HAL
open science

Des systèmes d'aide à la conduite au véhicule autonome connecté

Nolwenn Monot

▶ **To cite this version:**

Nolwenn Monot. Des systèmes d'aide à la conduite au véhicule autonome connecté. Automatique / Robotique. Université de Bordeaux, 2019. Français. NNT : 2019BORD0109 . tel-02347511

HAL Id: tel-02347511

https://theses.hal.science/tel-02347511v1

Submitted on 5 Nov 2019

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

PRÉSENTÉE POUR OBTENIR LE GRADE DE

DOCTEUR
DE L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

SPÉCIALITÉ : AUTOMATIQUE, PRODUCTIQUE, SIGNAL ET IMAGE

par

Nolwenn MONOT

DES SYSTÈMES D'AIDE À LA CONDUITE AU VÉHICULE AUTONOME CONNECTÉ

Sous la direction de : Xavier MOREAU
et le co-encadrement de : André BENINE-NETO

Soutenance le 9 juillet 2019

Membres du jury :

Mme	NOUVELIERE, Lydie	Maître de Conférences - HDR	Université d'Evry-Val-d'Essonne	Rapporteur
M.	RABHI, Abdelhamid	Maître de Conférences - HDR	Université de Picardie Jules Verne	Rapporteur
M.	BONNIFAIT, Philippe	Professeur	Université de Technologie de Compiègne	Président
M.	DUGARD, Luc	Directeur de Recherche CNRS	Laboratoire Gipsa-Lab	Examinateur
M.	MELCHIOR, Pierre	Professeur	Bordeaux INP	Examinateur
Mme	TALJ, Reine	Chargée de Recherche CNRS	Laboratoire Heudiasyc	Examinateur
M.	MOREAU, Xavier	Professeur	Université de Bordeaux	Directeur
M.	BENINE-NETO, André	Maître de Conférences	Université de Bordeaux	Co-encadrant
M.	AIOUN, François	Docteur Ingénieur	Groupe PSA	Co-encadrant
Mme	RIZZO, Audrey	Docteur Ingénieur	Groupe PSA	Co-encadrant

- 2019 -

Remerciements

Ces travaux ont été effectués en collaboration avec le Laboratoire d'Intégration du Matériau au Système (IMS) de Bordeaux, UMR 5218 du CNRS, et la Direction de la Recherche et de l'Innovation Automobile (DRIA) du groupe PSA dans le cadre de l'OpenLab « Electronics and Systems for Automotive ».

Je tiens à remercier très sincèrement mes encadrants pour leur confiance et leur soutien qui m'ont permis de mener à bien ces travaux de thèse.

Je commencerai ainsi par remercier M. Xavier Moreau, directeur de cette thèse, pour la confiance qu'il m'a donnée tout du long et même avant de commencer, pour m'avoir transmis sa passion pour la recherche et pour m'avoir apporté de nombreuses connaissances et ses bons conseils. Merci aussi à M. André Benine-Neto, co-encadrant, qui a parfaitement su m'accompagner et m'encourager, et dont les remarques et suggestions m'ont beaucoup aidée à avancer. Je remercie aussi M. François Aioun, co-encadrant, pour ses conseils avisés, son sens du détail et pour m'avoir conforté dans mes choix. Un immense merci à Mme Audrey Rizzo, co-encadrante, qui, avec à sa grande expertise sur le véhicule autonome, m'a guidée, conseillée et encouragée pendant ces trois années, et dont la bonne humeur indéfectible et l'optimisme à toute épreuve font du bien au moral.

En plus de mes encadrants, je tiens à adresser mes sincères remerciements à l'ensemble des membres du jury. Ainsi, j'exprime ma reconnaissance et mes remerciements à Mme Lydie Nouvelière et M. Abdelhamid Rabhi pour avoir accepté la charge de rapporter ces travaux de thèse, et pour toutes les remarques et questions pertinentes qu'ils ont su m'adresser à travers leurs rapports. De même, je témoigne toute ma reconnaissance à M. Philippe Bonnifait pour avoir accepté le rôle de président du jury. Je remercie également Mme Reine Talj, M. Pierre Melchior et M. Luc Dugard d'avoir accepté le rôle d'examinateur lors de la soutenance.

Je poursuivrai ces remerciements avec mes collègues de travail, pour leur aide, leurs conseils, mais aussi leur convivialité et les bons moments passés en leur compagnie au quotidien.

Du côté de l'IMS, je remercie M. Alain Oustaloup pour la rigueur scientifique qu'il a su me transmettre, pour sa joie de vivre, son sens du partage, et ses visites en fin d'après-midi au laboratoire qui raniment tout l'étage. Merci aussi à tous les permanents, que je retrouvais avec plaisir lors de mes semaines à Bordeaux : Christophe, Tudor, Alain, Pierre, Xavier, André, Patrick, Stéphane et Jocelyn. Enfin, mes plus chaleureux remerciements à Valérie Abel qui chouchoute ses doctorants, nous écoute, nous rassure et nous rend la vie beaucoup plus facile.

Pour le Groupe PSA, l'équipe s'agrandit de plus en plus et il y a beaucoup de monde à remercier. Je commencerai la Team Low Speed : Céline, Audrey R., Damien, Patrice, Elodie, Paul et Jean-François L. Si venir sur la région parisienne me rebutait un peu, le fait de travailler avec vous m'a fait changer d'avis. Merci beaucoup pour votre soutien, vos conseils, votre bonne humeur et votre convivialité, ne changez rien. Merci aussi à Rachid, Jean-Michel, Antonio et Florian T. pour les très nombreuses pauses cafés, inspirantes ou remotivantes, passées en votre compagnie. Je tiens aussi à remercier Jean-François C. pour sa confiance et pour m'avoir permis de continuer cette belle aventure. Plus généralement, merci à l'ensemble de l'équipe ADAS (anciens et nouveaux) pour ces trois années : Fabien, Loïc, Audrey M., Vincent D., Laurent, Marc, Stéphane, Mathieu D., Alain S., Delphine, Arthur, Romain P., Saleh, Patrick, Clément, Fred, Thibault F., Daniele, Stefano, Thibault G., Nada, Jonathan, Jean-Baptiste B., Florian M., Laura, Sofiane, Mark, Maxime, Mathieu P., Samuel, Adrien, Romain S., Corinne, François G. et Vincent A. ; et l'équipe SIEP : François A., Mathieu M. et Franck G.

Je terminerai par remercier mes amis doctorants (pour la plupart docteurs maintenant), parce qu'on se comprend et se soutient mutuellement. Un immense merci à Lamine, Elena, Sergey, Mathieu C., Baptiste, Jean-Louis, Alexandre, Julien, Emna, Jean-Baptiste R., Sara, Kendric, Vincent, Mélodie, Evgeny mais aussi Ghazi et Massinissa, pour la partie IMS. Cette équipe de doctorants est géniale, ces trois années ont été mémorables grâce à vous et j'aurais aimé venir à Bordeaux plus souvent.

Du côté de PSA et des Fous Thèses, merci à Elise pour avoir commencé à organiser les Afterworks qui nous ont rassemblés et merci à Amélie, Adrien, Anthony, Youna, Thomas, Florence, Shirley, Augustin, Jean-Philippe pour tous ces bons moments passés avec vous.

Sur le plan personnel, merci aux anciens de l'ENSEIRB-MATMECA que je suis toujours impatiente de retrouver, que ce soit à Bordeaux ou à Paris. Merci aussi à mes coéquipiers de tennis de table de l'USMV, qui ne liront probablement jamais ses lignes, mais m'ont permis de me vider la tête aux moments où j'en avais le plus besoin. Merci à l'ensemble de ma famille qui m'a toujours soutenue et encouragée pendant cette thèse, même s'ils ne comprenaient pas trop dans quoi je m'embarquais. Enfin, merci à Nicolas d'avoir toujours été présent, même à 600 km.

Dans la vie rien est à craindre, tout est à comprendre.

Marie Curie

Résumé

Cette thèse s'inscrit dans le développement et la conception de fonctions d'aide à la conduite pour les véhicules autonomes de niveau 3 et plus en milieu urbain ou péri urbain. Du fait d'un environnement plus complexe et de trajectoires possibles plus nombreuses et sinueuses, les algorithmes des véhicules autonomes développés pour l'autoroute ne sont pas adaptés pour le milieu urbain. L'objectif de la thèse est de mettre à disposition des méthodes et des réalisations pour permettre au véhicule autonome d'évoluer en milieu urbain. Cette thèse se focalise sur la proposition de solutions pour améliorer le guidage latéral des véhicules autonomes en milieu urbain à travers l'étude de la planification de trajectoire en situation complexe, l'analyse du comportement des usagers et l'amélioration du suivi de ces trajectoires complexes à faibles vitesses. Les solutions proposées doivent fonctionner en temps réel dans les calculateurs des prototypes pour pouvoir ensuite être appliquées sur route ouverte. L'apport de cette thèse est donc autant théorique que pratique.

Après une synthèse des fonctions d'aide à la conduite présentes à bord des véhicules et une présentation des moyens d'essais mis à disposition pour la validation des algorithmes proposés, une analyse complète de la dynamique latérale est effectuée dans les domaines temporel et fréquentiel. Cette analyse permet alors la mise en place d'observateurs de la dynamique latérale pour estimer des signaux nécessaires aux fonctions de guidage latéral et dont les grandeurs ne sont pas toujours mesurables, fortement dégradées ou bruitées. La régulation latérale du véhicule autonome se base sur les conclusions apportées par l'analyse de cette dynamique pour proposer une solution de type multirégulateur capable de générer une consigne en angle volant pour suivre une trajectoire latérale quelle que soit la vitesse. La solution est validée tant en simulation que sur prototype pour plusieurs vitesses sur des trajectoires de changement de voie. La suite de la thèse s'intéresse à la génération d'une trajectoire en milieu urbain tenant compte non seulement de l'infrastructure complexe (intersection/rond-point) mais également des comportements des véhicules autour. C'est pourquoi, une analyse des véhicules de l'environnement est menée afin de déterminer leur comportement et leur trajectoire. Cette analyse est essentielle pour la méthode de génération de trajectoire développée dans cette thèse. Cette méthode, basée sur l'algorithme A* et enrichie pour respecter les contraintes géométriques et dynamiques du véhicule, se focalise d'abord dans un environnement statique complexe de type parking ou rond-point. Des points de passage sont intégrés à la méthode afin de générer des trajectoires conformes au code de la route et d'améliorer le temps de calcul. La méthode est ensuite adaptée pour un environnement dynamique où le véhicule est alors capable, sur une route à double sens de circulation, de dépasser un véhicule avec un véhicule arrivant en sens inverse.

Mots clés : Automatique, dynamique du véhicule, assistance à la conduite, régulation, génération de trajectoire, prédiction de trajectoire

Abstract

This thesis is about the design of driving assistance systems for level 3 urban automated driving. Because of a more complex of the environment and a larger set of possible trajectories, the algorithms of highway automated driving are not adapted to urban environment. This thesis objective is to provide methods and algorithms to enable the vehicle to perform automated driving in urban scenarios, focusing on the vehicle lateral guidance and on the path planning. The proposed solutions operate in real-time on board of the automated vehicle prototypes. The contribution of this thesis is as theoretical as practical.

After a synthesis of the driving assistance systems available on current cars and a presentation of the prototypes used for the validation of the algorithms developed in the thesis, a complete analysis of the vehicle lateral dynamics is carried out in time and frequency domains. This analysis enables the design of observers of the lateral dynamics in order not only to estimate signals required for the lateral guidance functions but also to increase reliability of available measurements. Based on the conclusions from the analysis of lateral dynamic, a multi-controller solution has been proposed. It enables the computation of a steering wheel input to follow a trajectory at any longitudinal speed. The solution is validated in simulation and on real road traffic for lane change scenarios. Another contribution consist in an analysis on the other vehicles of the environment is conducted in order to identify their behaviors and which maneuver there are performing. This analysis is essential for the path planning function developed in the thesis. This method, based on the A* algorithm and extended to respect geometric and dynamic constraints, firstly focuses on static environment such as a parking lot. Waypoints are added to the method in order to compute trajectories compatible with traffic regulation and improve the computation time. The method is then adapted for dynamic environment where, in the end, the vehicle is able to perform overtaking manoeuvers in a complex environment.

Keywords : Automatic, vehicle dynamics, driving assistance system, control, path planning, trajectory prediction

Table des matières

Introduction générale **9**

Chapitre 1 - Évolution de la mécatronique à bord des véhicules **13**
1.1 La mécatronique et le secteur automobile 15
1.2 Les fonctions mécatroniques utilisant les données véhicule 16
1.2.1 Les fonctions de confort 16
1.2.2 Les fonctions de sécurité active 18
1.3 Les systèmes d'aide à la conduite 22
1.3.1 Les ADAS informatifs 22
1.3.2 Les ADAS actifs 24
1.4 Le véhicule autonome 29
1.4.1 Avantages, niveaux d'autonomies et réglementations du véhicule autonome 30
1.4.2 Les capteurs 33
1.4.3 Architecture 38
1.5 Présentation des prototypes 39
1.5.1 Architecture électrique et électronique pour la conduite autonome 39
1.5.2 Les prototypes utilisés pendant la thèse 39
1.6 Problématique et objectifs 40

Chapitre 2 - Modélisation et observation de la dynamique latérale **41**
2.1 Les modèles dynamiques du véhicule 43
2.1.1 Le modèle de validation : le modèle non linéaire 43
2.1.2 Les modèles simplifiés 46
2.1.3 Comparaison temporelle 50
2.1.4 Comparaison fréquentielle des modèles de la dynamique latérale du véhicule 55
2.2 Mise en évidence de l'influence de la vitesse longitudinale 59
2.2.1 Description du scénario 59
2.2.2 Résultats de la simulation pour le modèle de validation 60
2.2.3 Comparaison temporelle avec les modèles simplifiés 63
2.2.4 Analyse fréquentielle du modèle linéaire 66
2.2.5 Comparaison fréquentielle avec les autres modèles simplifiés 69
2.3 Observateur d'état des systèmes dynamiques : état de l'art 70
2.3.1 Les méthodes 71
2.3.2 Les modèles 72
2.3.3 Les variables observées 72
2.3.4 Les capteurs supplémentaires 73
2.4 Estimation de la dynamique latérale par approche multimodèle 73
2.4.1 Le modèle Takagi-Sugeno 74
2.4.2 Observabilité du modèle Takagi-Sugeno 76
2.4.3 Validation du modèle Takagi-Sugeno 76

2.4.4 Problème d'échantillonnage à basse vitesse 78
2.5 Validation et résultats pour l'estimation de la vitesse de lacet 81

Chapitre 3 - Guidage latéral du véhicule **86**
3.1 État de l'art : guidage latéral du véhicule 88
3.2 La conception de régulateurs PID et CRONE dans le domaine fréquentiel . 91
3.2.1 Le PID 91
3.2.2 La commande CRONE 92
3.3 Régulation latérale par PID 97
3.4 Régulation latérale par CRONE de 3ème génération 101
3.5 Approche multirégulateur 106
3.6 Régulation latérale par multi-PID 108
3.7 Théorie et apport du point de visée 113
3.7.1 Définition du point de visée 113
3.7.2 Influence du point de visée sur la dynamique latérale 114
3.8 Régulation latérale avec point de visée par CRONE de 3ème génération . . . 116
3.9 Régulation latérale avec point de visée par multi-PID 119
3.10 Validation et résultats sur prototype 124
3.10.1 Intégration du régulateur dans le prototype ADN1 124
3.10.2 Résultats sur route ouverte : maintien dans la voie 124
3.10.3 Résultats sur route ouverte : changement de voie 126
3.10.4 Bilan : avantages, inconvénients et perspectives 128

Chapitre 4 - Contrainte de l'environnement et prédiction de trajectoire des véhicules **130**
4.1 L'influence de l'environnement sur le calcul de la trajectoire : la conscience de situation 132
4.1.1 Le concept de la conscience de situation 132
4.1.2 État de l'art : détection d'intention 133
4.1.3 État de l'art : prédiction de trajectoire 134
4.1.4 Introduction à l'apprentissage automatique 136
4.1.5 Objectifs et méthodes 139
4.2 Les réseaux de neurones : fonctionnement et apprentissage 140
4.2.1 Fonctionnement d'un neurone 140
4.2.2 Propagation de l'information dans le réseau 141
4.2.3 Entraînement du réseau : la rétropropagation 143
4.2.4 Méthodologie de construction du réseau 146
4.3 Détection d'intention : changement de voie sur autoroute 148
4.3.1 Description du problème 148
4.3.2 Détection d'intention : méthode à base de règles 149
4.3.3 Méthode par réseau de neurones 154
4.3.4 Résultats 158
4.4 Prédiction de trajectoire : le cas du changement de voie 160
4.4.1 Description du problème 160
4.4.2 Le réseau Long-Short Term Memory 160
4.4.3 Structures de LSTM utilisées 162
4.4.4 Résultats 162
4.4.5 Perspectives pour la prédiction de trajectoire en environnement urbain . . 164

Chapitre 5 - Génération de trajectoire d'un véhicule autonome en environnement urbain **168**
5.1 Planification de trajectoire d'un véhicule autonome : état de l'art 170
5.1.1 En environnement statique 171
5.1.2 Approches locales : des approches réactives à court terme 176
5.1.3 En environnement dynamique 178
5.2 Recherche de chemin en environnement statique 180
5.2.1 A* 181
5.2.2 Recherche de chemin par A* sous contrainte géométrique : exemples applicatifs 186
5.2.3 Dijkstra 191
5.2.4 Cas applicatif : le carrefour giratoire 192
5.2.5 Résultats sur prototype en parking 199
5.3 Génération de trajectoire en environnement dynamique 200
5.3.1 Intégration de la dimension temporelle pour la recherche de trajectoire . . 200
5.3.2 Application : dépassement d'un vélo 202
5.3.3 Intégration des contraintes dynamiques dans le A* 204
5.3.4 Application : dépassement sur voie à double sens 206
5.3.5 Bilan : avantages, problématiques et perspectives 213

Conclusion générale et perspectives **214**

Annexe A - L'approche fréquentielle pour l'analyse des systèmes dynamiques et la conception de régulateurs **232**
A.1 Outils pour l'analyse fréquentielle des systèmes dynamiques 232
A.2 Les outils pour la conception de régulateurs et l'analyse de systèmes bouclés dans le domaine fréquentiel 235

Annexe B - Influence des variations paramétriques sur la dynamique latérale du véhicule **242**
B.1 Influence de la masse 242
B.2 Influence de l'adhérence 246

Annexe C - Quelques méthodes pour l'observation des systèmes dynamiques **250**
C.1 Observation des systèmes à modèles linéaires 250
C.2 Observations des systèmes à modèles non linéaires 253

Annexe D - Influence du choix des points de fonctionnement pour le multi-PID **256**
D.1 Les types de découpages possibles 256
D.2 L'influence des découpages sur la boucle ouverte dans le cas du guidage latéral 257

Abréviations et notations

Abréviation

ABS	Anti-blocage des roues
ACC	Adaptative Cruise Control
ADAS	Advanced Driver Assistance System
ADASIS	Advanced Driver Assistance Systems Interface Specifications
AEB	Automatic Emergency Braking
AFU	Aide au Freinage d'Urgence
ALV	Autonomous Land Vehicle
ASR/TSC	Système antipatinage
AVA	Autonomous Vehicule for All project
BPGA	Boîtier de Protection et de Gestion des Alimentations
BSD	Blind Spot Detection
BSI	Boîtier de Servitude Intelligent
BSM	Boîtier de Servitude Moteur
CAN	Controller Area Network
CL-RRT	Closed Loop Rapidly-exploring Random Tree
CMM	Calculateur Multifonction Moteur
CNN	Convolutional Neural Network
CRONE	Commande Robuste d'Ordre Non entier
DAE	Direction Assistée Électrique
DARPA	Defense Advanced Research Projects Agency
DMA	Driver Monitoring Assist
EBD	Electronic Brakeforce Distribution
ECU	Electronic Control Unit
ESP/ESC	Electronic Stability Program/Control
EuroNCAP	European New Car Assessment Program
FTBF	Fonction de Transfert en Boucle Fermée
FTBO	Fonction de Transfert en Boucle Ouverte
GPS	Global Positioning System
HMM	Hidden Markov Model
IHM	Interface Homme Machine

ISO	International Organization for Standardization
LDW/AFIL	Lane Danger Warning
LKA	Lane Keeping Assist
LPA	Lane Positioning Assist
LPI	Linéaire à Paramètres Invariants
LPV	Linéaire à Paramètres Variants
LSTM	Long-Short Term memory
LVV	Limiteur de Vitesse Variable
MABX	microAutoBox
MFRH	Matrice de Fusibles et de Relais Habitacle
MFRM	Matrice de Fusibles et de Relais Moteur
MLP	Multilayer Perceptron
MPC	Model Predictive Control
NGSIM	Next Generation Simulation
NHTSA	National Highway Traffic Safety Administration
NRMSE	Normalized Root Mean Square Error
ONU	Organisation des Nations Unies
PID	Proportionnel Intégral Dérivé
PSO	Particle Swarm Optimization
RNN	Recurrent Neural Network
RRT	Rapidly-exploring Random Tree
RVV	Régulateur de Vitesse Variable
SAE	Society of Automotive Engineers
SLAM	Simultaneous Localization and Mapping
SOTIF	Safety Of The Intended Functionality
SVM	Support Vector Machine
TIV	Temps Inter-Véhicule
UAV	Unmaned Aerial Vehicle

Notations

Grandeurs véhicule

Γ_t	accélération latérale dans le repère absolu
μ	adhérence de la route
β_v	angle de braquage des roues
Δ_G	angle de dérive du centre de gravité
Δ_p	angle de dérive d'un pneumatique
Ψ	angle de lacet
Θ_v	angle volant
c_{yr}	coefficient de rigidité de dérive des pneumatiques arrière
c_{yf}	coefficient de rigidité de dérive des pneumatiques avant
F_y	effort latéral d'un pneumatique
F_x	effort longitudinal d'un pneumatique
F_z	effort normal appliqué au pneumatique
λ	facteur de démultiplication entres les angles volant et roues
l_r	largeur de voie arrière
l_f	largeur de voie avant
L_r	longueur de l'empattement arrière
L_f	longueur de l'empattement avant
L	longueur du véhicule
M_t	masse totale du véhicule
I_z	moment d'inertie de l'axe vertical
Y_G	position latérale du centre de gravité dans le repère absolu
X_G	position longitudinale du centre de gravité dans le repère absolu
R	rayon de courbure
C_{Σ_z}	somme des couples extérieurs appliqués à l'axe vertical
F_{Σ_y}	somme des efforts transverse
V_y	vitesse latérale
V_x	vitesse longitudinale

Autres grandeurs physiques

t	temps
k	échantillon de temps
T_{θ_v}	période d'angle volant
T_e	période d'échantillonnage
ω	pulsation
ζ	facteur d'amortissement
M_ϕ	marge de phase

Notations mathématiques

$\dot{x}(t)$	dérivée temporelle de x
$\hat{x}$	estimation de x
$\mathcal{R}_0$	repère absolu
$\mathcal{R}$	repère véhicule
$\mathbb{R}$	ensemble des réels
$\mathbb{R}^+$	ensemble des réels positifs
$\mathbb{C}$	ensemble des complexes
I_n	matrice identité de taille n
s	constante de Laplace
A^{-1}	inverse de la matrice (ou du vecteur) A
A^T	transposée de la matrice (ou du vecteur) A
e^x	exponentiel de x
$\dfrac{\partial f}{\partial x}$	dérivée partielle de f par rapport à x
∇f	gradient de f
$\lvert x \rvert$	valeur absolue de x

Introduction générale

Contexte de la thèse

Cette thèse a été menée dans le cadre d'une Convention CIFRE « Convention Industrielle de Formation par la Recherche ». Elle s'est déroulée dans l'équipe CRONE du groupe Automatique du laboratoire IMS de l'Université de Bordeaux, Bordeaux INP, UMR 5218 CNRS, pour le côté académique et au sein des entités DSTF/SIEP et AOBS/ADAS de la DRIA (Direction de la Recherche et de l'Ingénierie Avancée) du Groupe PSA, pour le côté industriel. Cette thèse entre dans le cadre de l'OpenLab « Electronics and systems for Automotive » entre le laboratoire IMS et le Groupe PSA.

