figures 4.6 et 4.10 à des vitesses différentes. Comme on peut le constater, ces angles sont plus grands à faible rayon de courbure et ils sont toujours conservés entre les bornes souhaitées $\pm 6^\circ$ (limite de la zone d’adhérence pneumatique). En fait, les accélérations latérales sont importantes globalement le long de grands virages où le rayon de courbure est minimal, comme on peut l’observer sur les Figures 4.3(b) et 4.7(b) (cette accélération peut atteindre par exemple la valeur de 8 m.s^{-2} pour une vitesse $V_x = 10 \text{ m.s}^{-1}$).

Comme on peut le voir clairement sur les Figures 4.4 et 4.8, les erreurs de suivi de chemin latérales e_y et angulaires e_ψ sont assez importantes le long des grandes courbures et augmentent légèrement surtout lorsque les points extrêmes relatifs aux contraintes sont atteints (e.g., à l’abscisse curviligne 40 m , l’erreur latérale est de $0,7 \text{ m}$ pour le chemin en “Z” et $0,4 \text{ m}$ pour le chemin en “O” à $V_x = 10 \text{ m.s}^{-1}$). Cependant, la contrôlabilité et la stabilité latérale du véhicule sont assurées.

En conclusion, les résultats de la simulation numérique sont très satisfaisants. La commande MPC sous contraintes permet d’assurer un bon suivi de chemin tout en respectant à chaque

instant toutes les contraintes intrinsèques et physiques du véhicule. Cette loi de commande génère des entrées lisses et moins agressives avec des conditions d'adhérence roue/sol bien contrôlées. Cela joue un rôle primordial pour garantir la stabilité latérale et la sécurité du véhicule.

5 Résultats expérimentaux

Pour des raisons de sécurité et les petites dimensions du terrain d'essai, nous avons adopté des basses vitesses pour nos expériences réalisées sur le robot mobile "SPIDO" ($V_x = 2 \text{ m.s}^{-1}$ et $V_x = 3 \text{ m.s}^{-1}$). Néanmoins, pour avoir la même accélération latérale, nous avons choisi un chemin de référence à plus forte courbure et donc plus forte accélération latérale ($a_y = \rho V_x^2$).

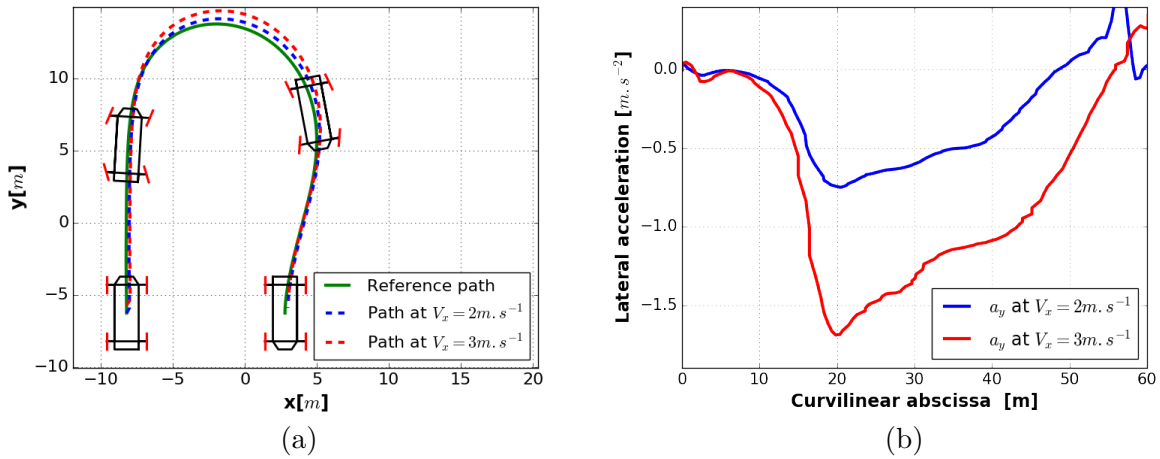


Figure 4.11. Résultats expérimentaux à $V_x = 2 \text{ m.s}^{-1}$ et $V_x = 3 \text{ m.s}^{-1}$: (a) Chemin de référence et chemins réalisés dans le plan (x, y) ; (b) Accélération latérale.

Nous définissons les paramètres suivants pour la partie expérimentale : Période d'échantillonnage : $T_d = 0, 2 \text{ s}$, contraintes relatives aux extrémums des angles de braquage : $\pm 10^\circ$, contraintes relatives aux extrémums des angles de glissement extrême : $\pm 6^\circ$, contraintes relatives aux extrémums de taux de changement d'angles de braquage : $\pm 3^\circ$ et un horizon de prédiction $N_p = 10$.

Le chemin de référence et les chemins réalisés pour $V_x = 2 \text{ m.s}^{-1}$ et $V_x = 3 \text{ m.s}^{-1}$ sont représentés sur la Figure 4.11(a). Comme on peut le constater, le véhicule suit bien le chemin désiré, mais quand la vitesse longitudinale croît le décalage entre le chemin réalisé et le chemin de référence augmente notamment le long du virage à cause de la grande accélération latérale qui varie proportionnellement au carré de cette vitesse longitudinale et à la courbure du chemin ($a_y = \rho V_x^2$). Cela pourrait être constaté clairement sur la Figure 4.11(b) où l'accélération latérale peut atteindre la valeur 2 m.s^{-2} à $V_x = 3 \text{ m.s}^{-1}$.

Les erreurs de suivi de chemin sont principalement importantes pour les fortes courbures du chemin comme on peut le voir sur la Figure 4.12. À l'abscisse curviligne 40 m , l'erreur latérale e_y atteint la valeur 1.1 m et l'erreur angulaire e_ψ atteint la valeur 6° à $V_x = 3 \text{ m.s}^{-1}$. Cette précision peut être améliorée dans nos futurs travaux par la fusion des données entre le GPS-RTK et l'odométrie.

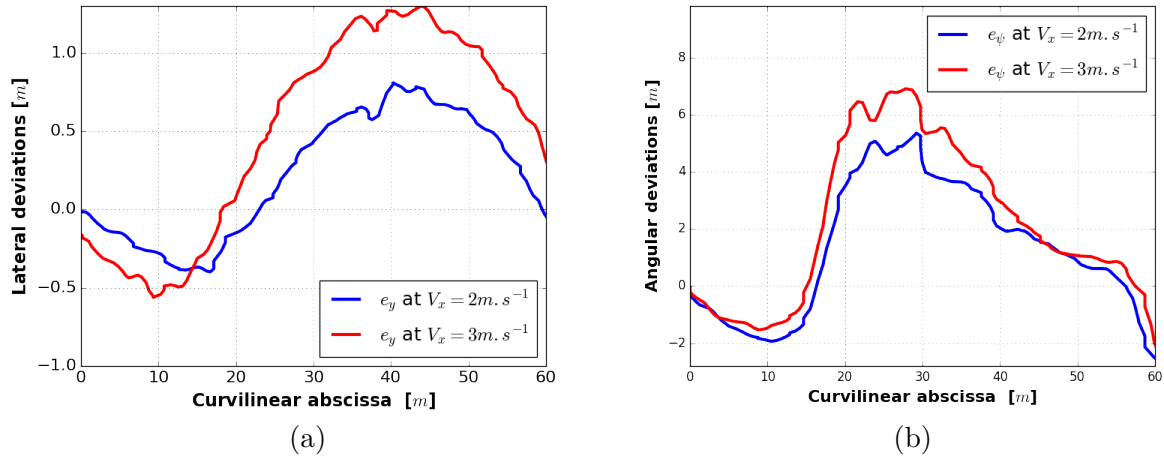


Figure 4.12. Résultats expérimentaux à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$: (a) Déviations latérales ; (b) Déviations angulaires.

En outre, les angles de braquage calculés par la commande MPC sous contraintes à $V_x = 2m.s^{-1}$ et $V_x = 3m.s^{-1}$ sont tracés dans les Figures 4.13(a) et (b). Ces angles sont toujours maintenus entre les bornes de braquages $\pm 10^\circ$. Ils sont de signes opposés et varient d'une façon lisse et moins agressive. D'où l'intérêt de la commande MPC sous contraintes à base d'un modèle dynamique qui prends en compte toutes les saturations physiques du véhicule².

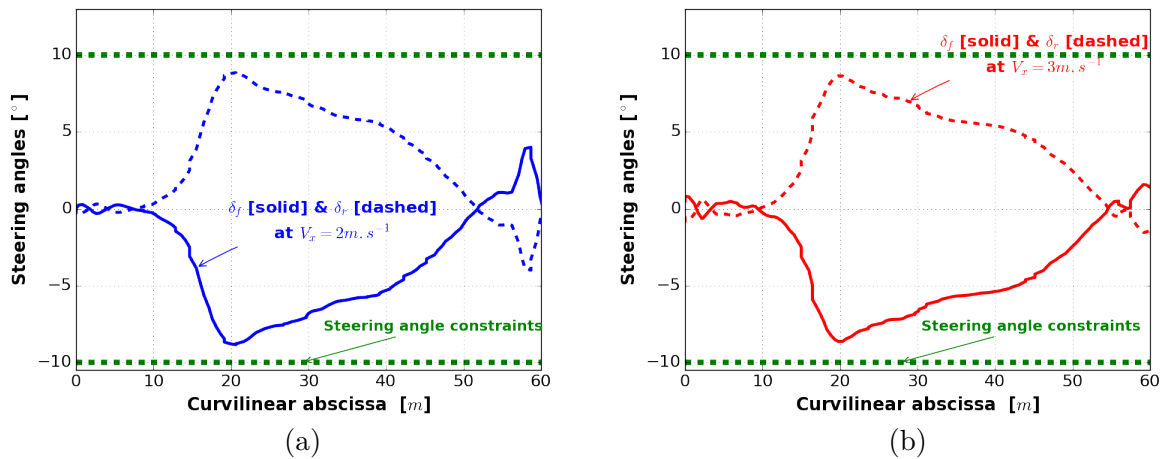


Figure 4.13. Résultats expérimentaux : (a) Angles de braquage à $V_x = 2m.s^{-1}$; (b) Angles de braquage à $V_x = 3m.s^{-1}$.

Le phénomène de glissement prédomine principalement durant le grand virage comme montré dans la Figure 4.14. Grâce à la commande MPC sous contraintes, les angles de glissement sont toujours limités entre les bornes de la zone de pseudo-glissement $\pm 6^\circ$. Par conséquent, la stabilité latérale et la contrôlabilité du véhicule peuvent être garanties à chaque instant par la commande MPC sous contraintes.

2. Voir la vidéo suivante des expérimentations de la commande MPC sous contraintes : [▶▶▶▶▶▶](#)

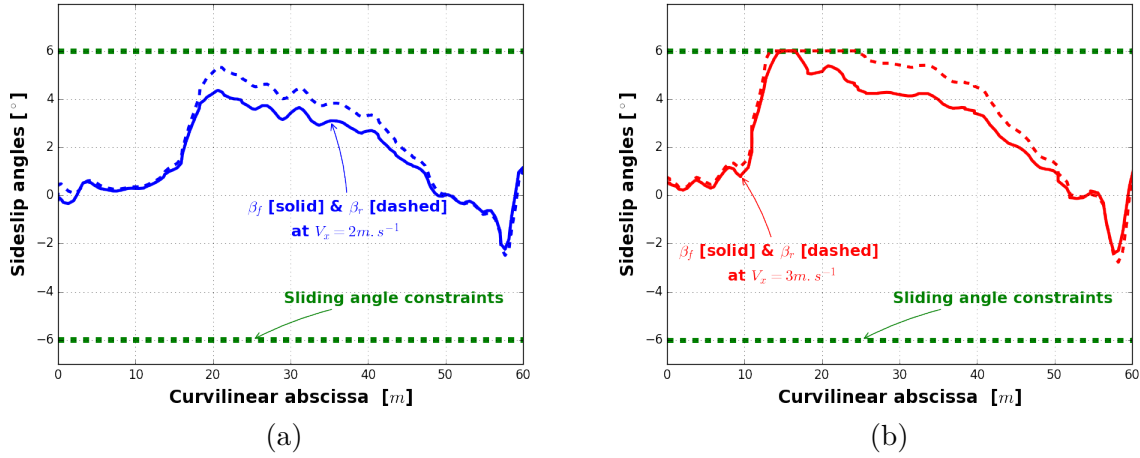


Figure 4.14. Résultats expérimentaux : (a) Angles de glissement à $V_x = 2 \text{ m.s}^{-1}$; (b) Angles de glissement à $V_x = 3 \text{ m.s}^{-1}$.

En résumé, les résultats fournis par notre commande MPC sous contraintes sont tout à fait satisfaisants. Ce contrôleur permet au véhicule de suivre différents types de chemins souhaités avec des angles de direction lisses et moins perturbants, tout en respectant aussi toutes les contraintes mécaniques et d'adhérence roue/sol. Les déviations latérales et angulaires de suivi de chemin deviennent légèrement plus élevées une fois que ces contraintes sont atteintes. Cela pourrait être amélioré en intégrant dans notre contrôleur (MPC sous contraintes) une autre commande de plus pour la vitesse longitudinale V_x , celle-ci doit être asservie et régulée en fonction de la courbure de chemin de référence.

6 Conclusion

Dans ce chapitre, nous avons développé un nouveau contrôleur de suivi de chemin (MPC sous contraintes) basé sur un modèle dynamique du véhicule, celui-ci est dédié aux robots mobiles à deux trains directeurs (4WS) qui pourrait être facilement appliqué à des véhicules à simple braquage moyennant quelques petites adaptations. Cette commande prend en compte les différentes contraintes de direction et d'adhérence au niveau de contact roue/sol pour assurer la sécurité et la stabilité latérale du véhicule. La stratégie de la commande prédictive sous contraintes est formulée dans la section 3 sous la forme d'un problème quadratique sous contraintes (QP) afin de trouver les angles de braquage avant et arrière optimaux nécessaires à la réalisation de la tâche du suivi de chemin.

À travers des simulations numériques avancées et des expériences sur la plateforme réelle, nous avons implémenté notre algorithme en utilisant le solveur CVXOPT afin de résoudre en temps réel le problème QP. Il a été prouvé à travers les résultats obtenus que le contrôleur proposé est efficace et permet au véhicule de suivre n'importe quel chemin désiré à différentes vitesses avec des angles de braquage lisses et moins agressifs et dans différentes conditions de travail (e.g., déplacement sur les pentes) avec une bonne précision qui diminue légèrement une fois les contraintes atteintes.

Les travaux en cours et futurs de ce chapitre portent essentiellement sur :

- L'approximation en ligne du coefficient de frottement $\mu \in [0, 1]$ lié au contact pneu/sol. Cette approximation permettra par la suite le calcul en ligne des limites relatives aux extrémums des angles de glissement latéraux avant et arrière ($\beta_{(f,r)}^{max} \approx \frac{\mu mg}{2C_{(f,r)}}$).
- Le contrôle de la dynamique longitudinale (contrôle du V_x) sera intégré au contrôleur MPC sous contraintes afin d'améliorer la précision et la convergence de nos algorithmes principalement le long des virages où le rayon du courbure est minimal. Cette vitesse longitudinale doit toujours remplir cette inégalité qui dépend surtout de la courbure du chemin désiré $V_x \leq \sqrt{\frac{\mu g}{\rho}}$.

Dans le prochain chapitre, nous allons présenter une nouvelle stratégie de commande haut-niveau qui a pour objectif de planifier localement un chemin de référence pour assurer la tâche d'évitement d'obstacle. Ce chemin planifié sera suivi par la commande bas-niveau en utilisant la commande MPC sous contraintes qui a été synthétisée dans ce chapitre.

Planification locale du chemin de contournement pour l'évitement d'obstacles

Sommaire

1	Introduction	85
2	État de l'art sur la planification de chemin pour l'évitement d'obstacles	85
3	Aperçu sur les courbes de Bézier	88
4	Détection d'obstacles	89
4.1	Introduction	89
4.2	Acquisition du nuage de points	89
4.3	Sous-échantillonnage des données	90
4.4	Algorithme de détection d'obstacles	91
4.5	Validation sous ROS-GAZEBO	94
4.6	Validation expérimentale	95
5	Planification locale du chemin pour l'évitement d'obstacle	96
5.1	Contrôleur haut-niveau	96
5.1.1	Distance de sécurité	96
5.1.2	Approche d'évitement d'obstacles via des courbes de Bézier	96
5.1.3	Extraction du point final et intermédiaire	97
5.1.4	Génération des courbes de Bézier cubiques	98
5.1.5	Sélection de la meilleure courbe de chaque courbe de Bézier	100
5.2	Résultats de simulations numériques	103
5.2.1	Chemin de référence rectiligne	104
5.2.2	Chemin en O	105
5.2.3	Chemin en S	106
5.2.4	Chemin en Z	107
5.2.5	Discussions des résultats de simulation	108
5.3	Résultats expérimentaux	109
6	Conclusion	112

1 Introduction

Une navigation autonome d'un robot mobile en milieu ouvert et non maîtrisé requiert (1) des techniques de perception de l'environnement afin de détecter d'éventuels obstacles fixes ou mobiles initialement non prévus, voire des humains (opérateurs, piétons) et (2) des algorithmes temps réel de génération de trajectoires qui évitent les obstacles qui se trouvent sur le chemin initialement prévu et de rallier ce chemin dès que possible.

Dans le présent chapitre, nous présenterons brièvement les principales approches réactives de planification locale de chemin destinées à l'évitement d'obstacles dans le domaine de la robotique mobile. À travers cette synthèse on tentera de dresser les limites de ces approches qui nous permettra de proposer une nouvelle méthode de génération de trajectoires compatibles avec les spécifications des applications robotiques en milieu ouvert et à grande vitesse. Avant de présenter cette partie de planification locale de chemin, on présentera le module de détection d'obstacles basé sur le traitement de nuage de points délivré par un capteur LiDAR 3D. Ensuite, nous allons présenter une nouvelle méthode de navigation destinée aux véhicules autonomes, qui a pour objectif de planifier localement un chemin de référence de classe C^3 (dont la courbure est continu) pour l'évitement d'obstacles en mettant en place des courbes de Bézier cubiques. Notre méthode sera couplée avec la commande MPC sous contraintes développée dans le chapitre précédent pour assurer la tâche de suivi de chemin local ou global. Nous validerons par la suite nos différents algorithmes par des simulations sous ROS-GAZEBO et par des expériences réelles sur la plateforme mobile "SPIDO".

2 État de l'art sur la planification de chemin pour l'évitement d'obstacles

Le contrôle des robots mobiles en environnement non maîtrisé est en général distribué sur deux niveaux de commande. Le premier niveau est un contrôleur de haut-niveau qui permet la perception de l'environnement entourant le véhicule et la planification de chemin ou trajectoire. Le deuxième niveau est un contrôleur de bas-niveau ayant une plus grande fréquence et qui ne dépend pas des mesures extéroceptives du robot. Il est dédié principalement à la poursuite du chemin planifié tout en contrôlant les actionneurs du véhicule. Le problème de la planification de chemin est généralement décrit de la manière suivante : on considère un véhicule se déplaçant dans un espace de travail, le but est de planifier les chemins ou trajectoires qui relient la position initiale du véhicule à sa position finale sans rentrer en collision avec les obstacles. Ces chemins générés doivent être réalisables par le véhicule, i.e., respecter les restrictions cinématiques du véhicule.

Khatib *et al.* a développé dans Khatib (1985) une technique réactive pour éviter les obstacles locaux, basée sur le champ potentiel. Cette approche considère le robot comme un point (dans l'espace des configurations) qui évolue dans un champ de forces virtuelles composé de forces répulsives autour des obstacles et une force attractive orientée vers la configuration finale du robot. Cette méthode est très élégante et présente l'avantage d'être efficace en temps réel et de ne pas nécessiter la connaissance précise des contours des obstacles. Cependant, dans Koren et Borenstein (1991) est montré que le robot peut rester bloqué dans un minimum local ou osciller lors du franchissement d'un passage étroit entre les obstacles. Aussi, cette méthode n'est

pas adaptée aux systèmes non-holonomes et les systèmes ayant des contraintes dynamiques. Wang *et al.* a proposé dans Wang (2011) une méthode de planification du chemin qui consiste à poursuivre parallèlement un mur en mettant en place la transformée de Hough. En revanche cette approche n'est pas performante si le chemin désiré est déformé, notamment le long de la tâche d'évitement d'obstacles.

D'autres méthodes d'évitement d'obstacles sont basées sur des approches probabilistes qui font partie de la grande famille des méthodes d'échantillonnage «sampling-based methods». L'idée de base des méthodes d'échantillonnage consiste à déterminer un ensemble fini de configurations sans collisions qui représentent adéquatement l'espace libre dans l'environnement, et utiliser ces configurations pour construire une “roadmap” (carte de route) entre deux positions assignées. Parmi ces méthodes probabilistes, on trouve la technique PRM (Probabilistic Roadmap) qui a été développée dans Kavraki *et al.* (1994) permettant de trouver un chemin sans collisions très rapidement entre deux configurations initiale et finale (à condition que l'algorithme PRM soit bien défini), notamment dans des espaces à très haute dimensionnalité. Cependant, la probabilité de trouver une solution par cet algorithme peut avoir un temps d'exécution qui tend vers l'infini s'il n'y a pas de solution. Aussi, la probabilité de placer un candidat dans une certaine région de l'espace libre est proportionnelle à son volume. Donc, il serait peu probable d'avoir un chemin à travers un passage étroit dans un délai raisonnable Bekris *et al.* (2003). Parmi aussi les algorithmes de planification qui sont couramment utilisés pour les systèmes holonomes et non-holonomes et qui peuvent être aussi mis en place pour l'évitement d'obstacles, la méthode RRT (Rapidly-exploring Random Trees) introduite par S. LaValle en 1998 LaValle (1998). Cette méthode consiste à faire évoluer un arbre partant de la configuration initiale afin d'atteindre une configuration but. Cette méthode RRT peut être utilisée facilement sur des robots avec des contraintes différentielles (non-holonomes et ciné-dynamiques). Trois algorithmes RRT bidirectionnels ont été développés dans Jun *et al.* (2016) pour planifier des trajectoires optimales de franchissement automatique d'obstacles d'un robot mobile dans des terrains accidentés. Une extension de cet algorithme appelée RRT* a été adoptée pour garantir une convergence rapide vers un parcours optimal Gammell *et al.* (2014). Cette technique RRT* a été appliquée dans hwan Jeon *et al.* (2011) sur un robot mobile rapide tout terrain comportant des contraintes différentielles et géométriques complexes avec des espaces d'état à grande dimension. Néanmoins, compte tenu des non-linéarités du système et la présence des contraintes non-holonomes, le taux de réussite de ces méthodes peut être faible si le nombre d'itération est limité, nécessitant parfois un temps de planification important.

Une autre méthode a été développée pour l'évitement d'obstacle qui consiste à utiliser un ensemble d'antennes virtuelles nommées “tentacules”. Ces dernières sont définies d'une façon égo-centrée autour du véhicule et sont représentées dans le repère véhicule. Dans Von Hundelshausen *et al.* (2008), les tentacules prennent la forme d'arcs de cercle, dont chaque arc correspond à un angle de braquage du véhicule. Cependant, cette forme d'arc de cercle n'est pas très appropriée pour les stratégies de navigation qui favorisent une variation continue de la courbure notamment en présence d'obstacles pour assurer la stabilité du véhicule et un changement lisse des angles de braquage. Pour améliorer cette méthode, Himmelsbach *et al.* (2011), Alia *et al.* (2015) et Ort *et al.* (2018) proposent d'utiliser des tentacules en forme de clothoïdes. Cette forme est complexe mais reste plus convenable aux véhicules à roues directrices vu que sa courbure varie linéairement en fonction de l'abscisse curviligne. Par contre, l'inconvénient principal de cette approche réside dans le fait d'avoir une certaine difficulté de raccordement des clothoïdes aux points de passage. D'autres approches d'évitement réactif d'obstacle pour des systèmes non-holonomes ont été

proposées dans [Gray et al. \(2012\)](#) et [Frazzoli et al. \(2000\)](#). Le principe général de leur méthode consiste à générer des primitives de mouvement (appelées trimes & manœuvres) couplées à un arbre de décision en utilisant le modèle dynamique du véhicule. Néanmoins, la détermination de la base de données des primitives n'est pas toujours facile et évidente vu la redondance cinématique et non-linéarité du système.

Afin de respecter la continuité de la courbure, plusieurs travaux se sont orientés vers les fonctions polynomiales pour définir des chemins ou trajectoires. En particulier, les courbes de type spline ou B-spline ont été fréquemment utilisées pour définir localement des trajectoires ou chemins, grâce à leurs simplicité de mise en œuvre et leurs propriétés de continuité d'ordre élevé (i.e., elles permettent de générer des courbures continues pour réaliser diverses manœuvres) [Muñoz et al. \(1992\)](#). Par exemple, des splines cubiques ont été adoptées dans [Haddad et al. \(2007\)](#) et [Eren et al. \(1999\)](#) pour la planification optimale des trajectoires libres entre deux points de passage d'un robot mobile destiné à se déplacer dans un espace de travail contraint. Aussi, [Hwang et al. \(2003\)](#) a utilisé des courbes de Bézier cubiques pour assurer la tâche d'évitement d'obstacles. En revanche, le chemin planifié ne passe pas exactement par les points du contrôle désirés vu qu'ils sont sélectionnés manuellement à travers une interface tactile. [Guechi et al. \(2008\)](#) a présenté une stratégie d'évitement d'obstacles dynamiques par l'intermédiaire de deux courbes de Bézier conçus en temps-réel selon la vitesse et la direction de l'obstacle mobile. Néanmoins, cette méthode est valide seulement si la vitesse et l'accélération du robot de type unicycle sont faibles. Enfin, [Nagatani et al. \(2001\)](#) a proposé une méthode d'évitement d'obstacles par le biais des courbes de Bézier pour un robot mobile de type voiture en tenant compte de la contrainte maximale sur le rayon de courbure que le robot peut réaliser.