Les travaux ont été effectués sous la direction de Xavier Moreau, Professeur à l'Université de Bordeaux, co-encadré par André Benine-Neto, Maître de Conférences à l'Université de Bordeaux, et sous la responsabilité en entreprise d'Audrey Rizzo et François Aioun, Docteurs et Ingénieurs de la DRIA.

Le véhicule autonome : contexte industriel

Depuis l'apparition de la Google Car fin 2010, la voiture autonome est passée du concept car futuriste à la réalité. Cette annonce n'est pas seulement un bouleversement technologique, il est aussi industriel. En étant le premier à annoncer une voiture autonome fonctionnelle, le géant du web devance tous les constructeurs automobiles historiques. La course à la mobilité autonome est lancée.

Les enjeux du véhicule autonome ne sont pas seulement économiques, ils sont aussi technologiques, sécuritaires, sociaux et éthiques. Le développement des véhicules autonomes devient possible grâce aux nouvelles technologies de capteurs et aux nouveaux processeurs dont les puissances de calcul augmentent d'année en année. La difficulté majeure réside dans le développement logiciel des fonctions autonomes. Il faut implanter dans les calculateurs les algorithmes qui vont agir sur les organes du véhicule pour reproduire une conduite humaine. Mais quelles règles donner au véhicule pour qu'il se déplace dans l'environnement tout en respectant le Code de la route et en interagissant avec les autres usagers ? Comment intégrer ces règles correctement ?

Viennent alors les enjeux sécuritaires. En enlevant les erreurs de conduite humaine liées à l'inattention du conducteur et aux mauvais choix de conduite, le véhicule autonome pourrait réduire drastiquement le nombre d'accidents de la route. La question qui se pose alors est comment être sûr que le véhicule autonome garantisse la sécurité des passagers et des autres usagers de la route ? Les capteurs sont-ils fiables et permettent-ils de retranscrire correctement l'environnement dans lequel circule le véhicule ? Les règles implantées peuvent-elles faire face à toutes les situations ?

Un véhicule autonome est un gain de temps considérable pour les utilisateurs. Le conducteur, en devenant passager, peut effectuer d'autres tâches tout en étant libéré du stress que peut provoquer la conduite pour certaines personnes. Cependant, avant de lâcher complètement le volant, une phase d'acceptation de la conduite autonome est nécessaire. Quelle stratégie adopter pour que les passagers aient confiance en cette technologie ?

Enfin, le véhicule autonome soulève des enjeux éthiques, notamment en cas d'accidents. Si un véhicule autonome est impliqué dans un accident, comment déterminer qui est en tort ? Qui devient responsable, le propriétaire ou le constructeur ?

Ce sont toutes ces questions, et bien d'autres, auxquelles les acteurs du secteur doivent répondre avant de pouvoir commercialiser une voiture autonome. Étant donné la complexité du problème et l'étendue des domaines scientifiques qu'il couvre, les acteurs se multiplient. Ainsi, les constructeurs s'allient aux entreprises du numérique pour mutualiser les recherches et aller plus vite dans le développement.

Les premiers véhicules pour particuliers seront partiellement autonomes, c'est-à-dire que la fonction autonome ne pourra être active que si certaines conditions sont respectées. De plus, le conducteur pourrait être sollicité en cas d'évènement inattendu que le véhicule ne sait pas gérer seul. La majorité des constructeurs aimerait proposer de telles fonctions sur autoroutes dès 2021. Il se pose cependant un problème de législation. En Europe, la majorité des pays sont signataires de la Convention de Vienne de 1968 qui oblige le conducteur à garder les mains sur le volant et rester maître de son véhicule. Pour faire évoluer la législation, un amendement de 2016 autorise le transfert de tâche de conduite au véhicule à condition que le conducteur puisse les remplacer ou les désactiver. Ensuite, en 2018, l'ONU a adopté une résolution pour le déploiement des véhicules hautement et complètement automatisés afin d'intégrer ces véhicules dans le Code de la route.

Objectifs de la thèse

C'est dans ce contexte que le Groupe PSA développe sa propre technologie de véhicule autonome. Au début de cette thèse en 2016, les fonctions d'automatisation de la conduite sur autoroute sont déjà bien avancées. L'objectif est alors l'amélioration des fonctions existantes et leur enrichissement pour la conduite autonome en environnement urbain.

Le passage à l'urbain est complexe car le nombre de situations de vie augmente considérablement. Le véhicule doit être capable de gérer tout type d'intersection, les routes à double sens de circulation, et les nouveaux usagers qui n'existaient pas sur autoroute comme les piétons ou les vélos. Cette thèse se focalise sur les fonctions de guidage latéral et de planification de trajectoire du véhicule. Les solutions proposées doivent être capables de fonctionner en temps réel dans les calculateurs des prototypes. L'apport de cette thèse doit donc être théorique et pratique.

Concernant le guidage latéral, l'objectif est de proposer une méthode calculant l'angle volant à appliquer à la direction pour suivre n'importe quel type de trajectoire, avec une vitesse longitudinale pouvant varier. Ainsi, le module de génération de trajectoire à développer doit être capable de calculer une trajectoire admissible dynamiquement par le véhicule. La solution proposée doit aussi tenir compte de l'infrastructure, du Code de la route et des objets dynamiques. Enfin, toutes les méthodes proposées doivent être implantées et validées sur prototype.

Contributions spécifiques et organisation de la thèse

Le mémoire de thèse concerne donc l'amélioration des systèmes d'aide à la conduite existants sur voies à chaussées séparées, et le développement de modules nécessaires à la conduite autonome en environnement urbain tels que le guidage latéral et la génération de trajectoire. Il est organisé comme suit en cinq chapitres.

Le **Chapitre 1** présente l'évolution des fonctions mécatroniques à bord des véhicules. Depuis l'allumage électronique, à la fonction d'aide au maintien dans la voie, le chapitre présente dans un premier temps les systèmes d'aide à la conduite présents aujourd'hui sur les véhicules. Puis les futures fonctions autonomes et leurs fonctionnements sont discutés. Le chapitre se termine par une présentation des prototypes du Groupe PSA sur lesquels ont été menées les expérimentations de cette thèse.

Le **Chapitre 2** étudie la modélisation et l'observation de la dynamique latérale du véhicule. Pour pouvoir traiter les problématiques de cette thèse que sont le guidage latéral et la planification de trajectoire, il convient de représenter correctement la dynamique latérale. Ainsi, le chapitre présente différents modèles et les compare. Dans cette comparaison, il apparaît que la vitesse longitudinale joue un rôle important sur la modélisation. L'observation de variables de la dynamique latérale est nécessaire lorsque ces variables ne sont pas mesurées ou lorsque les mesures sont trop bruitées. Une méthode d'estimation de la dynamique latérale par approche multimodèle, tenant compte de la vitesse longitudinale, est mise en place dans ce chapitre et validée sur les prototypes.

Basé sur la modélisation du chapitre 2, le **Chapitre 3** présente une méthode de régulation latérale du véhicule en mode autonome par multirégulateur. Étant donné que le calcul des régulateurs se fait dans le domaine fréquentiel, la conception de régulateur dans le domaine fréquentiel est présentée. Ensuite, plusieurs méthodes de régulation sont testées et comparées en simulation. La vitesse longitudinale impactant fortement la dynamique latérale, les méthodes « classiques » de type PID et même CRONE, ne permettent pas de garantir un respect du cahier des charges. La solution retenue, testée et validée sur prototype est un multi-PID utilisant une consigne devant le véhicule. Les scénarios ayant servi à la validation sont ceux du maintien dans la voie et du changement de voie.

Le **Chapitre 4** fait état de la prédiction d'intention et de trajectoire des autres véhicules de l'environnement. En effet, avant de définir la trajectoire de notre véhicule, il faut déterminer les actions que sont en train d'effectuer les autres usagers. D'autant plus qu'en environnement urbain, les comportements des objets sont plus complexes (piétons, vélos, scooters, etc.). Le chapitre présente donc dans un premier temps l'influence de l'environnement sur le calcul de la trajectoire. Puis, deux méthodes sont développées et comparées pour la détection de changement de voie. Une des méthodes est à base de règles tandis que l'autre utilise des réseaux de neurones. Enfin, une étude sur la prédiction de trajectoire par réseaux de neurones récurrents est présentée. Le but est de déterminer quelles variables sont nécessaires pour avoir une bonne prédiction de trajectoire avec ce type de méthode.

Le **Chapitre 5** concerne la génération de trajectoire pour un véhicule autonome en environnement urbain. Afin de développer une méthode capable d'être implantée sur prototype en temps réel, une attention particulière est apportée sur le temps de calcul. Un état de l'art complet sur les méthodes de planification de mouvement utilisées pour le véhicule autonome est

effectué dans un premier temps. Ensuite, ce chapitre présente une méthode pour la recherche de chemin avec l'intégration des contraintes géométriques du véhicule pour générer des chemins admissibles dans un environnement statique. Des points de passage sont aussi intégrés pour respecter le Code de la route. La méthode est testée en simulation et validée sur prototype. Enfin, la méthode pour l'environnement statique est enrichie pour que le véhicule se déplace dans un environnement dynamique tout en respectant ses propres contraintes dynamiques, qui dépendent de sa vitesse longitudinale. Le véhicule est alors capable de gérer en simulation, des situations de dépassement avec un véhicule arrivant en sens inverse sur une voie à double sens de circulation.

Chapitre 1

Évolution de la mécatronique à bord des véhicules

Sommaire

1.1 La mécatronique et le secteur automobile **15**
1.2 Les fonctions mécatroniques utilisant les données véhicule **16**
1.2.1 Les fonctions de confort 16
1.2.2 Les fonctions de sécurité active 18
1.3 Les systèmes d'aide à la conduite **22**
1.3.1 Les ADAS informatifs 22
1.3.2 Les ADAS actifs 24
1.4 Le véhicule autonome **29**
1.4.1 Avantages, niveaux d'autonomies et réglementations du véhicule autonome 30
1.4.2 Les capteurs 33
1.4.3 Architecture 38
1.5 Présentation des prototypes **39**
1.5.1 Architecture électrique et électronique pour la conduite autonome . . . 39
1.5.2 Les prototypes utilisés pendant la thèse 39
1.6 Problématique et objectifs **40**

Peu importe ce qu'on pourra vous dire, les mots et les idées peuvent changer le monde.

Robbin Williams

1.1 La mécatronique et le secteur automobile

La mécatronique est une démarche de conception qui combine plusieurs disciplines : la mécanique, l'électronique, l'automatique et l'informatique. En intégrant ces différents domaines, la mécatronique permet de concevoir et contrôler des systèmes automatiques, intelligents et complexes. Le terme mécatronique est apparu dans les années 70, mais n'est officialisé par son entrée dans le dictionnaire Larousse qu'en 2005.

La norme NF E01-010 de 2008 [2] décrit la mécatronique comme une « démarche visant l'intégration en synergie de la mécanique, l'électronique, l'automatique et l'informatique dans la conception et la fabrication d'un produit en vue d'augmenter et/ou d'optimiser sa fonctionnalité ». Le diagramme de Venn de la Figure 1.1 présente de manière schématique la pluridisciplinarité de la mécatronique et ses sous-domaines.

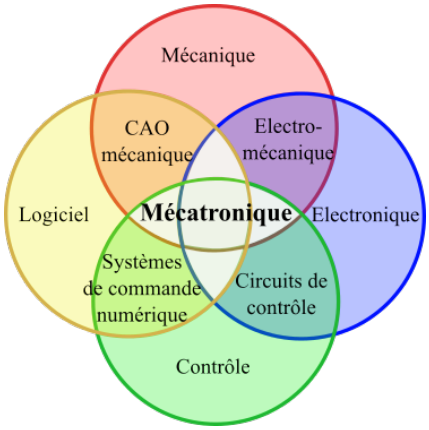


FIGURE 1.1 – Diagramme de Venn de la mécatronique

En conception, un système mécatronique se compose de trois sous-systèmes, tel que décrit en Figure 1.2. Le premier est la partie opérative du système, c'est-à-dire la partie mécanique avec les actionneurs : les éléments qui bougent et ceux qui les font bouger. Le deuxième est la partie commande, l'intelligence du système : ce qui contrôle les actionneurs. Le dernier est la partie IHM (Interface Homme Machine), le pilotage du système : ce qui dialogue avec l'opérateur. Ces trois sous-systèmes étant interdépendants, leur conception nécessite d'être simultanée.

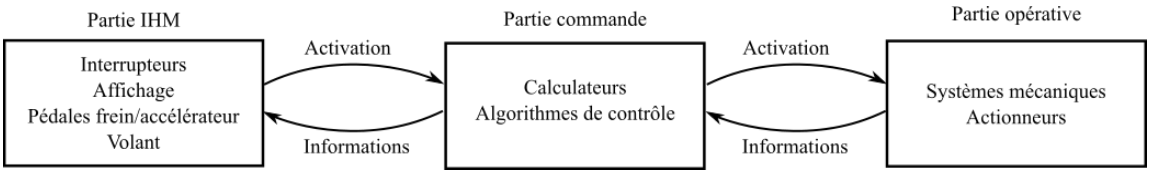


FIGURE 1.2 – Interactions entre les sous-systèmes mécatroniques

La mécatronique est présente dans tous les secteurs d'activité de l'industrie et dans les outils électroniques du quotidien. On peut par exemple citer l'autofocus des appareils photo ou les pilotes automatiques des avions, mais aussi les drones, les aspirateurs robots, les machines à laver intelligentes et les disques durs.

Dans l'automobile, la mécatronique fait son apparition à partir des années 1970. Avant cela, il y avait très peu de systèmes électroniques à bord des véhicules. Mis à part le système audio, tous les organes du véhicule étaient alors contrôlés mécaniquement. C'est l'apparition des microcontrôleurs au milieu des années 1970 qui permettra l'intégration et le développement de la mécatronique dans les véhicules.

L'allumage est le premier mécanisme à incorporer de la mécatronique pour améliorer les performances du véhicule. L'allumage est le mécanisme déclenchant l'amorce de la combustion du gaz inflammable présent dans la chambre de combustion de chaque cylindre. L'inflammation du mélange air-carburant se fait grâce à une étincelle électrique produite par une bougie d'allumage. De manière mécanique, le choix de la bougie se fait par un distributeur mécanique. L'ordre d'allumage est fixe et la vitesse d'allumage dépend du régime moteur et de l'appui du conducteur sur la pédale d'accélérateur. Le moment de l'allumage est important en termes de rendement, il faut choisir le bon moment pour profiter de la force maximum dégagée par l'augmentation de pression de la combustion. Les problèmes du système d'allumage mécanique sont son usure et ses imprécisions. De plus le système peut se dérégler.

L'allumage électronique permet de remplacer le système mécanique d'allumage des bougies par un système électronique qui calcule exactement le moment auquel provoquer une étincelle permettant un meilleur rendement énergétique. La pédale d'accélérateur est la partie IHM de la fonction. Avec les informations des capteurs de position du vilebrequin, de l'arbre à cames, de débit d'air, de position du papillon (qui gère l'arrivée de l'air), elle sert d'entrée au microcontrôleur qui est la partie intelligence du système. Le microcontrôleur calcule alors le point d'allumage et actionne la bougie d'allumage, c'est la partie opérative.

Par la suite, les fonctions mécatroniques n'ont fait qu'augmenter à bord des véhicules. Tirant profit de l'apparition de nouvelles informations sur l'état du véhicule grâce aux capteurs, les premières fonctions mécatroniques qui sont apparues visaient principalement à améliorer le confort des usagers et surtout à assurer la contrôlabilité du véhicule dans des situations d'urgence ou inhabituelles (chaussée glissante, évitement...). Un certain nombre d'exemples sont présentés dans la section 1.2. Par la suite, de nouveaux systèmes mécatroniques furent développés pour assister voire remplacer le conducteur dans certaines actions de sa tâche de conduite. Ce fut l'apparition des systèmes ADAS (« Advanced Driver Assistance System »), ou système d'aide à la conduite en français. L'évolution de ces systèmes, d'abord informatifs puis actifs, est présentée dans la section 1.3.

L'essor des systèmes mécatroniques connaît aujourd'hui une forte croissance dans le domaine automobile et atteindra dans les années à venir son apogée avec l'apparition des véhicules autonomes. Le paragraphe 1.4 détaille les enjeux ainsi que les caractéristiques du véhicule autonome.

1.2 Les fonctions mécatroniques utilisant les données véhicule

1.2.1 Les fonctions de confort

Les fonctions de confort permettent par exemple d'assurer l'isolation vibratoire de la caisse par rapport aux sollicitations de la route ou encore d'adapter la hauteur de la voiture pour faciliter son chargement ou le franchissement d'obstacle. Ces fonctions sont assurées par l'organe de suspension.

Correcteur d'assiette

Le contrôle de l'assiette de la voiture permet de régler automatiquement l'assiette de la voiture en cas de changement de chargement afin de garantir une garde au sol constante (voir Figure 1.3 (a)). Il permet également de modifier manuellement le niveau de la garde au sol du véhicule en cas de changement de situation de roulage, par exemple afin de faciliter le franchissement d'obstacles (voir Figure 1.3 (b)).

La correction automatique de l'assiette garantit un meilleur confort et une meilleure tenue de route du véhicule. Elle permet de recentrer le débattement des suspensions assurant ainsi la disponibilité du débattement complet sur sollicitation route (nid de poule, ralentisseur) et donc l'amélioration du confort lorsque le véhicule est chargé.

(a) (b)

FIGURE 1.3 – Publicités de la fonction du contrôle d'assiette automatique pour la Citroën CX (a) et réglage manuel de la garde au sol de la Citroën GSA (b) [Sources : (a) https://topworldauto.com/cars/citroen/citroen-cx/photos.html#gp/11, (b) http://www.gsaventure.com/gs-a-historique-et-documentations/les-gsa/les-publicit%C3%A9s-presse/]

Le correcteur d'assiette ayant besoin de la mesure du débattement, cela introduit l'utilisation de capteurs électroniques ou mécaniques situés sur chaque demi-train de suspension. Le calculateur (ECU, « Electronic Control Unit ») détermine à partir de ces valeurs la correction à effectuer et commande les actionneurs. En fonction des suspensions, les actionneurs peuvent être hydrauliques ou pneumatiques.

Suspensions pilotées

Il existe un certain nombre de suspension pilotées. L'objectif reste cependant toujours le même : améliorer la tenue de caisse tout en assurant un maximum de confort en isolant la caisse des perturbations de la route.

Une suspension est composée d'un ressort, qui peut être mécanique, pneumatique ou hydropneumatique, et d'un amortisseur. Une suspension pilotée s'appuie en plus sur des informations provenant de capteurs afin d'adapter l'effort dissipatif et/ou capacitif à appliquer. Souvent, ces capteurs sont des accéléromètres positionnés sur la caisse et sont couplés à des capteurs de débattement (position relative entre la caisse et les roues). Ils permettent alors de calculer l'effort à ajouter ou enlever pour tenir ou filtrer la caisse ou la roue.

Il existe un certain nombre de suspensions pilotées dans la littérature mais les plus communes sur les véhicules de série sont les suspensions à amortisseurs pilotés qui ne peuvent agir que sur l'effort dissipatif. L'ECU calcule la valeur d'effort que l'amortisseur est capable d'ajouter ou enlever à chaque instant en tenant compte de l'effort capacitif déjà fourni par le ressort.

1.2.2 Les fonctions de sécurité active

Les fonctions mécatroniques ont grandement contribué à la sécurisation des véhicules grâce à des fonctions dites de « sécurité active » . Le schéma de la Figure 1.4 montre que ces fonctions peuvent être corrélées à la réduction du nombre de décès sur les routes. Les fonctions de sécurité active sont des systèmes mécatroniques agissant directement et de manière automatique sur les actionneurs pour éviter les accidents et pour donner au conducteur un meilleur contrôle de son véhicule. Globalement, les fonctions de sécurité active existantes aujourd'hui agissent sur l'organe de freinage pour optimiser la motricité (système antipatinage), redonner de la dirigeabilité (ABS) ou gérer la stabilité (ESP, EDB).

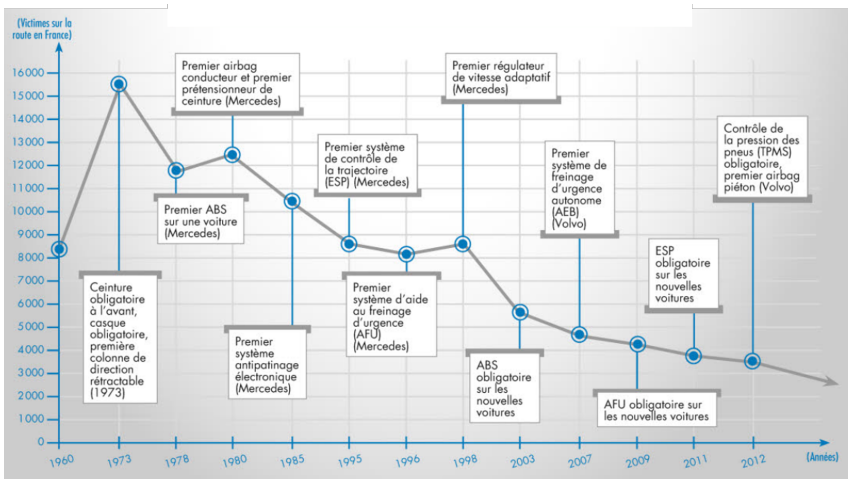


FIGURE 1.4 – Évolution du nombre de décès sur les routes françaises, associée à la disponibilité des systèmes d'aide à la conduite [Source : [148]]

Système antipatinage : ASR

Le système antipatinage, aussi appelé ASR de l'allemand « Antriebsschlupfregeling » ou TCS de l'anglais « Traction Control System » , est comme son nom l'indique un système permettant d'éviter que les roues du véhicule patinent sur route glissante. En effet, sur une chaussée glissante, les pneumatiques sont susceptibles de ne pas adhérer à la route. Les roues tournent donc dans le vide et le véhicule n'accélère pas, car les roues ne peuvent pas développer de force motrice.

Cette fonction permet donc un meilleur contrôle du véhicule et est particulièrement utile dans les situations où le véhicule doit accélérer sur une pente glissante ou roule sur une portion de route meuble causant la décélération du véhicule par perte d'adhérence.

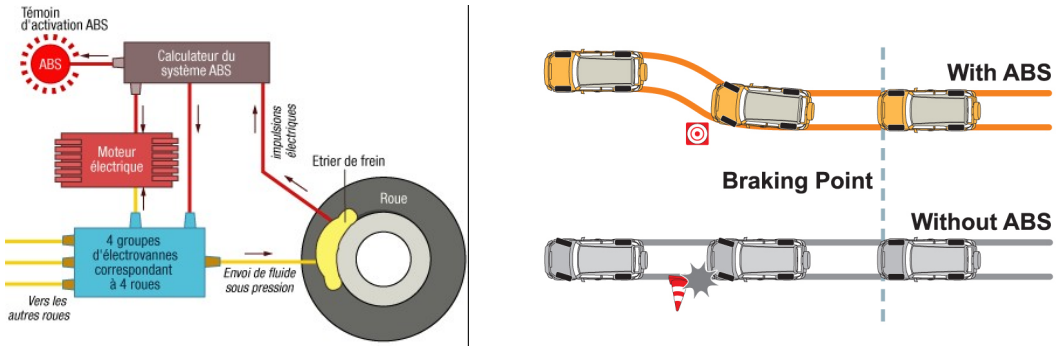
Le système antipatinage s'active lorsque la différence de vitesse de rotation entre les roues d'un même essieu ou entre les roues avant et arrière est trop importante. Lorsque la perte d'adhérence est détectée, les freins de la roue possédant le moins d'adhérence sont activés permettant d'avoir plus de couple sur la roue possédant le plus d'adhérence à la route et de retrouver de la motricité.

FIGURE 1.5 – Utilité du système antipatinage [Source : http://www.renault.pf/cars/newLaguna/overview2.html]

Antiblocage des roues : ABS

Le système antiblocage des roues, plus communément appelé ABS (de l'allemand « Antiblockiersystem »), est un système qui aide le conducteur à garder le contrôle de son véhicule lors de phases de freinage intenses. Lors de ces phases, les roues peuvent se bloquer et le risque est particulièrement élevé sur des chaussées glissantes (pluie, verglas, neige...). Ceci est dû à la sensibilité des efforts du pneumatique à l'adhérence de la route et à l'effort vertical. Si les roues sont bloquées, le véhicule devient difficile à contrôler, car il ne tourne plus. Le conducteur ne peut donc plus effectuer de manœuvre d'évitement. La législation européenne a rendu l'ABS obligatoire sur tous les nouveaux véhicules en 2004.

L'ABS intervient sur le freinage des roues pour redonner au conducteur le contrôle du véhicule. Pour cela, le système se compose d'un calculateur (ECU), de capteurs de rotations des roues et d'unités hydrauliques qui activent les freins par un envoi de fluide sous pression (voir Figure 1.6 (a)).



(a) (b)

FIGURE 1.6 – Schéma de fonctionnement de l'ABS (a) et son effet sur la conduite (b) [Sources : (a) https://www.testcoderoute.com/abs-systeme-anti-blocage-roues/, (b) http://www.mechanicalbooster.com/2017/08/anti-lock-braking-system.html]

L'ECU détecte le blocage des roues en analysant leur vitesse de rotation. Si celle-ci diminue de manière trop brusque, cela signifie que la roue est sur le point de se bloquer. En effet, lorsqu'une voiture freine pour s'arrêter, la vitesse de rotation des roues diminue progressivement. Dès que ce phénomène est détecté, l'ECU envoie un signal à l'unité hydraulique pour réduire la pression du circuit de freinage. La roue va alors se débloquer et recommencer à tourner. Le système effectue ces tâches de manière répétée. Les effets de l'ABS sur le regain de contrôlabilité du véhicule en

virage sont visibles sur la Figure 1.6 (b).

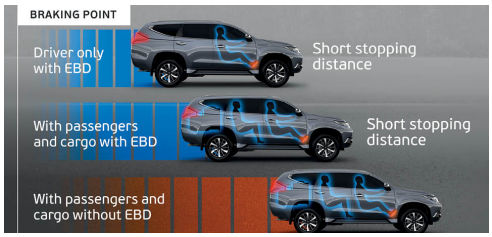
Répartiteur électronique de freinage : EBD

Le répartiteur électronique de freinage ou EBD « Electronic Brakeforce Distribution » a pour fonction de régler la répartition du freinage entre les roues de manière automatique en fonction du chargement du véhicule. Le répartiteur électronique de freinage est utile dans trois situations.

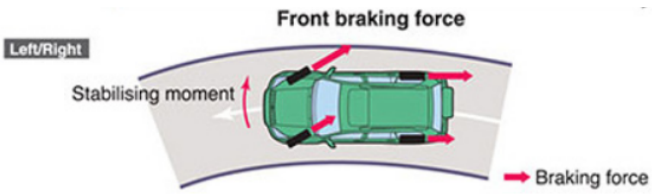
Le premier cas se produit si le chargement du véhicule change. En fonction du chargement du véhicule, la force de freinage peut s'avérer sous-optimale. En effet, si un véhicule est chargé à l'arrière et/ou dans le coffre, il faut apporter plus de puissance de freinage sur les roues arrière pour avoir une distance de freinage optimale. L'influence de l'EBD sur la distance de freinage pour une voiture chargée est visible sur la Figure 1.7 (a).

Le deuxième cas peut se produire lorsque la voiture décélère. Au moment du freinage, le poids du véhicule, et donc son centre de gravité, se déplacent vers l'avant. Le poids appliqué sur l'essieu arrière diminue pouvant devenir trop faible pour garantir une bonne adhérence aux pneumatiques arrière, entrainant leur dérapage voire leur blocage. L'EBD module alors la pression hydraulique des freins arrière pour limiter la force de freinage des roues en perte d'adhérence et augmente celle des freins avant pour compenser la perte de force de freinage.

Le dernier cas concerne l'influence de l'EBD lors d'un freinage en virage. Vu que le système permet de moduler la force de freinage de chaque pneumatique indépendamment, il peut augmenter la pression appliquée aux pneumatiques extérieurs au virage et diminuer celle des pneumatiques intérieurs (Figure 1.7 (b)). Cela permet d'éviter une perte de contrôle du véhicule dans le cas où les roues intérieures se bloqueraient.



(a)



(b)

FIGURE 1.7 – Influence du répartiteur de freinage en ligne droite sur un véhicule chargé (a) et illustration en virage (b) [Sources : (a) http://australiancar.reviews/_pdfs/Mitsubishi_Pajero_Mk4-NX_Brochure_201509.pdf P.11, (b) https://otomotif.kompas.com/read/2017/03/08/103200515/mengenal.fitur.keamanan.electronic.brake.distribution]

Le système EBD utilise le même ordinateur de bord que l'ABS et donc les mêmes capteurs et estimations de vitesse de roue, vitesse longitudinale, angle de dérive des roues, lacet et accélération. En fonction de ces informations, l'ECU est capable de calculer la répartition des masses et de déterminer si une roue est sur le point de déraper pour moduler la pression hydraulique des freins.

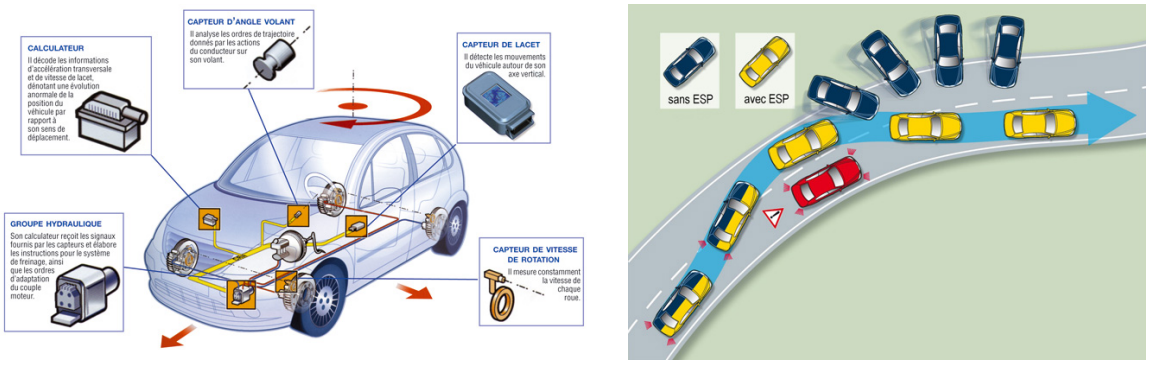
Correcteur électronique de trajectoire : ESP/C

Le correcteur électronique de trajectoire, plus communément appelé ESP (Electronic Stability Program) ou ESC (Electronic Stability Control), est une fonction de sécurité active d'anti dérapage aidant le conducteur à garder le contrôle de son véhicule par action sur le système de freinage. L'ESP est obligatoire sur les nouveaux véhicules de série depuis 2012 en Europe.

L'ESP agit sur les freins, comme l'ABS, ainsi que sur le couple moteur afin de corriger la trajectoire du véhicule. Il partage avec l'ABS et l'EBD le même ECU et les mêmes actionneurs hydrauliques agissant sur les freins des roues et les mêmes capteurs de vitesse de rotation des roues. L'ESP a cependant besoin de plus de capteurs pour réaliser sa fonction (voir Figure 1.8 (a)). Aux capteurs de l'ABS viennent s'ajouter un capteur d'angle volant qui mesure la position du volant pour estimer l'angle de braquage des roues, un capteur de vitesse de lacet, mesurant la vitesse de changement de cap, et un capteur d'accélération latérale.

Une situation de dérapage est détectée par l'ESP si la trajectoire actuelle du véhicule diffère de la trajectoire qu'aurait dû avoir le véhicule avec les entrées conducteurs actuelles : l'angle volant, la vitesse du véhicule et la position des pédales de frein et d'accélérateur. Pour cela, l'ESP utilise un modèle de la dynamique du véhicule pour estimer les valeurs de la vitesse de lacet et d'accélération latérale que devraient avoir le véhicule à partir des entrées conducteur. Ces valeurs sont comparées avec les valeurs réelles des capteurs embarqués sur le véhicule, à savoir un gyroscope et un accéléromètre.

L'ESP est particulièrement utile dans les cas de virages pris à grande vitesse ou d'évitement d'obstacle lors desquels le véhicule risque de partir en dérive, comme le montre la Figure 1.8 (b). Dans ces cas, la voiture peut soit sous-virer, soit survirer. Dans le cas d'un sous-virage, la dérive de l'essieu avant est supérieure à celle de l'essieu arrière, la voiture va tout droit au lieu de suivre la courbe du virage. L'ESP active alors le frein de la roue arrière intérieure au virage pour faire tourner la voiture. Dans le cas d'un survirage, la dérive de l'essieu arrière est supérieure à la dérive de l'essieu avant, la voiture tourne davantage que demandé et risque de partir en tête à queue. L'ESP active alors le frein de la roue avant extérieure pour redresser la voiture dans le virage.



(a) (b)

FIGURE 1.8 – Architecture mécatronique de la fonction ESP (a) et son effet sur la conduite lors d'un survirage (b) [Sources : (a) http://minute-auto.fr/guide/mecanique/l-esp-electronic-stability-program, (b) https://www.automobile-club.org/actualites/la-vie-de-l-aca/eurotest-enquete-des-automobile-clubs-sur-le-systeme-de-securite-esp]

1.3 Les systèmes d'aide à la conduite

Tout comme les fonctions de sécurité active ou les fonctions de confort, les systèmes d'aide à la conduite se multiplient avec l'émergence des nouvelles technologies et la baisse des prix des capteurs et ordinateurs embarqués. Les systèmes d'aide à la conduite n'agissent pas uniquement sur le véhicule en cas d'urgence, ils permettent d'apporter un confort à la conduite du véhicule en déléguant certaines tâches au véhicule.

1.3.1 Les ADAS informatifs

Système anticollision

Le système anticollision des véhicules est une fonction avertissant le conducteur en cas de danger, et plus particulièrement de collision. Pour cela, le système anticollision utilise un capteur frontal, usuellement un radar, capable de détecter un objet devant le véhicule pour calculer l'interdistance qui les sépare et la vitesse de l'objet. En fonction de ces informations, si une situation à risque est détectée, le système alerte le conducteur par un signal ou une combinaison de signaux : visuel sur le tableau de bord, sonore ou haptique par le volant ou la ceinture.

Alerte au franchissement de ligne : LDW

Le système d'alerte au franchissement de ligne, LDW « Lane Danger Warning » ou AFIL « Alerte au Franchissement Involontaire de Ligne » en français, avertit le conducteur en cas de franchissement involontaire de ligne. Le dispositif utilise pour cela un capteur infrarouge ou une caméra braquée vers le sol à l'avant de la voiture. Le capteur repère les lignes blanches à gauche et à droite du véhicule et calcule la position relative du véhicule par rapport à ces lignes. Dès que la voiture « mord » la ligne, une alerte se déclenche (voir Figure 1.9). Il peut s'agir soit d'une alerte sonore soit d'une alerte haptique avec la vibration du siège conducteur. Le caractère involontaire du franchissement de ligne est défini par l'activation ou non du clignotant. L'alerte se déclenche donc uniquement si le clignotant n'est pas allumé.

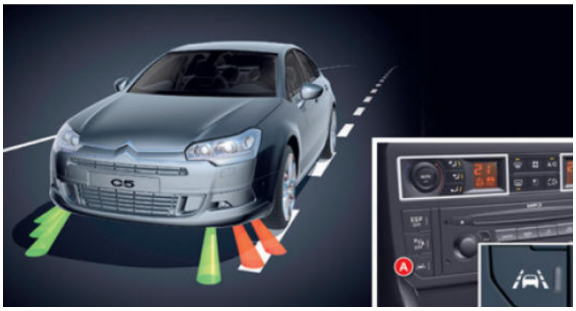


FIGURE 1.9 – Fonction alerte au franchissement de ligne [Source : http://www.c5manuel.com/alerte_de_franchissement_involontaire_de_ligne-355.html]

Aide au stationnement : ParkAssist

L'aide au stationnement est une fonction d'assistance au conducteur pour l'aider à se garer en lui indiquant, par des impulsions sonores et/ou par affichage sur un écran, la distance qui sépare le véhicule d'un objet.

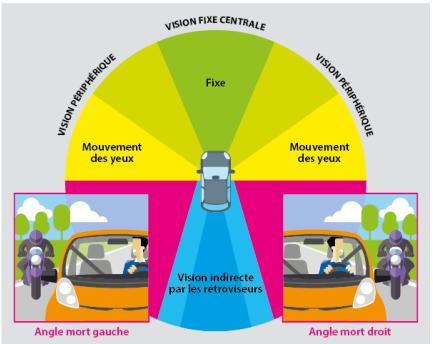
Pour ce faire, habituellement, des capteurs ultrasons sont installés dans les pare-chocs avant et arrière. Les capteurs évaluent en permanence la distance entre le pare-chocs et les objets

FIGURE 1.10 – Fonction d'aide au stationnement [Source : https://www.opel-accessories.com/opel_aoc/fr-FR/Corsa_D_-_5_portes/S%C3%A9curit%C3%A9_&_protection/Aide_au_stationnement/13365682]

lorsque la vitesse de la voiture est très faible et/ou lorsque la marche arrière est enclenchée (illustration Figure 1.10). Si l'information sur cette distance est retransmise au conducteur de manière sonore, plus la distance est faible et plus la fréquence des impulsions sonores est grande. Si le véhicule dispose d'une caméra de recul en plus des radars/ultrasons, l'image de la caméra est retransmise au conducteur sur un écran du tableau de bord. Sur cette image, des lignes correspondant au rayon de braquage de la voiture sont superposées, elles changent donc en fonction de l'angle volant mesuré par le système de direction et sont recalculées et transmises au conducteur en temps réel pour lui permettre de faire sa manœuvre plus confortablement.

Système de surveillance des angles morts : BSD

Les angles morts sont les zones non visibles par le conducteur. Le conducteur dispose de son champ de vision naturel vers l'avant et peut voir une partie de la zone arrière du véhicule grâce aux rétroviseurs. Il existe cependant deux zones proches du véhicule, une à droite et une à gauche, qui ne sont visibles par le conducteur que s'il tourne la tête, Figure 1.11 (a). La surveillance des angles morts, ou BDS « Blind Spot Detection », est indispensable dans plusieurs situations : sortie de rond-point, changement de voie, insertion sur voie rapide ou encore franchissement de voie réservée. Ces situations génèrent de nombreux accidents lorsque les angles morts ne sont pas vérifiés et qu'un véhicule, un vélo ou un piéton s'y trouve.



(a)

(b)

FIGURE 1.11 – Zones de visibilité et angles morts pour le conducteur (a) et indicateur lumineux du système de surveillance d'angles morts (b) [Sources : (a) https://www.ornikar.com/code/cours/mecanique-vehicule/technologie-assistance/angle-mort, (b) https://www.citroen.be/fr/vehicules-neufs/citroen/new-citroen-berlingo/configurateur/couleurs-et-equipements/options.html

Pour aider le conducteur dans sa tâche de surveillance des angles morts et éviter ces accidents, le système de surveillance des angles morts avertit le conducteur de la présence d'objet dans la zone invisible droite ou gauche par un voyant lumineux dans le rétroviseur respectif, voir Figure 1.11 (b). La détection des objets se fait à l'aide de radars ou d'ultrasons installés sous la carrosserie ou dans le bouclier.

Driver monitoring assist : DMA

À l'image de la fonction d'aide au stationnement, le driver monitoring n'agit pas directement sur le véhicule, mais donne une information au conducteur sur l'état de sa conduite. Dans le cas du driver monitoring, cette information concerne directement le conducteur et son activité au volant du véhicule. À l'aide d'une caméra placée au niveau du volant, la fonction peut avertir le conducteur s'il est en état de fatigue ou s'il ne regarde pas la route. Pour cela, un système d'analyse d'image (illustré en Figure 1.12) intégré à l'ECU analyse le comportement du conducteur et plus particulièrement la fréquence du battement de ses paupières, l'inclinaison de sa tête et la position de ses pupilles. Si un état de fatigue ou d'inattention est détecté, la fonction alerte le conducteur pour qu'il se concentre sur sa conduite. En cas d'extrême urgence, la fonction peut demander un arrêt complet du véhicule.

L'attention du conducteur peut aussi être analysée par son action sur le volant. L'utilisation d'un capteur volant résistif permet de détecter si un couple est appliqué au volant par le conducteur. Un capteur capacitif ou résistif dans le volant peut aussi être utilisé pour savoir si le conducteur tient le volant.

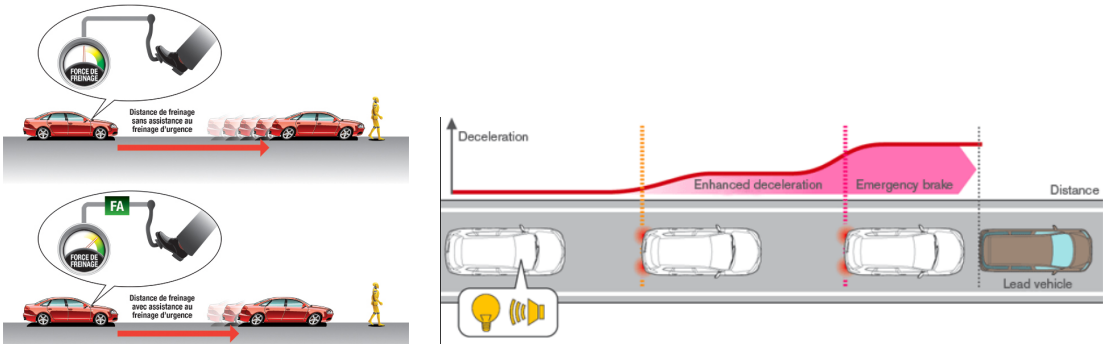
FIGURE 1.12 – Analyse de l'activité conducteur par la caméra de la fonction driver monitoring [Source : https://www.valeo.com/fr/driver-monitoring/]

1.3.2 Les ADAS actifs

Aide au freinage d'urgence et freinage automatique d'urgence : AFU et AEB

En cas de situation de freinage d'urgence, un conducteur a tendance à ne pas appuyer assez fort et assez rapidement sur la pédale de frein ce qui allonge la distance d'arrêt du véhicule. L'aide au freinage d'urgence (AFU) permet de réduire cette distance et donc d'éviter les collisions. Pour cela, l'AFU utilise les capteurs de position des pédales d'accélération et de frein pour détecter une situation de freinage d'urgence et appliquer une pression maximale au système de freinage, voir Figure 1.13 (a). Soit uniquement la position de la pédale de frein est utilisée pour repérer un freinage et c'est alors sa vitesse d'enfoncement qui indiquera si le déclenchement de l'AFU est nécessaire. Soit la position de la pédale d'accélération est utilisée en complément, et la transition entre les pédales d'accélération et de frein aide à compléter la détection d'un freinage

d'urgence. Chez certains constructeurs, l'activation de l'AFU enclenche par défaut les warnings du véhicule pour prévenir les véhicules suiveurs d'une situation à risque.



(a) (b)

FIGURE 1.13 – Fonctions d'aide au freinage d'urgence (AFU) (a) et de freinage automatique d'urgence (AEB) (b) [Sources : (a) https://www.carkeys.co.uk/guides/what-is-brake-assist, (b) https://www.nissan-global.com/EN/TECHNOLOGY/OVERVIEW/emergency_brake.html]

Le freinage automatique d'urgence ou AEB (Automatic Emergency Braking) en anglais est aussi une fonction d'aide au freinage d'urgence. Contrairement à l'AFU, elle peut s'activer automatiquement, sans l'aide du conducteur, et active les freins du véhicule si une collision imminente est détectée. Pour cela le véhicule est équipé d'un capteur frontal longue portée : caméra, laser, radar, ou combinaison de plusieurs de ces capteurs pour avoir une redondance d'information sur l'environnement autour du véhicule. Le capteur frontal détecte les véhicules de devant et est capable de calculer leur vitesse et leur distance à l'ego véhicule, *i.e* le véhicule qui va freiner et déclencher l'AEB. En fonction de la vitesse de l'ego véhicule et de la distance à l'objet, le risque d'une collision est évalué. Si ce risque est grand, le véhicule alerte le conducteur. En cas d'absence de réponse du conducteur ou d'une réponse trop lente, l'AEB enclenche automatiquement les freins, voir Figure 1.13 (b). L'AEB peut aussi anticiper le choc si la collision est inévitable en effectuant un freinage d'urgence pour réduire au maximum la vitesse et donc la puissance de l'impact et en ajustant les ceintures de sécurité. L'AEB est découpé en quatre catégories pour son évaluation EuroNCAP : haute vitesse, ville, piéton et cycliste.

L'EuroNCAP (European New Car Assessment Program) est un organisme d'évaluation consumériste des véhicules notant le niveau de sécurité des véhicules à l'aide de crash tests. Cette évaluation est non réglementaire, mais réalisée par un organisme indépendant des constructeurs, réunissant des organismes gouvernementaux et des associations de consommateurs européens. Les véhicules testés reçoivent une notation entre 1 et 5 étoiles. Les tests effectués par EuroNCAP sont mis à jour régulièrement et intègrent donc des tests sur les systèmes de sécurité active comme l'ESC et les systèmes d'aide à la conduite comme l'AEB. La volonté des constructeurs à avoir le maximum d'étoiles a entrainé, notamment grâce à l'évaluation de l'AEB, la démocratisation des radars sur les faces avant des voitures. Cela a permis de développer d'autres fonctions d'aides à la conduite telles que le régulateur de vitesse adaptatif ou le système de positionnement dans la voie, détaillées dans les paragraphes suivants.

Limiteur et régulateur de vitesse : LVV et RVV

Les premières fonctions de délégation de conduite ont pour objectif de remplacer le conducteur dans son action avec les pédales d'accélération et de frein. La toute première est le limiteur

de vitesse avec lequel le conducteur n'a plus besoin d'adapter la pression sur la pédale d'accélération pour respecter une vitesse limite.

En effet, le limiteur de vitesse, ou LVV pour « Limiteur de Vitesse Variable » est une fonction d'aide à la conduite qui limite la vitesse maximale du véhicule. C'est le conducteur qui active la fonction et choisit cette vitesse limite à ne pas dépasser. La vitesse choisie est comparée à la vitesse réelle du véhicule calculée à partir des capteurs de vitesse de rotation des roues ou à partir d'un capteur de vitesse situé au niveau de la sortie de la boîte de vitesse. En fonction de ce différentiel de vitesse et de l'appui du conducteur sur la pédale d'accélérateur, le limiteur de vitesse gère la puissance du moteur par l'intermédiaire du papillon des gaz ou de la pompe d'injection.

Ensuite, une évolution de délégation de conduite est apparue avec le régulateur de vitesse, RVV « Régulateur de Vitesse Variable » , qui permet au conducteur de ne plus avoir besoin de toucher la pédale d'accélérateur. Ce dernier fixe une vitesse cible que devra garder le véhicule. Le régulateur gère alors le régime de rotation du moteur pour maintenir cette vitesse. Il utilise pour cela les mêmes composants ECU, capteurs et actionneurs, que le limiteur de vitesse. Si le conducteur active la fonction alors que le véhicule n'a pas atteint la vitesse cible, une accélération est enclenchée automatiquement. Sur la plupart des véhicules, un système de sécurité désactive le régulateur de vitesse lorsque le conducteur appuie sur la pédale de frein. L'avantage du régulateur se trouve surtout sur les longs trajets où il apporte plus de confort au conducteur.

Le principal inconvénient de cette fonction est qu'elle n'adapte pas la consigne de vitesse aux objets et le conducteur doit encore freiner « au pied » lorsqu'il arrive sur un véhicule plus lent.

Régulateur de vitesse adaptatif : ACC

Grâce à la démocratisation de capteurs frontaux via les évaluations EuroNCAP, la fonction de régulation de vitesse adaptative a pu être aujourd'hui proposée permettant une délégation complète de l'appui sur les pédales de frein et d'accélérateur sur autoroute, en tenant compte également les objets autour du véhicule ego.