Dans cette étude bibliographique, il ressort que les approches basées sur l'utilisation des chemins polynomiaux (spline ou B-spline), telles que la mise en place des courbes de Bézier, représentent une piste prometteuse dans ce domaine. En effet, d'un point de vue théorique, les chemins polynomiaux sont infiniment dérivables et continues et sont par ailleurs facilement gérables par le biais des points de contrôle. D'un point de vue pratique, ils présentent l'avantage de fournir des courbures continues qui peuvent être ajustées à travers un placement judicieux des points de contrôle tout en tenant compte de la cinématique et la dynamique du véhicule.

Nous nous intéressons dans ce chapitre au développement d'une nouvelle approche permettant de calculer rapidement et en temps réel un chemin local de contournement pour l'évitement d'obstacles. Ce chemin est généré automatiquement en fonction de plusieurs points de passage basés sur les coordonnées de l'obstacle détecté et l'état du véhicule (e.g., direction du véhicule, positions actuelles, angles de braquage, etc.). Deux courbes de Bézier cubiques sont conçues pour relier ces points de passage de telle façon à éviter des grandes courbures et des variations agressives des angles de braquage (impliquant la garantie de la sécurité et stabilité du véhicule). Le suivi du chemin planifié, quant à lui, est assuré par la commande MPC sous contraintes développée dans le chapitre 4.

3 Aperçu sur les courbes de Bézier

La planification basée sur l'utilisation des courbes de Bézier est fréquemment utilisée dans la robotique mobile. Ces courbes ont été inventées vers la fin des années 1950 par Pierre Bézier, ingénieur chez Renault.

On considère $n+1$ points du plan ou de l'espace ($n \geq 1$, le plus souvent $n = 3$), P_0, P_1, \dots, P_n et on définit la courbe paramétrée de Bézier associée à ces points de contrôle par :

$$\gamma_c^n(v) = \sum_{k=0}^n P_k B_{n,k}(v), \quad v \in [0, 1], \quad (5.1)$$

où les $B_{n,k}(v) = C_n^k v^k (1-v)^{n-k}$ sont les polynômes de base de Bernstein.

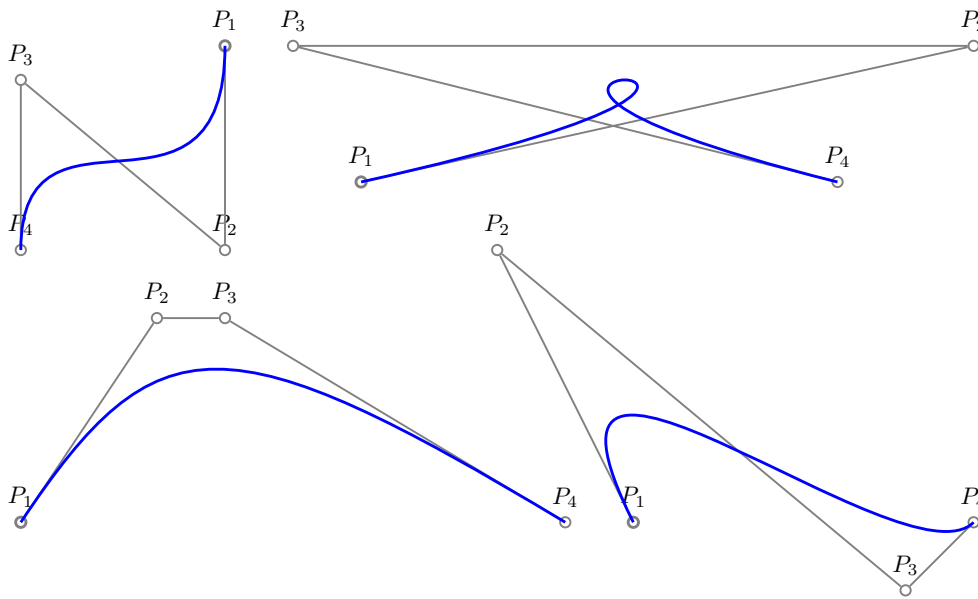


Figure 5.1. Courbes de Bézier cubiques avec différents positionnements des points de contrôle.

La courbe de Bézier passe par les points de contrôle initial et final P_0 et P_n et y est tangente aux vecteurs $\overrightarrow{P_0P_1}$ et $\overrightarrow{P_{n-1}P_n}$. La courbe est de classe C^∞ et elle est dans l'enveloppe convexe des points de contrôle. Le déplacement d'un seul de ces points modifie toute la courbe (Figure 5.1).

D'un point de vue pratique et vue leur variété (avec ou sans inflexion), les courbes Bézier cubiques (i.e., courbe paramétrée par quatre points de contrôle P_0, P_1, P_2, P_3) s'avèrent être un choix suffisant pour définir un chemin de contournement vérifiant les contraintes cinématiques d'un véhicule non-holonome. Des exemples de courbes de Bézier cubiques sont représentées sur la Figure 5.1 montrant l'importance de bien choisir le positionnement des points de contrôle. Pour plus de détail sur les courbes de Bézier et B-splines, on peut se référer à [Guillod \(2008\)](#).

4 Détection d'obstacles

4.1 Introduction

Depuis quelques années, grâce aux avancées technologiques dans le domaine de la télémétrie laser, plusieurs systèmes embarqués pour la perception de l'environnement sont commercialisés tels que les capteurs LiDAR ("Light Detection And Ranging") terrestres. Ces systèmes constituent un moyen rapide et précis de numériser en 3D (nuages de points) des environnements intérieurs et/ou extérieurs (ruraux et/ou urbains). La segmentation et l'interprétation de ces informations 3D est l'une des tâches les plus importantes pour la détection et l'identification des objets.

La plupart des recherches sur la détection d'obstacles s'appuie sur les méthodes d'exploitation et de traitement des nuages de points 3D. Becker et Haala (2007) et Stamos *et al.* (2006) ont développé des méthodes d'extraction du sol et d'approximation des façades par des plans en utilisant l'algorithme de RANSAC à partir d'un nuage de points 3D. Une méthode pour l'extraction et la modélisation des arbres à partir des données LiDAR a été introduite dans Reitberger *et al.* (2009). La croissance de région est une technique proposée dans Bethel (2004) pour les données LiDAR. Cette technique se compose de deux étapes : choisir les points de départ des régions et faire grossir les régions par agglomérations des pixels voisins. Aussi, une autre méthode de segmentation et d'interprétation du nuage de points au niveau du sol d'environnements urbains a été présentée dans Hernandez et Marcotegui (2013). Cette méthode a pour objectif de détecter des lampadaires, des voitures et des piétons ainsi que d'autres objets comme des panneaux de signalisation. Une autre approche pour détecter les obstacles consiste à utiliser les normales à une surface formée par un nuage de points Haala *et al.* (1998). L'orientation et la direction de ces normales permettent de détecter par exemple une pente importante, un objet, un mur, etc. Enfin, la transformée de Hough, proposée par Overby (2004) est une technique aussi de reconnaissance de formes et qui est utilisée dans le traitement d'images numériques pour la détection d'obstacles.

Dans cette section, nos travaux se centrent sur la détection rapide des obstacles autour du véhicule (e.g., piétons, arbres, bâtiments, cartons, etc.). Cette approche a pour but de constituer un pré-traitement rapide du nuage de points 3D délivré par le capteur LiDAR et par la suite une extraction et identification des obstacles. C'est pourquoi l'accent n'est pas mis seulement sur la qualité des résultats, mais sur un compromis entre qualité, temps de calcul et ressources nécessaires.

4.2 Acquisition du nuage de points

Notre système de perception est constitué uniquement d'un capteur LiDAR 3D "HDL-32E". C'est un capteur développé par la société Velodyne LiDAR notamment pour des applications en milieu ouvert tel que la voiture autonome. Il peut atteindre une portée de 120m et générer plus de 1.3 million de points par seconde. Ce capteur transmet les données brutes de l'environnement dans le plan du capteur au PC embarqué via une liaison Ethernet à la fréquence de 10Hz. Le driver ROS se charge de générer le nuage de points à partir de ces données brutes. L'information est alors prête pour être exploitée en fonction des besoins du projet. Pour la détection d'obstacles

en temps-réel, nous avons développé nos propres algorithmes en utilisant les filtres de la librairie PCL (Point Cloud Library) [Rusu et Cousins \(2011\)](#), de manière à réduire le temps de calcul et de traitement du nuage de points. Certaines hypothèses simplificatrices sont prises en compte pour faciliter la tâche de détection, qu'on résume ci-dessous,

Hypothèse 7. Dans un premier temps, nous ne considérerons que des obstacles statiques à positions fixes et invariantes dans l'environnement de travail.

Hypothèse 8. La profondeur de tous les obstacles est supposée limitée pour que tout objet détecté puisse s'insérer dans une sphère. De cette manière, la partie cachée de l'obstacle que le LiDAR ne peut pas percevoir n'influencera pas le résultat de notre stratégie de navigation.

Hypothèse 9. Le capteur Velodyne est supposé toujours parallèle au sol, pour que le nuage de points appartenant au sol soit facilement ignoré, i.e., a priori dans un premier temps, aucune pente sur le terrain de navigation n'est considérée.

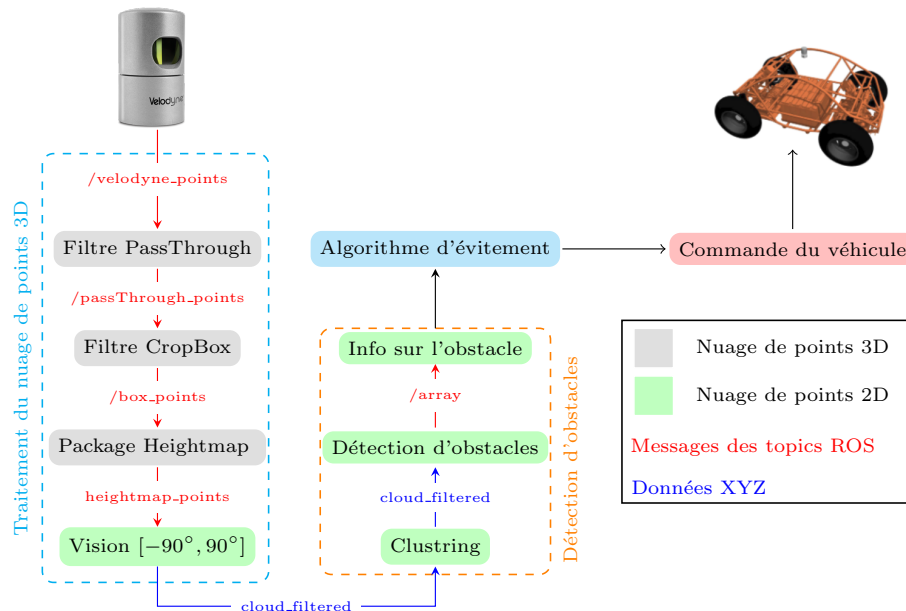


Figure 5.2. Chaîne d'information pour la détection d'obstacles.

4.3 Sous-échantillonnage des données

Une des premières opérations de traitement des données est le sous-échantillonnage (down-sampling) dont le but est de réduire la taille du nuage de points. Cette opération est primordiale en terme de temps de calcul. En effet, le capteur Velodyne peut fournir plus de 1.3 million de points par seconde, ce qui implique une charge de calcul considérable pour traiter l'information telle qu'elle. Nous désirons que notre robot puisse traiter rapidement les informations du capteur afin de les exploiter en temps-réel. Nous avons donc implémenté un nœud ROS utilisant les filtres de la bibliothèque PCL pour réduire la quantité d'information à traiter. La Figure 5.2 décrit les principales étapes pour la mise en œuvre de notre algorithme de détection d'obstacles. Pour ne pas alourdir le manuscrit de thèse, nous décrivons brièvement l'objectif de chaque partie de la chaîne de traitement du nuage de points.

Filtre à Voxel

Le filtre à voxel¹ permet de subdiviser l'espace en voxel, c'est à dire en cubes de dimensions choisies. Tous les points contenus dans un cube sont remplacés par leur barycentre. Cela permet de réduire grandement le nombre de points sans pour autant perdre en précision et en informations selon la taille choisie du cube. Plus la taille du cube est grande, moins on obtient de points en sortie et plus on perd d'informations.

Élimination du nuage de points appartenant au robot et au sol

Le filtre Pass-Through² est utilisé pour tronquer les points représentant le sol car l'une des hypothèses de cet algorithme de détection d'obstacles est que le terrain sur lequel le véhicule se déplace est plat. De plus, les points appartenant à une altitude légèrement supérieure à la hauteur du robot doivent être éliminés vu qu'ils ne gênent pas le déplacement du véhicule (e.g., les feuilles d'un arbre ne peuvent pas être prises en compte par notre algorithme).

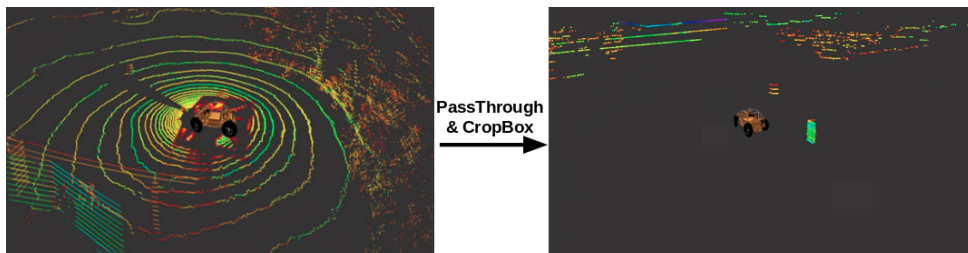


Figure 5.3. Élimination du nuage de points appartenant au robot et au sol.

Le filtre CropBox³, quant à lui, permet de retirer les points appartenant au robot mobile, en utilisant comme référence un parallélépipède 3D de dimensions bien définies caractérisant celles du véhicule. Ce filtre élimine tous les points situés à l'intérieur de ce parallélépipède pour que notre algorithme ne les considère pas comme des obstacles.

4.4 Algorithme de détection d'obstacles

Velodyne height map

Cette approche a été développée pour la DARPA challenge spécifiquement pour le capteur Velodyne *Jack O'Quin et al.*. Ce paquet ROS prend en amont le nuage de points 3D provenant du premier filtrage et renvoie en aval un nuage de point 2D dans le plan horizontal du LiDAR (i.e., $z = 0$) détectant les objets de taille supérieure à une certaine hauteur bien définie. Cette approche générique permet de détecter des zones non navigables, allant d'un objet posé sur le sol à une pente trop importante. Cet algorithme publie deux topics ROS : *velodyne_obstacles* qui représente les objets détectés, et *velodyne_clear* qui représente la partie libre sans obstacles

1. [Downsampling a PointCloud using a VoxelGrid filter](#)
2. [Filtering a PointCloud using a PassThrough filter](#)
3. [CropBox Filter](#)

(voir Figure 5.4). En résumé, cette méthode permet de segmenter l'environnement en espace occupé et son complément l'espace libre. Sur la Figure 5.4, on peut voir un résultat du nuage de points après l'utilisation du paquet height map : on représente en vert l'espace libre, et en rouge l'espace occupé.

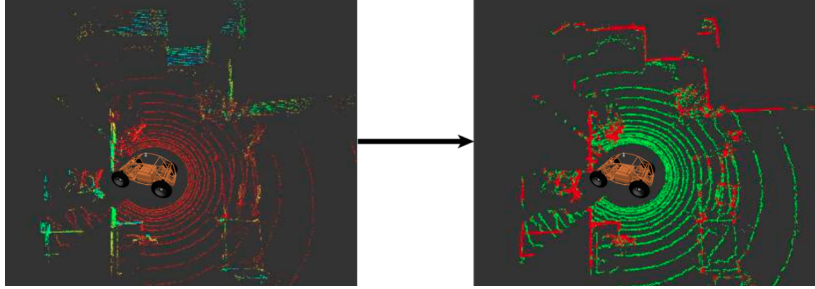


Figure 5.4. Velodyne height map.

Réduction du champ de vision

Afin de pouvoir traiter les obstacles en temps-réel, nous avons implémenté une stratégie qui consiste à limiter le champ de vision. En effet, le Velodyne fournit a priori un nuage de points couvrant 360° , soit l'intégralité de l'environnement du robot. Dans notre cas, il n'est pas nécessairement utile de récupérer l'intégralité de cette information. Par exemple, un véhicule avançant en marche avant peut naviguer en prenant en compte uniquement ce qui est situé devant le robot, jusqu'à une distance donnée. Le nuage de points fourni par le Velodyne est donné en coordonnées cartésiennes. Ainsi, tous les points dont les coordonnées sont telles que $x < 0$ seront ignorés, i.e., champ de vision est ainsi dans l'intervalle $[-90^\circ, 90^\circ]$ (cf. Figure 5.5), car les objets situés derrière le véhicule ne représentent plus un élément bloquant de la tâche de suivi de chemin et cela réduit considérablement le temps de traitement du nuage de point.

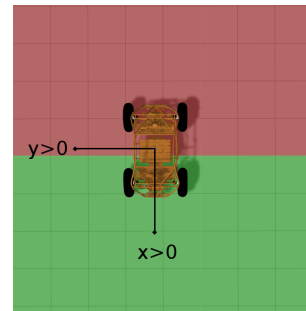


Figure 5.5. Restriction de la zone à traiter.

Segmentation en clusters

La segmentation du nuage de points est une étape cruciale dans la chaîne de traitement. Elle doit être la plus précise et la plus correcte possible. Nous utiliserons ici la technique de recherche Kd-tree pour pouvoir reconstruire les différents clusters. De cette manière, chaque objet est identifié et analysé individuellement et séparément par l'algorithme de détection d'obstacle. Cette méthode est basée sur la décomposition et la partition spatiale du nuage de points d'origine en parties plus petites afin de réduire le temps de traitement. Avec la méthode Kd-tree, l'espace de nuages de points est divisé en deux parties, puis chaque moitié est divisée en deux, etc.

La liste 5.1 indique les paramètres de la méthode Kd-tree⁴. Cette méthode de recherche des clusters prend en amant le nuage de points traité et donne en aval un tableau d'indice de chaque cluster détecté qui est enregistré dans le vecteur `cluster_indices`. Les paramètres à ajuster sont définis en choisissant la tolérance pour chaque cluster (`Set-ClusterTolerance`) et le nombre maximum et minimum de points à l'intérieur d'un cluster (`SetMaxClusterSize`, `SetMinClusterSize`).

Liste 5.1: Paramètres de la méthode Kd-tree

```
// Creating the KdTree object for the search method of the extraction
pcl::search::KdTree<pcl::PointXYZ>::Ptr tree (new pcl::search::KdTree<pcl::
PointXYZ>);
tree->setInputCloud((*cloud_filtered).makeShared());
std::vector<pcl::PointIndices> cluster_indices;
//Cluster based on the kd-tree algorithm, and returns a vector of indices
pcl::EuclideanClusterExtraction<pcl::PointXYZ> ec;
//kd-tree Library parameters
ec.setClusterTolerance (0.1); //10 cm
ec.setMinClusterSize (2);
ec.setMaxClusterSize (25000);
ec.setSearchMethod (tree);
ec.setInputCloud ((*cloud_filtered).makeShared());
ec.extract (cluster_indices);
```

Algorithme de détection d'obstacles

Une fois le nuage de points fourni par le Velodyne est traité, un ensemble de clusters est identifié contenant un certain nombre de groupes de nuage de points. Deux types d'obstacles sont distingués dans ce travail : des obstacles dangereux qui sont proches du véhicule et le chemin nominal et tous les autres (non dangereux) qui sont loins du chemin nominal.

Pour assurer la bonne détection de chaque obstacle et le classifier selon sa catégorie (dangereux ou pas), on procède tout d'abord par la détermination des coordonnées cartésiennes de son barycentre O_c dans le repère inertiel (un changement de repère est établi par des transformations homogènes entre le repère Velodyne et le repère inertiel). Après cette étape, le cercle contenant l'objet détecté est identifié (son rayon est noté R_o dans l'**algorithme 1**). Pour des raisons de sécurité, ce cercle est par la suite grossi d'une épaisseur égale à la demie-largeur du robot. Le rayon de ce cercle grossi est noté R_g et peut être exprimé comme suit,

$$R_g = R_o(1 + \alpha_g), \quad (5.2)$$

avec α_g est un coefficient d'augmentation relié au demi-largeur du véhicule ($\alpha_g = 0.3$).

Un autre cercle englobant (\mathcal{C}) est construit à partir du cercle grossi (cf. Figure 5.11). Ce cercle englobant sera utilisé par la suite par notre algorithme d'évitement d'obstacles pour extraire les points de passage. Son rayon est noté R_e et s'exprime de la façon suivante :

$$R_e = R_g(1 + \alpha_e), \quad (5.3)$$

4. [Euclidean Cluster Extraction](#)

où α_e est un coefficient d'augmentation ($\alpha_e = 0.4$).

Un objet détecté est classé comme un objet dangereux si et seulement si la distance entre son barycentre O_c et la position du véhicule (x, y) est inférieure à la distance de sécurité d_s qu'on définira dans la prochaine section.

L'**algorithme 1**, développé dans l'annexe B, donne plus de détails sur les différentes étapes effectuées pour la tâche de détection d'obstacles.

4.5 Validation sous ROS-GAZEBO

Nous validons notre algorithme de détection d'obstacles sous ROS-GAZEBO en utilisant la plateforme de simulation équipée d'un LiDAR virtuel qui génère un nuage de points 3D. On positionne autour du véhicule des obstacles (de formes cubique, sphérique et/ou cylindrique) comme on peut le voir sur la Figure 5.6.

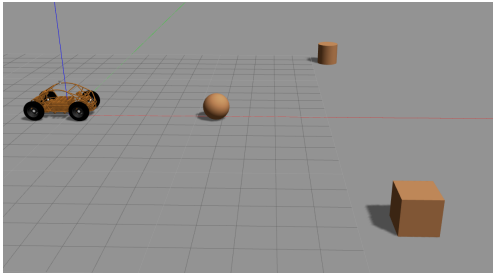


Figure 5.6. Validation en simulation de l'algorithme de détection.

tracé en orange) et des obstacles non dangereux donnés par un nuage en vert (si la plateforme est située à l'extérieur du cercle de sécurité).

La Figure 5.7 montre une première application pratique sur le simulateur GAZEBO de notre approche de perception de l'environnement pour la détection d'obstacles autour de la maquette virtuelle. On affiche en couleurs le nuage de points délivré par le capteur (Figure de gauche) et à droite le résultat final de l'algorithme de détection. La partie représentant l'espace libre n'est pas affichée ici pour une meilleure lisibilité. Deux types d'obstacles sont classifiés par cet algorithme : obstacles dangereux donnés par un nuage en rouge (quand la plateforme est située à l'intérieur de la périphérie de sécurité,

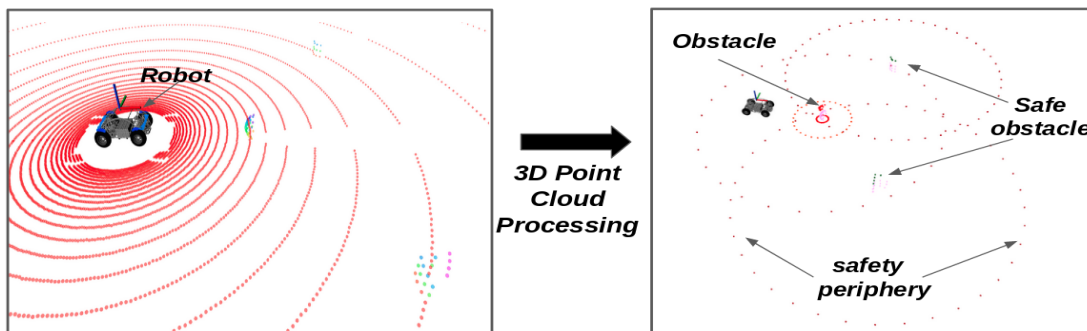


Figure 5.7. Approche de détection d'obstacles sous GAZEBO. [Gauche] : Nuage de points initial fourni par le LiDAR 3D virtuel. [Droite] : Résultats de traitement du nuage de points pour la détection d'obstacles.

4.6 Validation expérimentale

Nous avons effectué des essais sur le robot “*SPIDO*” commandé en toute sécurité par téléopération à l’aide de son Joystick (en boucle ouverte). Deux gros cartons ont été placés dans le terrain expérimental (cf. Figure 5.8) afin de valider notre algorithme de détection d’obstacle⁵.

Le nuage de points capturé par “*SPIDO*” a été enregistré à l’aide d’un ordinateur distant communiquant avec l’ordinateur embarqué NUC via le protocole SSH. Le résultat final de notre algorithme de détection d’obstacles est donné par la Figure 5.9. Dès qu’un obstacle est détecté, il est classé en fonction de la distance de sécurité d_s qu’on définira par la suite en tant que obstacle non dangereux ou dangereux. Les obstacles dangereux sont détectés proches du véhicule et de son chemin nominal, i.e., le véhicule se trouve à l’intérieur de la périphérie de sécurité (voir Figure 5.9).