Le régulateur de vitesse adaptatif ou ACC pour « Adaptive Cruise Control », est une fonction d'aide à la conduite de niveau 1 d'autonomie (voir section 1.4.1), qui régule automatiquement la vitesse du véhicule en fonction du trafic et en particulier de la voiture le précédent. Pour cela, l'ACC utilise un capteur frontal qui scanne l'environnement devant le véhicule. Ce capteur peut être un laser, un radar et/ou une caméra. Les informations reçues par ce capteur sont traitées par l'ECU pour identifier si un véhicule est devant. Si tel est le cas, sa vitesse ainsi que la distance qui sépare les deux véhicules sont calculées et le véhicule de devant est utilisé comme véhicule cible. C'est-à-dire que l'ACC va réguler sa vitesse en fonction de la dynamique de ce véhicule. Si aucun véhicule n'est détecté devant, la vitesse de référence de l'ACC est la vitesse maximale définie par le conducteur et la fonction ACC agit comme le régulateur de vitesse.

Pour réguler sa vitesse par rapport au véhicule devant, celle-ci est asservie sur la distance inter-véhicule. Cette distance est plus couramment convertie en temps inter-véhicule, aussi appelé TIV. D'après l'article R412-12 du Code de la route, le TIV, afin de conserver de bonnes distances de sécurité, doit être supérieur à 2 secondes : « Lorsque deux véhicules se suivent, le conducteur du second doit maintenir une distance de sécurité suffisante pour pouvoir éviter une collision en cas de ralentissement brusque ou d'arrêt subit du véhicule qui le précède. Cette

distance est d'autant plus grande que la vitesse est plus élevée. Elle correspond à la distance parcourue par le véhicule pendant un délai d'au moins deux secondes. »

Pour garantir un certain confort aux passagers du véhicule, l'ACC est limité en accélération longitudinale et ne peut donc pas intervenir en cas de freinage d'urgence du véhicule de devant. L'AEB peut prendre le relais, mais c'est au conducteur de rester attentif à son environnement. De plus comme le conducteur reste maître de son véhicule, les régulateurs de vitesse adaptatifs peuvent être réglables, comme illustré en Figure 1.14, avec plusieurs modes de TIV : faible, moyen, grand.

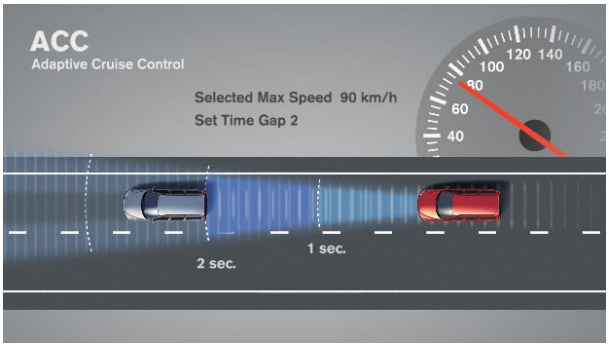
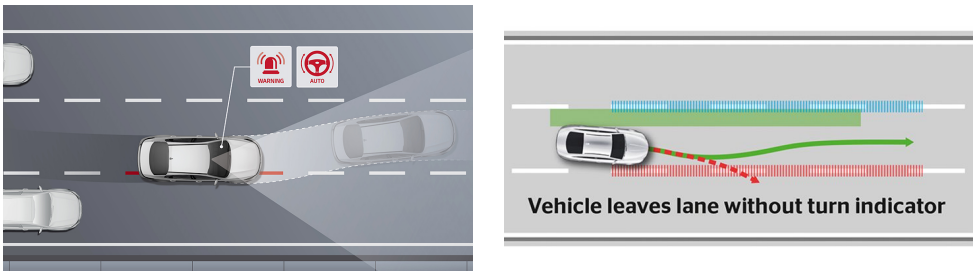


FIGURE 1.14 – Fonction ACC réglable [Source : http://crankydriver.com/blog/images/Reports/ADAS/acc_gap.jpg

Systèmes d'aide au maintien et d'aide au positionnement dans la voie : LKA et LPA

Le système d'aide au maintien dans la voie aussi appelé « Lane Keeping Assist » (LKA) est le prolongement de la fonction d'alerte au franchissement de ligne. Là où l'alerte au franchissement de ligne prévient le conducteur d'une sortie de route, l'aide au maintien dans la voie agit sur le système de direction par lecture des marquages au sol grâce à une caméra pour aider le conducteur à se remettre dans la voie. Ici, lorsqu'une sortie de voie est détectée, l'ECU calcule le couple nécessaire à envoyer à la Direction Assistée Électrique (DAE) en fonction de sa position dans la voie pour repositionner le véhicule au centre. Cette fonction n'agit que si le véhicule s'approche « trop » de la ligne. Une fois le véhicule recentré, elle s'éteint jusqu'à la prochaine détection de franchissement de ligne.



(a) (b)

FIGURE 1.15 – Fonctions d'aide au maintien dans la voie (a) et d'aide au positionnement dans la voie (b) [Sources : (a) https://preprod-eu.kia.com/eu/about-kia/experience-kia/technology/drive-wise/, (b) https://www.continental-automotive.com/en-gl/Passenger-Cars/Chassis-Safety/Software-Functions/Active-Safety/Lane-Assist]

La fonction d'aide au positionnement dans la voie ou « Lane Positioning Assist » (LPA) va encore plus loin en agissant constamment sur le couple volant pour maintenir le véhicule au centre de la voie. Chez la plupart des constructeurs, le LPA ne peut être actif que si le conducteur a les mains sur le volant. La Figure 1.15 illustre ces deux systèmes.

Fonction de stationnement automatique

La fonction de stationnement automatique permet au conducteur de garer sa voiture sans qu'il ait besoin de toucher à son volant. La fonction permet de réaliser les manœuvres en marche arrière : créneau, rangement épis et rangement bataille.

Les capteurs nécessaires à la réalisation de cette fonction sont des ultrasons et/ou des caméras carrosseries. La vitesse étant limitée en parking, il n'est pas nécessaire de « voir » très loin. Ces capteurs sont souvent appelés capteurs « cocoon », car ils forment une ceinture autour de la voiture et permettent de voir l'environnement à 360°. En utilisant les informations de ces capteurs, l'ordinateur de bord est capable de calculer l'angle volant nécessaire à appliquer à la DAE pour réaliser la manœuvre.

FIGURE 1.16 – Fonction de stationnement automatique [Source : http://volvo.custhelp.com/app/answers/detail/a_id/9559/~/park-assist-pilot]

Le stationnement automatisé se déroule en plusieurs étapes. D'abord, lorsque le conducteur active la fonction, les capteurs vont analyser l'espace et déterminer si l'espace est suffisant pour se garer. Ensuite, la fonction demande au conducteur d'activer la marche arrière. Une fois la marche arrière enclenchée et dès que le conducteur lâche la pédale de frein, l'ordinateur de bord

prend le contrôle du volant en calculant la trajectoire à suivre et en envoyant les instructions à la DAE. Le conducteur reste au contrôle de son véhicule puisqu'il doit engager la marche avant ou arrière lorsqu'un changement de sens est demandé par la fonction. De plus, le conducteur peut mettre fin à la manœuvre en freinant ou reprenant le contrôle du volant en cas d'objet non détecté par les capteurs (trottoir, piéton...).

Ces fonctions sont les premières briques vers le véhicule autonome offrant un niveau 1 d'autonomie, détaillé à la section 1.4.

1.4 Le véhicule autonome

Les premiers véhicules autonomes ont commencé à être expérimentés à la fin des années 1970 où plusieurs projets majeurs ont été mis en place. En 1977, le laboratoire Tsukuba Mechanical Engineering au Japon a construit un prototype capable de se diriger seul dans une voie à l'aide d'une caméra détectant les lignes blanches [180]. Sur le projet Eureka PROMETHEUS Project [23], une initiative de recherche européenne commencée en 1985 et terminée 10 ans plus tard, les prototypes étaient capables à terme de se déplacer sur voies à chaussées séparées avec un trafic dense. Un autre projet important qui a contribué à l'essor du véhicule autonome à la même période est le projet ALV (Automous Land Vehicle) [112], financé par la DARPA (Defense Advanced Research Projects Agency), agence du département de la Défense des États-Unis. Comparé aux autres projets cités, l'objectif de celui-ci est la conception de véhicule autonome dans des environnements non structurés.

Des concours autour du véhicule autonome ont émergé dans les années 2000, avec notamment en 2004 le DARPA Grand Challenge [176]. Les véhicules devaient parcourir en mode autonome, et donc sans recevoir d'instruction d'un conducteur, plus de 200 km en moins de 10 heures dans le désert des Mojaves. Même si aucun des participants n'a réussi à franchir la ligne d'arrivée, la première édition de ce concours a connu un réel succès et a montré un grand engouement des secteurs industriels et académiques sur le sujet du véhicule autonome. Le concours a donc été reconduit plusieurs années et a été transformé en 2007 en DARPA Urban Challenge [185] où l'objectif était de concevoir un véhicule capable de se déplacer en mode autonome dans un environnement urbain, et donc capable de gérer les intersections. Pour plus de détails sur toutes les expérimentations et projets menés autour du véhicule autonome, les thèses [128] et [193] présentent un état de l'art complet et récent sur le sujet.

L'enthousiasme autour du véhicule autonome a réellement débuté avec l'annonce de Google et de sa Google car, qui en 2010 était déjà capable de se déplacer sans conducteur en condition urbaine. Depuis, tous les grands constructeurs automobiles ont engagé des programmes de développement sur le véhicule autonome avec leurs propres prototypes. De nombreuses coopérations entre les géants d'internet, les acteurs de mobilités et les constructeurs sont menées pour développer au plus vite ces voitures autonomes et connectées qui constituent un véritable enjeu stratégique et économique. Parmi ces coopérations, la plus connue est certainement celle entre UBER et Volvo, mais il existe également les coopérations entre Apple et Volkswagen ou Waymo (anciennement Google Car) et Fiat Chrysler.

1.4.1 Avantages, niveaux d'autonomies et réglementations du véhicule autonome

Les avantages

L'atout principal des véhicules automatisés est de libérer du temps au conducteur et de lui permettre de réaliser d'autres tâches que celle de conduire dans des situations de conduite rébarbatives. Parmi les bénéfices identifiés, les éléments notables sont :

- la simplification de la conduite
- le soulagement physique et allègement mental
- la valorisation du temps qui mobilise peu le cognitif
- la mise à disposition de ce qui se fait de mieux en termes de technologie.

Les niveaux d'autonomie

Le terme véhicule autonome désigne aussi bien un véhicule pouvant se garer seul qu'un véhicule pouvant circuler sans intervention humaine sur autoroute ou dans un environnement spécifique. Afin de catégoriser ces véhicules robotisés et automatisés en fonction de leurs capacités, des organismes tels que SAE International (Society of Automotive Engineers) ou NHTSA (National Highway Traffic Safety Administration) ont mis en place une échelle d'autonomie des véhicules. Ainsi, les véhicules sont répartis en 6 niveaux d'autonomie comme l'illustre la Figure 1.17.

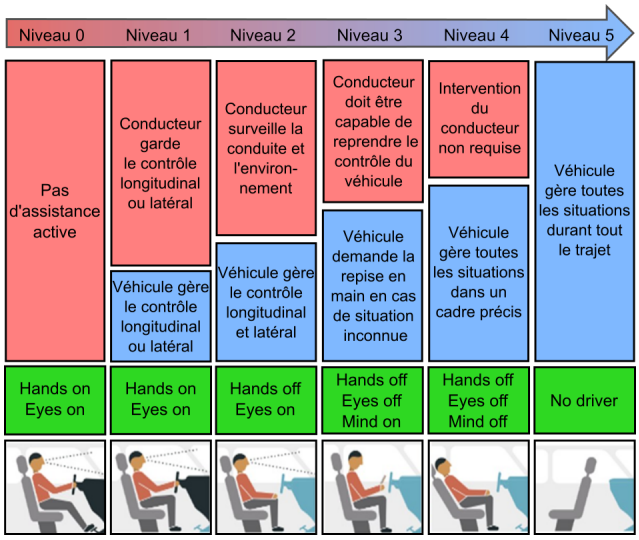


FIGURE 1.17 – Les niveaux d'autonomie de la norme SAE

- Niveau 0 : la voiture est complètement manuelle. Tous les organes de conduite (moteur, frein, volant) sont contrôlés par le conducteur.
- Niveau 1 : le véhicule gère de manière automatique soit le contrôle longitudinal, soit le contrôle latéral. L'exemple le plus parlant de ce niveau d'autonomie est l'ACC.
- Niveau 2 : le véhicule gère automatiquement les modes longitudinal et latéral sans intervention du conducteur. Le conducteur reste tout de même maître de son véhicule et doit superviser la conduite de manière à intervenir en cas de problème. La fonction d'aide à la conduite la plus répandue actuellement est la fonction de stationnement automatique.

- Niveau 3 : dans ce mode d'autonomie, le conducteur n'est plus obligé de superviser la conduite, mais doit être capable de reprendre la main si le véhicule le lui demande en cas d'événement non géré par le véhicule en mode autonome.
- Niveau 4 : l'intervention du conducteur n'est plus du tout requise dans un cadre précis, il délègue totalement la conduite au véhicule.
- Niveau 5 : le véhicule gère toutes les situations de vie, le volant et les pédales ne sont plus nécessaires dans l'habitacle.

Les réglementations

Actuellement, l'organisme EuroNCAP ne prend pas en compte la conduite autonome dans la notation des véhicules. Des discussions sont en cours à l'EuroNCAP pour ajouter les fonctions de conduite autonome, notamment à l'aide de bonus.

La principale réglementation qui pose problème pour le déploiement de véhicules autonomes de niveau 3 et plus dans la rue est la convention de Vienne. En effet, les fonctions de conduite autonome de niveau 3 ne sont pas compatibles avec plusieurs textes réglementaires.

Les systèmes ADAS et le futur véhicule autonome nécessitent des évolutions dans la réglementation mondiale et nationale notamment sur la maîtrise par le conducteur de son véhicule. La Convention de Vienne est un traité international qui permet l'harmonisation des règles de circulation routière et des codes de la route entre 71 pays du monde [182]. Cette convention était de plus en plus en décalage avec les évolutions technologiques des véhicules. Par conséquent, un amendement a été adopté suite à l'appui des constructeurs automobiles. Avant cet amendement, la Convention de Vienne exigeait en particulier le contrôle du véhicule par son conducteur en toutes circonstances. L'évolution réglementaire, proposée en 2014, autorise la conduite automatisée transférant des tâches de conduite au véhicule, à condition qu'elle soit conforme au règlement de l'ONU sur les véhicules ou que le conducteur puisse les remplacer ou les désactiver [183].

Cet amendement est entré en vigueur en avril 2016 pour être transposé dans les différents codes de la route des pays signataires de la Convention de Vienne. Mais cela n'est pas suffisant pour le véhicule autonome de niveau 3 et plus. Un nouveau groupe de travail a été créé sous pilotage de la France en vue de compléter à nouveau cette convention. Son objectif sera de permettre la conduite en mode autonome (commercialisation du véhicule autonome à l'horizon 2020-2030).

Ce groupe de travail est tombé d'accord en mars 2017 sur les évolutions suivantes, en vue de proposer un nouvel amendement dont voici quelques extraits [184] :

— « Lorsque le véhicule est conduit par des systèmes embarqués qui ne nécessitent pas d'activité de conduite de la part du conducteur, celui-ci peut se livrer à des activités autres que la conduite à condition que :

— Principe 1 : ces activités n'empêchent pas le conducteur de prendre la direction du véhicule si les systèmes embarqués l'exigent, et

— Principe 2 : ces activités soient compatibles avec les prescriptions relatives à l'emploi et aux fonctions des systèmes embarqués. »

En septembre 2017, ces principes ont été confirmés, et il n'y aura pas besoin d'un nouvel amendement. À partir de maintenant, les états pourront transposer ce texte dans un code de la route qui autoriserait la conduite en niveaux 3 et 4.

Les problématiques liées à la délégation de conduite

L'objectif principal de la fonction autonome est donc de permettre au conducteur de ne plus avoir à superviser en permanence l'environnement, le système ou les tâches de conduite et de pouvoir réaliser certaines autres activités (profiter du système infotainment du véhicule pour téléphoner, regarder des vidéos, etc.). Par contre, dans le cas d'un niveau d'autonomie de niveau 3, le conducteur doit rester en mesure de reprendre le contrôle du véhicule si le système le lui demande.

Cela a plusieurs conséquences sur les exigences de sûreté de fonctionnement à porter par ces fonctions de délégation de conduite. En effet, toutes les fonctions ADAS développées jusqu'ici reposent sur la supervision des tâches de conduite et de l'environnement par le conducteur, qui doit prendre le relais en cas de défaillance ou de limite de performance du système. Cette supervision du conducteur permet, par exemple, de limiter les exigences de performances (sûreté de fonctionnement notamment) sur la détection d'objets : on se repose sur le conducteur pour voir et éviter un objet non identifié par le système. Ainsi, dans le développement des fonctions d'aide à la conduite de niveau 2, la conception avait pour objectif de réduire au maximum les phénomènes intempestifs venant perturber le bon fonctionnement de la fonction, quitte à trop filtrer les obstacles, car le conducteur supervise la conduite. Alors que pour la conception des fonctions de niveau 3, il faut éviter les intempestifs sans rater d'objets.

Afin de pallier cette absence de supervision du conducteur, les décisions et actions du véhicule autonome doivent être sûres, ce qui impose une connaissance fiable de l'environnement du véhicule. Les principaux impacts sont les suivants :

- Comme le conducteur ne supervise plus en permanence la conduite, la sécurité du véhicule ne peut plus lui incomber (comme c'est le cas sur les fonctions ADAS actuelles) : les défaillances du système ne doivent pas provoquer d'événement redouté, et en cas de doute, le système doit réaliser une action sûre, par exemple s'arrêter dans la voie en cas de défaillance majeure d'un capteur ou d'un actionneur.
- Comme le conducteur n'est pas attentif au moment de la demande de reprise en main, il ne sera pas en capacité de reprendre la main instantanément (pour gérer une situation d'urgence par exemple) : la fonction doit donc lui laisser un temps plus long que les fonctions de niveau 2 pour revenir en conduite manuelle. De nouvelles exigences sont donc nécessaires.

D'autre part, la complexité des fonctions ADAS (capteurs et algorithmes de décision) impose de traiter à la fois :

- Le dysfonctionnel : application de la norme ISO 26262 pour couvrir les défaillances (les fautes aléatoires matérielles et les fautes systématiques)
- Le fonctionnel sûr (SOTIF : Safety Of The Intended Functionality) : s'assurer que le véhicule, en état de fonctionnement nominal, ne fait pas courir à ses utilisateurs des risques en usage normal. En particulier, le SOTIF doit adresser une spécificité des capteurs de perception qui, même en fonctionnement nominal, peuvent générer des informations erronées.

Ainsi, pour la perception plusieurs types de technologies sont utilisées, chacune ayant ses propres limites de performances.

1.4.2 Les capteurs

La majorité des prototypes autonomes développés aujourd'hui présentent plusieurs ceintures de capteurs de différentes technologies permettant de scanner l'environnement autour du véhicule. La capacité du véhicule à percevoir son environnement et à se situer dans celui-ci est fortement dépendante des technologies de capteurs qui le composent. Ces capteurs peuvent être des caméras, des ultrasons, des radars ou des lidars. Les prototypes, mais aussi les futurs véhicules commercialisés, utilisent les capteurs déjà présents en série pour les fonctions d'aide à la conduite décrites précédemment et récapitulés dans le Tableau 1.1.

	Extéroceptifs				Proprioceptifs		
	Caméra frontale	Ultrasons	Radar	Caméra carrosserie	Accélération latérale	Vitesse	Angle volant
ABS et ESP					x	x	x
AEB	(x)		x			x	
LKA et LPA	x				(x)	x	x
ACC	x		x			x	
Stationnement automatique		x		x		x	x

TABLE 1.1 – Capteurs nécessaires pour chaque fonction mécatronique. Les parenthèses signifient que l'utilisation du capteur n'est pas systématique et dépend du choix du constructeur

Les capteurs permettent de représenter l'environnement, c'est le système de perception. Pour le véhicule autonome, l'objectif de la perception est de s'assurer que sa représentation de l'environnement est conforme à la réalité afin de garantir un comportement sûr et robuste des fonctions autonomes, même lors de perte d'information. Pour cela, plusieurs technologies de capteurs sont utilisées sur une même voiture (Figure 1.18) et une fusion de toutes les données capteurs est réalisée selon les critères suivants :

- caractéristiques de chaque capteur : champ de détection (ouverture et portée), limites de détection, suivi de détection ;
- occultation : en fonction de la distance des objets, un capteur peut ne pas être capable de détecter un autre objet dans son champ de vision si ce dernier est masqué ;
- en fonction du type d'objet (forme, réflexion, mouvement, contraste ...) et de sa position (proche des barrières de sécurité ...), chaque capteur possède une qualité de détection différente. Par exemple, une caméra a de bonnes performances pour des objets dont elle peut reconnaitre la forme et les radars sont tout à fait adaptés pour la détection d'objets métalliques.

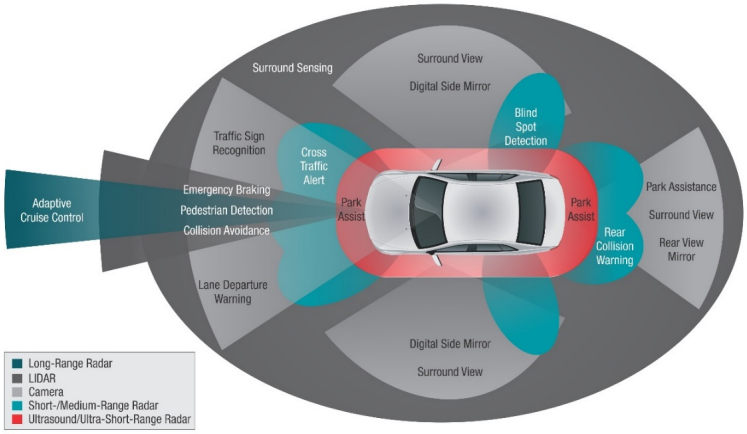


FIGURE 1.18 – Illustration de la couverture en capteurs des véhicules équipés de fonctions ADAS et autonomes [Source : https://e2e.ti.com/blogs_/b/behind_the_wheel/archive/2014/02/04/advanced-safety-and-driver-assistance-systems-paves-the-way-to-autonomous-driving

Les technologies de chaque type de capteurs et leurs apports par rapport à l'autonomisation du véhicule sont expliqués ci-après.

Caméra

Une caméra est constituée d'un imageur, d'une optique et d'un ECU pour le traitement des images permettant de détecter, reconnaitre et estimer les marquages au sol (position et courbure des lignes), mais aussi de suivre et classifier des objets (voiture, camion, deux-roues, piéton) sur des bases d'apprentissages de caractéristiques typiques. Les techniques d'apprentissage profond, ou Deep-Learning, sont de plus en plus intégrées pour la classification et la segmentation des objets.

Les véhicules ayant des fonctionnalités d'aide à la conduite sont quasiment tous équipés d'au moins une caméra frontale positionnée entre le rétroviseur intérieur et le pare-brise. Leur résolution d'image est comprise entre 1 et 2 Mpixel et leur champ de vision horizontal entre 50° et 100°. La portée de détection des objets et marquages peut aller jusqu'à 120 m pour des conditions météo favorables (les caméras sont très impactées en performances de nuit, en situation de brouillard ou fortes pluies). Certains véhicules haut de gamme sont équipés de systèmes stéréovision permettant de construire une information de profondeur et distance plus précise par triangulation. D'autres types de caméras sont aussi utilisés aujourd'hui pour les aides visuelles en parking. Ces caméras sont généralement dites « grand-angles » avec un grand champ d'ouverture (180° horizontal), mais restreintes en portée.

La tendance est de multiplier le nombre de caméras à l'avant, mais aussi tout autour du véhicule avec des focales multiples pour couvrir tout l'environnement avec des portées variables. Le coût de l'imageur et de l'optique n'est pas très élevé, car on retrouve les mêmes composants utilisés par l'électronique grand public. C'est généralement l'ECU qui amène un coût pour gérer la multitude de fonctions de détection (objets, panneaux, lignes, ...).

(a) (b) (c)

FIGURE 1.19 – Exemples de caméras : mono caméra (a), stéréovision (b) et caméra de parking (c)

FIGURE 1.20 – Exemple des capacités de détection et classification d'une caméra frontale de Mobileye [Source : https://www.mobileye.com/our-technology/]

Ultrason

Les capteurs ultrasons utilisent une onde acoustique variable inaudible pour nous (généralement entre 43 et 60 kHz). Ces capteurs sont capables de mesurer des distances pour un coût très faible. Du fait de leur portée assez réduite (jusqu'à 2,5 m), ils ont essentiellement et très largement déployés sur les véhicules pour les applications d'aide au stationnement. Une extension à la surveillance d'angle mort est aussi disponible pour certains véhicules pour limiter les coûts de cette fonction en remplacement de radars, mais avec des performances moindres. Leur champ de vue est assez limité et leur coût permet de les mettre en ceinture sur les boucliers avant et arrière des véhicules (il est assez courant d'avoir des ceintures de 12 capteurs sur les véhicules de série).

FIGURE 1.21 – Exemple d'un capteur ultrasons utilisé pour les fonctions de parking et de surveillance d'angle mort

La tendance est de les généraliser sur tous les véhicules et d'avoir des évolutions technologiques améliorant la portée jusqu'à 5 m.

Radar

Un radar exploite les ondes radio (24 GHz au début des applications ADAS et 77 GHz aujourd'hui) pour effectuer une télémétrie de la cible et mesurer directement sa vitesse relative par effet Doppler. Selon les fournisseurs, différents types de modulations et d'antennes sont utilisés pour déterminer la distance, la vitesse et la position angulaire de la cible. Les radars en 77 GHz deviennent la référence et permettent d'obtenir des champs plus importants et des résolutions de plus en plus fines.

La portée des radars permet d'aller jusqu'à 250 m avec les technologies longues portées tout en garantissant de très bonnes estimations des distances et vitesses des objets dans quasiment toutes les conditions météo.

Le radar a notamment été introduit pour les applications de type ACC du fait de ses performances d'estimation de la distance et de la vitesse du véhicule qui précède. Il permet également d'aider à l'anticipation de la cible suivante du fait que la technologie permet de « voir au travers » des objets. Pour les véhicules plus haut de gamme, les radars sont également utilisés dans les coins arrière du véhicule pour alerter ou assister le changement de voie en détectant la présence de véhicules arrivant dans les voies adjacentes.

Comme pour les caméras, la tendance est de percevoir l'environnement à 360° autour du véhicule avec au moins un radar en frontal et un radar dans chaque coin.

(a) (b)

FIGURE 1.22 – Exemples de radars Bosch dits Middle Range (a) et Long Range (b)

Lidar

Un lidar utilise la lumière (généralement infrarouge autour de 905 nm) pour la télémétrie des objets (temps entre l'émission et la réflexion d'une impulsion connaissant la vitesse de la lumière). Il est capable de déterminer le contour des obstacles statiques et dynamiques et d'en extraire les mêmes paramètres que ceux obtenus avec des radars ou caméras, mais avec des précisions plus importantes.

Le coût élevé de ce capteur avec une technologie basée sur un système rotatif de miroirs ou modules lasers n'a pas encore permis son déploiement largement sur les véhicules de série. Il est apparu en 2018 chez Audi avec le fournisseur Valeo sur des véhicules haut de gamme pour les premières applications autonomes de niveau 3. Il est toutefois très utilisé par les sociétés qui travaillent sur les navettes autonomes et robots taxi pour couvrir à 360° la détection des objets et faire de la localisation avec des techniques de type SLAM (Simultaneous Localisation and Mapping). L'arrivée de nouvelles technologies dites « Solid State » (sans système mécanique tournant) va permettre une réduction des coûts et un déploiement beaucoup plus large sur les véhicules de série.

(a) (b) (c)

FIGURE 1.23 – Illustration du capteur Scala de Valeo utilisé par Audi pour l'autonomie de niveau 3 (a), d'un capteur Velodyne utilisé pour les robots taxi (b) et d'un lidar Solid State (c) d'Innoviz pour les futurs véhicules série

FIGURE 1.24 – Exemple de ce que peut percevoir un lidar 360° sur le toit d'un robot taxi [Source : https://news.voyage.auto/an-introduction-to-lidar-the-key-self-driving-car-sensor-a7e405590cff]

Avantages et inconvénients de chaque technologie

La combinaison de différentes technologies de capteurs est la clé pour réussir à percevoir le maximum d'informations de l'environnement en tirant profit des avantages de chacune de ces différentes technologies tout en amenant des possibilités de redondance en cas de défaillances. Le tableau en Figure 1.25 illustre les différents avantages et inconvénients des technologies utilisées pour les ADAS et fonctions autonomes.

	Radar		Caméra		Lidar		Ultrason
	Long range	Middle range	Mono	Stéréo	Monofaisceau	Multifaisceau	
Portée	5	4	4	3	2	4	1
Performances longitudinale	5	5	3	4	4	5	3
Performances latérale	3	3	5	5	2	5	2
Météo	4	4	2	2	3	3	3
Visibilité de nuit	5	5	2	2	5	5	5
Objet statique	3	2	5	5	5	5	5
Intégration	3	3	4	2	3	1	4
Calibration	4	3	4	2	4	2	4
Maturité	5	5	5	4	5	3	5
Coût	4	3	4	3	5	2	5

FIGURE 1.25 – Tableau récapitulatif des avantages et inconvénients de chaque capteur, notés de 1 à 5

1.4.3 Architecture

Pour la conduite en mode autonome, plusieurs fonctions sont nécessaires pour convertir les informations issues des capteurs en commande des actionneurs du véhicule. Ces étapes sont représentées sur la Figure 1.26.

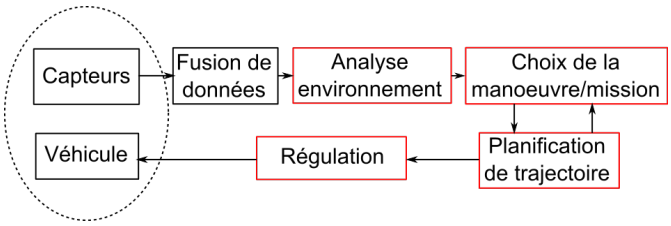


FIGURE 1.26 – Le fonctionnement d'un véhicule autonome

Les capteurs installés sur le véhicule enregistrent en permanence des informations sur l'environnement. Du fait des différentes technologies utilisées pour la redondance, plusieurs capteurs peuvent détecter le même objet en même temps, mais pas forcément exactement au même endroit. De même, dans certains cas, un objet dans le champ de vision de plusieurs capteurs peut ne pas être détecté par tous les capteurs. Pour traiter toutes ces informations pouvant être contradictoires, des algorithmes de fusion sont utilisés [72, 61]. Ces algorithmes effectuent une synthèse des données d'entrée en extrayant les objets, l'espace libre autour de la voiture, la position des objets, etc.

À partir de la fusion de l'ensemble des données capteurs, une analyse de l'environnement est possible, permettant d'extraire les informations utiles pour décider de la trajectoire du véhicule autonome. Tout comme un conducteur classique, le véhicule autonome doit être capable de déterminer si un véhicule va changer de voie, tourner à une intersection, freiner ou accélérer. Le véhicule doit déterminer dans quel état se trouvent les objets de l'environnement [164, 62, 66], prendre une décision et générer une trajectoire [132, 160, 124] qu'il devra suivre dans l'espace libre autour de lui tout en respectant le Code de la route, les autres usagers de la route [27] et la dynamique du véhicule [165, 59].

La dernière étape est le contrôle des actionneurs afin de mettre le véhicule en mouvement pour qu'il suive la trajectoire définie précédemment. Les méthodes de régulation sont multiples. Pour gérer la dynamique longitudinale et donc l'action sur le groupe motopropulseur et le groupe hydraulique des freins, on peut par exemple citer les méthodes de gains adaptatifs [159], la commande prédictive [133], ou la commande CRONE [129]. Pour la dynamique latérale et donc l'action sur la colonne de direction des méthodes comme la commande H_∞ [136], ou la logique floue [52] sont aussi utilisées.

Le travail de cette thèse porte sur les modules, ou une partie des modules, en rouge sur la Figure 1.26. Les modules travaillés sont ensuite intégrés dans les calculateurs des prototypes robotisés pour la conduite autonome du Groupe PSA. L'objectif est de proposer des solutions intégrables sur prototype à l'équipe innovation pour traiter des scénarios urbains. Comme jusqu'à présent, 80% des développements concernent des situations de vie sur autoroute, un certain nombre de ces solutions sera testé sur des scénarios autoroutiers afin de les comparer aux algorithmes existants ou pour faciliter les tests sur routes ouvertes.

1.5 Présentation des prototypes

Dans un mode autonome de niveau 2 ou plus, ce sont les ECU qui pilotent en temps réel l'ensemble des actionneurs du véhicule : le moteur, la direction et le système de freinage. Pour réaliser les fonctions d'aide à la conduite décrites dans la section 1.3, les véhicules de série équipés de ces fonctions possèdent donc les ordinateurs de bord permettant le contrôle des actionneurs. Les prototypes du projet AVA « Autonomous Vehicle for All » du Groupe PSA sont des véhicules avec des actionneurs et calculateurs série, auxquels ont été rajoutés des capteurs et des calculateurs. Pour pouvoir contrôler le groupe moto propulseur, tous les véhicules ont une boîte de vitesse automatique.

FIGURE 1.27 – Les prototypes du Groupe PSA

1.5.1 Architecture électrique et électronique pour la conduite autonome

CONFIDENTIEL

1.5.2 Les prototypes utilisés pendant la thèse

CONFIDENTIEL

1.6 Problématique et objectifs

Depuis les premières intégrations de capteurs électroniques à bord des véhicules, leur nombre n'a fait qu'augmenter au même rythme que les fonctions d'aide à la conduite proposées. Les fonctions d'autonomisation de la conduite de niveau 3 et plus sont la suite logique des évolutions des fonctions mécatroniques à bord des véhicules. Ces fonctions, dont l'architecture de fonctionnement a été présentée dans la section 1.4, font l'objet de beaucoup de sujets de recherche pour chacun des modules : perception, analyse de l'environnement, décision, planification de trajectoire et contrôle des actionneurs.

Les recherches sont déjà très avancées en ce qui concerne les situations de vie sur les routes à chaussées séparées. Cela fait déjà plusieurs années que les expérimentations ont commencé et plusieurs constructeurs, dont Audi et Tesla, ont même annoncé qu'ils étaient prêts à la commercialisation de fonctions d'aide à la conduite de niveau 3.

L'équipe innovation du Groupe PSA dans laquelle s'est déroulée la thèse concentre donc une partie de ses travaux sur la conduite autonome en environnement urbain. Par rapport aux travaux précédents de l'équipe en environnement autoroutier, la conduite autonome en environnement urbain se complexifie en raison d'une augmentation des situations de vie possibles, un changement des usagers, avec l'apparition de piétons et de vélos notamment, et un changement du type d'infrastructure (intersections, ronds-points, passages piétons, ...). L'objectif de cette thèse est donc, à partir des connaissances et des algorithmes développés pour les fonctions autonomes sur autoroute, d'améliorer et d'enrichir les fonctions d'autonomie existantes pour l'environnement urbain et de créer les modules supplémentaires nécessaires pour gérer les nouvelles situations de vie.

Parmi les modules à modifier pour le cas urbain, cette thèse se focalise sur la planification de trajectoire et le guidage latéral du véhicule. Ces deux points nécessitent d'une part une bonne analyse de l'environnement, ce qui sera décrit plus tard à travers la prédiction de trajectoire des véhicules entourant l'ego véhicule, et d'autre part une compréhension complète de la dynamique latérale du véhicule. Il est donc nécessaire de choisir la méthode la plus adaptée et de développer des fonctions respectant les contraintes liées à cette dynamique. Le Chapitre 2 se focalise donc sur la modélisation de la dynamique latérale et son observation.

Chapitre 2

Modélisation et observation de la dynamique latérale

Sommaire

2.1 Les modèles dynamiques du véhicule **43**
2.1.1 Le modèle de validation : le modèle non linéaire 43
2.1.2 Les modèles simplifiés 46
2.1.3 Comparaison temporelle 50
2.1.4 Comparaison fréquentielle des modèles de la dynamique latérale du véhicule 55
2.2 Mise en évidence de l'influence de la vitesse longitudinale **59**
2.2.1 Description du scénario 59
2.2.2 Résultats de la simulation pour le modèle de validation 60
2.2.3 Comparaison temporelle avec les modèles simplifiés 63
2.2.4 Analyse fréquentielle du modèle linéaire 66
2.2.5 Comparaison fréquentielle avec les autres modèles simplifiés 69
2.3 Observateur d'état des systèmes dynamiques : état de l'art **70**
2.3.1 Les méthodes 71
2.3.2 Les modèles 72
2.3.3 Les variables observées 72
2.3.4 Les capteurs supplémentaires 73
2.4 Estimation de la dynamique latérale par approche multimodèle . . **73**
2.4.1 Le modèle Takagi-Sugeno 74
2.4.2 Observabilité du modèle Takagi-Sugeno 76
2.4.3 Validation du modèle Takagi-Sugeno 76
2.4.4 Problème d'échantillonnage à basse vitesse 78
2.5 Validation et résultats pour l'estimation de la vitesse de lacet . . . **81**

On n'a jamais fait de grande découverte sans hypothèse audacieuse.

Isaac Newton

Le déplacement longitudinal et le déplacement latéral du véhicule sont régis par des lois physiques. Il est alors possible de les représenter à partir d'équations. Pour la conduite autonome, ses équations sont nécessaires pour les modules de génération de trajectoire et de guidage latéral. En effet, un conducteur humain sait par expérience comment va se comporter le véhicule en fonction des efforts appliqués par les actionneurs. Pour la conduite autonome, la modélisation physique du véhicule est indispensable pour être certain que la trajectoire est une trajectoire admissible et que les consignes envoyées aux actionneurs mènent au comportement désiré.

La thèse se concentrant sur le guidage latéral, seule la modélisation de la dynamique latérale est traitée. La première partie du chapitre présente donc les différents modèles utilisés : un modèle de validation, qui servira de référence, et des modèles simplifiés. Ces modèles sont ensuite comparés dans les domaines fréquentiels et temporels dans le cas d'une simulation de changement de voie. Puis, l'influence de la dynamique longitudinale est mise en avant afin de montrer les limites des différentes modélisations en fonction de ce paramètre.

À partir des modélisations, il est possible d'estimer des variables de la dynamique latérale non mesurables. En utilisant en plus des données capteurs disponibles, ces estimations sont plus précises, car diminuent les incertitudes liées à une mauvaise modélisation ou une mauvaise calibration du modèle. Dans le cadre du véhicule autonome, et notamment les fonctions de guidage latéral et de localisation, une estimation précise des variables de la dynamique latérale est nécessaire. Ainsi la deuxième partie du chapitre se focalise sur l'observation de la dynamique latérale du véhicule avec un état de l'art des observateurs et la présentation d'une méthode d'estimation de la dynamique latérale du véhicule par approche multimodèle. La méthode est ensuite validée et comparée à d'autres techniques pour l'estimation de la vitesse de lacet sur prototype.

2.1 Les modèles dynamiques du véhicule

Il existe différents modèles dynamiques pour représenter plus ou moins fidèlement la dynamique latérale d'un véhicule. Un modèle dynamique peut être choisi en fonction de son utilisation ou pour son compromis entre fidélité et simplicité. On verra par exemple dans le Chapitre 3 que pour chaque type d'observateur un certain type de modèle est requis. Sont présentés dans cette section, les modèles simplifiés usuels de la dynamique latérale d'un véhicule utilisés pour l'estimation ou la régulation de variables de cette dynamique. Ces modèles simplifiés sont dérivés du modèle de validation, qui est le modèle complet non linéaire représentant le plus fidèlement la dynamique latérale du véhicule. Cette section exprime donc dans un premier temps comment s'écrit le modèle de validation, puis présente les hypothèses menant aux modèles simplifiés.

2.1.1 Le modèle de validation : le modèle non linéaire

L'équation décrivant la dynamique de lacet $\Psi(t)$ du véhicule est donnée par :

$$\dot{\Psi}(t) = \frac{1}{I_z}\int_0^t C_{\Sigma z}(\tau)d\tau + \dot{\Psi}(0), \tag{2.1}$$

où I_z est le moment d'inertie et $C_{\Sigma z}(t)$ la somme des couples extérieurs appliqués à l'axe vertical du véhicule telle que :

$$\begin{aligned} C_{\Sigma z}(t) =& L_f\left(F_{y11}(t) + F_{y12}(t)\right) - L_r\left(F_{y21}(t) + F_{y22}(t)\right) \\ &+ l_f\left(F_{x12}(t) - F_{x11}(t)\right) + l_r\left(F_{x22}(t) - F_{x21}(t)\right), \end{aligned} \tag{2.2}$$

avec L_f et L_r les empattements avant et arrière, et l_f et l_r les demi-voies avant et arrière. On note également $\beta_v(t)$, l'angle de braquage des roues avant. Les forces $F_x(t)$ et $F_y(t)$ utilisées dans (2.2) sont les forces longitudinales et latérales générées par les contacts entre les pneumatiques et la chaussée. Chaque indice correspond à un pneumatique spécifique comme illustré sur la Figure 2.1.

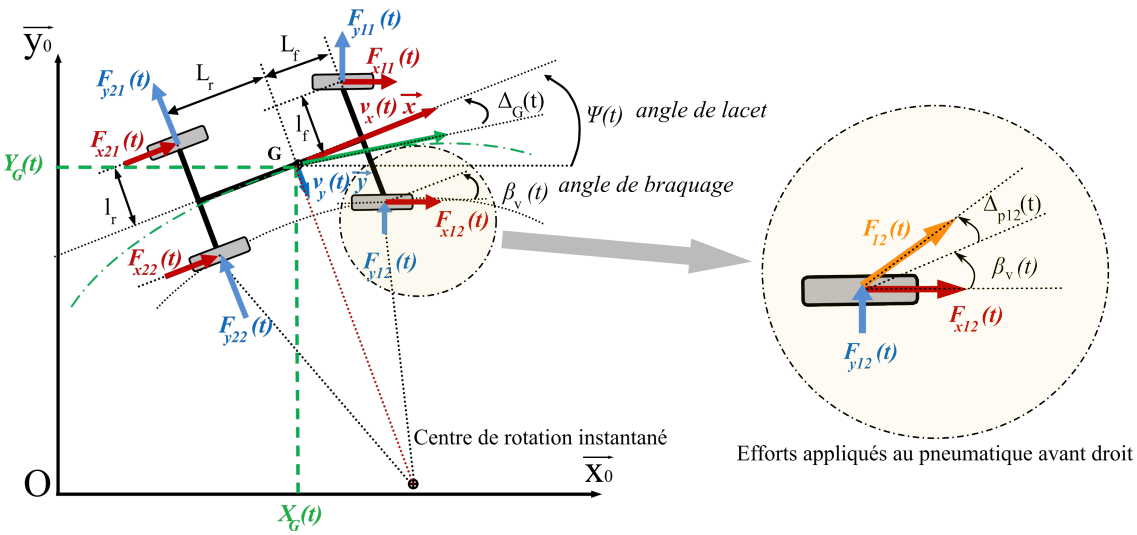


FIGURE 2.1 – Représentation schématique du véhicule et des forces appliquées

Sachant que le véhicule roule à une vitesse longitudinale $V_x(t)$, sa vitesse transversale s'exprime :

$$V_y(t) = \frac{1}{M_t}\int_0^t F_{\Sigma y}(\tau)d\tau - \int_0^t V_x(t)\dot{\Psi}(\tau)d\tau + V_y(0), \tag{2.3}$$

avec M_t la masse totale du véhicule et $F_{\Sigma y}(t)$ la somme des forces transversales extérieures appliquées au centre de gravité telle que :

$$F_{\Sigma y}(t) = F_{y11}(t) + F_{y12}(t) + F_{y21}(t) + F_{y22}(t) + f_{0y}(t), \tag{2.4}$$

où $f_{0y}(t)$ représente l'effort résultant d'une éventuelle rafale de vent, appliqué au centre de gravité du véhicule.

L'utilisation de la deuxième loi de Newton permet d'exprimer $\Gamma_t(t)$, l'accélération latérale dans le repère absolu $\mathcal{R}_0 = (O, \overrightarrow{x_0}, \overrightarrow{y_0})$, telle que :

$$M_t\Gamma_t(t) = F_{y11}(t) + F_{y12}(t) + F_{y21}(t) + F_{y22}(t) + f_{0y}(t). \tag{2.5}$$

Plusieurs modèles de représentation des pneumatiques existent [169]. Ici, le modèle de Pacejka est utilisé pour mettre en équation les efforts des pneumatiques [143]. Il permet d'exprimer l'effort latéral $F_{yij}(t)$ en fonction de l'angle de dérive $\Delta_{pij}(t)$:

$$F_{yij}(t) = \mu D_{yij}\sin\left[C_{yij}\arctan\left(B_{yij}\left[(1-E_{yij})\Delta_{pij}(t) + \frac{E_{yij}}{B_{yij}}\arctan(B_{yij}\Delta_{pij}(t))\right]\right)\right], \tag{2.6}$$

les paramètres D_{yij}, C_{yij}, B_{yij} et E_{yij} dépendent des micros paramètres des pneumatiques, de l'effort normal $F_z(t)$ et du coefficient d'adhérence de la route μ.

Les angles de dérive $\Delta_{pij}(t)$ utilisés dans l'équation (2.6) sont donnés par les relations :

$$\begin{aligned}
\Delta_{p11}(t) &= \beta_v(t) - \text{atan}\left(\frac{V_y(t) + L_f\dot{\Psi}(t)}{V_x(t) - l_f\dot{\Psi}(t)}\right) \\
\Delta_{p12}(t) &= \beta_v(t) - \text{atan}\left(\frac{V_y(t) + L_f\dot{\Psi}(t)}{V_x(t) + l_f\dot{\Psi}(t)}\right) \\
\Delta_{p21}(t) &= -\text{atan}\left(\frac{V_y(t) - L_r\dot{\Psi}(t)}{V_x(t) - l_r\dot{\Psi}(t)}\right) \\
\Delta_{p22}(t) &= -\text{atan}\left(\frac{V_y(t) - L_r\dot{\Psi}(t)}{V_x(t) + l_r\dot{\Psi}(t)}\right).
\end{aligned} \tag{2.7}$$

À noter que l'angle de braquage des roues avant $\beta_v(t)$ est proportionnel à l'angle volant $\Theta_v(t)$, soit :

$$\Theta_v(t) = \lambda\beta_v(t), \tag{2.8}$$

où λ est considéré comme une constante qui représente le rapport de réduction de la direction.