Figure 5.8. Validation expérimentale de l’algorithme de détection d’obstacles.

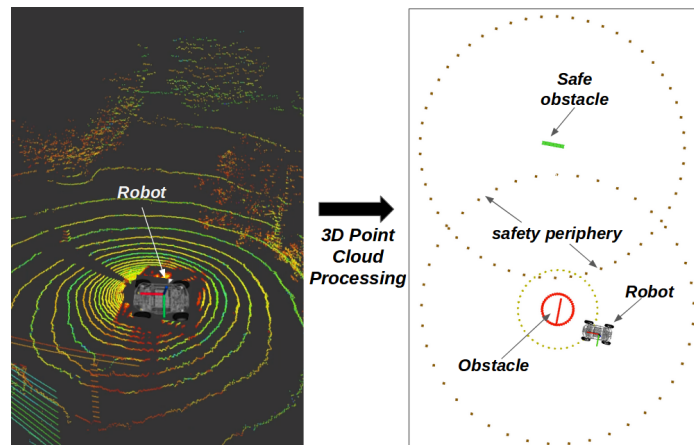


Figure 5.9. Approche de détection d’obstacles. [Gauche] : Nuage de points brut fournie par le LiDAR 3D. [Droite] : Résultats de traitement du nuage de points pour la détection d’obstacles, la distance entre chaque obstacle et le centre de gravité du véhicule est calculée. Notez que deux types d’obstacles sont distingués dans ce travail, les plus sûrs (nuages verts) et les dangereux (nuages rouges) proches du chemin de référence global.

Les résultats obtenus montrent que l’algorithme retenu permet bien de détecter les différents obstacles présents sur la zone d’essai. Pour améliorer cet algorithme dans nos futurs travaux, nous pensons qu’il serait peut-être très utile et préférable d’utiliser une ellipse englobante des objets détectés au lieu d’un cercle englobant, cela permettra de convenir à la forme de l’obstacle et d’être moins conservatif dans le cas d’obstacles très allongés (un mur par exemple). Par ailleurs, l’algorithme de détection d’obstacle proposé a donné des résultats satisfaisants.

⁵. Visionner la vidéo suivante sur la validation expérimentale de l’algorithme de détection d’obstacles : [▶▶▶▶▶▶▶▶](#)

5 Planification locale du chemin pour l'évitement d'obstacle

Notre stratégie vise à éviter les obstacles détectés et maintenir le véhicule à proximité du chemin de référence global (noté Γ). Notre approche consiste à extraire les points les plus significatifs que le véhicule doit traverser une fois un obstacle est détecté autour de Γ , puis générer deux courbes admissibles et réalisables par le véhicule. Nous allons utiliser les courbes de Bézier pour relier deux points de passage consécutifs de manière à ce que ce chemin ait des courbures minimales et que le véhicule puisse suivre ce chemin avec un taux de variation minimal qui sollicite le braquage des roues le moins possible.

5.1 Contrôleur haut-niveau

5.1.1 Distance de sécurité

La gamme de variation de la vitesse longitudinale V_x du robot est comprise entre 0 et 12 m.s^{-1} . La distance de sécurité est exprimée comme suit,

$$d_s = d_{min} + k_1 V_x^2 + k_2 a_{x_{max}}, \quad (5.4)$$

avec d_{min} est la distance minimale de sécurité calculée en prenant en compte les dimensions géométriques du véhicule et la position du capteur LiDAR sur le robot, elle est ajustée sur $d_{min} = 3\text{m}$. Deux coefficients k_1 et k_2 utilisés pour régler cette distance de sécurité ($k_1 = 0.2$, $k_2 = 0.5$), et a_{max} est la décélération longitudinale maximale du véhicule reflétant l'efficacité de freinage du véhicule, dans notre implémentation $a_{x_{max}} = 1.5 \text{ m.s}^{-2}$. La Figure 5.10 montre la variation de cette distance de sécurité en fonction de la vitesse longitudinale.

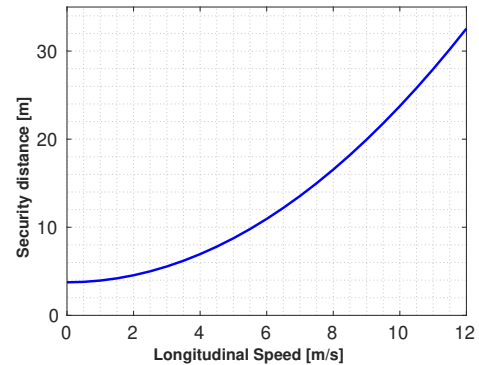


Figure 5.10. Distance de sécurité

5.1.2 Approche d'évitement d'obstacles via des courbes de Bézier

Supposons que le robot suive parfaitement le chemin de référence global Γ en utilisant la commande MPC sous contraintes développée dans le chapitre 4. Bien que la distance entre le centre de gravité du véhicule et le barycentre de l'obstacle détecté (noté O_c) soit supérieure à la distance de sécurité d_s , le robot continue de suivre son chemin de référence global Γ . Quand cette distance de sécurité est non respectée, le robot dévie de son chemin de référence Γ en suivant un autre chemin généré rapidement et en temps-réel afin de contourner les obstacles détectés. Ce nouveau chemin est composé de deux courbes cubiques de Bézier. La première courbe est utilisée pour éviter les obstacles détectés et la deuxième pour rejoindre le chemin de référence global Γ . Toutes les deux devraient être continues et dérivables en tous points et particulièrement aux points de transition entre les deux courbes de Bézier et avec le chemin de référence global Γ (cf. Figure 5.11).

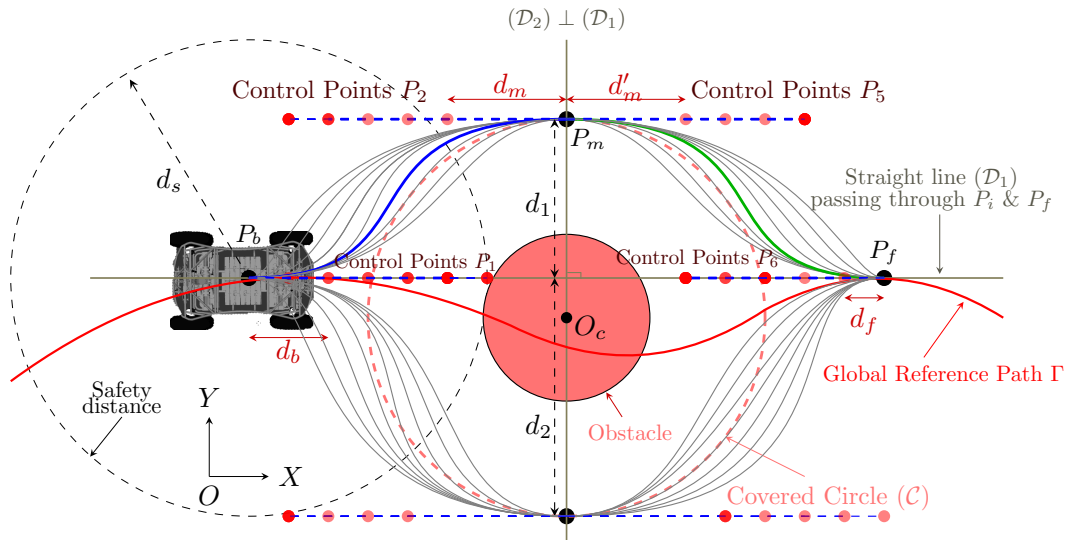


Figure 5.11. Stratégie d'évitement d'obstacle.

Ce chemin local relie trois points de passage importants :

- La position initiale du véhicule $P_b(x_b, y_b, \psi_b)$ où la distance de sécurité d_s est violée, avec x_b et y_b sont les positions instantanées du véhicule dans le repère inertiel et ψ_b est son angle de cap.
- Le point intermédiaire noté $P_m(x_m, y_m, \psi_m)$ reliant les deux courbes cubiques de Bézier. Ce point est choisi de telle façon à rester proche du chemin de référence global Γ et à assurer la convergence rapide vers ce chemin.
- La position finale du véhicule $P_f(x_f, y_f, \psi_f)$ qui se trouve sur le chemin de référence global Γ .

5.1.3 Extraction du point final et intermédiaire

Nous avons supposé que la profondeur des obstacles détectés est limitée (c'est-à-dire que l'épaisseur des obstacles, qui correspond à la partie non visible par le LiDAR, est limitée - voir Hypothèse 8). Le dernier point de contrôle $P_f(x_f, y_f, \psi_f)$ est situé dans le chemin de référence global Γ . Ce point est choisi tel que sa distance avec le barycentre de l'obstacle détecté O_c est supérieure à la distance de sécurité d_s ($\|\overrightarrow{O_c P_f}\| \geq d_s$).

Après la sélection du point final P_f , une droite (\mathcal{D}_1) reliant les points P_b et P_f et sa perpendiculaire (\mathcal{D}_2) passant par le barycentre O_c sont déterminées. En se basant sur les dimensions de l'obstacle gonflé, le cercle englobant (\mathcal{C}) de rayon R_e est calculé par l'équation (5.3) (voir Figure 5.11).

Une fois le cercle englobant fixé, deux intersections de ce cercle avec (\mathcal{D}_2) sont déterminées, et ensuite, deux distances d_1 et d_2 par rapport à la droite (\mathcal{D}_1) sont calculées. Enfin, le point d'évitement P_m est fixé en fonction du minimum de ces distances (voir **Algorithme 2** dans l'annexe B).

5.1.4 Génération des courbes de Bézier cubiques

Les obstacles détectés, qui entrent en collision avec le chemin de référence global Γ , seront évités en générant en temps-réel un nouveau chemin local de référence afin de contourner ces obstacles. Ce nouveau chemin de contournement est composé de deux courbes de Bézier cubiques, la première est générée entre P_b et P_m pour éviter l'obstacle détecté, et la seconde connecte P_m et P_f pour revenir au chemin de référence global Γ (chemin de ralliement).

Sur la Figure 5.12, on donne les différents scénarios de la stratégie haut-niveau selon les conditions de l'environnement de navigation afin de générer la bonne consigne à exécuter par le véhicule.

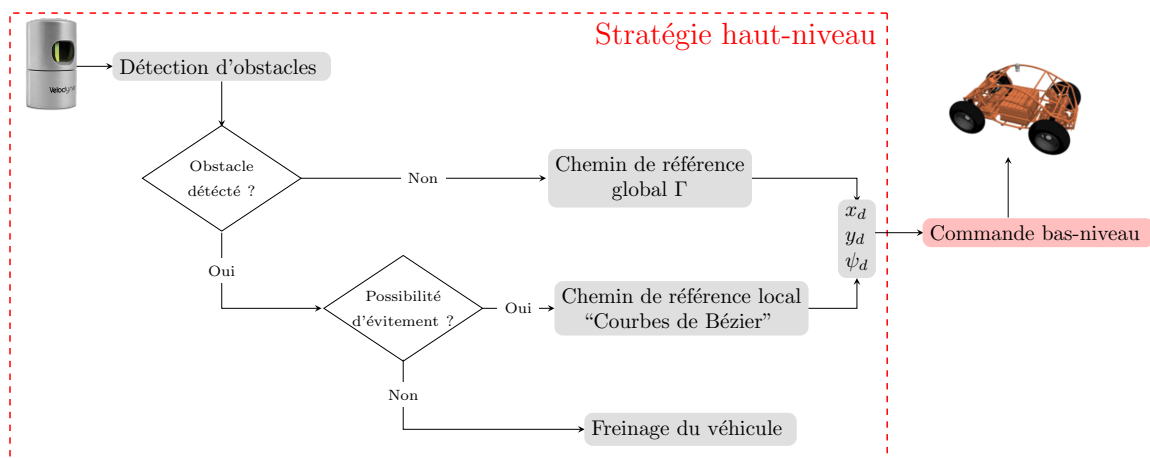


Figure 5.12. Stratégie haut-niveau pour l'évitement d'obstacles.

Le premier segment du chemin local nécessite quatre points de contrôle : P_b , P_1 , P_2 et P_m . Le premier point de contrôle P_b est le point de départ (c'est-à-dire la position actuelle du véhicule lorsque la distance de sécurité est non respectée), et le dernier P_m est le point final. Les deux points ont été déjà extraits et le nouveau chemin local les traverse. Le deuxième et troisième points de contrôle $P_1(x_1, y_1)$ et $P_2(x_2, y_2)$ sont conçus de manière à respecter les contraintes cinématiques et dynamiques du véhicule et assurer la continuité du chemin.

Pour assurer la continuité du chemin de référence aux points de passage, les points P_1 et P_2 doivent être sélectionnés en respectant la posture initiale et finale du véhicule. Par conséquent, P_1 est placé sur la tangente du chemin Γ au point P_b (noté (\mathcal{T}_b)), qui dépend du cap du véhicule ψ_b et ses positions instantanées x_b et y_b où la distance de sécurité n'est pas respectée. De même, P_2 doit varier dans la deuxième tangente au point P_m (noté (\mathcal{T}_m)), qui est choisie parallèle à la droite (\mathcal{D}_1) (voir Figure 5.11). Ainsi, leurs coordonnées sont exprimées comme suit :

$$\begin{cases} x_1 = x_b + d_b \cos \psi_b, & y_1 = y_b + d_b \sin \psi_b, \\ x_2 = x_m - d_m \cos \psi_m, & y_2 = y_m - d_m \sin \psi_m, \end{cases} \quad (5.5a)$$

avec la longueur d_b correspond à la distance entre la position initiale P_b et le premier point de contrôle intermédiaire P_1 , $d_b = d(P_b, P_1)$, et la longueur d_f est la distance entre P_m et le deuxième point de contrôle intermédiaire P_2 , $d_m = d(P_2, P_m)$, avec $d()$ symbolise la distance Euclidienne.

Un ensemble de courbes de Bézier cubiques est généré pour la première section du chemin de contournement en faisant varier les distances d_b et d_m . Chaque distance appartient à l'intervalle $[0, d_s]$ qui est échantillonné en n intervalles égaux. Ensuite, n^2 courbes de Bézier cubiques sont générées. L'expression paramétrique de chaque courbe de Bézier cubique est donnée par :

$$\begin{cases} x_d(v_1) = x_b(1 - v_1)^3 + 3x_1v_1(1 - v_1)^2 + 3x_2v_1^2(1 - v_1) + x_mv_1^3, \\ y_d(v_1) = y_b(1 - v_1)^3 + 3y_1v_1(1 - v_1)^2 + 3y_2v_1^2(1 - v_1) + y_mv_1^3, \end{cases} \quad (5.6a)$$

avec $v_1 \in [0, 1]$ est un paramètre dépendant du temps qu'on peut définir par, $v_1 = \frac{T - T_i}{\Delta T_1}$, où $T \in [T_i, T_{f1}]$ est le temps courant, T_i est l'instant quand l'obstacle est détecté, T_{f1} est le temps final et $\Delta T_1 = T_{f1} - T_i$ est la durée nécessaire pour traverser la première courbe de Bézier.

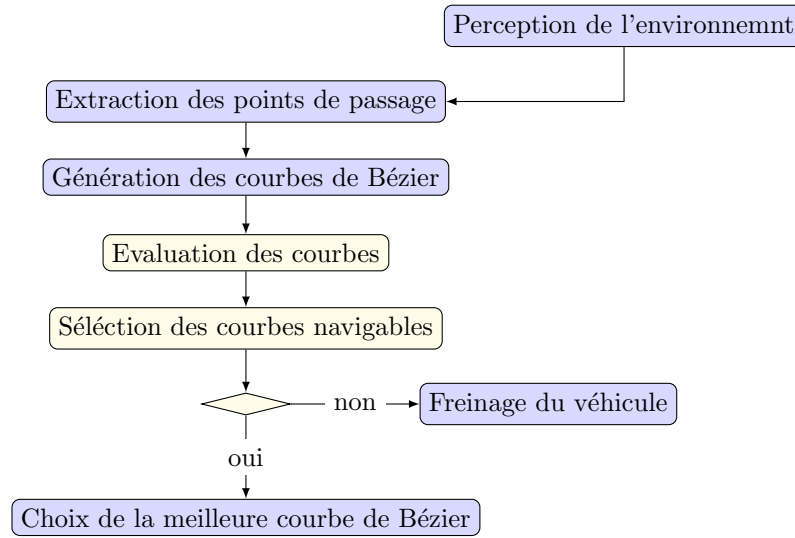


Figure 5.13. Principales étapes de la planification locale du chemin de contournement.

La deuxième partie du chemin local de contournement est également déterminée de la même manière. Cette deuxième partie de chemin local relie les points de contrôle suivants : P_m , P_5 , P_6 et P_f . Le point de départ P_m et le point final P_f ont été déjà sélectionnés. De plus, P_5 appartient à la même tangente que celle du point P_2 (\mathcal{T}_m), ce qui permet d'assurer la continuité et la finesse de ce nouveau chemin au point d'évitement P_m . Enfin, P_6 est choisi sur la dernière tangente de Γ au point de passage P_f (noté (\mathcal{T}_f)) en respectant l'orientation finale du robot ψ_f et ses coordonnées (x_f et y_f). Ils peuvent être exprimés comme suit,

$$\begin{cases} x_5 = x_m + d'_m \cos \psi_m, & y_5 = y_m + d'_m \sin \psi_m, \\ x_6 = x_f - d_f \cos \psi_f, & y_6 = y_f - d_f \sin \psi_f, \end{cases} \quad (5.7a)$$

avec les distances $d'_m = d(P_m, P_5)$ et $d_f = d(P_6, P_f)$.

De la même manière, n^2 courbes de Bézier cubiques sont générées pour cette deuxième tranche du chemin local d'évitement en faisant varier d'une façon régulière n fois les distances d'_m et d_f dans l'intervalle $[0, d_s]$. Chaque courbe est calculée à partir de la formule suivante,

$$\begin{cases} x_d(v_2) = x_m(1 - v_2)^3 + 3x_5v_2(1 - v_2)^2 + 3x_6v_2^2(1 - v_2) + x_fv_2^3, \\ y_d(v_2) = y_m(1 - v_2)^3 + 3y_5v_2(1 - v_2)^2 + 3y_6v_2^2(1 - v_2) + y_fv_2^3, \end{cases} \quad (5.8a)$$

avec $v_2 \in [0, 1]$ est un paramètre de temps défini de la façon suivante : $v_2 = \frac{T-T_{f1}}{\Delta T_2}$, où $T \in [T_{f1}, T_{f2}]$, et $\Delta T_2 = T_{f2} - T_{f1}$ est la durée nécessaire pour traverser la seconde courbe de ralliement vers le chemin global Γ , T_{f1} et T_{f2} sont les instants de début et fin nécessaires pour effectuer la deuxième courbe.

5.1.5 Sélection de la meilleure courbe de chaque courbe de Bézier

Pour sélectionner les deux meilleures courbes de Bézier, les distances d_b , d_m , d'_m et d_f doivent être sélectionnées de sorte que les courbes choisies puissent garantir les contraintes cinématiques et dynamiques du véhicule. La meilleure courbe choisie doit remplir ces critères :

- La posture initiale du véhicule doit être respectée, notamment les angles de braquage courants.
- La variation de la courbure doit être continue car les angles de braquage doivent changer d'une façon lisse afin de favoriser la stabilité du véhicule.
- Les fortes courbures que le véhicule ne peut pas exécuter doivent être évitées.

Les angles de direction avant et arrière actuels $\delta_{(f,r)}$ au point P_b doivent être pris en compte pour empêcher tout changement brusque de ces angles et permettre au véhicule d'éviter les mouvements qui peuvent causer des comportements indésirables et des oscillations le long de chemin, puis la courbure initiale ρ_0 de la première partie du chemin de contournement peut être approchée selon la géométrie du véhicule par la formule suivante (voir Figure 5.14) :

$$\rho_0 \simeq \frac{2 \tan \delta_{(f,r)}}{L}, \tag{5.9}$$

avec L est l'empattement du véhicule.

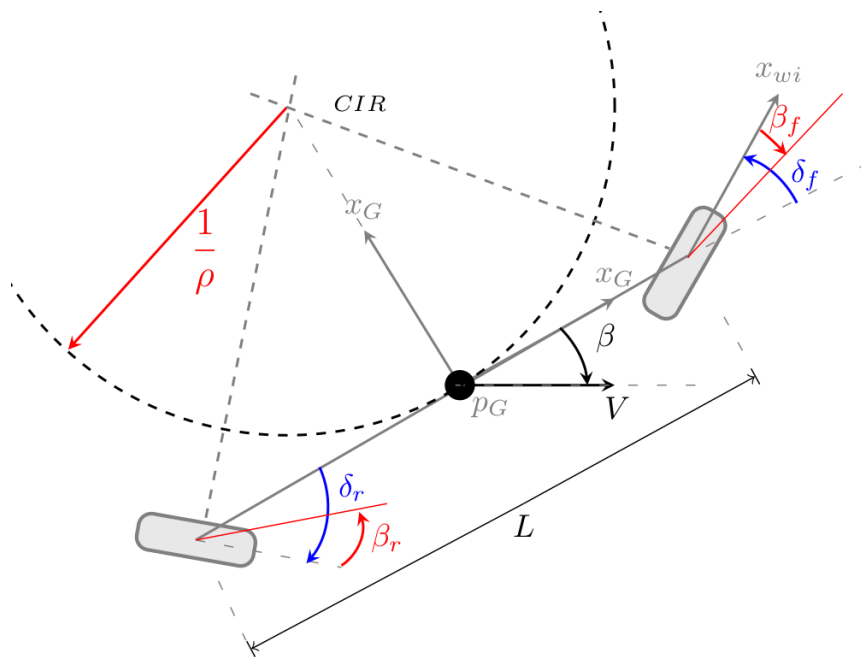


Figure 5.14. Centre instantané de rotation

Rappelons la définition de la courbure $\rho_{j,k}$ de la courbe de Bézier S_k (avec $k \in \llbracket 1, n^2 \rrbracket$) au point $(x_{d_{j,k}}, y_{d_{j,k}}) \in S_k$ qui vaut le rapport entre la variation de l'angle de lacet $d\psi_{j,k}$ à ce point et la longueur de la courbe $ds_{j,k}$. Symboliquement,

$$\rho_{j,k} = \frac{d\psi_{j,k}}{ds_{j,k}} = \frac{x'_{d_{j,k}}(v)y''_{d_{j,k}}(v) - y'_{d_{j,k}}(v)x''_{d_{j,k}}(v)}{\left(x'_{d_{j,k}}(v)^2 + y'_{d_{j,k}}(v)^2\right)^{\frac{3}{2}}}, \quad (5.10)$$

La courbure maximale de chaque courbe de Bézier S_k est notée,

$$|\rho_k| = \max_{\forall j} |\rho_{j,k}|, \quad k \in \llbracket 1, n^2 \rrbracket, \quad (5.11)$$

La stabilité latérale du véhicule est liée à son accélération latérale qui doit être limitée. En effet, l'accélération latérale maximale que le véhicule peut atteindre le long de chaque courbe de Bézier dépend de sa courbure maximale ρ_k et de la vitesse longitudinale du véhicule V_x , qu'on peut exprimer par la formule suivante :

$$a_{y_{\max k}} = \rho_k V_x^2, \quad (5.12)$$

La limitation de cette accélération revient à limiter la courbure maximale de chaque courbe de Bézier générée ρ_k entre deux bornes $\pm \rho_{\max}$. Cette limite maximale de la courbure ρ_{\max} peut être exprimée géométriquement en fonction de l'angle de braquage maximal du véhicule δ_{\max} sous la forme suivante :

$$\rho_{\max} = \frac{2 \tan \delta_{\max}}{L}, \quad (5.13)$$

Par ailleurs, cette courbure maximale est liée aussi aux conditions d'adhérence au niveau du contact roue/sol. En effet, la force de frottement F_{frott} maximale peut s'exprimer en fonction de la force de gravité de la manière suivante :

$$F_{frott} = mg\mu \geq ma_y, \quad (5.14)$$

où m est la masse du robot, g la constante de gravitation et μ le coefficient de frottement.