La trajectoire du véhicule est décrite par ses coordonnées $X_G(t)$ et $Y_G(t)$ de son centre de gravité dans un repère $\mathcal{R}_0$. $\mathcal{R}_0 = (O, \overrightarrow{x_0}, \overrightarrow{y_0})$ est le repère absolu et $\mathcal{R} = (G, \overrightarrow{x}, \overrightarrow{y})$ est le repère véhicule comme illustré sur la Figure 2.1. En robotique, la mesure du déplacement est appelée odométrie. Ces coordonnées sont calculées en intégrant la vitesse du centre de gravité de la voiture :

$$\begin{aligned}
\overrightarrow{V}_{G/R_0} = V_x(t)\overrightarrow{x} + V_y(t)\overrightarrow{y} &= \left[\frac{d\overrightarrow{OG}}{dt}\right]_{R_0} \\
&= \left[\frac{d}{dt}\left(X_G(t)\overrightarrow{x_0} + Y_G(t)\overrightarrow{y_0}\right)\right]_{R_0} \\
&= \dot{X}_G(t)\overrightarrow{x_0} + \dot{Y}_G(t)\overrightarrow{y_0}.
\end{aligned} \tag{2.9}$$

Sachant que les équations permettant le passage du repère $\mathcal{R}_0$ vers $\mathcal{R}$ sont :

$$\begin{cases}
\overrightarrow{x} = \cos\left(\Psi(t)\right)\overrightarrow{x_0} + \sin\left(\Psi(t)\right)\overrightarrow{y_0} \\
\overrightarrow{y} = -\sin\left(\Psi(t)\right)\overrightarrow{x_0} + \cos\left(\Psi(t)\right)\overrightarrow{y_0},
\end{cases} \tag{2.10}$$

les dérivées des coordonnées du centre de gravité dans le repère absolu sont donc :

$$\begin{cases}
\dot{X}_G(t) = V_x(t)\cos\left(\Psi(t)\right) - V_y(t)\sin\left(\Psi(t)\right) \\
\dot{Y}_G(t) = V_x(t)\sin\left(\Psi(t)\right) + V_y(t)\cos\left(\Psi(t)\right).
\end{cases} \tag{2.11}$$

Les paramètres nominaux de la dynamique latérale du véhicule, utilisés pour les simulations sont répertoriés dans la Table 2.1. M_f et M_r sont respectivement les masses appliquées aux essieux avant et arrière telles que :

$$L_f = \frac{M_r}{M_t}L \quad \text{et} \quad L_r = \frac{M_f}{M_t}L. \tag{2.12}$$

M_t	1759 kg
M_f	1319 kg
M_r	440 kg
I_z	2638,5 kg.m^2
L_f	0,71 m
L_r	2,13 m
c_{yf} $(\mu = 1)$	94446 N.rad^{-1}
c_{yr} $(\mu = 1)$	48699 N.rad^{-1}
λ	16

TABLE 2.1 – Paramètres nominaux du véhicule

2.1.2 Les modèles simplifiés

Modèle cinématique

Le modèle cinématique du véhicule est défini dans [153]. En négligeant la dérive des pneumatiques, le modèle de la dynamique latérale défini précédemment se ramène alors à un modèle purement géométrique comme le montre la Figure 2.2.

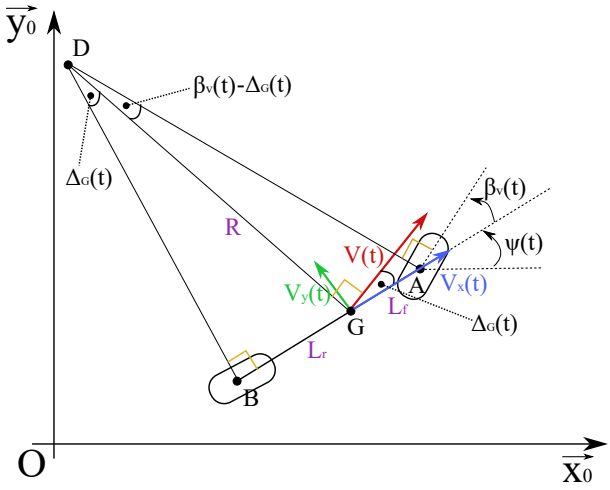


FIGURE 2.2 – Représentation schématique du véhicule bicyclette cinématique

Ici, le centre de rotation instantané D est défini comme l'intersection des droites (DA) et (DB) perpendiculaires aux roues avant et arrière. Le véhicule a alors un rayon de courbure R, qui est la distance entre le centre de gravité G et D. L'angle de dérive du centre de gravité est noté $\Delta_G(t)$ défini tel que :

$$\Delta_G(t) = \text{atan}\left(\frac{V_y(t)}{V_x(t)}\right). \tag{2.13}$$

La loi des sinus dans le triangle DAG donne :

$$\frac{\sin(\beta_v(t) - \Delta_G(t))}{L_f} = \frac{\sin(\pi/2 - \beta_v(t))}{R}. \tag{2.14}$$

En développant, on obtient :

$$\frac{\sin(\beta_v(t))\cos(\Delta_G(t)) - \sin(\Delta_G(t))\cos(\beta_v(t))}{L_f} = \frac{\cos(\beta_v(t))}{R}, \tag{2.15}$$

et en multipliant par $\dfrac{L_f}{\cos(\beta_v(t))}$ les deux membres de (2.15), il vient :

$$\tan(\beta_v(t))\cos(\Delta_G(t)) - \sin(\Delta_G(t)) = \frac{L_f}{R}. \tag{2.16}$$

Il est ensuite possible d'exprimer le sinus de $\Delta_G(t)$ grâce au triangle DGB :

$$\sin(\Delta_G(t)) = \frac{L_r}{R}. \tag{2.17}$$

En transposant (2.17) dans (2.16) :

$$\tan(\beta_v(t))\cos(\Delta_G(t)) = \frac{L}{R}, \tag{2.18}$$

avec $L = L_f + L_r$ la longueur du véhicule.

Le modèle cinématique suppose que le véhicule roule à faible vitesse. Ainsi, le rayon de courbure du véhicule évolue lentement et la vitesse de lacet $\dot{\Psi}(t)$ est égale à la vitesse angulaire du véhicule :

$$\dot{\Psi}(t) = \frac{V(t)}{R}. \tag{2.19}$$

Avec les équations (2.18) et (2.19) :

$$\dot{\Psi}(t) = \frac{V(t)}{L}\tan(\beta_v(t))\cos(\Delta_G(t)), \tag{2.20}$$

et donc :

$$\dot{\Psi}(t) = \frac{V_x(t)}{L}\tan(\beta_v(t)). \tag{2.21}$$

L'équation décrivant le mouvement du centre de gravité dans le repère absolu est :

$$\dot{Y_G}(t) = V(t)\sin(\Psi(t) + \Delta_G(t)), \tag{2.22}$$

soit en développant :

$$\dot{Y_G}(t) = V(t)\left[\sin(\Psi(t))\cos(\Delta_G(t)) + \sin(\Delta_G(t))\cos(\Psi(t))\right], \tag{2.23}$$

et en simplifiant :

$$\dot{Y_G}(t) = V_x(t)\sin(\Psi(t)) + V_y(t)\cos(\Psi(t)). \tag{2.24}$$

Dans [153], la variable $V(t)$, avec ses composantes $V_x(t)$ et $V_y(t)$ est supposée connue. Ici, on va faire l'hypothèse que $V_x(t)\sin(\Psi(t)) \gg V_y(t)\cos(\Psi(t))$ pour pouvoir exprimer les variables de vitesse de lacet et position latérale uniquement en fonction de la vitesse longitudinale et l'angle volant.

Les équations décrivant la dynamique latérale pour le modèle cinématique sont donc :

$$\begin{aligned} \dot{\Psi}(t) &= \frac{V_x(t)}{L}\tan(B_v(t)) \\ \dot{Y_G}(t) &= V_x(t)\sin(\Psi(t)). \end{aligned} \tag{2.25}$$

Le modèle linéaire

Afin d'analyser la dynamique latérale du véhicule dans le domaine fréquentiel et de synthétiser ensuite la commande, un modèle linéaire de la dynamique de ce système est nécessaire. Parmi les équations ci-dessus décrivant la dynamique latérale du véhicule, (2.6), (2.7) et (2.11) sont les seules équations non linéaires.

En supposant que la vitesse de lacet est faible devant la vitesse longitudinale : $V_x(t) = V_{x0} >> |l_f\dot{\Psi}(t)|$, $V_{x0} >> |l_r\dot{\Psi}(t)|$ et que les angles sont petits, les expressions (2.7) des angles de dérive des pneumatiques deviennent pour $j \in \{1;2\}$:

$$\Delta_{p1j}(t) = \Delta_{pf}(t) = \beta_v(t) - \frac{V_y(t) + L_f\dot{\Psi}(t)}{V_{x0}},\ \Delta_{p2j}(t) = \Delta_{pr}(t) = -\frac{V_y(t) - L_r\dot{\Psi}(t)}{V_{x0}}. \tag{2.26}$$

De la même manière, (2.11) peut se simplifier de la manière suivante :

$$\dot{Y}_G(t) = V_{x0}\Psi(t) + V_y(t). \tag{2.27}$$

Lorsque l'angle de dérive $\Delta_{pij}(t)$ d'une roue est suffisamment faible, compris entre $[-1°; 1°]$, l'effort latéral $F_{yij}(t)$ exprimé par (2.6) peut se simplifier sous une forme linéaire de la forme :

$$F_{yij}(t) = c_{yij}\Delta_{pij}(t), \tag{2.28}$$

avec c_{yij} la rigidité de dérive du pneumatique associée telle que :

$$c_{yij} = \mu B_{yij}C_{yij}D_{yij}. \tag{2.29}$$

En supposant que les pneumatiques sont identiques et que l'effort normal est réparti de manière symétrique sur les côtés gauche et droit du véhicule, il vient alors :

$$F_{y1j}(t) = F_{yf}(t) = c_{yf}\Delta_{pf}(t),\ F_{y2j}(t) = F_{yr}(t) = c_{yr}\Delta_{pr}(t), \tag{2.30}$$

avec c_{yf} et c_{yr} les rigidités de dérive des pneumatiques avant et arrière.

Maintenant que toutes les équations ont été linéarisées, il est possible d'écrire les expressions de l'accélération de lacet $\ddot{\Psi}(t)$, de la dérivée de la vitesse latérale dans le repère $\mathcal{R}$, $\dot{V}_y(t)$, et de l'accélération latérale dans le repère $\mathcal{R}_0$, Γ_t, linéairement en fonction de l'angle de braquage, de la vitesse latérale et de la vitesse de lacet.

En dérivant (2.1) et en considérant que les masses de chaque essieu sont réparties de manière uniforme sur les côtés gauche et droit de (2.2), il vient :

$$I_z\ddot{\Psi}(t) = 2L_fF_{yf}(t) - 2L_rF_{yr}(t). \tag{2.31}$$

En utilisant les relations (2.30) entre les efforts latéraux des pneumatiques et leur angle de dérive, (2.31) se réécrit :

$$I_z\ddot{\Psi}(t) = 2L_fc_{yf}\Delta_{pf}(t) - 2L_rc_{yr}\Delta_{pr}(t), \tag{2.32}$$

et l'utilisation de (2.26) permet d'obtenir la relation linéaire de la dérivée de l'angle de lacet en fonction des autres composantes :

$$\ddot{\Psi}(t) = 2\frac{L_f c_{yf}}{I_z}\beta_v(t) + 2\frac{-L_f c_{yf} + L_r c_{yr}}{V_{x0} I_z} V_y(t) - 2\frac{L_f^2 c_{yf} + L_r^2 c_{yr}}{V_{x0} I_z}\dot{\Psi}(t). \tag{2.33}$$

Pour exprimer la dérivée de la vitesse latérale, l'équation de départ est la dérivée de (2.3) donnant avec l'hypothèse précédente sur la répartition des masses :

$$\dot{V}_y(t) = \frac{2}{M_t}(F_{yf}(t) + F_{yr}(t)) - V_{x0}\dot{\Psi}(t). \tag{2.34}$$

En suivant la même démarche que précédemment avec (2.30), $\dot{V}_y(t)$ s'écrit :

$$\dot{V}_y(t) = \frac{2}{M_t}\left(c_{yf}\Delta_{pf}(t) + c_{yr}\Delta_{pr}(t)\right) - V_{x0}\dot{\Psi}(t), \tag{2.35}$$

et avec (2.26) :

$$\dot{V}_y(t) = 2\frac{c_{yf}}{M_t}\beta_v(t) - 2\frac{c_{yf} + c_{yr}}{M_t V_{x0}} V_y(t) + \left[2\frac{-c_{yf} L_f + c_{yr} L_r}{M_t V_{x0}} - V_{x0}\right]\dot{\Psi}(t). \tag{2.36}$$

Enfin, la linéarisation de l'expression de l'accélération latérale utilise les équations (2.5) et (2.30) pour obtenir

$$M_t \Gamma_t(t) = 2c_{yf}\Delta_{pf}(t) + 2c_{yr}\Delta_{pr}(t), \tag{2.37}$$

qui, avec (2.26), devient :

$$\Gamma_t(t) = 2\frac{c_{yf}}{M_t}\beta_v(t) - 2\frac{c_{yf} + c_{yr}}{M_t V_{x0}} V_y(t) + 2\frac{-c_{yf} L_f + c_{yr} L_r}{M_t V_{x0}}\dot{\Psi}(t). \tag{2.38}$$

L'hypothèse des petits angles donne l'équation de la dérivée de la position latérale linéaire :

$$\dot{Y}_G(t) = V_{x0}\Psi(t) + V_y(t). \tag{2.39}$$

Le modèle linéaire en régime circulaire uniforme

En plus des hypothèses précédentes sur les petits angles et le régime linéaire des pneumatiques, le véhicule est supposé être en mouvement circulaire uniforme. Les hypothèses d'un mouvement circulaire uniforme sont :

$$\ddot{\Psi}(t) = 0, \tag{2.40}$$

et

$$\dot{\Psi}(t) = \frac{\Gamma_t(t)}{V_{x0}}. \tag{2.41}$$

Avec ces deux hypothèses, il devient possible d'exprimer l'accélération latérale directement en fonction de l'angle de braquage des roues.

Les équations (2.40), (2.31) et (2.5) permettent d'écrire les relations entre les efforts latéraux des pneumatiques et l'accélération latérale :

$$F_{yf}(t) = \frac{L_r M_t}{2L}\Gamma_t(t), \quad F_{yr}(t) = \frac{L_f M_t}{2L}\Gamma_t(t). \tag{2.42}$$

Sachant que (2.26) permet d'écrire :

$$\Delta_{pf}(t) - \Delta_{pr}(t) = \beta_v(t) - \frac{L}{V_{x0}}\dot{\Psi}(t), \tag{2.43}$$

avec (2.30) :

$$\frac{F_{yf}(t)}{c_{yf}} - \frac{F_{yr}(t)}{c_{yr}} = \beta_v(t) - \frac{L}{V_{x0}}\dot{\Psi}(t), \tag{2.44}$$

et avec (2.42) :

$$M_t\left(\frac{L_r c_{yr} - L_f c_{yf}}{2Lc_{yr}c_{yf}}\right)\Gamma_t(t) = \beta_v(t) - \frac{L}{V_{x0}}\dot{\Psi}(t). \tag{2.45}$$

Et finalement, avec la deuxième hypothèse du mouvement circulaire uniforme (2.41) :

$$\boxed{\beta_v(t) = \left[M_t\left(\frac{L_r c_{yr} - L_f c_{yf}}{2Lc_{yr}c_{yf}}\right) + \frac{L}{V_{x0}^2}\right]\Gamma_t(t)} \tag{2.46}$$

2.1.3 Comparaison temporelle

Le scénario

Les équations de la dynamique de chaque modèle décrit précédemment sont utilisées pour modéliser un changement de voie du véhicule afin de comparer les modèles simplifiés au modèle de validation non linéaire servant de référence. Cette simulation va ainsi permettre d'analyser la pertinence des modèles simplifiés.

Le changement de voie est effectué à une vitesse $V_{x0} = 90$ km/h. Le chemin suivi par le véhicule débute par une ligne droite de 5 m, puis un changement de voie sur 200 m. L'amplitude de l'angle volant en entrée est calculée de manière qu'au bout de ces 200 m, le véhicule s'est déplacé latéralement de 3,5 m. L'allure de la trajectoire désirée est tracée à la Figure 2.3. De plus, le véhicule est déjà à la vitesse cible au début de la simulation ($X_G = 0$ m).

Afin de garantir une trajectoire lisse et naturelle pour un conducteur lambda, telle que présentée en Figure 2.3, l'angle volant choisi est une période d'un sinus. La période T_{θ_v} de ce sinus est calculée avec (2.47) tel que le changement de voie dure 200 m. L'amplitude du signal est ensuite adaptée afin d'atteindre les 3,5 m latéraux au bout des 200 m.

$$T_{\theta_v} = \frac{D_x}{V_{x0}}, \tag{2.47}$$

avec $D_x = 200$ m la distance parcourue lors du changement de voie.

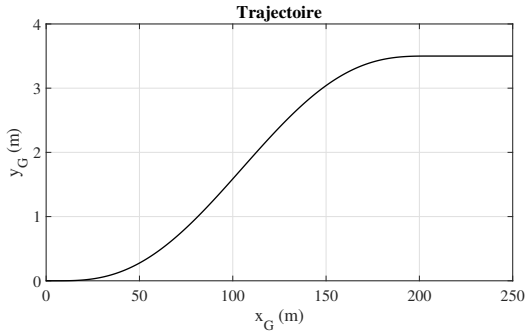


FIGURE 2.3 – Trajectoire du changement de voie

Les résultats de la simulation

Les variables de la dynamique latérale du modèle linéaire sont comparées à celles des modèles simplifiés de la section précédente. Les variables des modèles simplifiés, tracées en Figure 2.5 et 2.6, correspondent aux équations encadrées de la section 2.1.2. En entrée de chaque modèle est utilisé l'angle volant présenté Figure 2.4 et la vitesse longitudinale fixée à $V_{x0} = 90$ km/h.

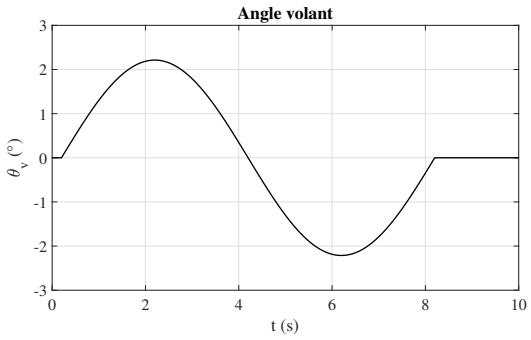
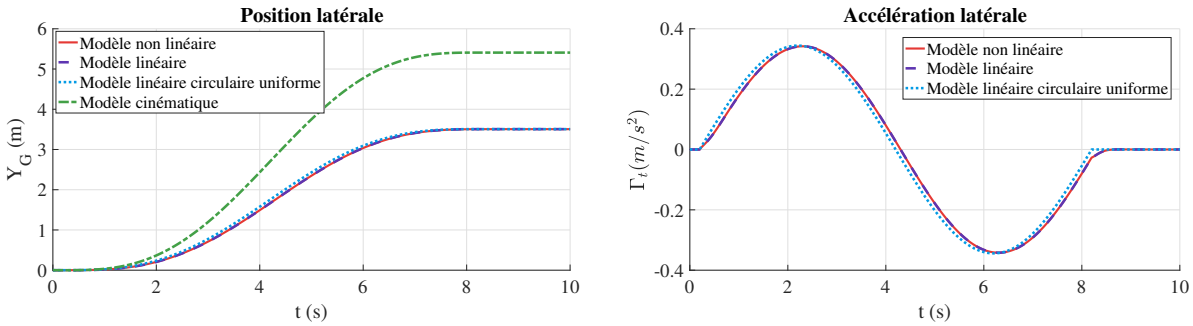


FIGURE 2.4 – Évolution de l'angle volant

Les simulations temporelles montrent qu'à une vitesse constante de 90 km/h, dans le cas d'un changement de voie, le modèle linéaire (courbes violettes) est fidèle au modèle non linéaire (courbes rouges). L'erreur en position latérale est de maximum 0.7mm (Figure 2.5 (a)) et celle en vitesse de lacet de 10^{-4} °/s (Figure 2.6 (b)).



(a) (b)

FIGURE 2.5 – Position latérale du véhicule (a) et accélération latérale (b)

En ajoutant les hypothèses de mouvement circulaire uniforme aux hypothèses des petits angles et des zones de fonctionnement des pneumatiques, le modèle linéaire circulaire uniforme reste tout de même très représentatif de la dynamique latérale du véhicule. En effet, pour cette modélisation simplifiée et dans ce cas d'étude, l'erreur en position est de maximum 9 cm (Figure 2.5 (a)) et celle en vitesse de lacet de 0,05°/s (Figure 2.6 (b)). Cependant, le modèle présente une avance de 0,1 s en accélération latérale (Figure 2.5 (b)).

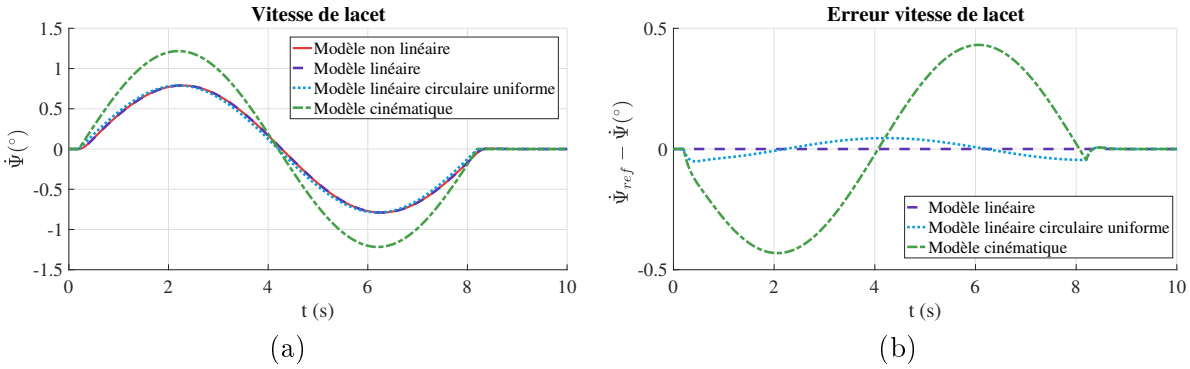


FIGURE 2.6 – Vitesse de lacet (a) et erreur de vitesse de lacet par rapport au modèle de validation (b)

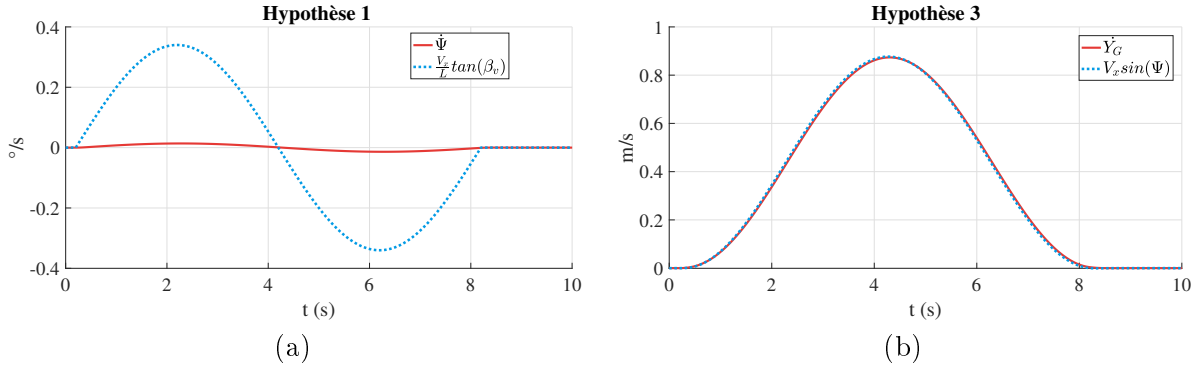
Quant au modèle cinématique, ce dernier ne permet pas une bonne représentation du modèle dans ce cas. L'erreur finale en position est de quasiment 2 m et l'erreur maximale en vitesse de est de 0,4°/s, ce qui correspond à des erreurs relatives de plus de 40 %. Ce résultat est prévisible du fait que la simulation se fait avec une vitesse de 90 km/h alors que ce modèle n'est valide qu'à faible vitesse.

Analyse des hypothèses

Afin de comprendre pourquoi les modèles simplifiés restent plus ou moins fidèles au modèle non linéaire, les hypothèses de simplification sont analysées avec les variables du modèle de validation. Pour rappel, les hypothèses utilisées pour chaque modèle sont listées ci-après.