Par la suite, la courbure maximale peut s'exprimer autrement par la formule suivante :

$$\rho_{\max} = \frac{\mu g}{V_x^2}, \quad (5.15)$$

Enfin, la contrainte sur la courbure maximale ρ_k le long de chaque courbe de Bézier S_k peut être déduite comme suit,

$$|\rho_k| \leq \min \left[\frac{\mu g}{V_x^2}, \frac{2 \tan \delta_{\max}}{L} \right], \quad \forall k \in \llbracket 1, n^2 \rrbracket, \quad (5.16)$$

Une fois qu'un ensemble de m courbes de Bézier cubiques satisfaisant la contrainte (5.16) est déterminé ($m \leq n^2$), le meilleur chemin local sélectionné pour éviter l'obstacle est celui qui a une courbure minimale parmi les m courbes choisies.

$$k^* = \arg \min_{\forall k \in \llbracket 1, m \rrbracket} |\rho_k|, \quad (5.17)$$

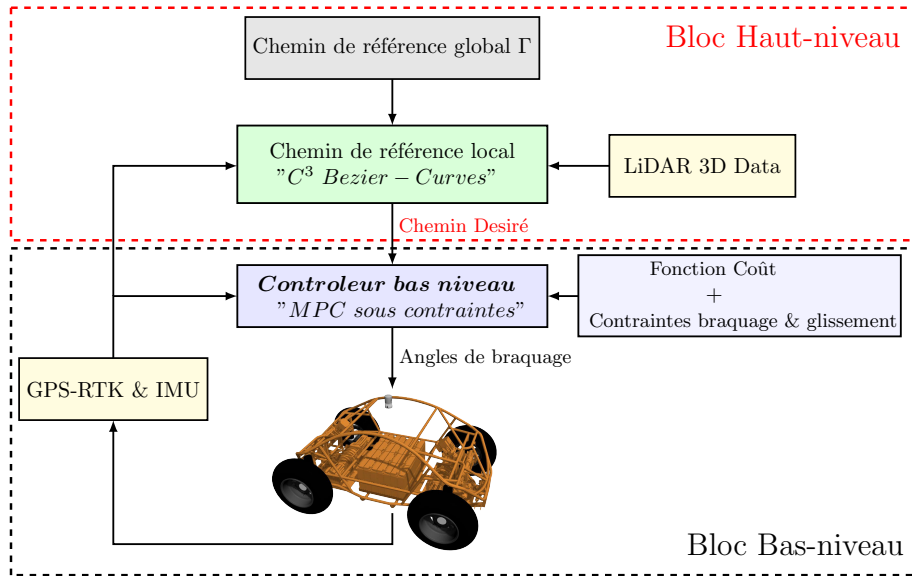


Figure 5.15. Schéma global du contrôle. Le contrôleur de haut-niveau permet la perception de l'environnement en utilisant le capteur Velodyne pour la détection d'obstacles, puis la génération d'un chemin de référence local pour la tâche d'évitement. Ce chemin planifié est utilisé comme consigne à exécuter par le contrôleur de bas-niveau qui calcule les bons angles de direction.

La variation de courbure $\Delta\rho_{j,k}$ doit être également limitée afin de respecter les limites en vitesse des actionneurs. Néanmoins, le contrôleur bas-niveau applique déjà des contraintes sur le taux de changement des angles de direction avant et arrière $\Delta\delta_{(f,r)}$ (voir la section 3.2.2) ce qui reste suffisant. Le meilleur chemin de contournement pour la première partie est le chemin d'indice k^* (S_{k^*}). Les mêmes critères de sélection sont appliqués sur la deuxième tranche du chemin de contournement.

La tâche de suivi de ce chemin de contournement est assurée par la commande MPC sous contraintes. L'architecture globale de nos contrôleurs est illustrée par la Figure 5.15.

Les **algorithmes 2 & 3**, donnés dans l'annexe B, récapitulent notre stratégie de planification locale de chemin de contournement qui se base principalement sur la détermination faisable des points de passage (P_b , P_m et P_f); ceci fait l'objet de l'**algorithme 2**, puis le calcul de deux meilleures courbes de Bézier cubiques reliant ces points de passage (P_b , P_m et P_f) tout en respectant les contraintes cinématiques et dynamiques du véhicule (**algorithme 3**).

Dans les parties qui viennent, nous présenterons la validation de notre stratégie de navigation. Tout d'abord, nous introduisons les résultats obtenus par simulation sous ROS-GAZEBO, puis ensuite les résultats expérimentaux en utilisant le robot "SPIDO".

5.2 Résultats de simulations numériques

Dans un premier temps, nous validons notre stratégie de navigation avec la plateforme virtuelle sous ROS-GAZEBO de la Figure 1.4(a). Afin d'évaluer la robustesse de nos algorithmes, les obstacles sont positionnés d'une manière aléatoire dans l'environnement de simulation autour du chemin de référence global (e.g., Figure 5.6).

Pour la partie de simulation, nous définissons les paramètres suivants : période d'échantillonnage : $T_d = 0,2s$, contraintes relatives aux extrémums des angles de direction : $\pm 10^\circ$, contraintes relatives aux extrémums de la variation de la commande : $\pm 3^\circ$ et l'horizon de prédiction $N_p = 30$.

Le coefficient de frottement a été choisi constant ($\mu = 0,35$). Les rigidités de dérive avant et arrière C_f et C_r ont été supposées connues et sont observées grâce à l'observateur non linéaire (NLO) proposé dans le chapitre 3. Ils convergent vers la même valeur constante ($C_{(f,r)} \approx 16\,000\text{ N.rad}^{-1}$). Ainsi, les contraintes relatives aux extrémums sur les angles de glissement transversaux peuvent être approximées par l'intermédiaire de la force verticale F_z représentant le poids total du véhicule équitablement réparti entre les essieux avant et arrière, celles-ci peuvent être exprimées par la formule suivante :

$$\beta_{(f,r)}^{\max} = \frac{\mu F_z}{2C_{(f,r)}} = \frac{\mu mg}{2C_{(f,r)}} \approx 6^\circ, \quad (5.18)$$

Nous testons nos algorithmes sur quatre chemins de référence globaux qui ont été définis hors-ligne $(x, y, \psi)_d$ ⁶ :

1. "**Chemin rectiligne**" (cf. Figure 5.16(a))
2. "**Chemin en O**" de faible rayon de courbure afin d'exciter la dynamique du véhicule (cf. Figure 5.19(a)).
3. "**Chemin en S**" comprenant deux virages lisses gauche-droite à faible rayon courbure (cf. Figure 5.22(a)).
4. "**Chemin en Z**" comprenant deux manœuvres agressives gauche-droite (cf. Figure 5.25(a)).

Nous commençons d'abord par l'évaluation du temps de calcul nécessaire pour générer le chemin de contournement (constitué de deux courbes de Bézier cubiques). Plusieurs tests ont été réalisés en variant le nombre de courbes de Bézier cubique généré sur chaque tranche (n^2) qu'on résume dans le Tableau 5.1. En effet, ce temps de calcul dépend de deux facteurs (hors du langage de programmation utilisé (Python) et de la capacité calculatoire de la machine utilisée - GPU, carte graphique). Le premier est le nombre de courbes générés n^2 qui correspond au nombre de fois de variation des points de contrôle intermédiaires. Le second est le temps nécessaire pour évaluer ces chemins générés des deux tranches pour en sélectionner le meilleur qui respecte les contraintes cinématiques et dynamique du véhicule.

Tableau 5.1. Temps de calcul du chemin de contournement.

Nombre de courbes n^2	Temps de calcul (s)
25	0.029
100	0.061
225	0.105
400	0.183
625	0.262
900	0.351

6. Visionner la vidéo suivante sur la validation en simulation numérique de la stratégie de planification locale de chemin d'évitement sur la maquette virtuelle : [▶▶▶▶▶▶▶▶▶▶](#)

5.2.1 Chemin de référence rectiligne

Nous allons tester dans un premier temps nos contrôleurs de haut-niveau et bas-niveau présentés par la Figure 5.15 sur un "*Chemin rectiligne*" sur lequel on place arbitrairement des obstacles.

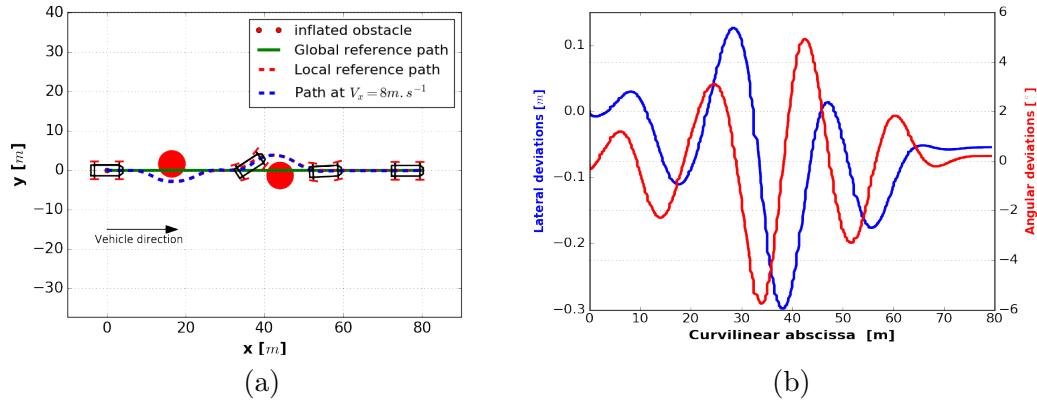


Figure 5.16. Résultats sur le "*Chemin rectiligne*" à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé ; (b) Écart latéral et angulaire de suivi de chemin.

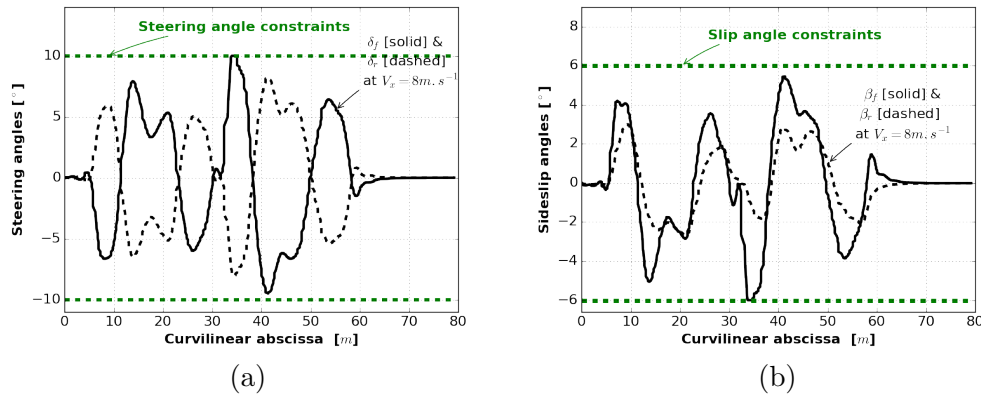


Figure 5.17. Résultats de simulation sur le "*Chemin rectiligne*" à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière ; (b) Angles de glissement avant et arrière.

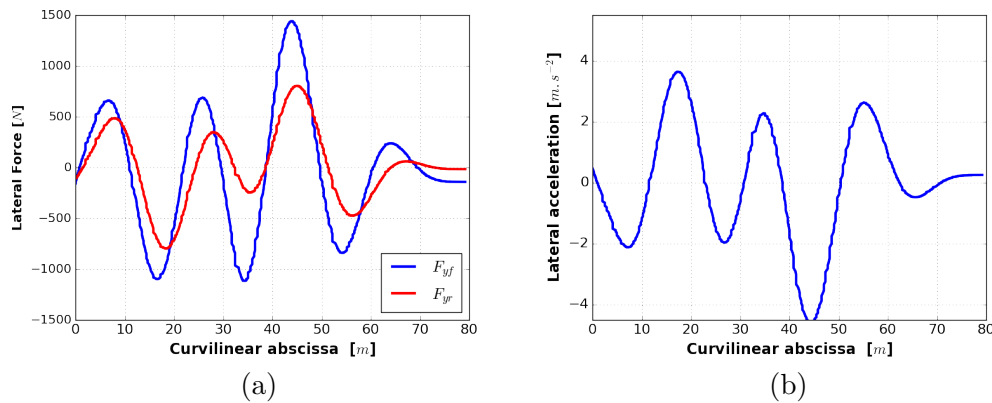


Figure 5.18. Résultats de simulation sur le "*Chemin rectiligne*" à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière ; (b) Accélération latérale.

5.2.2 Chemin en O

Notre stratégie de navigation est appliquée aussi sur un autre "*Chemin en O*" dont la courbure est importante afin de montrer l'efficacité des contrôleurs haut-niveau et bas-niveau de générer le chemin de contournement et d'assurer sa poursuite.

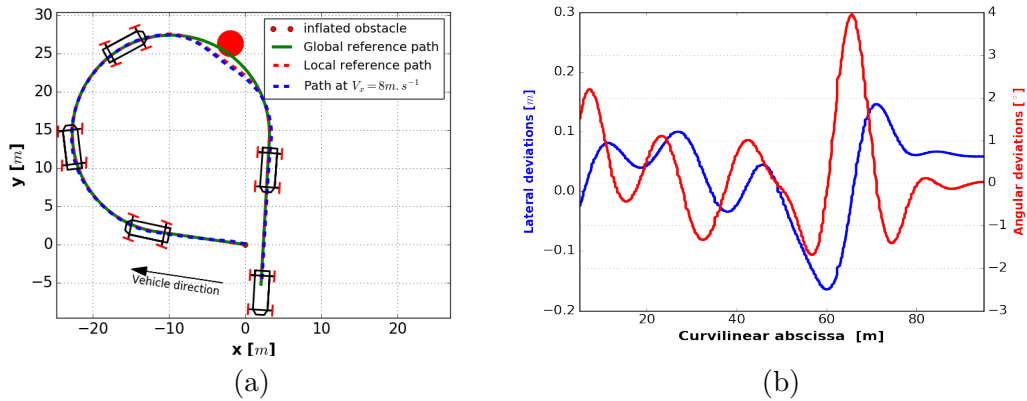


Figure 5.19. Résultats sur le "*Chemin en O*" à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé ; (b) Écart latéral et angulaire de suivi de chemin.

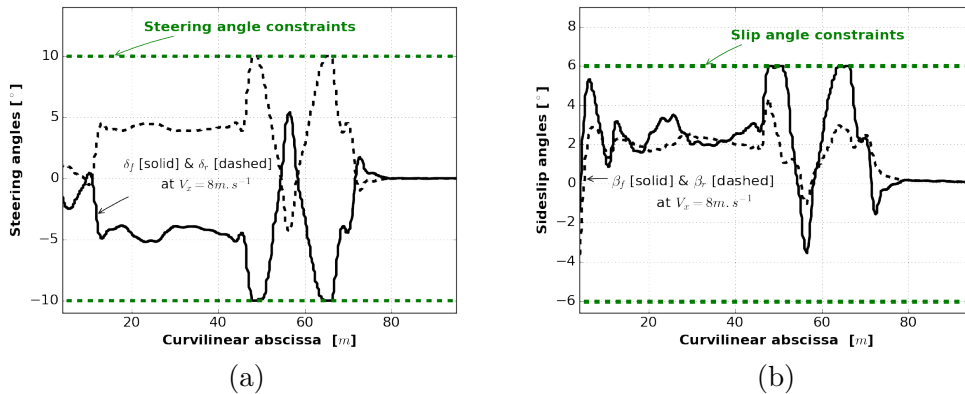


Figure 5.20. Résultats de simulation sur le "*Chemin en O*" à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière ; (b) Angles de glissement avant et arrière.

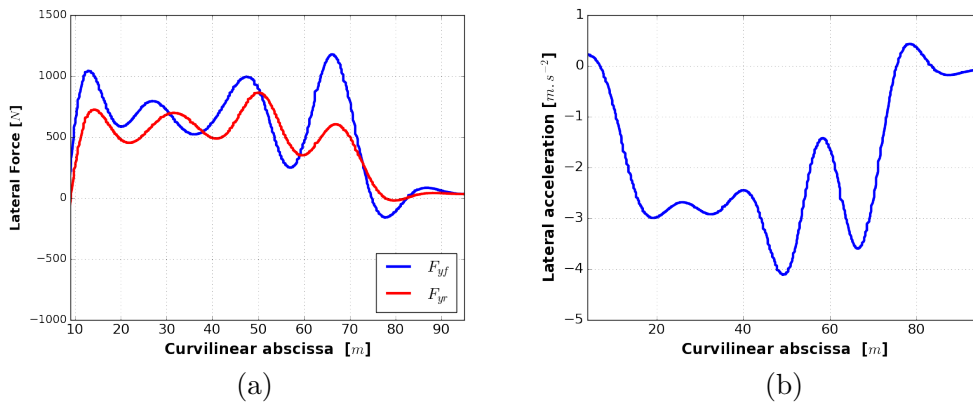


Figure 5.21. Résultats de simulation sur le "*Chemin en O*" à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière ; (b) Accélération latérale.

5.2.3 Chemin en S

Nous allons tester un autre "*Chemin en S*" avec deux virages lisses à gauche et droite pour évaluer nos algorithmes d'évitement d'obstacle et de suivi de chemin.

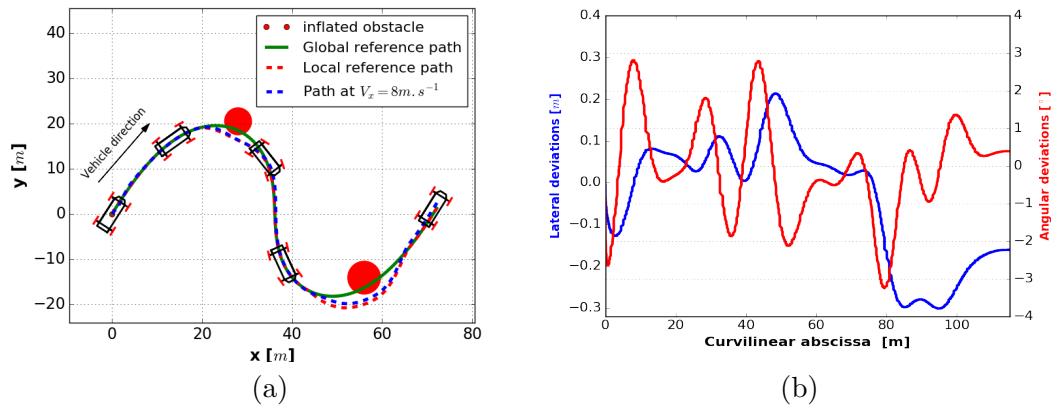


Figure 5.22. Résultats sur le "*Chemin en S*" à $V_x = 8 \text{ m.s}^{-1}$: (a) Chemin de référence global et local et chemin réalisé ; (b) Écart latéral et angulaire de suivi de chemin.

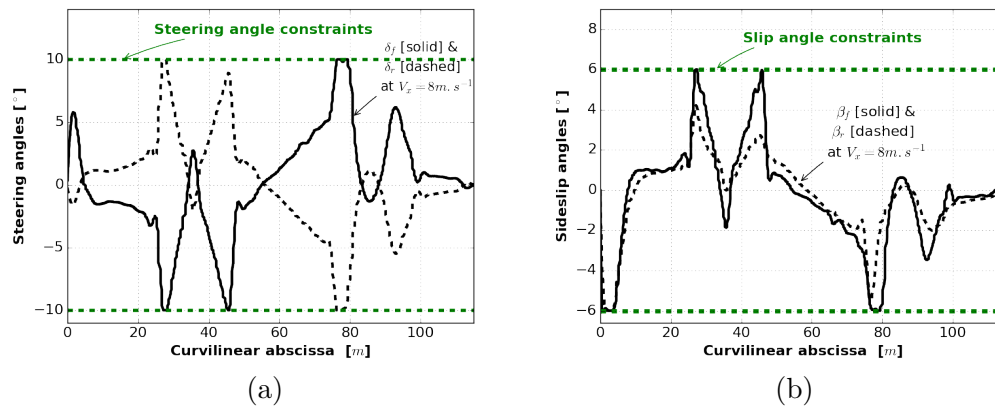


Figure 5.23. Résultats de simulation sur le "*Chemin en S*" à $V_x = 8 \text{ m.s}^{-1}$: (a) Angles de braquage avant et arrière ; (b) Angles de glissement avant et arrière.

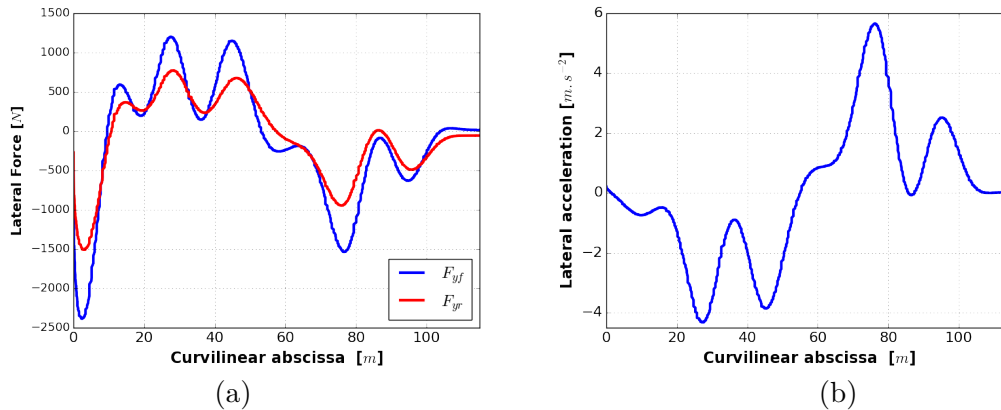


Figure 5.24. Résultats de simulation sur le "*Chemin en S*" à $V_x = 8 \text{ m.s}^{-1}$: (a) Efforts latéraux avant et arrière ; (b) Accélération latérale.

5.2.4 Chemin en Z

Pour aussi mettre en évidence l'apport de notre stratégie de navigation avec évitement d'obstacles, on a choisi un autre "*Chemin en Z*" pour pouvoir tester le braquage à droite et à gauche et le changement rapide et agressif de la courbure sur nos algorithmes.

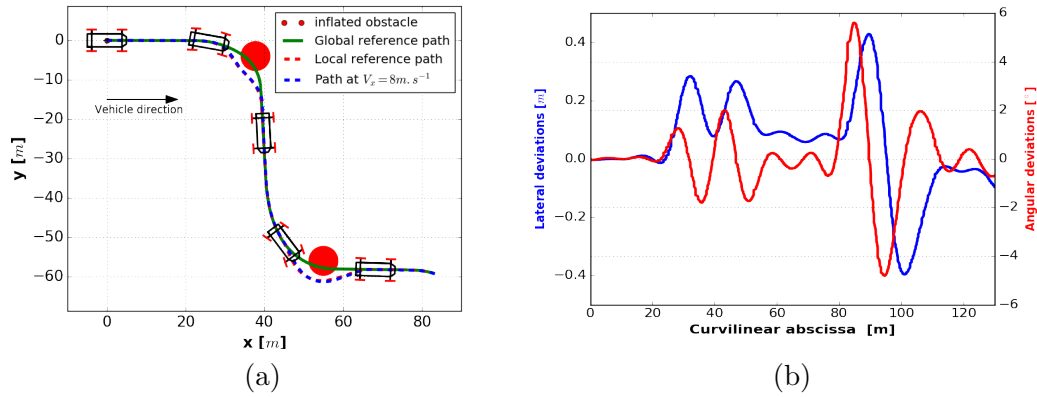


Figure 5.25. Résultats sur le "*Chemin en Z*" à $V_x = 8m.s^{-1}$: (a) Chemin de référence global et local et chemin réalisé ; (b) Écart latéral et angulaire de suivi de chemin.

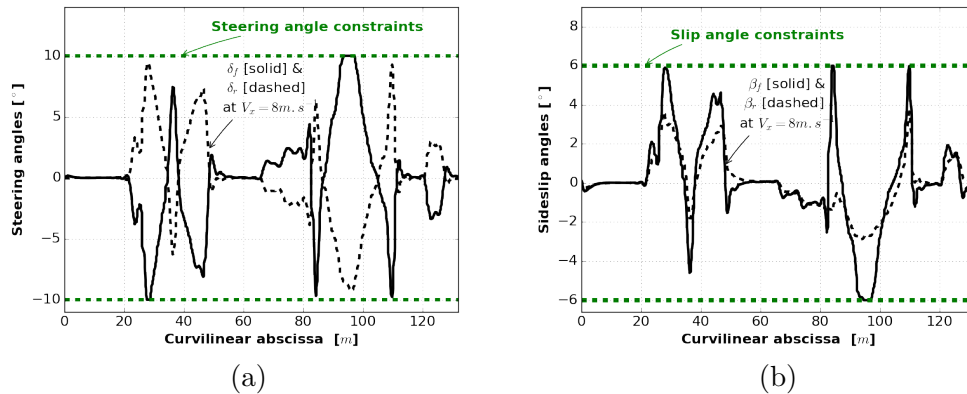


Figure 5.26. Résultats de simulation sur le "*Chemin en Z*" à $V_x = 8m.s^{-1}$: (a) Angles de braquage avant et arrière ; (b) Angles de glissement avant et arrière.

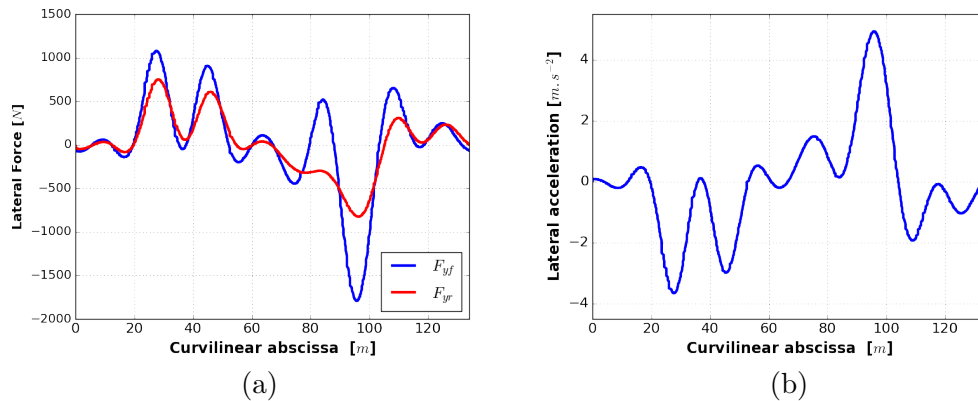


Figure 5.27. Résultats de simulation sur le "*Chemin en Z*" à $V_x = 8m.s^{-1}$: (a) Efforts latéraux avant et arrière ; (b) Accélération latérale.

5.2.5 Discussions des résultats de simulation

La réduction du nombre de courbes de Bézier cubiques n^2 permet de diminuer le temps nécessaire pour avoir un chemin de contournement réalisable par le véhicule. Pour notre implémentation, nous choisissons $n = 10$ qui correspond à un temps de calcul de l'ordre de $61ms$ (voir Tableau 5.1). Ce temps de calcul représente une distance parcourue de $0.4m$ quand la vitesse du robot est de $V_x = 8m.s^{-1}$.

Notre algorithme d'évitement d'obstacles a été testé sur la maquette virtuelle sous ROS-GAZEBO avec un nombre de courbes de Bézier cubiques $n^2 = 100$ générés automatiquement et en temps-réel une fois la distance de sécurité violée pour chaque partie du chemin local de contournement. Ces courbes sont générées à partir de la posture initiale et finale du véhicule en respectant la courbure maximale que le véhicule peut exécuter et en évitant les variations agressives des angles de braquage qui pourraient entraîner l'instabilité du véhicule. Il s'avère aussi nécessaire de choisir la meilleure courbe favorisant des courbures faibles ainsi que la convergence rapide vers le chemin de référence global.

Les Figures 5.16(a), 5.19(a), 5.22(a) et 5.25(a) montrent que l'algorithme fonctionne convenablement. Le véhicule évite les obstacles placés arbitrairement autour des chemins de référence globaux puis revient sur ces chemins après la tâche d'évitement. Les deux tâches principales (suivi de chemin et évitement d'obstacles) sont ainsi réalisées en utilisant les deux meilleures courbes de Bézier cubiques qui respectent les contraintes cinématiques et dynamiques du véhicule. Nous notons aussi que l'algorithme de bas-niveau (MPC sous contraintes) assure un bon suivi des différents chemins de référence désirés à $V_x = 8m.s^{-1}$.

Ce suivi de chemin est assuré avec des faibles erreurs latérales et angulaires pour les différents scénarios étudiés, comme on peut le constater sur les Figures 5.16(b), 5.19(b), 5.22(b) et 5.25(b). Nous remarquons que ces erreurs sont importantes le long du chemin local de contournement où la distance de sécurité est non respecté. Ceci est dû aux grandes accélérations latérales le long des petits rayons du courbures, comme on peut le notifier pour chaque scénario sur les Figures 5.18(b), 5.21(b), 5.24(b) et 5.27(b). Par ailleurs, les forces latérales au niveau du contact roue/sol sont élevées le long des chemins de contournement à cause de la prédominance du phénomène de glissement (cf. Figures 5.18(a), 5.21(a), 5.24(a) et 5.27(a)).

En générale, nous constatons que l'évitement est convenablement réalisé. Le véhicule revient vers le chemin de référence global après la tâche d'évitement. La commande MPC sous contraintes permet au robot de suivre efficacement les différents chemins désirés. La courbure du chemin local d'évitement varie de manière continue tout en respectant les contraintes cinématiques et dynamiques du véhicule.

Les Figures 5.17(a), 5.20(a), 5.23(a) et 5.26(a) illustrent les angles de direction avant et arrière pour chacun des chemins globaux testés. Ces angles sont calculés par le contrôleur bas-niveau pour assurer la tâche de suivi de chemin. Nous constatons que ces angles de braquage varient de manière lisses et moins agressifs. Ainsi ils sont toujours limités entres les bornes fixées $\pm 10^\circ$ à chaque pas de calcul. Nous notons également que ces limites sont atteintes principalement le long du chemin local planifié qui nécessite un braquage parfois important pour assurer une déviation rapide auprès de l'obstacle détecté et la convergence tout aussi rapide vers le chemin de référence global.

les Figures 5.28(b) et 5.31(b) (à l'abscisse curviligne 30m, la déviation latérale est de 0.8 m à $V_x = 2.5 \text{ m.s}^{-1}$ et de 1,1 m à $V_x = 4 \text{ m.s}^{-1}$, alors que la déviation angulaire reste aux alentours de 5° à $V_x = 2,5 \text{ m.s}^{-1}$ et $V_x = 4 \text{ m.s}^{-1}$).

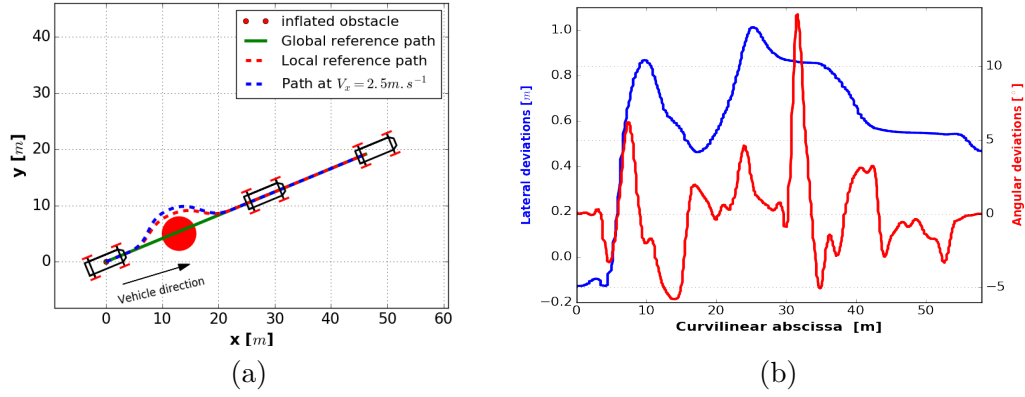


Figure 5.28. (a) Chemin de référence global et local et chemin réalisé à $V_x = 2.5 \text{ m.s}^{-1}$; (b) Écart latéral et angulaire de suivi de chemin à $V_x = 2.5 \text{ m.s}^{-1}$.

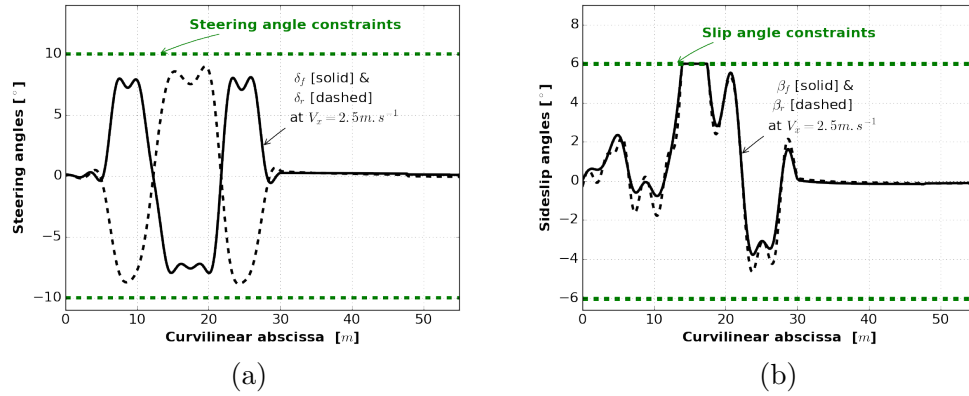


Figure 5.29. (a) Angles de braquage avant et arrière à $V_x = 2.5 \text{ m.s}^{-1}$; (b) Angles de glissement avant et arrière à $V_x = 2.5 \text{ m.s}^{-1}$.

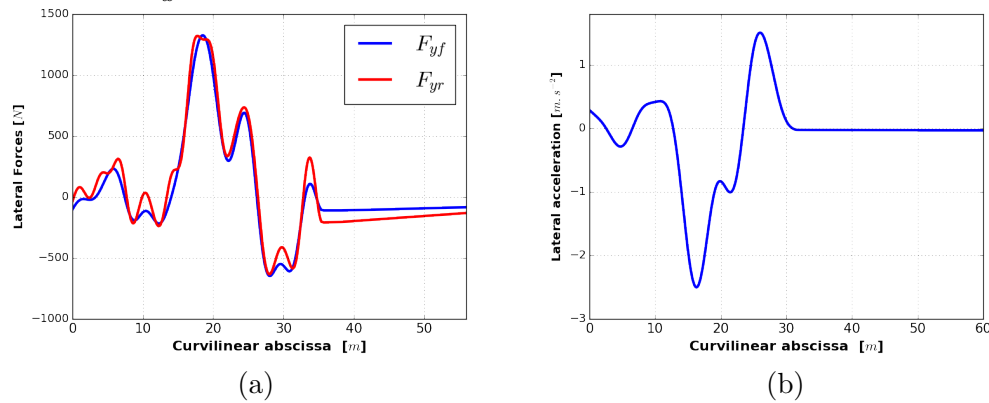


Figure 5.30. (a) Efforts latéraux avant et arrière à $V_x = 4 \text{ m.s}^{-1}$; (b) Accélération latérale à $V_x = 4 \text{ m.s}^{-1}$.

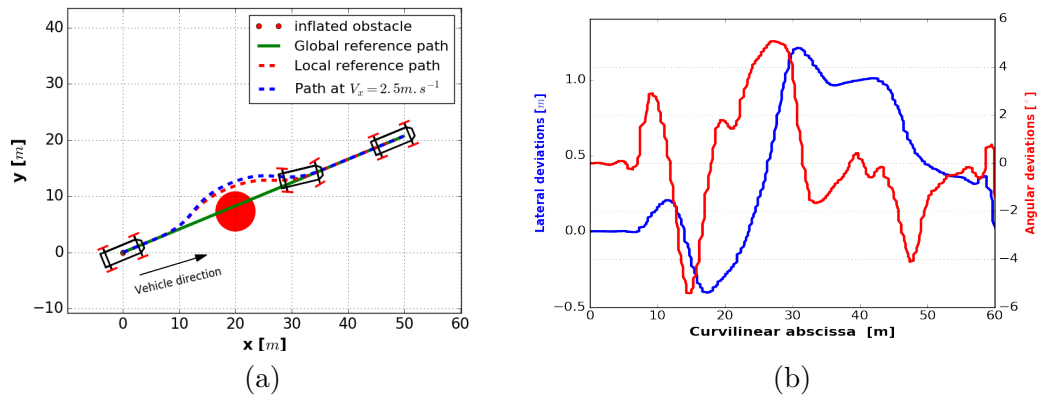


Figure 5.31. (a) Chemin de référence global et local et chemin réalisé à $V_x = 4m.s^{-1}$; (b) Écart latérale et angulaire de suivi de chemin à $V_x = 4m.s^{-1}$.

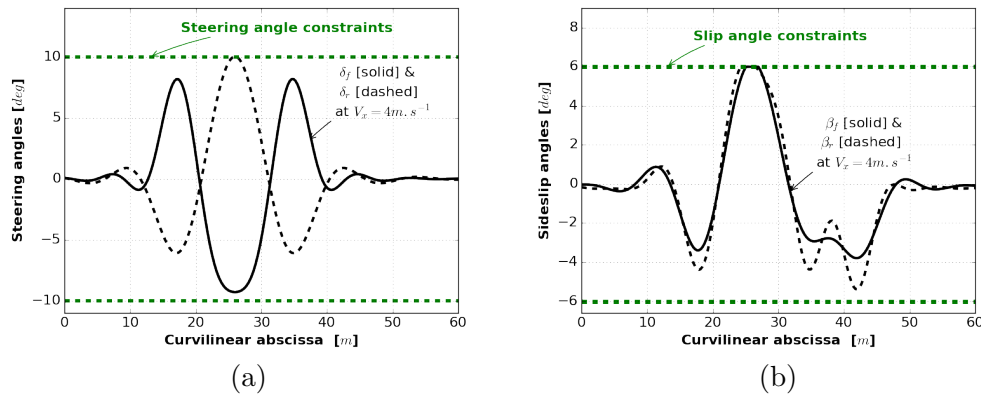


Figure 5.32. (a) Angles de braquage avant et arrière à $V_x = 4m.s^{-1}$; (b) Angles de glissement avant et arrière à $V_x = 4m.s^{-1}$.

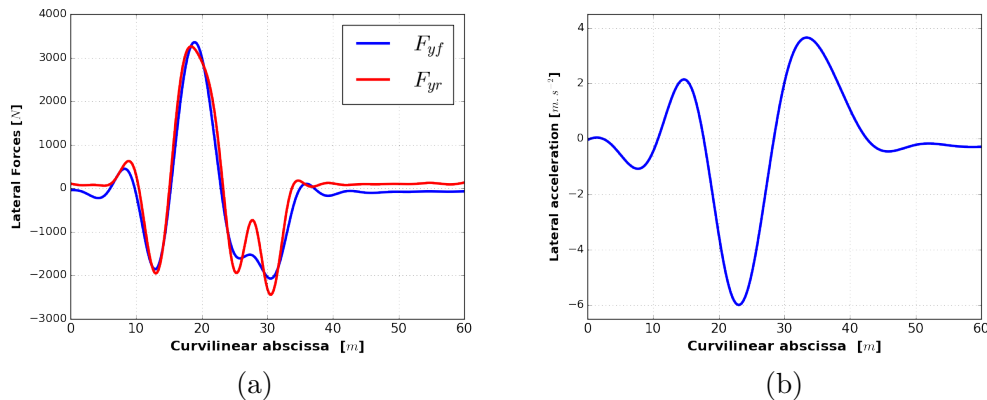


Figure 5.33. (a) Efforts latéraux avant et arrière à $V_x = 4m.s^{-1}$; (b) Accélération latérale à $V_x = 4m.s^{-1}$.

Les angles de braquage avant et arrière calculés par le contrôleur MPC sous contraintes aux vitesses longitudinales $V_x = 2,5 m.s^{-1}$ et $V_x = 4 m.s^{-1}$ sont illustrés par les Figures 5.29(a) et 5.32(a). Ils varient d'une façon lisse et douce le long des deux courbes de Bézier. Aussi, les angles de dérive avant et arrière, quant à eux, sont tracés sur les Figures 5.29(b) et 5.32(b) à $V_x = 2,5 m.s^{-1}$ et $V_x = 4 m.s^{-1}$. Comme on peut le remarquer, ces angles sont plus grands le long du chemin planifié en raison de la haute accélération latérale. Celle-ci est importante quand

le robot cherche à dévier l'obstacle détecté comme montré par les Figures 5.30(b) et 5.33(b), ce qui génère aussi des fortes forces latérales au niveau du contact roue/sol (cf. Figures 5.30(a) et 5.33(a)).

Pour résumer, les résultats fournis par notre approche de navigation sont assez satisfaisants. Cette technique permet au véhicule de gérer avec succès différents types de suivi de chemin à différentes vitesses de déplacement tout en évitant les collisions et en respectant les limites des actionneurs et les contraintes de glissement. Grâce au système de double direction (4WS), le robot peut éviter facilement les obstacles détectés avec des angles de direction petits et lisses, ainsi qu'avec un glissement latéral faible.

6 Conclusion

En ce qui concerne la détection d'obstacles, nous avons développé un algorithme basé sur les données d'un LiDAR Velodyne. Afin de pouvoir évaluer l'apport de cet algorithme de détection, nous avons calculé le temps de calcul nécessaire pour le traitement du nuage de points. Ce temps reste aux alentours de $0.096s$, cela correspond à un déplacement de $0.35m$ à la vitesse $V_x = 4m.s^{-1}$. Cet algorithme a été validé en boucle ouverte en simulation et expérimentalement sur la plateforme réelle "SPIDO". Par ailleurs, nous envisageons à court terme de développer une autre méthode de détection d'obstacles moins gourmande en temps (sans passer par la librairie PCL) qui a pour objectif d'identifier en temps-réel les surfaces verticales dans l'environnement afin de détecter les obstacles présents (e.g., murs, piétons, tronc d'arbre, etc) à partir des données LiDAR dont les informations de base sont des points situés et classés dans des plans verticaux. Cela permettrait une réduction significative de la charge calculatoire de l'ensemble de la chaîne de perception de l'environnement.

En ce qui concerne la planification locale des chemins, nous avons développé un algorithme permettant le contournement efficace des obstacles par le biais de deux courbes de Bézier cubiques conçues en ligne et automatiquement dès que la distance de sécurité est violée. Ce nouvel algorithme d'évitement d'obstacle est couplé au contrôleur bas-niveau (MPC sous contraintes) afin de permettre au véhicule d'assurer la tâche de suivi de chemin. Nos différentes stratégies ont été validées, tout d'abord, en simulation sous ROS-GAZEBO via une maquette virtuelle sur quatre chemins globaux avec des caractéristiques différentes, ensuite, des validations expérimentales sur la plateforme rapide réelle. Les résultats obtenus montrent que nos approches sont efficaces vis-à-vis des objectifs attendus d'évitement réactif d'obstacles et de suivi de chemin.

Par ailleurs, nous allons tester notre algorithme d'évitement dans le cas de présence des nouveaux obstacles sur le chemin de contournement, cela nécessitera d'améliorer la technique d'extraction des points de passage. D'autre part, nous allons étendre cette technique de planification locale pour éviter en temps réel des obstacles mobiles, cela dépendra de la vitesse et de la direction de l'obstacle dynamique.

Conclusions et perspectives

Au cours de ce travail de thèse, nous nous sommes intéressés à la dynamique d'un rover rapide à deux trains directeurs destiné à se déplacer rapidement sur des terrains extérieurs qui peuvent présenter localement des irrégularités (pente, par exemple) ou des obstacles francs et non prévus initialement sur le chemin de consigne. Nous avons développé plusieurs algorithmes de commande intégrant les contraintes du système ainsi que l'estimation temps réel des paramètres du terrain. Nous avons également développé un algorithme de navigation pour l'évitement réactif d'obstacle. Cet ensemble de fonctions permet d'assurer des déplacements autonome d'un véhicule rapide tout en garantissant sa stabilité en virage et en pente, le maintien du véhicule dans la zone d'adhérence et la non-collision avec les obstacles avoisinants. Chacune de ces fonctions a été évaluée et validée en simulation puis lors d'expérimentations sur le démonstrateur "*SPIDO*".

1 Synthèse des travaux réalisés

Afin d'asservir avec précision le robot mobile sur le chemin désiré, il a été nécessaire de prendre en compte les dynamiques importantes (liées à la vitesse) et variables (liées à l'interaction variable avec l'environnement) dans un modèle représentant la dynamique de véhicule et prenant en compte un modèle linéaire du pneumatique ainsi que les effets de gravité dus aux irrégularités du terrain. Afin de disposer d'une commande de suivi de chemin pour valider les algorithmes d'estimation des paramètres géométriques et physiques du sol, une commande basée sur l'approche LQR et mixée avec un observateur d'état de Kalman-Bucy a été synthétisée et développée dans le chapitre 2. Les résultats de simulation et expérimentaux ont montré que cette technique permet d'avoir une bonne qualité de suivi de chemin. Néanmoins, sur un plan expérimental, cette commande LQR s'est avérée assez oscillante. Par ailleurs, elle ne prend en compte aucune contrainte intrinsèque du système. C'est pourquoi, nous avons opté pour une commande prédictive (MPC) basée sur la prédiction de la sortie sur un horizon de temps fini

afin de pouvoir anticiper sur la position future du robot et compenser le retard du système. D'autre part, ce type de commande permet d'intégrer aisément les différentes contraintes. Dans le chapitre 4, une commande MPC sous contraintes a été développée pour répondre à ce besoin. Elle prend en compte les contraintes intrinsèques du système et d'adhérence du contact roue/sol (i.e., limites des angles de braquage et de glissement). Cette stratégie est présentée sous la forme d'un problème quadratique (QP) qui minimise une fonction coût bien définie soumise aux différentes contraintes intrinsèques et physiques du véhicule (limites des angles de braquage et leurs vitesses ainsi que le respect des limites de la zone d'adhérence du pneu). Ce problème QP a été implémenté en utilisant un solveur d'optimisation convexe qui donne une solution à chaque pas de calcul. Cette commande a été validée en simulation et expérimentalement. Les résultats ont montré que cette stratégie assure un suivi de chemin précis et stable tout en respectant toutes les contraintes imposées du système.

Au cours d'un déplacement rapide dans des environnements accidentés où les conditions d'adhérence sont fortement variables, les performances de suivi de chemin peuvent être dégradées si les paramètres physiques et géométriques du sol ne sont pas bien identifiés en temps-réel et compensés dans les contrôleurs de suivi de chemin. Ainsi, il est indispensable d'avoir accès à une estimation en ligne de ces paramètres intervenant dans la dynamique du véhicule lors de son évolution et de reconstruire toutes ces variables non directement mesurables via des techniques d'observation et d'estimation. Pour ce faire, deux observateurs de paramètres géométriques (angle de pente et angle de dévers) et physiques (liés à l'adhérence des roues sur le sol) sont synthétisés dans le chapitre 3. Le premier observateur est un observateur non linéaire (NLO) qui a été développé en se basant sur le modèle dynamique en lacet du véhicule et d'un modèle linéaire de pneumatique. Celui-ci a pour objectif de reconstruire indépendamment et en temps réel les rigidités de dérive avant et arrière à la liaison roue/sol tout en exploitant les variables mesurées ou estimées de la vitesse latérale, des angles de braquage et de la vitesse de lacet. Les valeurs estimées de rigidité de dérive sont ensuite utilisées en temps réel dans les algorithmes de suivi de chemin. Par ailleurs, pour pouvoir compenser les effets liés à la gravité dus à la géométrie locale du terrain, un deuxième observateur a été synthétisé pour estimer en temps réel l'angle de pente et de dévers du véhicule. Cet observateur est basé sur la théorie de Luenberger et utilise les mesures inertielles pour estimer les angles de la géométrie locale du terrain qui sont utilisés dans les lois de commande de suivi de chemin. Les essais ont montré que l'intégration de ces observateurs permet une estimation correcte des paramètres de contact et de sol, ce qui améliore grandement la précision du suivi.

À un plus haut niveau de commande, nous avons proposé un générateur local de trajectoires permettant la synthèse de courbes géométriques plusieurs fois dérivables qui respectent la courbure admissible par le rover ainsi que les conditions de raccordement au chemin de référence global. L'évitement d'obstacles est basé sur la génération automatique de deux courbes de Bézier cubiques (un chemin d'évitement et un chemin de raccordement). Ces deux courbes permettent d'avoir un chemin de contournement à courbure minimale tout en respectent les contraintes cinématiques et dynamiques du rover et les conditions de continuité avec le chemin nominal. Elles sont générées automatiquement en fonction de plusieurs points de passage qui sont extraits en temps-réel selon la position de l'obstacle détecté et l'état du véhicule (e.g., direction du véhicule, positions actuelles, angles de braquage, etc.). Cet algorithme local d'évitement est couplé à la commande MPC sous contraintes développée dans le chapitre 4 pour assurer la tâche de suivi de chemin local ou global. Les résultats de la simulation sur plusieurs scénarios de chemins de référence globaux et à des vitesses allant jusqu'à $8m.s^{-1}$ montrent que nos algorithmes donnent de

bons résultats vis-à-vis des objectifs attendus tant au niveau de l'évitement qu'au niveau du suivi du chemin local ou global. Enfin, une évaluation expérimentale des algorithmes développés a été réalisée avec succès sur le démonstrateur expérimental avec des vitesses allant jusqu'à $4m.s^{-1}$. Cette évaluation montre que dès qu'un obstacle est détecté sur le chemin nominal du véhicule, les algorithmes choisissent deux meilleures courbes de Bézier cubiques (évitement & ralliement) qui correspondent aux restrictions cinématiques et dynamiques du véhicule, montrant ainsi la faisabilité et l'efficacité des solutions proposées.

2 Perspectives

Plusieurs améliorations et développements restent à poursuivre pour aboutir à une navigation rapide et autonome de rover en milieu naturel.

Amélioration de la localisation rapide du véhicule

Au cours de nos expérimentations, nous avons pu noter que, au niveau de la géolocalisation, les informations de positionnement apportées actuellement par un GPS-RTK ne sont que partielles et demeurent insatisfaisantes en tant que données extéroceptives seules (présence de bâtiments et arbres, masquages satellites, pertes de précisions, changement de constellation, apparitions des données aberrantes, etc.). Il est donc nécessaire de pallier à ces défauts pour avoir à tout moment une estimée de position localement précise du véhicule. Dans une perspective d'amélioration de suivi de chemin sur le rover "*SPIDO*", une implémentation d'une fusion de données odométriques, inertielle et GPS par le biais d'un filtre de Kalman étendu (EKF) a été initiée (voir annexe C) mais n'a pas encore pu être testée.

Amélioration des performances des observateurs

Au niveau estimation des paramètres d'environnement, le temps de convergence de l'observateur non linéaire des coefficients de rigidité de dérive $C_{(f,r)}$, synthétisé dans le chapitre 3, dépend largement des conditions initiales données à ces coefficients. Afin de pouvoir réduire le temps de réponse de cet observateur NLO, il serait judicieux d'utiliser une caméra installée sur le robot afin d'estimer le type de sol à l'avant du robot grâce à une analyse de la texture des images collectés couplée à une classification par un apprentissage hors-ligne.

Il serait également intéressant de synthétiser un estimateur de coefficient de frottement μ au niveau de la liaison roue/sol en utilisant un modèle longitudinal du véhicule et un modèle non linéaire du contact. Cette estimation permettrait, d'une part, d'approximer en temps réel les bornes de la zone d'adhérence des pneumatiques ($\beta_{(f,r)}^{max} \approx \frac{\mu mg}{2C_{(f,r)}}$) qui sont utilisées en ligne dans le problème QP de la commande MPC sous contraintes. D'autre part, ce coefficient peut être utilisé pour calculer la contrainte imposée sur la vitesse longitudinale V_x (voir ci-dessous).