- Modèle cinématique
  - Hypothèse 1 : $\dot{\Psi}(t) = \dfrac{V_x(t)}{L}\tan(\beta_v(t))$
  - Hypothèse 2 : $V_x(t)\sin(\Psi(t)) \gg V_y(t)\cos(\Psi(t))$
  - Hypothèse 3 : $\dot{y_G}(t) = V_x(t)\sin(\Psi(t))$
- Modèle linéaire
  - Hypothèse 4 : $V_{x0} \gg |l_f\dot{\Psi}(t)|$ et $V_{x0} \gg |l_r\dot{\Psi}(t)|$
  - Hypothèse 5 : $\Psi(t)$ faible
  - Hypothèse 6 : les pneumatiques sont dans leurs zones linéaires de fonctionnement
- Modèle linéaire en régime circulaire uniforme
  - Hypothèses 4, 5 et 6
  - Hypothèse 7 : $\ddot{\Psi}(t) = 0$
  - Hypothèse 8 : $\dot{\Psi}(t) = \dfrac{\Gamma_t(t)}{V_{x0}}$

Pour le modèle cinématique, il faut vérifier l'hypothèse 2 ainsi que les hypothèses 1 et 3 représentées par (2.25). Les deux membres de la première équation de (2.25) sont tracés sur la Figure 2.7 (a) et ceux de la deuxième équation sur la Figure 2.7 (b).



(a) (b)

FIGURE 2.7 – Première hypothèse du modèle cinématique sur la vitesse de lacet (a) et troisième hypothèse sur la dérivée de la position latérale (b)

Alors que la première hypothèse sur l'estimation de la vitesse de lacet en fonction de l'angle de braquage n'est pas du tout représentative de la réalité avec une erreur de $0,3°/\text{s}$, 95% en relatif, lorsque l'angle volant est maximal, l'hypothèse 3 sur la dérivée de la position est vérifiée avec une erreur relative de maximum 5%.

Pour vérifier l'hypothèse 2, c'est-à-dire que $V_y(t)\cos(\Psi(t))$ est négligeable devant $V_x(t)\sin(\Psi(t))$, ces variables sont tracées sur la Figure 2.8. Au maximum, $\max(V_y(t)\cos(\Psi(t))) = 0,01$ m/s. C'est en moyenne 50 fois inférieur à $V_x(t)\sin(\Psi(t))$ ce qui rend l'hypothèse valide.

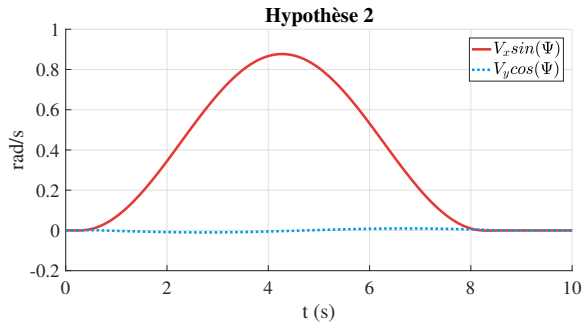


FIGURE 2.8 – Deuxième hypothèse du modèle cinématique

Finalement, c'est la non-vérification de la première hypothèse qui conduit aux erreurs d'estimation de la position latérale et de la vitesse de lacet observées précédemment pour le modèle cinématique.

Plusieurs hypothèses ont été posées pour écrire le modèle linéaire. La première (hypothèse 4) est que $V_x(t) = V_{x0} >> |l_f\dot{\Psi}(t)|$ et $V_{x0} >> |l_r\dot{\Psi}(t)|$. Numériquement, $V_{x0} = 25$ m/s, $l_f\max(\dot{\Psi}(t)) = 0,022$ m/s et $l_r\max(\dot{\Psi}(t)) = 0,022$ m/s. Avec un rapport supérieur à 1000 entre les expressions, les inégalités sont vérifiées.

La deuxième hypothèse (hypothèse 5) est celle d'un petit angle de lacet pour simplifier les relations trigonométriques de l'équation de la trajectoire (2.11). L'angle de lacet issu du modèle de validation est tracé sur la Figure 2.9. Lors de la simulation, $\Psi(t)$ reste inférieur à 2° ce qui permet de valider l'hypothèse.

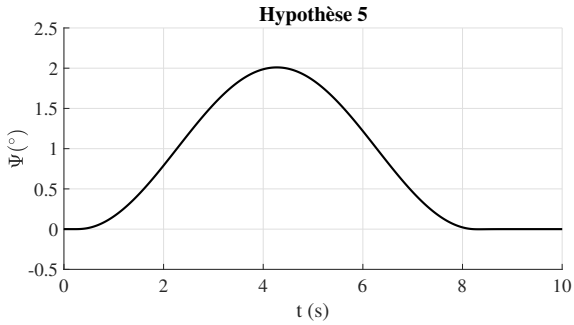
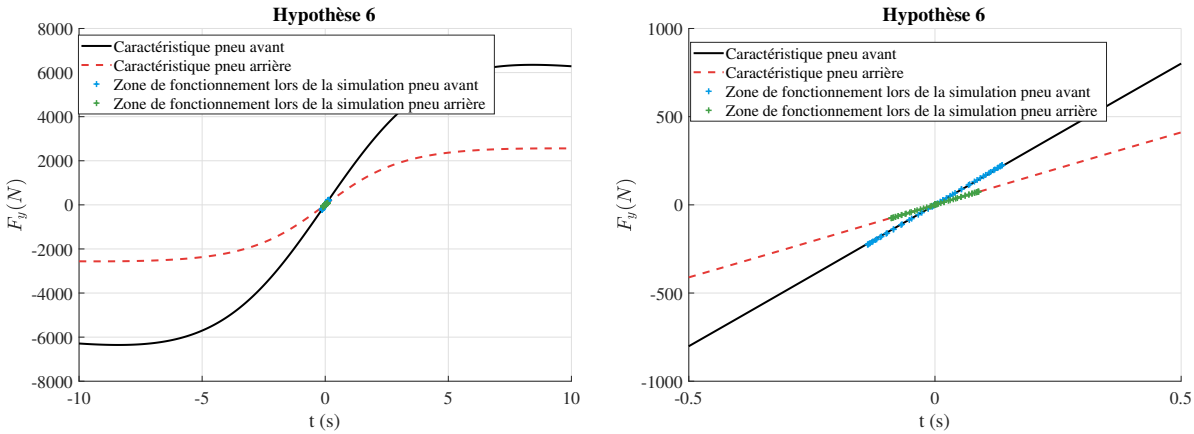


FIGURE 2.9 – Évolution de l'angle de lacet pour l'hypothèse des petits angles

La dernière hypothèse du modèle linéaire (hypothèse 6) concerne la zone de fonctionnement des pneumatiques. C'est la supposition d'une relation linéaire entre l'effort latéral appliqué au pneumatique et l'angle de dérive de ce dernier qui mène à la relation (2.30). Pour vérifier cette supposition, les caractéristiques des pneumatiques avant et arrière ainsi que leurs zones de fonctionnement lors de la simulation du modèle de validation sont tracés sur la Figure 2.10. Pour un changement de voie à 90 km/h, les efforts des pneumatiques sont bien dans leur zone linéaire, car les angles de dérive des pneumatiques restent inférieurs à $0,5°$, l'hypothèse est validée.

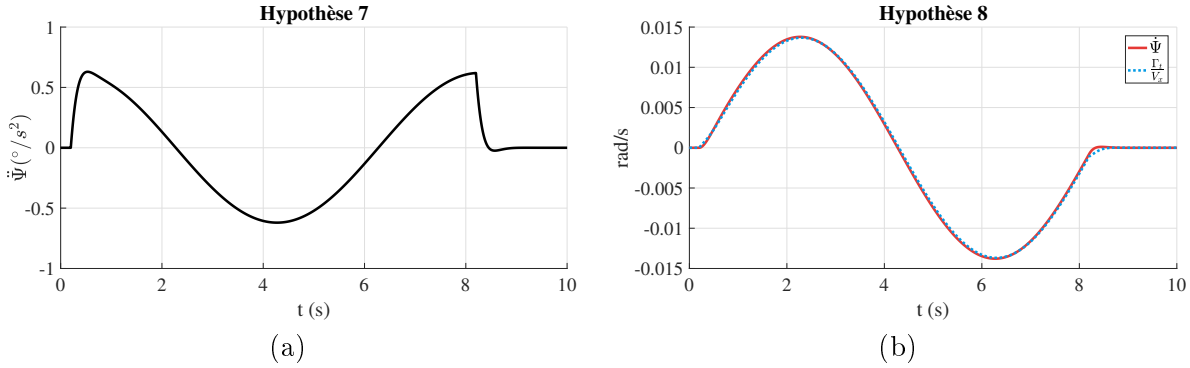


(a) (b)

FIGURE 2.10 – Variation de l'effort latéral du pneumatique en fonction de l'angle de dérive et zone de fonctionnement des pneumatiques (a), zoom (b)

Toutes les hypothèses du modèle linéaire sont validées, c'est pourquoi ce modèle reste fidèle au modèle de validation.

Pour le modèle linéaire circulaire uniforme, les hypothèses (2.40) (hypothèse 7) et (2.41) (hypothèse 8) sont rajoutées aux hypothèses 4, 5 et 6 validées du modèle linéaire. L'hypothèse 7 d'accélération de lacet nulle est représentée par la Figure 2.11 (a) et l'hypothèse 8 d'un lien constant entre la vitesse de lacet et l'accélération latérale par la Figure 2.11 (b).



(a) (b)

FIGURE 2.11 – Accélération de lacet (a) et hypothèse sur la relation entre la vitesse de lacet et l'accélération latérale (b)

Sur la Figure 2.11 (a), l'accélération de lacet est nulle uniquement lorsque l'angle volant est nul ou à son maximum/minimum. L'accélération de lacet varie entre $-0,6°/s^2$ et $0,6°/s^2$ ce qui ne permet pas de valider l'hypothèse. Cependant, sur la Figure 2.11 (b), la vitesse de lacet et son estimation par l'accélération latérale sont très proches avec une erreur inférieure à $6\,10^{-4}$ rad/s et un retard de maximum 0,04 s. L'hypothèse est validée.

Concernant le modèle linéaire en régime circulaire uniforme, comme pour le modèle cinématique, une seule hypothèse n'est pas vérifiée : l'hypothèse 7 ici. Cependant, pour le cas de ce changement de voie, la non-validation de l'hypothèse 1 sur la vitesse de lacet pour le modèle cinématique a beaucoup plus d'impact que l'hypothèse 7 sur l'accélération de lacet du modèle linéaire en régime circulaire uniforme.

2.1.4 Comparaison fréquentielle des modèles de la dynamique latérale du véhicule

À l'aide des équations de la section 2.1.2, il devient possible d'écrire les relations entrées/sorties des modèles sous la forme de fonctions de transfert et de mener ainsi une analyse fréquentielle de la dynamique latérale. L'approche fréquentielle ainsi que les outils utilisés sont présentés en Annexe A Pour tous les modèles, l'entrée est l'angle volant $\theta_v(t)$. Suite à la simulation temporelle, les sorties comparables des modèles sont la position latérale $Y_G(t)$ et la vitesse de lacet $\dot{\Psi}(t)$. Les fonctions de transfert $\dfrac{s\Psi(s)}{\theta_v(s)}$ et $\dfrac{Y_G(s)}{\theta_v(s)}$ sont calculées pour chaque modèle.

Les fonctions de transfert peuvent être calculées de deux manières différentes : soit en utilisant une représentation d'état de type

$$\begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) + D(t)u(t) \end{cases} \tag{2.48}$$

avec

- $x \in \mathbb{R}^n$, le vecteur d'état ;
- $y \in \mathbb{R}^q$, le vecteur de sortie ;
- $u \in \mathbb{R}^p$, le vecteur de commande ;
- $A(t) \in \mathbb{R}^{nxn}$, la matrice d'état ;
- $B(t) \in \mathbb{R}^{nxp}$, la matrice de commande ;

— $C(t) \in \mathbb{R}^{qxn}$, la matrice d'observation ;
— $D(t) \in \mathbb{R}^{qxp}$, la matrice d'action directe,

et la relation

$$H(s) = C(sI_n - A)^{-1}B + D, \tag{2.49}$$

avec $H(s)$ la fonction de transfert du procédé représenté par (2.48). Soit en utilisant directement la transformée de Laplace sur les équations différentielles de la dynamique.

Fonctions de transfert du modèle cinématique

Le modèle cinématique étant simple, il n'est pas nécessaire de passer par une représentation d'état pour trouver les fonctions de transfert. En utilisant la simplification des petits angles vérifiée par la simulation temporelle et la relation entre l'angle de braquage et l'angle volant (2.8), les transformées de Laplace de (2.25) à $V_x(t) = V_{x0}$ constante donnent :

$$\begin{aligned} s\Psi(s) &= \frac{V_{x0}}{\lambda L}\theta_v(s) \\ sY_G(s) &= V_{x0}\Psi(s). \end{aligned} \tag{2.50}$$

Avec (2.50) il est alors possible d'écrire les fonctions de transfert :

$$\boxed{\begin{aligned} G_c^{(1)}(s) &= \frac{s\Psi(s)}{\theta_v(s)} = \frac{V_{x0}}{\lambda L} \\ G_c^{(2)}(s) &= \frac{Y_G(s)}{\theta_v(s)} = \frac{V_{x0}^2}{\lambda L}\frac{1}{s^2} \end{aligned}} \tag{2.51}$$

Fonctions de transfert du modèle linéaire

À partir des équations du modèle linéaire, la dynamique latérale du véhicule peut être décrite par une représentation d'état linéaire. Le vecteur d'état $\underline{x}(t)$ du véhicule se compose de l'angle de lacet, de la vitesse de lacet, de la vitesse latérale et de la position latérale tel que $\underline{x}(t) = \begin{pmatrix} \psi(t) \\ \dot{\psi}(t) \\ V_y(t) \\ Y_G(t) \end{pmatrix}$. L'entrée en commande est l'angle volant $u(t) = \theta_v(t)$. Comme les sorties qui vont être analysées sont la vitesse de lacet et la position latérale, la sortie $\underline{y}(t)$ est définie par $\underline{y}(t) = \begin{pmatrix} \dot{\Psi}(t) \\ Y_G(t) \end{pmatrix}$ et la représentation d'état s'écrit alors :

$$\begin{cases} \dot{\underline{x}}(t) = A_L\underline{x}(t) + B_L u(t) \\ \underline{y}(t) = C_L\underline{x}(t), \end{cases} \tag{2.52}$$

avec

$$A_L = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -2\dfrac{({L_f}^2 c_{yf} + {L_r}^2 c_{yr})}{I_z V_{x0}} & -2\dfrac{L_f c_{yf} - L_r c_{yr}}{I_z V_{x0}} & 0 \\ 0 & -2\dfrac{L_f c_{yf} - L_r c_{yr}}{M_t V_{x0}} - V_{x0} & -2\dfrac{(c_{yf} + c_{yr})}{M_t V_{x0}} & 0 \\ V_{x0} & 0 & 1 & 0 \end{bmatrix},$$

$$B_L = \begin{bmatrix} 0 \\ 2\dfrac{c_{yf} L_f}{\lambda I_z} \\ 2\dfrac{c_{yf}}{\lambda M_t} \\ 0 \end{bmatrix} \text{ et } C_L = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.53}$$

À partir de (2.49), on obtient les fonctions de transfert :

$$G_L(s) = \begin{bmatrix} G_L^{(1)}(s) \\ G_L^{(2)}(s) \end{bmatrix} = \begin{bmatrix} \dfrac{s\Psi(s)}{\theta_v(s)} \\ \dfrac{Y_G(s)}{\theta_v(s)} \end{bmatrix} = C_L(sI_{4\text{x}4} - A_L)^{-1} B_L. \tag{2.54}$$

Elles sont de la forme

$$\boxed{\begin{aligned} G_L^{(1)}(s) &= \frac{s\Psi(s)}{\theta_v(s)} = K_0 \frac{1 + s/\omega_1}{1 + 2\zeta_0(s/\omega_0) + s^2/\omega_0^2} \\ G_L^{(2)}(s) &= \frac{Y_G(s)}{\theta_v(s)} = \frac{K_0'}{s^2} \frac{1 + 2\zeta_1'(s/\omega_1') + s^2/\omega_1'^2}{1 + 2\zeta_0'(s/\omega_0') + s^2/\omega_0'^2}, \end{aligned}} \tag{2.55}$$

avec, pour $G_L^{(1)}(s)$:

$$\begin{aligned} K_0 &= \frac{2}{\lambda} \frac{c_{yf} c_{yr} V_{x0} L}{2 c_{yf} c_{yr} L^2 - M_t {V_{x0}}^2 (L_f c_{yv} - L_r c_{yr})}, \\ \zeta_0 &= \frac{M_t (L_f^2 c_{yf} + L_r^2 c_{yr}) + I_z (c_{yf} + c_{yr})}{\sqrt{2 I_z M_t (2 c_{yf} c_{yr} L^2 - M_t {V_{x0}}^2 (L_f c_{yv} - L_r c_{yr}))}}, \\ \omega_0 &= \sqrt{\frac{2}{I_z M_t {V_{x0}}^2}} \sqrt{2 c_{yf} c_{yr} L^2 - M_t V_{x0}^2 (L_f c_{yv} - L_r c_{yr})}, \\ \omega_1 &= \frac{2 c_{yr} L}{L_f M_t V_{x0}}, \end{aligned} \tag{2.56}$$

et, pour $G_L^{(2)}(s)$

$$
\begin{aligned}
K_0' &= \frac{2}{\lambda}\frac{c_{yf}c_{yr}V_{x0}^2 L}{2c_{yf}c_{yr}L^2 - M_t {V_{x0}}^2(L_f c_{yv} - L_r c_{yr})},\\
\zeta_0' &= \frac{M_t(L_f^2 c_{yf} + L_r^2 c_{yr}) + I_z(c_{yf} + c_{yr})}{\sqrt{2I_z M_t(2c_{yf}c_{yr}L^2 - M_t {V_{x0}}^2(L_f c_{yv} - L_r c_{yr}))}},\\
\omega_0' &= \sqrt{\frac{2}{I_z M_t {V_{x0}}^2}}\sqrt{2c_{yf}c_{yr}L^2 - M_t V_{x0}^2(L_f c_{yv} - L_r c_{yr})},\\
\zeta_1' &= \frac{L_r}{V_{x0}}\sqrt{\frac{c_{yr}L}{2I_z}},\\
\omega_1' &= \sqrt{2\frac{c_{yr}L}{I_z}}.
\end{aligned}
\tag{2.57}
$$

Fonctions de transfert du modèle linéaire en régime circulaire uniforme

Comme pour le modèle cinématique, il est plus rapide de calculer les fonctions de Laplace et d'en déduire les fonctions de transfert. Les fonctions de Laplace déduites de (2.41) et (2.46) sont :

$$
\begin{aligned}
s\Psi(s) &= \frac{\Gamma_t(s)}{V_{x0}}\\
\beta_v(s) &= \left[M_t\left(\frac{L_r c_{yr} - L_f c_{yf}}{2Lc_{yr}c_{yf}}\right) + \frac{L}{{V_{x0}}^2}\right]\Gamma_t(s)\\
sY_G(s) &= V_{x0}\Psi(s).
\end{aligned}
\tag{2.58}
$$

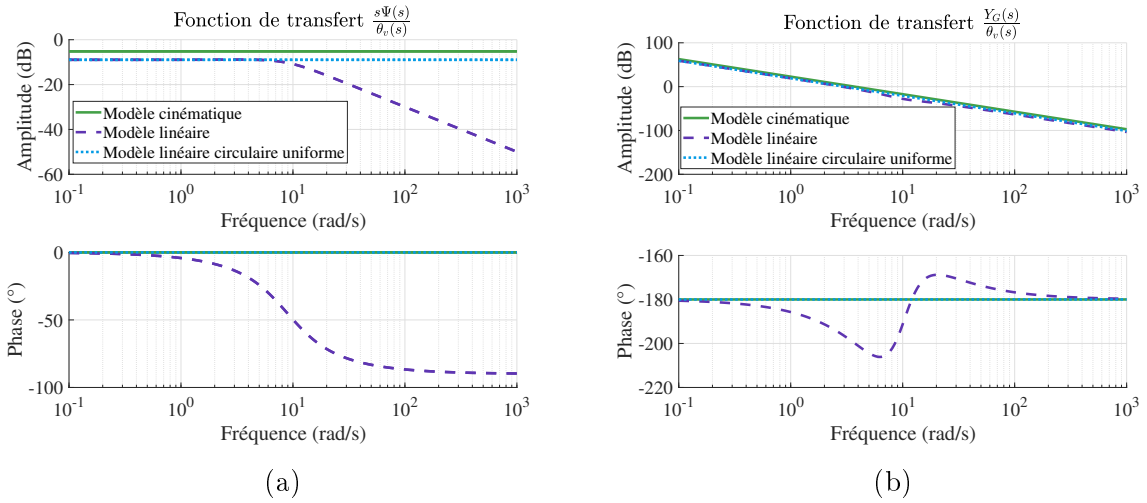
$$
\begin{aligned}
G_{LU}^{(1)}(s) &= \frac{s\Psi(s)}{\theta_v(s)} = \frac{2}{\lambda}\frac{c_{yf}c_{yr}V_{x0}L}{2c_{yf}c_{yr}L^2 - M_t {V_{x0}}^2(L_f c_{yv} - L_r c_{yr})} = K_0\\
G_{LU}^{(2)}(s) &= \frac{Y_G(s)}{\theta_v(s)} = \frac{1}{s^2}\frac{2}{\lambda}\frac{c_{yf}c_{yr}{V_{x0}}^2 L}{2c_{yf}c_{yr}L^2 - M_t {V_{x0}}^2(L_f c_{yv} - L_r c_{yr})} = \frac{K_0'}{s^2}
\end{aligned}
\tag{2.59}
$$

Comparaison des diagrammes de Bode

Les diagrammes de Bode des fonctions de transfert calculées précédemment sont tracés Figure 2.12.

La dualité temps-fréquence permet de retrouver les mêmes observations que la simulation temporelle. La première observation est que, pour la vitesse de lacet Figure 2.12 (a) ou la position latérale Figure 2.12 (b), les amplitudes en basses fréquences du modèle cinématique sont supérieures à celles du modèle linéaire qui se rapproche le plus du modèle de validation. Cela explique donc le comportement temporel du modèle cinématique où les variables de position latérale et de vitesse de lacet ont des valeurs plus grandes qu'attendu.

La deuxième observation est que le modèle linéaire en régime circulaire uniforme reste très proche du modèle linéaire et du modèle de validation. En effet, les fonctions de transfert $G_L^{(1)}(s)$



(a) (b)

FIGURE 2.12 – Diagrammes de Bode des fonctions de transfert $\dfrac{s\Psi(s)}{\theta_v(s)}$ (a) et $\dfrac{Y_G(s)}{\theta_v(s)}$ (b)

et $G_{LU}^{(1)}(s)$ et les fonctions de transfert $G_L^{(2)}(s)$ et $G_{LU}^{(2)}(s)$, (2.55) et (2.59), ont les mêmes gains. Par contre, les fonctions de transfert représentant le modèle linéaire en régime circulaire uniforme ne possèdent pas les fractions du second ordre avec les fréquences propres des fonctions de transfert. Ces fréquences propres modulent l'évolution dynamique de la position latérale et de la vitesse de lacet ; c'est pourquoi on observe que la position latérale du véhicule et la vitesse de lacet du modèle linéaire en régime circulaire uniforme sont en avance par rapport au modèle de validation Figures 2.5 (a) et 2.6 (a).

Pour le scénario proposé dans cette section, le modèle linéaire est celui qui se rapproche le plus du modèle de validation. En supposant un régime circulaire uniforme, la modélisation reste représentative, mais présente un léger retard. En revanche, le modèle cinématique n'est pas un bon choix de modèle pour le changement de voie à 90 km/h. D'ailleurs, ce modèle est surtout utilisé à basse vitesse. Comme la vitesse longitudinale est une composante importante de la dynamique latérale, il est nécessaire d'évaluer son impact sur cette dynamique.

2.2 Mise en évidence de l'influence de la vitesse longitudinale

2.2.1 Description du scénario

Le même scénario de changement de voie est utilisé afin d'analyser l'influence de la vitesse longitudinale sur la dynamique latérale. Ce changement de voie est donc effectué pour différentes vitesses : 30 km/h, 50 km/h, 70 km/h, 90 km/h, 110 km/h, 130 km/h qui correspondent aux limitations de vitesse rencontrées usuellement, plus un cas très basse vitesse à 10 km/h pour représenter les embouteillages.