Prise en compte de la dynamique longitudinale et de roulis

Au niveau de la commande, la dynamique longitudinale (contrôle de V_x) n'a pas été considérée dans cette étude. Il n'est pas tout à fait exact qu'elle soit découplée de la dynamique latérale. Les forces de traction générées par les roues contribuent aussi bien à l'avancement qu'à la manœuvrabilité. A notre connaissance, ce problème n'a jamais été traité dans sa totalité et a toujours été abordé d'une façon découplée. Le contrôle de la dynamique longitudinale peut être intégré au contrôleur MPC sous contraintes afin d'améliorer ses performances, notamment la précision et la convergence de nos algorithmes principalement le long des virages où la courbure est maximale (régime transitoire). Une telle commande présente également un intérêt en terme d'évitement d'obstacles en permettant au robot de décélérer une fois un obstacle est détecté. Cette vitesse longitudinale doit toujours remplir une inégalité qui dépend surtout de la courbure ρ du chemin désiré et le coefficient de frottement μ : $V_x \leq \sqrt{\frac{\mu g}{\rho}}$.

Enfin, une autre perspective de ce travail serait de développer une nouvelle commande MPC, intégrant les actions sur les deux barres anti-roulis, dans un paradigme de contrôle corps complet qui ne découple pas les dynamiques et les couplages inertiels intrinsèques. Un résultat intéressant serait d'anticiper le virage pour décaler le centre de masse vers l'intérieur de celui-ci afin de réduire le transfert de charge latérale (TCL), qui représente la différence entre les efforts normaux exercés sur les deux roues d'un même essieu. Cette commande nécessiterait de prendre en compte un modèle à 4 roues soit grâce à une linéarisation sur un horizon de prédiction court soit par l'utilisation de "solver" d'optimisation non-linéaire.

Exploitation des données 3D LiDAR pour la détection d'obstacles et leur classification

L'algorithme de détection développé dans cette thèse dans le chapitre 5 utilise les données LiDAR 3D sous forme des données 2D après la projection dans le plan horizontal du capteur via la librairie PCL et le package Height map. Nous envisageons de développer un algorithme qui permet d'identifier en temps-réel les surfaces verticales autour du véhicule en travaillant directement avec les données brutes de LiDAR 3D (sans passer par la librairie PCL) afin de réduire la charge calculatoire de l'ensemble de la chaîne de perception. Un challenge est de pouvoir le faire d'une façon efficace et rapide en temps-réel et sans sous-échantillonnage.

Enfin, il serait extrêmement intéressant d'étendre ce travail au sol irrégulier et d'étudier la traversabilité d'une zone donnée compte tenue de sa géométrie et de sa nature physique. En effet, la notion d'obstacle dans un milieu naturel est une notion difficile à définir, car elle dépend de la capacité du robot à franchir, voire aussi de ses paramètres d'état tels que sa vitesse et son angle de cap. Les méthodes à tentacules tels que décrites dans cette thèse peuvent être étendues avec une prise en compte d'un modèle statique ou quasi-dynamique du véhicule, où chaque tentacule sera évaluée sur la base de critères qui caractérisent la stabilité du véhicule et sa mobilité le long de cette trajectoire. Ces deux critères peuvent être simplement évalués grâce à un modèle de prédiction des forces normales et tangentielles aux contacts roue-sol.

Plate-formes numérique et expérimentale

Dans cette annexe, un aperçu sur les principales caractéristiques de la maquette virtuelle et le rover rapide “*SPIDO*” sont exposées. Ces deux plateformes sont utilisées pour la validation de nos différents algorithmes développés par simulation sous ROS/GAZEBO, puis, par des expérimentations sur la plateforme réelle.

1 Environnement de simulation

Les simulations numériques sont réalisées à partir du simulateur physique Gazebo¹ afin d’évaluer les performances de nos différents algorithmes de contrôle. Tous nos programmes sont implémentés en Python ou en langage C++. Une première étape de la modélisation du robot est la création du modèle *URDF*² (Unified Robot Description Format) du robot. Ce modèle a été généré à partir d’un modèle CAO de la plateforme conçue avec SolidWorks (Figure A.1). Il s’agit de la description géométrique cinématique et inertielle du robot (articulations, masse, inertie, etc.).

La Figure A.3 montre l’interface graphique du Gazebo, qui permet de créer des scénarios de simulation complexes (type terrain très accidenté et déformé) et de simuler plusieurs types d’environnements virtuels proches de la réalité avec des structures et caractéristiques variées.

1. gazebosim.org

2. [Unified Robot Description Format - URDF](#)

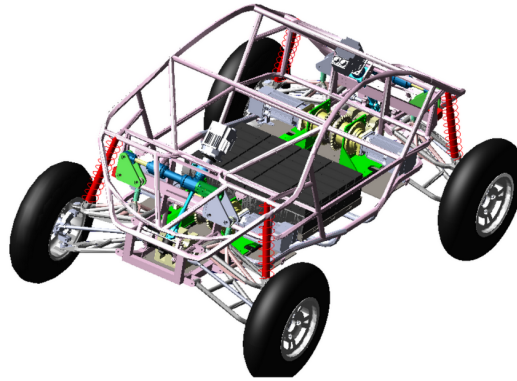


Figure A.1. Le modèle CAO de la maquette virtuelle de la simulation.

1.1 Modèle cinématique de la plate-forme virtuelle

L'architecture cinématique de la plate-forme virtuelle consiste à modéliser la suspension et le braquage des roues d'avant et arrière. Pour simplifier le modèle de simulation, nous allons opter pour un modèle simplifié. La suspension entre la roue et le châssis du robot est modélisée par une liaison prismatique en série avec une liaison pivot représentant le braquage de la roue et enfin la rotation de la roue autour de son axe (voir Figure A.2). L'architecture globale de la cinématique du robot est représentée par l'outil de visualisation temps-réel Rviz illustré par la Figure A.3.

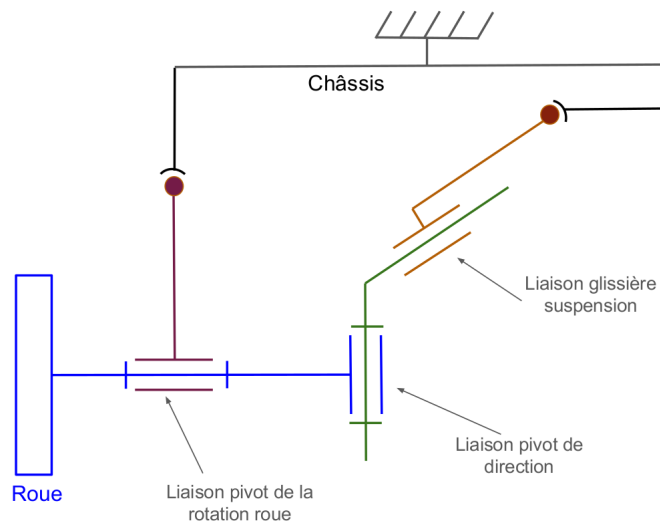


Figure A.2. Modèle cinématique au niveau de la roue i .

1.2 Modèle dynamique de la plate-forme virtuelle

Une fois la cinématique du robot décrite par le fichier *URDF*, il faut intégrer les propriétés dynamiques du robot. On définit ainsi les masses, positions des centres de gravité et matrices d'inertie de chaque pièce. On complète le modèle en définissant les coefficients de frottement

et en ajoutant les saturations des articulations prismatiques ou rotoïdes comme par exemple la limitation des angles de direction (e.g., $\pm 3 \text{ rad}$).

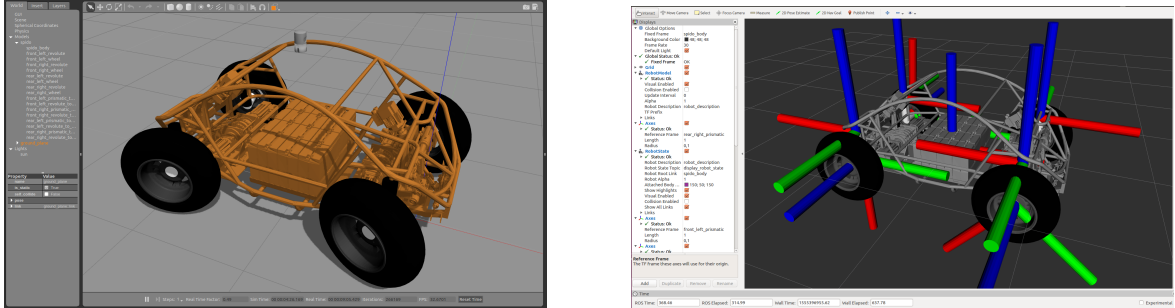


Figure A.3. [Gauche] : Vue principale de l'interface GAZEBO de la plateforme “SPIDO”; [Droite] : Architecture des articulations sur Rviz.

2 Plate-forme expérimentale

2.1 Caractéristiques générales de la plate-forme expérimentale

Le rover “SPIDO” est un véhicule électrique à deux trains directeurs de type Ackermann et à quatre roues motrices indépendantes (Figure A.4). Conçu et fabriqué par la société RoboSoft, il est capable d'évoluer jusqu'à la vitesse 12m.s^{-1} . Doté de 4 suspensions indépendantes à double triangle avec ressort-amortisseur, il est capable de rouler sur des sols accidentés et maintenir les contacts au sol sur les 4 roues (Figure A.5). Ses caractéristiques générales sont présentées dans la tableau A.1. Il est équipé de six servomoteurs dont quatre sont utilisés pour actionner les quatre roues et deux sont utilisés pour le braquage avant et arrière grâce à 2 mécanismes de type Ackermann (ou épure de Jeantaud) (cf. Figures A.5 & A.6). Le véhicule comporte aussi une installation hydraulique (vérins hydrauliques) qui permet d'actionner les deux barres anti-roulis actives destinées au contrôle de l'angle de roulis du rover. Elle permet de minimiser le risque de renversement du véhicule en réduisant l'angle de roulis de la masse suspendue, notamment durant les phases de virage ou en dévers (cf. Figure A.6).

Tableau A.1. Caractéristiques techniques générales de “SPIDO”

Caractéristique	Valeur
Empattement avant a	0.85 m
Empattement arrière b	0.85 m
Demie-voie d	0.45 m
Masse m	880 kg
Moment d'inertie I_z	300 kg.m^2
Braquage maximum	13°
Vitesse maximale	12m.s^{-1} (43km.h^{-1})

D'un point de vue architecture logicielle, un PC embarqué est chargé de récupérer les données du robot et de transmettre les ordres aux actionneurs. Piloté en mode manuel par un Joypad. En mode autonome, un second PC (doté d'un logiciel temps réel embarqué (intel NUC)) est chargé du calcul des consignes de commande (calcul des lois de commande) et du traitement des capteurs extéroceptifs.

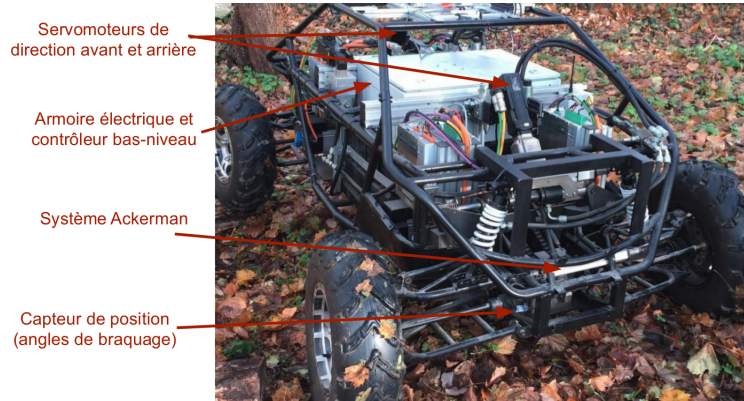


Figure A.4. “*SPIDO*”, véhicule électrique 4WS utilisé pour évaluer les algorithmes proposés.



Figure A.5. L'architecture mécanique de la plateforme expérimentale “*SPIDO*”.

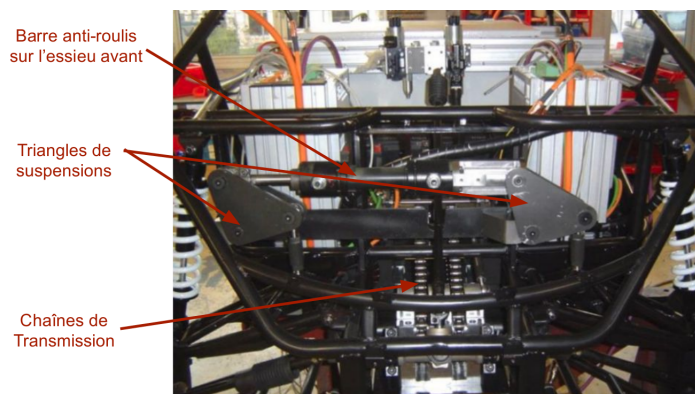


Figure A.6. La barre anti-roulis intégrée sur le robot “*SPIDO*” pour la stabilisation du roulis.

2.2 Capteurs disponibles

Nous décrivons dans cette section les différents matériels disponibles sur la plateforme mobile.

2.2.1 Mesures extéroceptives

Capteur GPS-RTK

Le principal capteur utilisé est un capteur GPS cinématique temps réel “*FLEX6-D2L-B0G*” commercialisé par la société Novatel. Ce capteur se compose d’un récepteur mobile placé sur le véhicule et d’une base fixe dont la position est parfaitement connue (cf.figure A.7). Le positionnement de ce genre de GPS fonctionne grâce à la méthode de trilatération (similaire à la triangulation, mais n’utilisant qu’un calcul de distance, sans calcul d’angle). L’antenne GPS reçoit le signal de plusieurs satellites (leurs positions dans l’espace et le temps d’envoi du signal), il est donc nécessaire de recevoir au moins le signal de quatre satellites : trois pour la position, et un supplémentaire pour la synchronisation. Ce dispositif permet d’obtenir une localisation du véhicule avec une précision centimétrique $\pm 2cm$ à la fréquence $10Hz$.

Capteur Xsens IMU

La centrale inertielle (IMU, Inertial Measurement Unit) utilisée dans cette thèse est de référence “*MTi-G-28*” et de marque Xsens (cf. Figure A.7). Ce dispositif informe le véhicule sur son accélération linéaire (accéléromètre) et sa vitesse de rotation (gyroscope) suivant les trois axes de l’espace. Il comporte également un magnétomètre qui donne la direction du nord magnétique en trois dimensions. Une fusion des données capteurs par un filtre de Kalman étendue (EKF) permet d’avoir les angles d’orientation du véhicule par rapport au repère inertiel avec une précision de $0,1^\circ.s^{-1}$ (3 angles d’Euler : lacet, roulis, tangage).

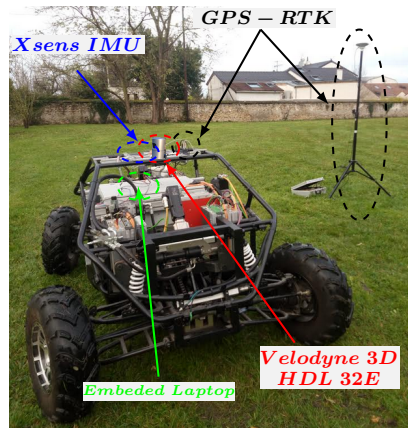


Figure A.7. Capteurs extéroceptifs embarqués sur le robot “*SPIDO*”.

Capteur Velodyne 3D

Nous utilisons un Velodyne LiDAR 3D qui permet au robot de percevoir son environnement et donc la détection et l'identification des objets présents dans son champ proche. Le capteur disponible pour ce projet est un LiDAR Velodyne "*HDL-32E*" (cf.figure A.7) qui comporte 32 lasers qui décrit un champ visuel de 40° en élévation et 360° en azimuth. Ce capteur génère un nuage de 1.3 millions de points chaque seconde, avec une portée de $120m$ et une précision de $\pm 2cm$ à la fréquence $10Hz$. Ce type de matériel est couramment utilisé en robotique mobile d'extérieur, par exemple sur les véhicules de la DARPA urban challenge et sur les Google cars. Cette technologie permet une utilisation en extérieur, y compris dans des conditions météorologiques défavorables et de visibilité réduite (e.g., brouillard, pluie, nuit, etc.).

2.2.2 Mesures proprioceptives

Capteurs odométriques

Les codeurs optiques sont des capteurs rotatifs placés au niveau des roues et donnant une information sur le sens et la vitesse de leurs rotations. Ces données odométriques nous permettront d'estimer le déplacement de la plateforme.

Capteurs d'angle de braquage

Les angles de braquage avant et arrière sont mesurés par des capteurs de position placés sur les essieux avant et arrière. Ces mesures seront exploitées pour estimer en ligne les paramètres physiques du terrain.

2.3 Architecture logicielle et hardware

Le matériel (bas-niveau) permet de piloter les actionneurs (moteurs, direction et barre anti-roulis). Cette partie a été développée par la société RoboSoft en utilisant le contrôleur PURE. L'ordinateur bas-niveau est directement connecté à l'ordinateur embarqué (intel NUC) par une liaison Ethernet. L'ordinateur embarqué fonctionne sous Linux et sur lequel était implémenté tous nos différents programmes permettant d'effectuer le suivi de chemin et la planification des chemins de franchissement. Le véhicule peut fonctionner en mode manuel (Joystick) ou en mode automatique (ordinateur embarqué) (voir Figure A.8).

L'ordinateur distant est relié par WiFi au mini PC embarqué, il permet principalement d'assurer la sécurité de la plateforme et de modifier les lois de commandes à distance. Tous les algorithmes, incluant ceux de planification et de commande présentés dans ce mémoire, sont implémentés en C++ ou Python. La communication entre les différentes parties est assurée par ROS³ (Robot Operating System).

3. [Robot Operating System \(ROS\)](#)

- **Nœud-Haut-Niveau** : Ce nœud permet principalement le traitement des données capteurs et la synthèse des lois de commande pour le suivi de chemin et l'évitement réactif d'obstacles.
 1. Calcul des consignes : À partir des données capteurs - ROS Drivers (LiDAR, GPS-RTK et Centrale inertielle), les consignes de direction sont calculées (angles de braquage) en prenant en compte toutes les contraintes du système. Celles-ci sont envoyées par la suite aux actionneurs.
 2. Planification des chemins d'évitement : À partir des données LiDAR, le robot est capable d'apercevoir les obstacles existants autour de la plateforme. De nouvelles consignes sont générées permettant l'évitement réactif des obstacles et la convergence rapide vers le chemin nominal.

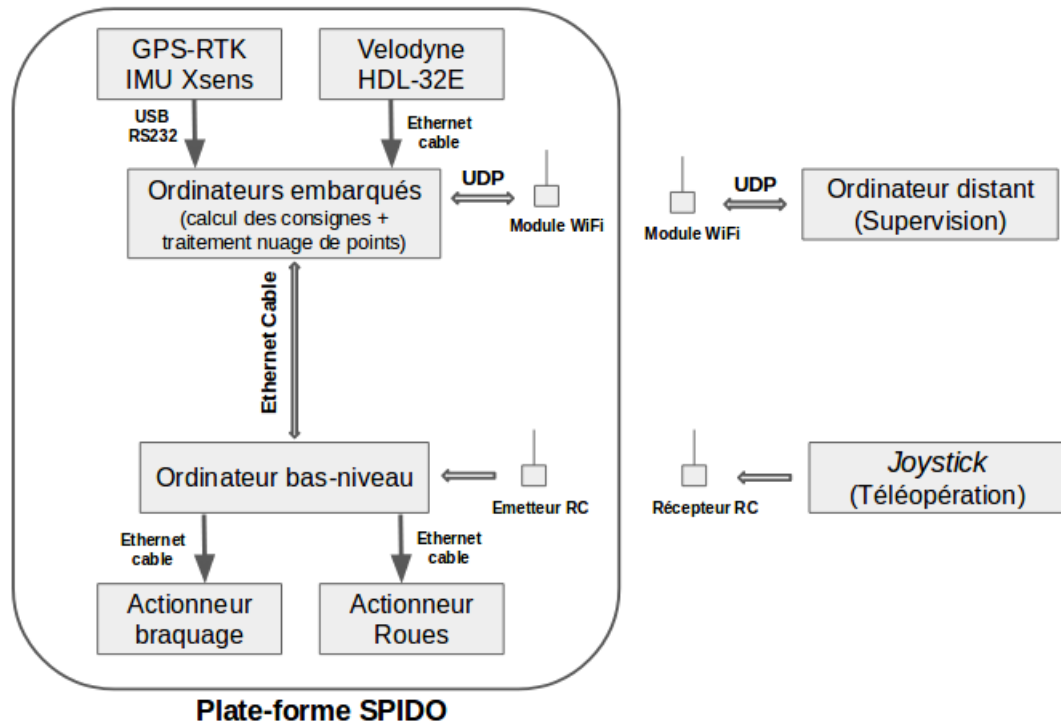


Figure A.8. Architecture matérielle embarquée du "SPIDO".

- **Nœud-Bas-Niveau** : Ce nœud permet principalement la communication avec le bas-niveau (Nœud : Spido Pure-Interface).
 1. Transmission des commandes : Les commandes de direction calculées par le nœud précédent sont envoyées à l'ordinateur embarqué puis aux actionneurs.
 2. Acquisition données proprioceptives : Les mesures des capteurs proprioceptifs sont aussi stockées et remontées vers le premier nœud Nœud-Haut-Niveau.
 3. Toutes les mesures des capteurs proprioceptifs et extéroceptifs sont stockées dans un fichier *.tex* pour des analyses ultérieures.

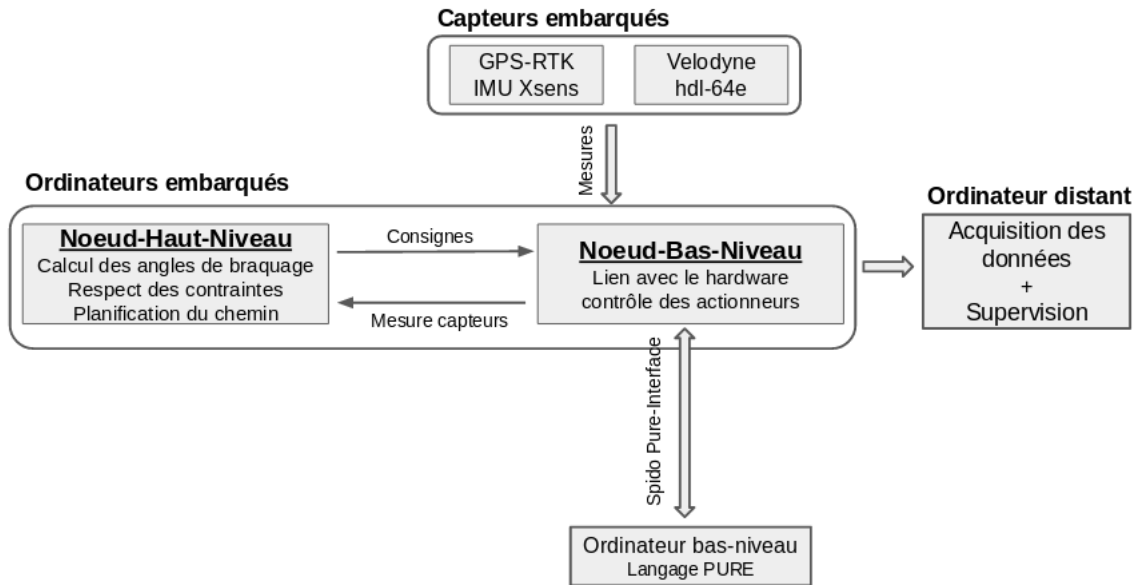


Figure A.9. Architecture logicielle embarquée du “SPIDO”.

2.4 Robot Operating System : ROS

L’intégralité de la partie logicielle du projet est développée sous ROS⁴ (Robot Operating System). Cet ensemble d’outils au code source ouvert, devenu très populaire dans la communauté robotique, permet de faciliter le développement de programmes et leurs interactions. ROS est un middleware proposant de nombreux outils pour faciliter le développement logiciel pour les robots, il intègre toute une série de fonctionnalités comme une infrastructure de communication inter-processus et inter-machine avec des messages standardisés, les outils de simulation, la possibilité d’enregistrer et de rejouer les messages, etc. L’architecture ROS considère chaque exécutable comme un nœud (comme par exemple les drivers des capteurs disponibles par le biais de la communauté, etc.), qui peut envoyer ou recevoir des messages sur des topics (données capteurs, valeur de la commande en temps réel, etc.). Lorsque plusieurs executables sont lancés, ils sont alors capables de communiquer entre eux par l’intermédiaire de ces topics.

Dans ce projet, l’ensemble des programmes correspondant à la gestion des capteurs et des actionneurs ont été réalisés à partir de cet environnement ROS.

2.5 Point Cloud Library (PCL)

Point Cloud Library (PCL⁵) permet de fournir aux développeurs et aux roboticiens des outils de traitement des nuages de points fournis par le capteur LiDAR. Un nuage de point est une structure de données utilisée pour représenter une collection de points multi-dimensions. En plus des coordonnées spatiales (x, y, z) chaque point peut également contenir des informations sur l’intensité du signal echo et la réflectance (RGB, etc.).

4. [Robot Operating System \(ROS\)](#)

5. [Point Cloud Library \(or PCL\)](#)

PCL met à disposition une multitude de fonction allant de la simple copie et concaténation du nuage de points jusqu'à la segmentation. Et grâce à ROS il est facile et rapide d'obtenir un affichage en temps réel de ce nuage de points grâce aux outils RQT et Rviz. Nos algorithmes de détection d'obstacles regroupent plusieurs filtres spatiaux provenant de cette librairie (e.g., downsampling, passThrough, etc.) [Rusu et Cousins \(2011\)](#).

2.6 Site d'expérimentation

Les expérimentations en extérieur sur le robot “*SPIDO*” ont lieu sur le site de Sorbonne Université de Saint-Cyr-l'École. Nous disposons d'un espace relativement dégagé, avec des obstacles naturels (murs, arbres, bute de terre, ...), les zones navigables sont relativement planes et horizontales (Figure A.10).



Figure A.10. Le champs utilisé pour les expérimentations [Google Maps].

Détection et évitement d'obstacles

Cette annexe a pour objectif de présenter quelques algorithmes de détection et d'évitement d'obstacles. La partie détection se focalise sur le traitement du nuage de points délivré par le LiDAR 3D (Velodyne HDL-32E) (**Algorithme 1**). La partie évitement, quant à elle, se base sur l'extraction des points de contrôle (**Algorithme 2**) et la génération des deux meilleures courbes de Bézier cubiques entre ces points, l'une pour assurer l'évitement d'obstacle et l'autre pour converger vers le chemin de référence global (**Algorithme 3**).

1 Algorithme de détection d'obstacles

Algorithme 1 Algorithme de détection d'obstacles

$C_j \leftarrow$ Clusters de nuage de points détecté
 $p_i^j \leftarrow$ Point i appartenant au cluster de nuage de points C_j
 $array \leftarrow$ Publisher ROS donnant des informations sur l'obstacle détecté // cette variable sera exploitée par **Algorithme 2** de l'évitement d'obstacle
 $d_s \leftarrow$ Distance de sécurité
 $R_o \leftarrow$ Rayon de cercle contenant l'obstacle détecté (contenant tous les points p_i^j)
 $R_g \leftarrow$ Rayon de cercle de l'obstacle gonflé
 $R_e \leftarrow$ Rayon de cercle englobant (\mathcal{C})
 $\alpha_g \leftarrow$ Coefficient d'augmentation pour le cercle gonflé
 $\alpha_e \leftarrow$ Coefficients d'augmentation pour le cercle englobant
 $O_c(x_c, y_c) \leftarrow$ Barycentre de l'obstacle détecté
 $x, y \leftarrow$ Positions du centre de gravité du véhicule
for tout cluster C_j **faire**
- Trouver les points extrêmes du cluster ($\forall p_i^j \in C_j$), $x_{i \max}$, $x_{i \min}$, $y_{i \max}$, $y_{i \min}$
if $\frac{x_{i \max} - x_{i \min}}{2} \geq \frac{y_{i \max} - y_{i \min}}{2}$ **then**
- $R_o = \frac{x_{i \max} - x_{i \min}}{2}$
else
- $R_o = \frac{y_{i \max} - y_{i \min}}{2}$
Fin if
- $R_g \leftarrow R_o(1 + \alpha_g)$
- $R_e \leftarrow R_g(1 + \alpha_e)$
- $x_c \leftarrow \frac{x_{i \max} + x_{i \min}}{2}$
- $y_c \leftarrow \frac{y_{i \max} + y_{i \min}}{2}$
// Partie traitée quand un obstacle est détecté
if $\sqrt{(x - x_b)^2 + (y - y_b)^2} - d_s \leq 0$ **then**
- $array.flag \leftarrow$ flag // flag indiquant qu'un obstacle est détecté (prend la valeur 0 ou 1)
- $array.x \leftarrow x - x_b$ // Distance longitudinale entre le véhicule et l'obstacle
- $array.y \leftarrow y - y_b$ // Distance latérale entre le véhicule et l'obstacle
- $array.R_o \leftarrow R_o$
- $array.R_g \leftarrow R_g$
- Publication des information sauvegardées dans le topic ROS $array$
Fin if
Fin for

2 Évitement d'obstacles

2.1 Algorithme d'évitement d'obstacles

Algorithme 2 Algorithme d'extraction des points de passage et d'évitement d'obstacles

```

 $\Gamma \leftarrow$  Chemin désiré global défini hors-ligne
 $index \leftarrow$  Indice de chaque donnée de  $\Gamma$  (numéro de ligne du tableau  $\Gamma$ )
 $P_k \leftarrow$  Point défini par  $(x_k, y_k)$ 
 $O_c(x_c, y_c) \leftarrow$  Barycentre de l'obstacle détecté
 $d_s \leftarrow$  Distance de sécurité
 $R_c \leftarrow$  Rayon de cercle contenant l'obstacle détecté
 $R_g, R_e \leftarrow$  Rayons de l'obstacle gonflé et cercle englobant ( $\mathcal{C}$ )
 $\alpha_g, \alpha_e \leftarrow$  Coefficients d'augmentation pour le cercle gonflé et le cercle englobant
 $flag\_new\_path \leftarrow 1$  //  $flag\_new\_path$  indiquant l'existence ou non d'un obstacle
 $k \leftarrow 0$ 
// Extraction du point initiale et finale  $P_b$  et  $P_f$  du chemin de contournement
if  $flag\_new\_path = 1$  then
   $flag\_new\_path \leftarrow 0$ 
   $k \leftarrow index$ 
   $P_b \leftarrow P_k$ 
  for tout  $P_k \in \Gamma$  faire
    if  $\|O_b P_f\| \leq d_s$  then
       $k ++$ 
    Fin if
  Fin for
   $j \leftarrow k$ 
   $P_f \leftarrow P_j$ 
  // Extraction du point d'évitement  $P_m$ 
  -  $(\mathcal{D}_1) \leftarrow$  Droite reliant les points  $P_s$  et  $P_f$ 
  -  $(\mathcal{D}_2) \leftarrow$  Droite perpendiculaire à  $(\mathcal{D}_1)$ , passant par  $O_b$ 
  - Calcul des intersections de  $(\mathcal{D}_2)$  avec le cercle ( $\mathcal{C}$ ), en retournant les points  $P_{m1}$  et  $P_{m2}$ 
  if  $\|\overrightarrow{O_c P_{m1}}\| \leq \|\overrightarrow{O_c P_{m2}}\|$  then
     $P_m = P_{m1}$ 
  else
     $P_m = P_{m2}$ 
  Fin if
  // Extraction des points de contrôle intermédiaires
   $P_1, P_2 \leftarrow$  lancer Algorithme 3 pour la première courbe de Bézier
   $P_5, P_6 \leftarrow$  lancer Algorithme 3 pour la seconde courbe de Bézier
Fin if
Meilleures courbes de Bézier cubiques
Mise à jour du chemin de référence entre  $P_b$  and  $P_f$ 

```

2.2 Algorithme de choix des points de contrôle

Algorithme 3 Choix des meilleurs points de contrôle

$P_b(x_b, y_b, \psi_b), P_m(x_m, y_m, \psi_m), P_f(x_f, y_f, \psi_f) \leftarrow$ variables venant de **Algorithme 2**
 $d_s \leftarrow$ Distance de sécurité
bezier_réalisable \leftarrow Variable pour stocker les Béziers qui sont réalisables par le véhicule
// Détermination des équations cartésiennes des tangentes
- La tangente au point P_b est d'équation (\mathcal{T}_b) : $y = \tan \psi_b(x - x_b) + y_b$
- La tangente au point P_f est d'équation (\mathcal{T}_f) : $y = \tan \psi_f(x - x_f) + y_f$
- La tangente au point P_m est parallèle à (\mathcal{D}_1) et d'équation (\mathcal{T}_m) : $y = \tan \psi_m(x - x_m) + y_m$
// Détermination de la première courbe de Bézier cubique (fixation du P_2 et P_3)
 $n \leftarrow 10$ // Nombre de fois pour faire varier les points P_2 et P_3 sur l'intervalle $[0, d_s]$
 $d_b, d_m \leftarrow d_s$
for $i < n$ **faire**
- Variation du P_1 sur la tangente (\mathcal{T}_b), avec $d(P_b, P_1) = d_b \in [0, d_s]$
for $j < n$ **faire**
- Variation du P_2 sur la tangente (\mathcal{T}_m), avec $d(P_2, P_m) = d_m \in [0, d_s]$
- Calcul de la courbure maximale de la courbe de Bézier générée par les points de contrôle P_b, P_1, P_2 & P_m , notée $|\rho_j|$
if $|\rho_j| \leq \min \left[\frac{\mu g}{V_x^2}, \frac{2 \tan \delta_{\max}}{L} \right]$ **then**
- **bezier_réalisable** $\leftarrow P_b, P_1, P_2, P_m$ et la valeur de la courbure $|\rho_j|$
Fin if
 $d_m \leftarrow d_m - \frac{d_s}{n}$
Fin for
 $d_b \leftarrow d_b - \frac{d_s}{n}$
 $d_m \leftarrow d_s$
Fin for
// Sélection de la meilleure courbe de Bézier pour la première section
 $m \leftarrow \text{length}(\text{bezier_réalisable})$ // $m \leq n^2$
for $k = 1 < m - 1$ **faire**
if $|\rho_k| \leq |\rho_{k+1}|$ **then**
 $k^* \leftarrow k$
else
 $k^* \leftarrow k + 1$
Fin if
Fin for
- La meilleure courbe est d'indice k^* qui minimise la courbure et satisfait les contraintes cinématiques et dynamiques du véhicule
// Sélection de la meilleure courbe de Bézier pour la première section
- Même étapes appliquées sur la deuxième section du chemin local (choix du P_5 & P_6)
Return : Meilleur positionnement des points de contrôle intermédiaires P_1, P_2, P_5 & P_6

Localisation rapide du véhicule

La connaissance d'une bonne estimée de l'état du robot (notamment sa position) lors de ses déplacements est essentielle. C'est en effet le seul moyen de garantir le suivi correct du chemin nominal. Si le GPS-RTK permet de garantir une localisation à long terme, sa précision est localement insuffisante et aisément perturbée par les bâtiments et les arbres existant dans l'environnement de navigation. Il est donc nécessaire de pallier ces défauts pour connaître à tout moment une estimée de position localement précise : c'est l'objet de cet annexe qu'on implémentera dans nos futures travaux. Pour cela nous proposons de développer un système de localisation qui estime la position $[x, y, \psi]^T$ du véhicule en fusionnant les informations fournies par le GPS-RTK, la centrale inertielle et les codeurs odométriques. On mettra en œuvre un filtre de Kalman étendu (EKF) afin d'améliorer la localisation du véhicule dans son environnement d'intervention.

1 Modèle cinématique des robots à 4 roues directrices

Les équations de mouvement du véhicule dans le repère inertiel absolu \mathcal{R}_i sont,

$$\dot{x} = V_x \cos \psi - V_y \sin \psi, \quad (\text{C.1a})$$

$$\dot{y} = V_x \sin \psi + V_y \cos \psi, \quad (\text{C.1b})$$

$$\dot{\psi} = V_\psi, \quad (\text{C.1c})$$

avec \dot{x} et \dot{y} sont respectivement la vitesse longitudinale et latérale dans le repère inertiel \mathcal{R}_i , ψ est l'angle d'orientation du véhicule dans le plan de lacet. V_x et V_y sont respectivement la vitesse longitudinale et latérale dans le repère véhicule \mathcal{R}_G .

Le modèle cinématique (C.1) peut être exprimé en fonction du rayon de la roue R et la distance

entre la roue droite et roue gauche $2d$ ainsi que les vitesses de rotation ω_i avec $i \in \{1, 2, 3, 4\}$.

$$\dot{x} = \frac{R \sum_1^4 \omega_i}{4} (\cos \delta_f \cos \psi - \sin \delta_r \sin \psi), \quad (\text{C.2a})$$

$$\dot{y} = \frac{R \sum_1^4 \omega_i}{4} (\cos \delta_f \sin \psi + \sin \delta_r \cos \psi), \quad (\text{C.2b})$$

$$\dot{\psi} = \frac{R}{8d} (\omega_1 - \omega_2 + \omega_3 - \omega_4), \quad (\text{C.2c})$$

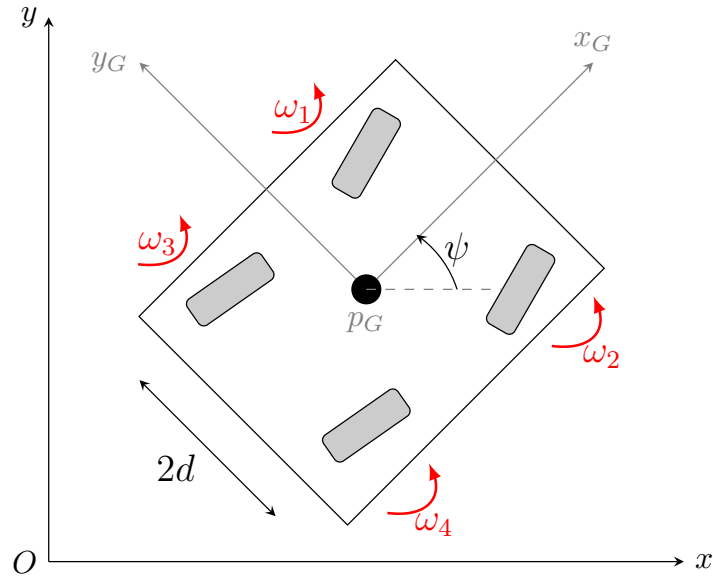


Figure C.1. Vitesses de rotation des roues.

Les équations C.2 peuvent se mettre sous la forme d'un système non linéaire à temps continu sous la forme suivante :

$$\dot{X}_c = \mathbf{f}_c(X_c, U_c), \quad (\text{C.3})$$

avec $X_c = [x, y, \psi]^T$ est le vecteur d'état et $U_c = [\omega_1, \omega_2, \omega_3, \omega_4, \delta_f, \delta_r]^T$ le vecteur de commande.

Une discrétisation approchée au premier ordre à la période T_c de cette équation cinématique (C.3) en utilisant la méthode d'Euler peut s'exprimer sous la forme suivante :

$$X_{k+1} = X_k + T_c \cdot \mathbf{f}_c(X_k, U_k) = \mathbf{f}(X_k, U_k), \quad (\text{C.4})$$

2 Mise en œuvre d'un EKF pour la localisation rapide

On procède par la linéarisation exacte du modèle (C.4) en supposant que nous avons une estimation \hat{X}_k du vecteur d'état et la commande U . Donc,

$$\begin{cases} X_{k+1} \simeq A_k \cdot (X_k - \hat{X}_k) + B_k \cdot (U_k - U), \\ Y_k \simeq C_k (X_k - \hat{X}_k), \end{cases} \quad (\text{C.5})$$

avec Y_k est le vecteur mesure venant des capteurs embarqués (GPS et centrale inertielle), $C_k = I_{3 \times 3}$ est la matrice d'observation, et

$$A_k = \left. \frac{\partial \mathbf{f}(X, U)}{\partial X} \right|_{(X=\hat{X}_k, U=U_k)}, B_k = \left. \frac{\partial \mathbf{f}(X, U)}{\partial U} \right|_{(X=\hat{X}_k, U=U_k)},$$

En rajoutant des bruits blancs gaussiens indépendants sur le modèle et la mesure α_k et β_k , le système (C.5) devient,

$$\begin{cases} X_{k+1} = A_k \cdot (X_k - \hat{X}_k) + B_k \cdot (U_k - U) + \alpha_k, \\ Y_k = C_k (X_k - \hat{X}_k) + \beta_k, \end{cases} \quad (\text{C.6})$$

Supposant que le système (C.6) est observable, les équations du filtre de Kalman étendu (EKF) s'expriment comme suit,

$$\hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + K_{k+1} \cdot \tilde{Y}_{k+1}, \quad (\text{C.7a})$$

$$\tilde{Y}_{k+1} = Y_{k+1} - C_k \hat{X}_{k+1|k}, \quad (\text{C.7b})$$

$$\Gamma_{k+1|k} = A_k \cdot \Gamma_{k|k} \cdot A_k^T + B_k \cdot \Gamma_{\alpha_k} \cdot B_k^T, \quad (\text{C.7c})$$

$$\Gamma_{k+1|k+1} = (I - K_k C_k) \Gamma_{k+1|k}, \quad (\text{C.7d})$$

$$K_{k+1} = \Gamma_{k+1|k} C_k^T (C_k \Gamma_{k+1|k} C_k^T + \Gamma_{\beta_k})^{-1}, \quad (\text{C.7e})$$

Ce filtre EKF prend en amont Y_k , A_k , B_k , C_k , Γ_{α_k} et Γ_{β_k} (matrices de covariance sur le modèle et la mesure), et en aval, on obtient ces quantités : $\hat{X}_{k+1|k+1}$ (estimée de l'état corrigée), $\Gamma_{k+1|k+1}$ (matrice de covariance) et K_k (gain du Kalman). Ce filtre sera implémenté et validé dans nos futurs travaux afin d'améliorer la précision de la position de la plateforme dans son environnement d'intervention.

Bibliographie

- Juergen ACKERMANN, Jürgen GULDNER, Wolfgang SIENEL, Reinhold STEINHAUSER et Vadim I UTKIN : Linear and nonlinear controller design for robust automatic steering. *IEEE Transactions on Control Systems Technology*, 3(1):132–143, 1995. *Cité page 17*
- Amit AILON, Nadav BERMAN et Shai AROGETI : On controllability and trajectory tracking of a kinematic vehicle model. *Automatica*, 41(5):889–896, 2005. *2 citations pages 6 et 17*
- Chebly ALIA, Tagne GILLES, Talj REINE et Charara ALI : Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method. *In 2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 674–679. IEEE, 2015. *2 citations pages 10 et 86*
- Martin S ANDERSEN, Joachim DAHL et Lieven VANDENBERGHE : Cvxopt : Python software for convex optimization, 2013. *Cité page 75*
- Rusty ANDERSON et David M BEVLY : Estimation of tire cornering stiffness using gps to improve model based estimation of vehicle states. *In IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, pages 801–806. IEEE, 2005. *Cité page 43*
- Boualem BADJI : *Caractérisation du comportement non linéaire en dynamique du véhicule.* Thèse de doctorat, Université de Technologie de Belfort-Montbéliard, 2009. *Cité page 43*
- Guillaume BAFFET, Ali CHARARA et Daniel LECHNER : Estimation of vehicle sideslip, tire force and wheel cornering stiffness. *Control Engineering Practice*, 17(11):1255–1264, 2009. *Cité page 44*
- Susanne BECKER et Norbert HAALA : Combined feature extraction for façade reconstruction. *In Proceedings of the ISPRS Workshop Laser Scanning*, pages 241–247, 2007. *Cité page 89*
- Kostas E BEKRIS, Brian Y CHEN, Andrew M LADD, Erion PLAKU et Lydia E KAVRAKI : Multiple query probabilistic roadmap planning using single query planning primitives. *In Pro-*

- ceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*(Cat. No. 03CH37453), volume 1, pages 656–661. IEEE, 2003. *Cité page 86*
- Tarun Kumar BERA, K BHATTACHARYA et Arun K SAMANTARAY : Evaluation of antilock braking system with an integrated model of full vehicle system dynamics. *Simulation Modelling Practice and Theory*, 19(10):2131–2150, 2011. *Cité page 17*
- Jim E BERNARD et Chris L CLOVER : Tire modeling for low-speed and high-speed calculations. *SAE transactions*, pages 474–483, 1995. *Cité page 18*
- James BETHEL : Detailed building reconstruction from airborne laser data using a moving surface method. 2004. *Cité page 89*
- David M BEVLY, Jihan RYU et J Christian GERDES : Integrating ins sensors with gps measurements for continuous estimation of vehicle sideslip, roll, and tire cornering stiffness. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):483–493, 2006. *Cité page 43*
- Pierre-Alexandre BLIMAN et Michel SORINE : A system-theoretic approach of systems with hysteresis. application to friction modelling and compensation. *In Proceedings of the 2nd European control conference*, pages 1844–1849, 1993. *Cité page 5*
- Michael BLUNDELL et Damian HARTY : *Multibody systems approach to vehicle dynamics*. Elsevier, 2004. *Cité page 5*
- Beatriz L BOADA, Daniel GARCIA-POZUELO, Maria Jesus L BOADA et Vicente DIAZ : A constrained dual kalman filter based on pdf truncation for estimation of vehicle parameters and road bank angle : Analysis and experimental validation. *IEEE Transactions on Intelligent Transportation Systems*, 18(4):1006–1016, 2016. *Cité page 29*
- Johann BORENSTEIN, Yoram KOREN *et al.* : The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE transactions on robotics and automation*, 7(3):278–288, 1991. *Cité page 9*
- Francesco BORRELLI, Paolo FALCONE, Tamas KEVICZKY, Jahan ASGARI et Davor HROVAT : Mpc-based approach to active steering for autonomous vehicle systems. *International Journal of Vehicle Autonomous Systems*, pages 265–291, 2005. *2 citations pages 43 et 66*
- Nicolas BOUTON : *Stabilité dynamique des véhicules légers tout-terrain. Nouvelles solutions. Application aux véhicules légers de type quad*. Thèse de doctorat, Université Blaise Pascal-Clermont-Ferrand II, 2009. *2 citations pages 2 et 4*
- Nicolas BOUTON, Roland LENAIN, Benoit THUILOT et Philippe MARTINET : Backstepping observer dedicated to tire cornering stiffness estimation : Application to an all terrain vehicle and a farm tractor. *In 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2007)*, pages 1763–1768. IEEE, 2007. *6 citations pages 6, 8, 17, 29, 42 et 44*
- Stephen BOYD et Lieven VANDENBERGHE : *Convex optimization*. Cambridge university press, 2004. *Cité page 66*
- Christophe CARIOU, Roland LENAIN, Benoit THUILOT et Michel BERDUCAT : Automatic guidance of a four-wheel-steering mobile robot for accurate field operations. *Journal of Field Robotics*, 26(6-7):504–518, 2009. *3 citations pages 4, 5 et 28*

- Roland CHAPUIS, Jean LANEURIT, Romuald AUFRÈRE, Frédéric CHAUSSE et Thierry CHATEAU : Accurate vision based road tracker. *In Intelligent Vehicle Symposium, 2002. IEEE*, volume 2, pages 666–671. IEEE, 2002. *Cité page 8*
- Charles R CUTLER et Brian L RAMAKER : Dynamic matrix control a computer control algorithm. *In joint automatic control conference*, numéro 17, page 72, 1980. *Cité page 63*
- Phil R DAHL : A solid friction model. Rapport technique, Aerospace Corp El Segundo Ca, 1968. *Cité page 5*
- Alessandro DE LUCA, Giuseppe ORIOLO et Claude SAMSON : Feedback control of a nonholonomic car-like robot. *In Robot motion planning and control*, pages 171–253. Springer, 1998. *Cité page 28*
- C Canudas DE WIT et Panagiotis TSIOTRAS : Dynamic tire friction models for vehicle traction control. *In Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304)*, volume 4, pages 3746–3751. IEEE, 1999. *Cité page 5*
- Moustapha DOUMIATI, Alessandro VICTORINO, Ali CHARARA et Daniel LECHNER : A method to estimate the lateral tire force and the sideslip angle of a vehicle : Experimental validation. *In Proceedings of the 2010 American Control Conference*, pages 6936–6942. IEEE, 2010. *3 citations pages iii, 7 et 8*
- Moustapha DOUMIATI, Alessandro Correa VICTORINO, Ali CHARARA et Daniel LECHNER : On-board real-time estimation of vehicle lateral tire–road forces and sideslip angle. *IEEE/ASME Transactions on Mechatronics*, 16(4):601–614, 2011. *Cité page 44*
- Noel E DU TOIT et Joel W BURDICK : Robotic motion planning in dynamic, cluttered, uncertain environments. *In 2010 IEEE International Conference on Robotics and Automation*, pages 966–973. IEEE, 2010. *Cité page 10*
- Howard DUGOFF, Paul S FANCHER et Leonard SEGEL : Tire performance characteristics affecting vehicle response to steering and braking control inputs. Rapport technique, 1969. *2 citations pages 5 et 18*
- Ray EATON, H POTA et Jayantha KATUPITIYA : Path tracking control of agricultural tractors with compensation for steering dynamics. *In Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 7357–7362. IEEE, 2009. *Cité page 28*
- Magnus EGERSTEDT, Xiaoming HU, Henrik REHBINDER et A STOTSKY : Path planning and robust tracking for a car-like robot. Citeseer, 1997. *Cité page 6*
- Halit EREN, Chun Che FUNG et Jeromy EVANS : Implementation of the spline method for mobile robot path control. *In IMTC/99. Proceedings of the 16th IEEE Instrumentation and Measurement Technology Conference (Cat. No. 99CH36309)*, volume 2, pages 739–744. IEEE, 1999. *Cité page 87*
- Paolo FALCONE, Francesco BORRELLI, Jahan ASGARI, Hongtei Eric TSENG et Davor HROVAT : Predictive active steering control for autonomous vehicle systems. *IEEE Transactions on control systems technology*, 15(3):566–580, 2007. *3 citations pages 29, 63 et 66*

- Wendelin FEITEN, Rudolf BAUER et Gisbert LAWITZKY : Robust obstacle avoidance in unknown and cramped environments. *In Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2412–2417. IEEE, 1994. *Cité page 10*
- Mohamed FNADI, Bachir MENKOUZ, Frédéric PLUMET et Faiz BENAMAR : Path tracking control for a double steering off-road mobile robot. *In ROMANSY 22–Robot Design, Dynamics and Control*, pages 441–449. Springer, 2018. *Cité page 13*
- Mohamed FNADI, Frédéric PLUMET et Faiz BENAMAR : Model predictive control based dynamic path tracking of a four-wheel steering mobile robot. *In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2019)*. IEEE/RSJ, 2019a. *Cité page 13*
- Mohamed FNADI, Frédéric PLUMET et Faiz BENAMAR : Nonlinear tire cornering stiffness observer for a double steering off-road mobile robot. *In IEEE International Conference on Robotics and Automation (ICRA2019)*. IEEE, 2019b. *Cité page 13*
- Mohamed FNADI, Frédéric PLUMET et Faiz BENAMAR : Road bank and road grade angles estimation for a double steering off-road mobile robot. *In IFToMM World Congress on Mechanism and Machine Science*, pages 1771–1780. Springer, 2019c. *Cité page 13*
- Mohamed FNADI, Frédéric PLUMET et Faiz BENAMAR : Local obstacle-skirting path planning for a fast bi-steerable rover using bézier curves. *In IEEE International Conference on Robotics and Automation (ICRA2020)*. IEEE (Soumis), 2020. *Cité page 14*
- Gilles Tagne FOKAM : *Commande et planification de trajectoires pour la navigation de véhicules autonomes*. Thèse de doctorat, 2014. *Cité page 10*
- Emilio FRAZZOLI, Munther A DAHLEH et Eric FERON : Robust hybrid control for autonomous vehicle motion planning. *In Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, volume 1, pages 821–826. IEEE, 2000. *Cité page 87*
- Jonathan D GAMMELL, Siddhartha S SRINIVASA et Timothy D BARFOOT : Informed rrt* : Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. *In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2997–3004. IEEE, 2014. *Cité page 86*
- Yiqi GAO, Theresa LIN, Francesco BORRELLI, Eric TSENG et Davor HROVAT : Predictive control of autonomous ground vehicles with obstacle avoidance on slippery roads. 2010. *Cité page 66*
- Carlos E GARCIA et AM MORSHEDI : Quadratic programming solution of dynamic matrix control (qdmc). *Chemical Engineering Communications*, 46(1-3):73–87, 1986. *Cité page 63*
- Thomas D. GILLESPIE : *Fundamentals of vehicle dynamics*. Society of Automotive Engineers, Warrendale, PA, 4. printing édition, 1992. OCLC : 24907034. *2 citations pages 17 et 18*
- David GONZÁLEZ, Joshué PÉREZ, Vicente MILANÉS et Fawzi NASHASHIBI : A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):1135–1145, 2015. *Cité page 10*
- Andrew GRAY, Yiqi GAO, Theresa LIN, J Karl HEDRICK, H Eric TSENG et Francesco BORRELLI : Predictive control for agile semi-autonomous ground vehicles using motion primitives. *In 2012 American Control Conference (ACC)*, pages 4239–4244. IEEE, 2012. *2 citations pages 66 et 87*

- Håvard Fjær GRIP, Lars IMSLAND, Tor A JOHANSEN, Thor I FOSSEN, Jens C KALKKUHL et Avshalom SUISSA : Nonlinear vehicle side-slip estimation with friction adaptation. *Automatica*, 44(3):611–622, 2008. *Cité page 42*
- El-Hadi GUECHI, Jimmy LAUBER et Michel DAMBRINE : On-line moving-obstacle avoidance using piecewise bezier curves with unknown obstacle trajectory. In *2008 16th Mediterranean Conference on Control and Automation*, pages 505–510. IEEE, 2008. *2 citations pages 10 et 87*
- Thomas GUILLOD : Interpolations, courbes de bézier et b-splines. 2008. *Cité page 88*
- Jurgen GULDNER et Vadim I UTKIN : Stabilization of non-holonomic mobile robots using lyapunov functions for navigation and sliding mode control. In *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, volume 3, pages 2967–2972. IEEE, 1994. *Cité page 28*
- Fredrik GUSTAFSSON : Slip-based tire-road friction estimation. *Automatica*, 33(6):1087–1099, 1997. *Cité page 44*
- Norbert HAALA, Claus BRENNER et Karl-Heinrich ANDERS : 3d urban gis from laser altimeter and 2d map data. 1998. *Cité page 89*
- M HADDAD, T CHETTIBI, S HANCHI et HE LEHTIHET : A random-profile approach for trajectory planning of wheeled mobile robots. *European Journal of Mechanics-A/Solids*, 26(3):519–540, 2007. *Cité page 87*
- Jin-Oh HAHN, Rajesh RAJAMANI et Lee ALEXANDER : Gps-based real-time identification of tire-road friction coefficient. In *ASME 2002 International Mechanical Engineering Congress and Exposition*, pages 767–776. American Society of Mechanical Engineers, 2002. *3 citations pages 9, 18 et 22*
- Jorge HERNANDEZ et Beatriz MARCOTEGUI : Segmentation et interprétation de nuages de points pour la modélisation d’environnements urbains. *arXiv preprint arXiv :1306.3084*, 2013. *Cité page 89*
- Michael HIMMELSBACH, Thorsten LUETTEL, Falk HECKER, Felix von HUNDELSHAUSEN et Hans-Joachim WUENSCHKE : Autonomous off-road navigation for mucar-3. *KI-Künstliche Intelligenz*, 25(2):145–149, 2011. *2 citations pages 10 et 86*
- Rami Y HINDIYEH et J Christian GERDES : Equilibrium analysis of drifting vehicles for control design. In *ASME 2009 Dynamic Systems and Control Conference*, pages 181–188. American Society of Mechanical Engineers, 2009. *Cité page 18*
- Emmanuel HUGO et Pierre COUBLE : Robot paysan made in auvergne - irstea. La Montagne, Quotidien régional - Clermont-Ferrand, 2013. *Cité page 4*
- Jeong hwan JEON, Sertac KARAMAN et Emilio FRAZZOLI : Anytime computation of time-optimal off-road vehicle maneuvers using the rrt. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 3276–3282. IEEE, 2011. *Cité page 86*
- Jung-Hoon HWANG, Ronald C ARKIN et Dong-Soo KWON : Mobile robots at your fingertip : Bezier curve on-line trajectory generation for supervisory control. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, volume 2, pages 1444–1449. IEEE, 2003. *Cité page 87*

- O'Quin JACK O'QUIN, Claridge DAVID et Quinlan MICHAEL : velodyne height map - https://github.com/jack-oquin/velodyne_height_map. *Cité page 91*
- Luc JAULIN : *La robotique mobile cours et exercices*. ISTE editions, London, 2015. ISBN 978-1-78406-087-9. OCLC : 1005733205. *Cité page 32*
- J-S JO, S-H YOU, Ji Yoel JOENG, Kyo Il LEE et Kyongsu YI : Vehicle stability control system for enhancing steerability, lateral stability, and roll stability. *International Journal of Automotive Technology*, 9(5):571, 2008. *Cité page 55*
- Kichun JO, Junsoo KIM et Myoungcho SUNWOO : Real-time road-slope estimation based on integration of onboard sensors with gps using an immpda filter. *IEEE Transactions on Intelligent Transportation Systems*, 14(4):1718–1732, 2013. *2 citations pages 8 et 55*
- Jae-Yun JUN, Jean-Philippe SAUT et Faiz BENAMAR : Pose estimation-based path planning for a tracked mobile robot traversing uneven terrains. *Robotics and Autonomous Systems*, 75:325–339, 2016. *Cité page 86*
- Matthias KALDENBACH, Philipp SPICHAERTZ et Constantinos SOURKOUNIS : Assessment of methods for estimating the maximum coefficient of friction between road and tire. In *2017 IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–6. IEEE, 2017. *Cité page 8*
- Rudolph Emil KALMAN : A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960. *Cité page 32*
- Lydia KAVRAKI, Petr SVESTKA et Mark H OVERMARS : *Probabilistic roadmaps for path planning in high-dimensional configuration spaces*, volume 1994. Unknown Publisher, 1994. *Cité page 86*
- Tamás KEVICZKY, Paolo FALCONE, Francesco HBORRELLI, Jahan ASGARI et Davor HROVAT : Predictive control approach to autonomous vehicle steering. In *2006 American control conference*, pages 6–pp. IEEE, 2006. *Cité page 66*
- Oussama KHATIB : Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 500–505. IEEE, 1985. *2 citations pages 9 et 85*
- Dongshin KIM, Kwangil KIM, Woogab LEE et Inyong HWANG : Development of mando esp (electronic stability program). Rapport technique, SAE Technical Paper, 2003. *Cité page 17*
- Yoram KOREN et Johann BORENSTEIN : Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1398–1404. IEEE, 1991. *Cité page 85*
- Mohamed KRID et Faiz BENAMAR : Design and control of an active anti-roll system for a fast rover. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 274–279. IEEE, 2011. *2 citations pages 6 et 63*
- Mohamed KRID, Faiz BENAMAR et Roland LENAIN : A new explicit dynamic path tracking controller using generalized predictive control. *International Journal of Control, Automation and Systems*, 15(1):303–314, 2017. *4 citations pages 7, 29, 43 et 65*

- Mohamed Larbi KRID : *Commande en suivi de chemin et en roulis des robots mobiles rapides en présence de glissements et d'instabilités*. Thèse de doctorat, Université Pierre et Marie Curie-Paris VI, 2012. *3 citations pages 4, 17 et 18*
- F KÜNHE, J GOMES et W FETTER : Mobile robot trajectory tracking using model predictive control. In *II IEEE latin-american robotics symposium*, volume 51, 2005. *Cité page 63*
- Walter Fetter LAGES et J Gomes da SILVA JR : Model predictive control of a mobile robot using linearization. In *Proceedings of the 2005 IEEE international conference on robotics and automation (ICRA2005)*, pages 885–890. Citeseer, 2004. *Cité page 65*
- Steven M LAVALLE : Rapidly-exploring random trees : A new tool for path planning. 1998. *Cité page 86*
- Roland LENAIN, Mathieu DEREMETZ, Jean-Baptiste BRACONNIER, Benoit THUILOT et Vincent ROUSSEAU : Robust sideslip angles observer for accurate off-road path tracking control. *Advanced Robotics*, 31(9):453–467, 2017. *Cité page 42*
- Roland LENAIN et Benoît THUILOT : Mobile robot control on uneven and slippery ground : An adaptive approach based on a multi-model observer. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1141–1148. IEEE, 2012. *Cité page 55*
- Roland LENAIN, Benoit THUILOT, Christophe CARIOU et Philippe MARTINET : Adaptive control for car like vehicles guidance relying on rtk gps : Rejection of sliding effects in agricultural applications. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 1, pages 115–120. IEEE, 2003. *2 citations pages 6 et 28*
- Roland LENAIN, Benoit THUILOT, Christophe CARIOU et Philippe MARTINET : Model predictive control for vehicle guidance in presence of sliding : application to farm vehicles path tracking. In *Proceedings of the 2005 IEEE international conference on robotics and automation (ICRA2005)*, pages 885–890. IEEE, 2005. *2 citations pages 7 et 65*
- Roland LENAIN, Benoit THUILOT, Christophe CARIOU et Philippe MARTINET : Sideslip angles observer for vehicle guidance in sliding conditions : Application to agricultural path tracking tasks. In *Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA2006)*., pages 3183–3188. IEEE, 2006. *Cité page 4*
- Roland LENAIN, Benoit THUILOT, Christophe CARIOU et Philippe MARTINET : Adaptive and predictive path tracking control for off-road mobile robots. *European journal of control*, 13(4):419–439, 2007. *2 citations pages 28 et 65*
- Roland LENAIN, Benoit THUILOT, Christophe CARIOU et Philippe MARTINET : Mixed kinematic and dynamic sideslip angle observer for accurate control of fast off-road mobile robots. *Journal of Field Robotics*, 27(2):181–196, 2010. *4 citations pages 6, 8, 17 et 42*
- Li LI et Fei-Yue WANG : *Advanced motion control and sensing for intelligent vehicles*. Springer Science & Business Media, 2007. *Cité page 17*
- Chia-Shang LIU et Huei PENG : Road friction coefficient estimation for vehicle path prediction. *Vehicle system dynamics*, 25(S1):413–425, 1996. *2 citations pages 22 et 43*
- Éric LUCET : *Contribution à la commande dynamique des robots mobiles rapides à roues sur sols glissants*. Thèse de doctorat, 2010. *Cité page 17*

- Eric LUCET, Roland LENAIN et Christophe GRAND : Dynamic path tracking control of a vehicle on slippery terrain. *Control Engineering Practice*, 42:60–73, 2015. *Cité page 4*
- Saïd MAMMAR, Nicoleta Minoiu ENACHE, Sébastien GLASER, Benoit LUSETTI et Andre Benine NETO : Lane keeping automation at tire saturation. *In Proceedings of the 2010 American Control Conference*, pages 6466–6471. IEEE, 2010. *Cité page 28*
- F MANCOSU, R SANGALLI, F CHELI, G CIARLARIELLO et F BRAGHIN : A mathematical-physical 3d tire model for handling/comfort optimization on a vehicle : comparison with experimental results. *Tire Science and Technology*, 28(4):210–232, 2000. *Cité page 18*
- Riccardo MARINO, Stefano SCALZI et Mariana NETTO : Nested pid steering control for lane keeping in autonomous vehicles. *Control Engineering Practice*, 19(12):1459–1467, 2011. *Cité page 29*
- Alain MICAELLI et Claude SAMSON : Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. 1993. *Cité page 4*
- Christian MICHELONI, Gian Luca FORESTI, Claudio PICIARELLI et Luigi CINQUE : An autonomous vehicle for video surveillance of indoor environments. *IEEE Transactions on Vehicular Technology*, 56(2):487–498, 2007. *Cité page 2*
- Manfred MORARI : *Model predictive control*. 1993. *Cité page 64*
- Nicolas MORETTE : *Contribution à la Navigation de robots mobiles : approche par modèle direct et commande prédictive*. Thèse de doctorat, 2009. *Cité page 10*
- Pascal MORIN et Claude SAMSON : Trajectory tracking for nonholonomic vehicles. *In Robot Motion and Control*, pages 3–23. Springer, 2006. *Cité page 28*
- VF MUÑOZ, JL MARTINEZ et A OLLERO : New continuous-curvature local path generators for mobile robots. *IFAC Proceedings Volumes*, 25(6):455–460, 1992. *Cité page 87*
- Keiji NAGATANI, Yosuke IWAI et Yutaka TANAKA : Sensor based navigation for car-like mobile robots using generalized voronoi graph. *In Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, volume 2, pages 1017–1022. IEEE, 2001. *Cité page 87*
- Amin NIKOUBIN et Reza HAGHIGHI : Lyapunov-based nonlinear disturbance observer for serial n-link robot manipulators. *Journal of Intelligent and Robotic Systems*, 55(2-3):135–153, 2009. *Cité page 45*
- Akitaka NISHIO, Kenji TOZU, Hiroyuki YAMAGUCHI, Katsuhiro ASANO et Yasushi AMANO : Development of vehicle stability control system based on vehicle sideslip angle estimation. *SAE transactions*, pages 115–122, 2001. *Cité page 55*
- Ange NIZARD, Benoit THUILOT, Roland LENAIN et Youcef MEZOUAR : Nonlinear path tracking controller for bi-steerable vehicles in cluttered environments. *IFAC-PapersOnLine*, 49(15):19–24, 2016. *Cité page 4*
- Lydie NOUVELIERE *et al.* : Experimental vehicle longitudinal control using a second order sliding mode technique. *Control Engineering Practice*, 15(8):943–954, 2007. *Cité page 28*

- A OLLERO et O AMIDI : Predictive path tracking of mobile robots. application to the cmu navlab. *In Proceedings of the Mechatronics and Robotics*, pages 525–530. IEEE, 2004. *Cité page 65*
- Teddy ORT, Liam PAULL et Daniela RUS : Autonomous vehicle navigation in rural environments without detailed prior maps. *In 2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2040–2047. IEEE, 2018. *Cité page 86*
- Jens OVERBY : Automatic 3d building reconstruction from airborne laser scanning and cadastral data using hough transform. 2004. *Cité page 89*
- Hans PACEJKA : *Tire and vehicle dynamics*. Elsevier, 2005. *2 citations pages 5 et 18*
- Stephen R PASTOR et Gordon L TIERNEY : Method and apparatus for estimating incline and bank angles of a road surface, août 29 1995. US Patent 5,446,658. *Cité page 55*
- AI PROPOI : Application of linear programming methods for the synthesis of automatic sampled-data systems. *Automat. i Telemekh.*, 24:912–920, 1963. *Cité page 63*
- Xiangjun QIAN, Arnaud DE LA FORTELLE et Fabien MOUTARDE : A hierarchical model predictive control framework for on-road formation control of autonomous vehicles. *In 2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 376–381. IEEE, 2016. *Cité page 66*
- Rajesh RAJAMANI : *Vehicle dynamics and control*. Springer Science & Business Media, 2011. *Cité page 17*
- Laura R RAY : Nonlinear tire force estimation and road friction identification : simulation and experiments. *Automatica*, 33(10):1819–1833, 1997. *Cité page 44*
- Josef REITBERGER, Cl SCHNÖRR, Peter KRZYTEK et Uwe STILLA : 3d segmentation of single trees exploiting full waveform lidar data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(6):561–574, 2009. *Cité page 89*
- Jacques RICHALET, A RAULT, JL TESTUD et J PAPON : Model predictive heuristic control. *Automatica (Journal of IFAC)*, 14(5):413–428, 1978. *Cité page 63*
- Mathieu RICHIER, Roland LENAIN, Benoit THUILOT et Christophe DEBAIN : On-line estimation of a stability metric including grip conditions and slope : Application to rollover prevention for all-terrain vehicles. *In 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4569–4574. IEEE, 2011. *Cité page 55*
- Radu Bogdan RUSU et Steve COUSINS : 3d is here : Point cloud library (pcl). *In 2011 IEEE international conference on robotics and automation (ICRA)*, pages 1–4. IEEE, 2011. *2 citations pages 90 et IX*
- Jihan RYU et J Christian GERDES : Estimation of vehicle roll and road bank angle. *In Proceedings of the 2004 American control conference*, volume 3, pages 2110–2115. IEEE, 2004a. *3 citations pages 8, 55 et 56*
- Jihan RYU et J Christian GERDES : Integrating inertial sensors with global positioning system (gps) for vehicle dynamics control. *Journal of Dynamic Systems, Measurement, and Control*, 126(2):243–254, 2004b. *Cité page 55*
- Hideaki SASAKI et Takatoshi NISHIMAKI : A side-slip angle estimation using neural network for a wheeled vehicle. Rapport technique, SAE Technical Paper, 2000. *Cité page 55*

- Yazid SEBSADJI, Sébastien GLASER, Saïd MAMMAR et Mariana NETTO : Vehicle roll and road bank angles estimation. *IFAC Proceedings Volumes*, 41(2):7091–7097, 2008. *Cité page 55*
- Chouki SENTOUH, Yazid SEBSADJI, Saïd MAMMAR et Sébastien GLASER : Road bank angle and faults estimation using unknown input proportional-integral observer. *In 2007 European Control Conference (ECC)*, pages 5131–5138. IEEE, 2007. *2 citations pages 6 et 55*
- Payman SHAKOURI, A ORDYS, Dina Shona LAILA et Mohamad ASKARI : Adaptive cruise control system : comparing gain-scheduling pi and lq controllers. *IFAC Proceedings Volumes*, 44(1):12964–12969, 2011. *Cité page 28*
- Zvi SHILLER, Yu-Rwei GWO *et al.* : Dynamic motion planning of autonomous vehicles. *IEEE Transactions on Robotics and Automation*, 7(2):241–249, 1991. *Cité page 10*
- Shahrul Naim SIDEK et N SARKAR : *Dynamic modeling and control of nonholonomic wheeled mobile robot subjected to wheel slip*. Thèse de doctorat, Vanderbilt University, 2008. *Cité page 5*
- C SIERRA, E TSENG, A JAIN et H PENG : Cornering stiffness estimation based on vehicle lateral dynamics. *Vehicle System Dynamics*, 44(sup1):24–38, 2006. *Cité page 43*
- Kenneth Renny SIMBA, Naoki UCHIYAMA et Shigenori SANO : Real-time smooth trajectory generation for nonholonomic mobile robots using bézier curves. *Robotics and Computer-Integrated Manufacturing*, 41:31–42, 2016. *Cité page 10*
- Reid SIMMONS : The curvature-velocity method for local obstacle avoidance. *In Proceedings of IEEE international conference on robotics and automation*, volume 4, pages 3375–3382. IEEE, 1996. *Cité page 10*
- Jarrold M SNIDER *et al.* : Automatic steering methods for autonomous automobile path tracking. 2009. *Cité page 17*
- OJ SORDALEN : Conversion of the kinematics of a car with n trailers into a chained form. *In [1993] Proceedings IEEE International Conference on Robotics and Automation*, pages 382–387. IEEE, 1993. *Cité page 28*
- Ioannis STAMOS, Gene YU, George WOLBERG et Siavash ZOKAI : 3d modeling using planar segments and mesh elements. *In Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pages 599–606. IEEE, 2006. *Cité page 89*
- Todor STOYANOV, Rasoul MOJTAHEDZADEH, Henrik ANDREASSON et Achim J LILIENTHAL : Comparative evaluation of range sensor accuracy for indoor mobile robotics and automated logistics applications. *Robotics and Autonomous Systems*, 61(10):1094–1105, 2013. *Cité page 2*
- JL TESTUD, J RICHALET, A RAULT et J PAPON : Model predictive heuristic control : Applications to industrial processes. *Automatica*, 14(5):413–428, 1978. *Cité page 63*
- Sebastian THRUN, Mike MONTEMERLO, Hendrik DAHLKAMP, David STAVENS, Andrei ARON, James DIEBEL, Philip FONG, John GALE, Morgan HALPENNY, Gabriel HOFFMANN *et al.* : Stanley : The robot that won the darpa grand challenge. *Journal of field Robotics*, 23(9):661–692, 2006. *Cité page 17*
- Benoît THUILOT : *Contribution à la modélisation et à la commande de robots mobiles à roues*. Thèse de doctorat, École Nationale Supérieure des Mines de Paris, 1995. *Cité page 18*

- Hongtei Eric TSENG : Dynamic estimation of road bank angle. *Vehicle system dynamics*, 36 (4-5):307–328, 2001. *2 citations pages 9 et 55*
- Ari TUONONEN : Vehicle lateral state estimation based on measured tyre forces. *Sensors*, 9 (11):8761–8775, 2009. *Cité page 43*
- Iwan ULRICH et Johann BORENSTEIN : Vfh+ : Reliable obstacle avoidance for fast mobile robots. In *Proceedings. 1998 IEEE international conference on robotics and automation (Cat. No. 98CH36146)*, volume 2, pages 1572–1577. IEEE, 1998. *Cité page 10*
- Iwan ULRICH et Johann BORENSTEIN : Vfh/sup* : Local obstacle avoidance with look-ahead verification. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 3, pages 2505–2511. IEEE, 2000. *2 citations pages 9 et 10*
- Ali Y UNGOREN, Huei PENG et HE TSENG : Experimental verification of lateral speed estimation methods. 2002. *Cité page 55*
- Ali Y UNGOREN, Huei PENG et HE TSENG : A study on lateral speed estimation methods. *International Journal of Vehicle Autonomous Systems*, 2(1-2):126–144, 2004. *Cité page 55*
- HA VAN ESSEN et Henk NIJMEIJER : Non-linear model predictive control for constrained mobile robots. In *2001 European Control Conference (ECC)*, pages 1157–1162. IEEE, 2001. *Cité page 66*
- Felix VON HUNDELSHAUSEN, Michael HIMMELSBACH, Falk HECKER, Andre MUELLER et Hans-Joachim WUENSCHÉ : Driving with tentacles : Integral structures for sensing and motion. *Journal of Field Robotics*, 25(9):640–673, 2008. *Cité page 86*
- Liuping WANG : *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media, 2009. *Cité page 64*
- Rongrong WANG et Wang JUNMIN : Tire–road friction coefficient and tire cornering stiffness estimation based on longitudinal tire force difference generation. *Control Engineering Practice*, 21(1):65–75, 2013. *2 citations pages 8 et 22*
- Ruijie WANG, Guodong YIN et Xianjian JIN : Robust adaptive sliding mode control for nonlinear four-wheel steering autonomous vehicles path tracking systems. In *2016 IEEE 8th international power electronics and motion control conference (IPEMC-ECCE Asia)*, pages 2999–3006. IEEE, 2016. *2 citations pages 7 et 28*
- Xueming WANG : *Wall Following Algorithm for a Mobile Robot Using Extended Kalman Filter*. Thèse de doctorat, 2011. *Cité page 86*
- John C WHITEHEAD : Four wheel steering : Maneuverability and high speed stabilization. *SAE Transactions*, pages 668–679, 1988. *Cité page 4*
- Pierre-Brice WIEBER : Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pages 137–142. IEEE, 2006. *Cité page 63*
- S YAMAZAKI, O FURUKAWA et T SUZUKI : Study on real time estimation of tire to road friction. *Vehicle System Dynamics*, 27(S1):225–233, 1997. *Cité page 22*

Jong-Min YANG et Jong-Hwan KIM : Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots. *IEEE Transactions on robotics and automation*, 15(3):578–587, 1999. *Cité page 28*

Pan ZHAO, Jiajia CHEN, Tao MEI et Huawei LIANG : Dynamic motion planning for autonomous vehicle in unknown environments. *In 2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 284–289. IEEE, 2011. *Cité page 29*