Afin d'avoir une comparaison pertinente entre les différentes vitesses, le chemin à suivre est le même. Ce chemin débute par une ligne droite de 5 m, puis effectue un changement de voie sur 200 m. L'amplitude de l'angle volant est calculée pour qu'au bout de ces 200 m, le véhicule se soit déplacé latéralement de 3,5 m. L'allure de la trajectoire désirée est tracée à la Figure 2.3. De plus, le véhicule est déjà à la vitesse cible au début de la simulation ($X_G = 0$m).

Vu qu'il n'y a aucune régulation latérale pour cette simulation, les trajectoires du véhicule ne sont pas exactement les mêmes pour chaque vitesse et l'entrée en angle volant pour effectuer cette trajectoire doit être bien choisie. Comme pour la simulation de la section 2.1.3, l'angle volant choisi est une période d'un sinus. La période T_{θ_v} de ce sinus est calculée avec (2.47) de sorte que le changement de voie dure 200 m avec une amplitude permettant d'atteindre les 3,5 m latéraux au bout des 200 m pour le modèle de validation. Avec la période et donc la fréquence du sinus connue pour chaque vitesse, il est important de vérifier que celles-ci sont conformes à une fréquence de rotation admissible pour un conducteur. Les périodes et fréquences de l'angle volant sont répertoriées dans la Table 2.2. Ce sont ces mêmes angles volant, calculés pour le modèle de validation à différentes vitesses V_{x0}, qui sont utilisés en entrée des simulations de changement de voie des modèles simplifiés afin d'analyser l'écart entre les modèles.

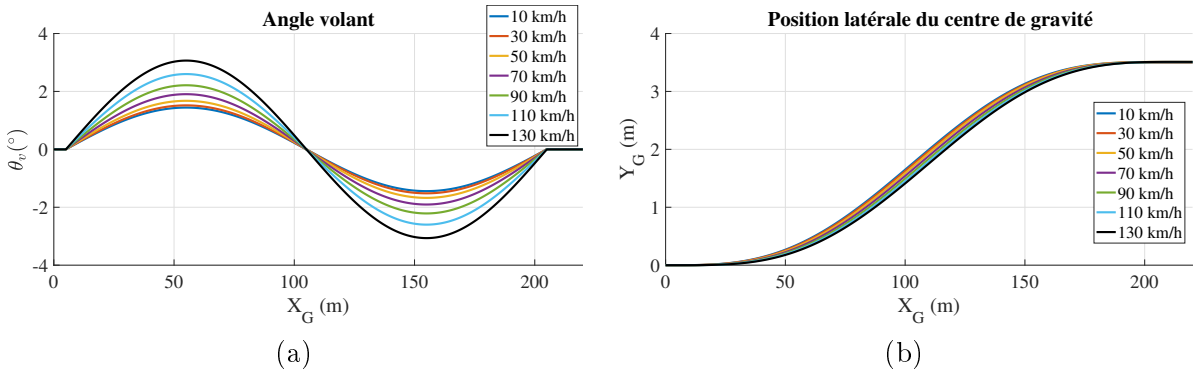
TABLE 2.2 – Périodes et fréquences des entrées en angles volant

V_{x0} (km/h)	T_{θ_v} (s)	$F_{\theta_v}(Hz)$
10	72	0,014
30	24	0,042
50	14,4	0,07
70	10,3	0,10
90	8	0,13
110	6,5	0,15
130	5,5	0,18

En moyenne, un conducteur tourne le volant au maximum à 1,5 Hz, avec une fréquence souvent inférieure à 0,5 Hz [31]. Les valeurs d'angle volant utilisés pour les simulations sont donc conformes à la réalité.

2.2.2 Résultats de la simulation pour le modèle de validation

Les entrées en angle volant et les trajectoires en sortie sont tracées pour chaque vitesse sur la Figure 2.13. Afin d'avoir les signaux des simulations à différentes vitesses sur la même figure, la position longitudinale est utilisée en abscisse des figures.

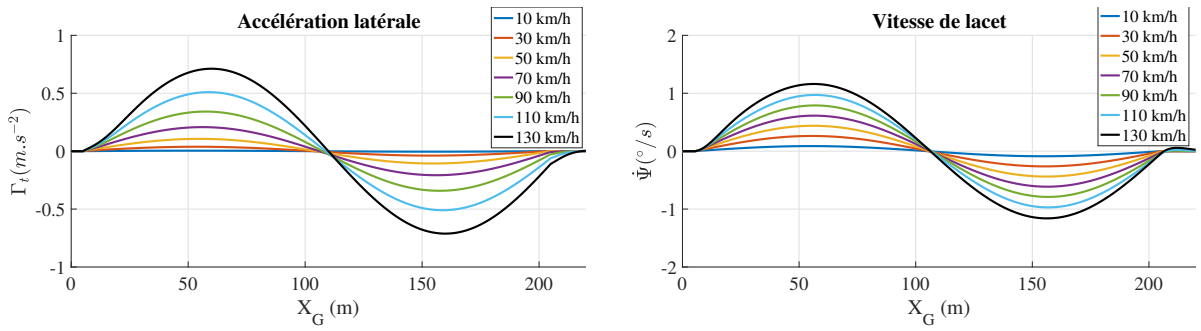


(a) (b)

FIGURE 2.13 – Entrée angle volant (a) et trajectoire (b) pour les différentes vitesses

Les angles volant respectent tous le comportement conducteur, mais il est aussi important de regarder l'accélération latérale. En effet, pour un véhicule particulier, l'accélération latérale ne doit pas excéder les 2 m.s^{-2} pour garantir un bon confort. Dans le cas de cette simulation,

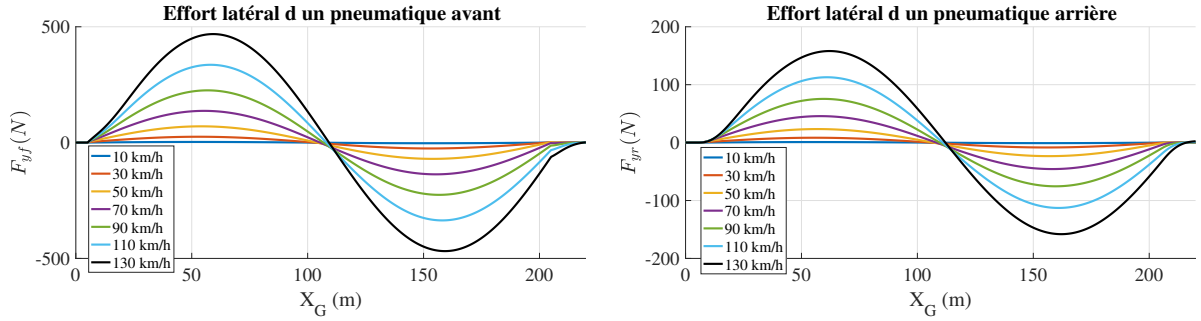
l'accélération latérale est inférieure à 0,75 m.s^{-2} et est donc réaliste.



(a) (b)

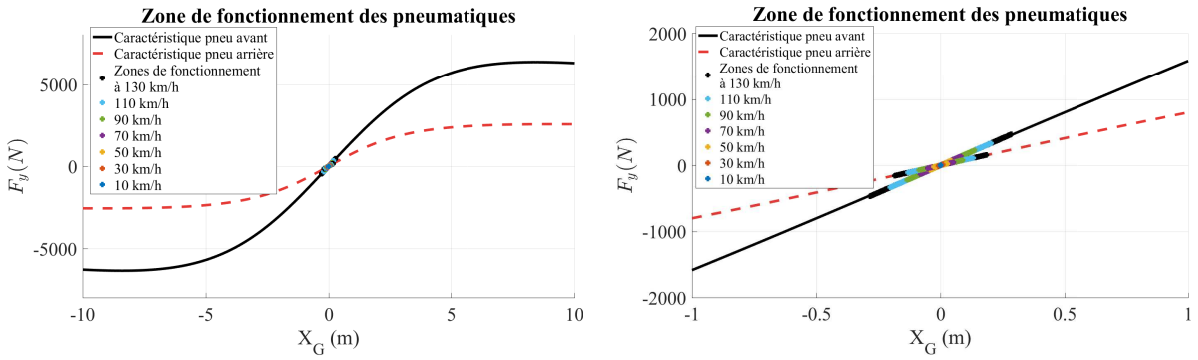
FIGURE 2.14 – Variation de l'accélération latérale (a) et de la vitesse de lacet (b) en fonction de $X_G(t)$

Étant donné que les pneumatiques du véhicule ont un impact important sur les dynamiques longitudinales comme transversales, il est important d'analyser les efforts résultants de ces pneumatiques. La Figure 2.15 représente la variation de ces efforts pour les pneumatiques avant (a) et arrière (b) en fonction de la position longitudinale pour les différentes vitesses. La Figure 2.16, quant à elle, trace ces efforts en fonction de la dérive du pneumatique considéré. Sur cette figure, les courbes en bleu et en rouge pointillé représentent respectivement les caractéristiques des pneumatiques arrière et avant des efforts transversaux en fonction de l'angle volant et les croix de différentes couleurs les points de la caractéristique atteints pendant la simulation, chaque couleur correspondant à une vitesse.



(a) (b)

FIGURE 2.15 – Variation des efforts latéraux appliqués à un pneumatique avant (a) et un pneumatique arrière (b) en fonction de $X_G(t)$

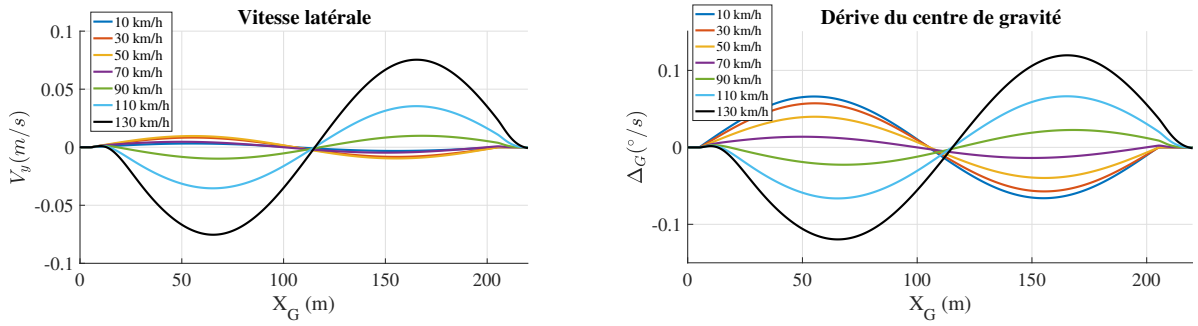


(a) (b)

FIGURE 2.16 – Variation des efforts latéraux appliqués aux pneumatiques en fonction de l'angle de dérive (a) et un zoom autour de la zone linéaire de fonctionnement des pneumatiques (b)

Dans cette situation et pour toutes les vitesses, les efforts transversaux des pneumatiques sont dans la zone linéaire des pneumatiques, Figure 2.16 (a). Cette observation est cruciale pour la suite de l'étude et notamment pour la linéarisation des équations de la dynamique présentées précédemment.

Il est intéressant de noter un changement dans le comportement de l'angle de dérive du véhicule en fonction de la vitesse, tracé sur la Figure. 2.17.



(a) (b)

FIGURE 2.17 – Variation de la vitesse latérale (a) et de la dérive au centre de gravité (b) en fonction de $X_G(t)$

Pour une même vitesse, la vitesse latérale Figure. 2.17 (a) et la dérive du centre de gravité Figure 2.17 (b) ont la même forme vu que V_{x0} est constant, mais n'ont pas la même amplitude. Pour $10\,\text{km/h} \leq V_{x0} \leq 70\,\text{km/h}$, à $\theta_v(t)$ maximal, les amplitudes de $V_y(t)$ et $\Delta_G(t)$ décroissent quand la vitesse augmente. Alors que pour $90\,\text{km/h} \leq V_{x0} \leq 130\,\text{km/h}$ leurs amplitudes augmentent. De plus, les signes de $V_y(t)$ et $\Delta_G(t)$ s'inversent lorsque V_{x0} devient supérieur à 90 km/h.

Afin de visualiser l'impact du changement de signe de la vitesse latérale sur le véhicule, l'allure des vecteurs vitesse est tracée à basse et haute vitesses pour une même trajectoire sur la Figure 2.18. Il est important de noter que ces vecteurs ne sont pas à l'échelle. En effet, comme à toutes les vitesses $V_{x0} >> V_y$, tracer ces vecteurs proportionnellement à leurs valeurs réelles n'aurait pas rendu le vecteur de la vitesse latérale de norme V_y visible.

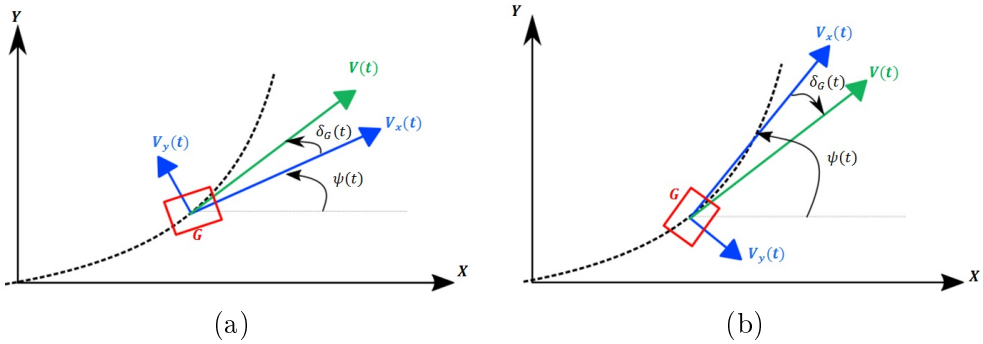


FIGURE 2.18 – Schémas des vecteurs vitesse à basse vitesse (a) et à haute vitesse (b)

À petite vitesse, le vecteur de la vitesse latérale pointe vers le centre de rotation instantané. Au contraire, plus la vitesse augmente et plus ce vecteur se dirige en sens inverse, attirant le véhicule à l'extérieur de la trajectoire.

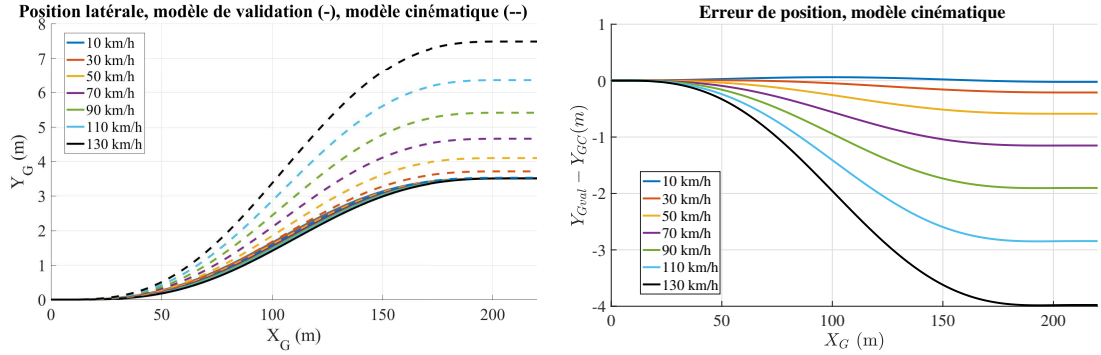
2.2.3 Comparaison temporelle avec les modèles simplifiés

De la même manière que pour la simulation de changement de voie à 90 km/h, les variables temporelles de position latérale et de vitesse de lacet des différents modèles sont comparées pour des changements de voie à différentes vitesses. Pour ne pas surcharger les figures avec les simulations des 4 modèles pour 7 vitesses, chaque modèle simplifié est comparé séparément au modèle de validation non linéaire qui est le modèle de référence.

Comparaison avec le modèle cinématique

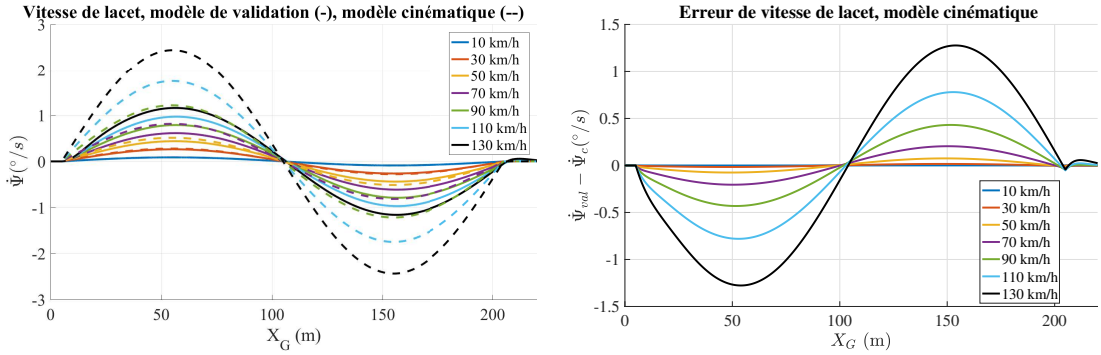
Précédemment, nous avons vu que l'hypothèse géométrique sur la vitesse de lacet ne permettait pas une représentation fidèle de la dynamique latérale à 90 km/h. En changeant la vitesse de changement de voie, il apparaît que plus la vitesse longitudinale diminue, plus l'erreur de vitesse de lacet, comparée au modèle de validation, diminue, Figure 2.20.

Cette constatation peut aussi être faite sur la position latérale où plus la vitesse augmente et plus la position latérale en régime permanent du modèle géométrique augmente. Alors qu'à 10 km/h, la position latérale finale est à 3,52 m correspondant à une erreur relative de 0,6 %, à 130 km/h, la position finale est de 7,49 m et l'erreur relative de 114 %, voir Figure 2.19.



(a) (b)

FIGURE 2.19 – Variations de la position latérale des modèles de validation (trait plein) et cinématique (trait pointillé) (a), et de l'erreur de position latérale du modèle cinématique par rapport au modèle de validation (b) pour différentes vitesses



(a) (b)

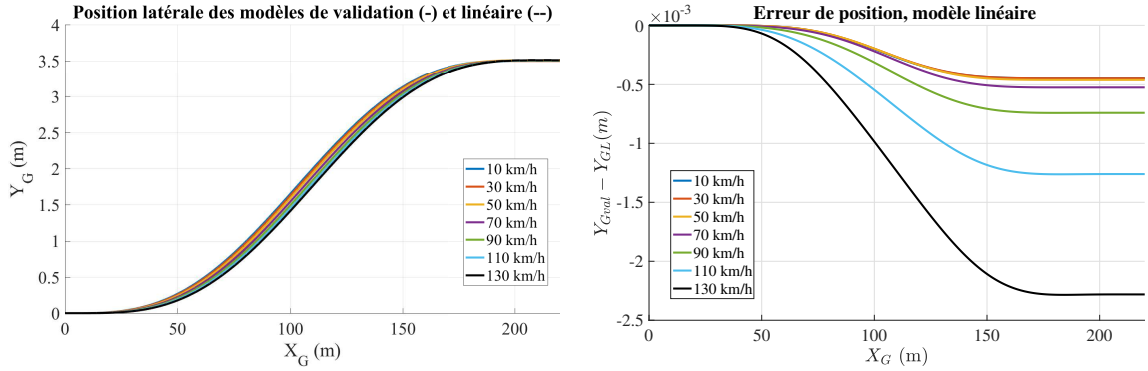
FIGURE 2.20 – Variations de la vitesse de lacet des modèles de validation (trait plein) et cinématique (trait pointillé) (a), et de l'erreur de vitesse de lacet du modèle cinématique par rapport au modèle de validation (b) pour différentes vitesses

Ces simulations mettent bien en évidence le fait que le modèle géométrique n'est pas utilisable pour des vitesses longitudinales supérieures à 50 km/h. À partir de cette vitesse, l'erreur relative entre le modèle cinématique et le modèle de validation est supérieure à 15 %. L'analyse fréquentielle de la section 2.2.4 permettra une analyse plus fine du modèle en fonction de la vitesse.

Comparaison avec le modèle linéaire

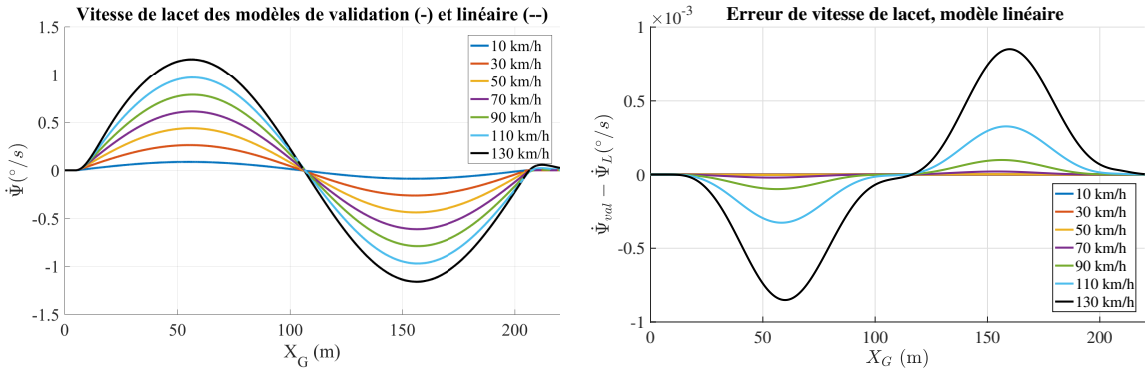
De tous les modèles simplifiés, à 90 km/h, le modèle linéaire est celui qui se rapproche le plus du modèle de validation. Même en changeant de vitesse, le modèle reste très représentatif de la dynamique latérale comme on peut le voir sur les Figures 2.21 et 2.22. En effet, l'erreur en position en régime permanent est inférieure à 2,5 mm (0,7 %) et celle en vitesse de lacet inférieure à 1.10^{-3} $°/s$ (0,8 %).

Ceci s'explique par le fait que, dans le cadre de ce changement de voie et peu importe la vitesse, les pneumatiques restent dans leur zone linéaire, Figure 2.16, et les angles de braquage restent faibles.



(a) (b)

FIGURE 2.21 – Variations de la position latérale des modèles de validation (trait plein) et linéaire (trait pointillé) (a), et de l'erreur de position latérale du modèle linéaire par rapport au modèle de validation (b) pour différentes vitesses



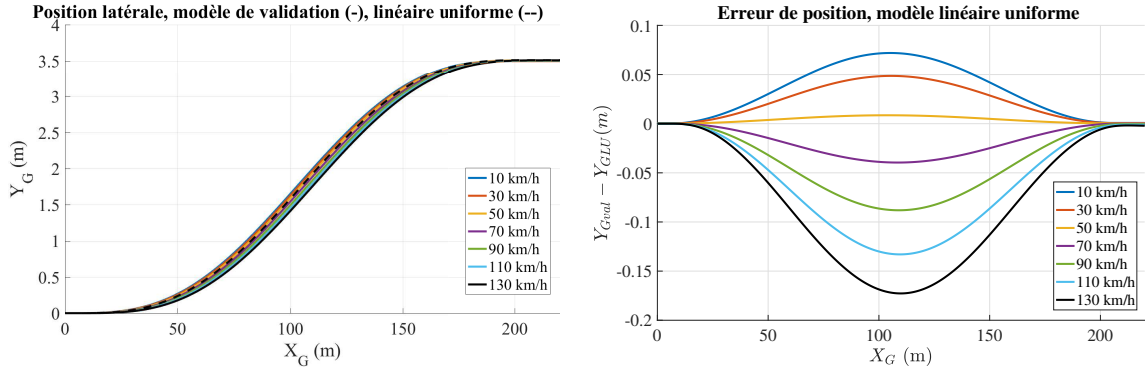
(a) (b)

FIGURE 2.22 – Variations de la vitesse de lacet des modèles de validation (trait plein) et linéaire (trait pointillé) (a), et de l'erreur de vitesse de lacet du modèle linéaire par rapport au modèle de validation (b) pour différentes vitesses

Comparaison avec le modèle linéaire en régime circulaire uniforme

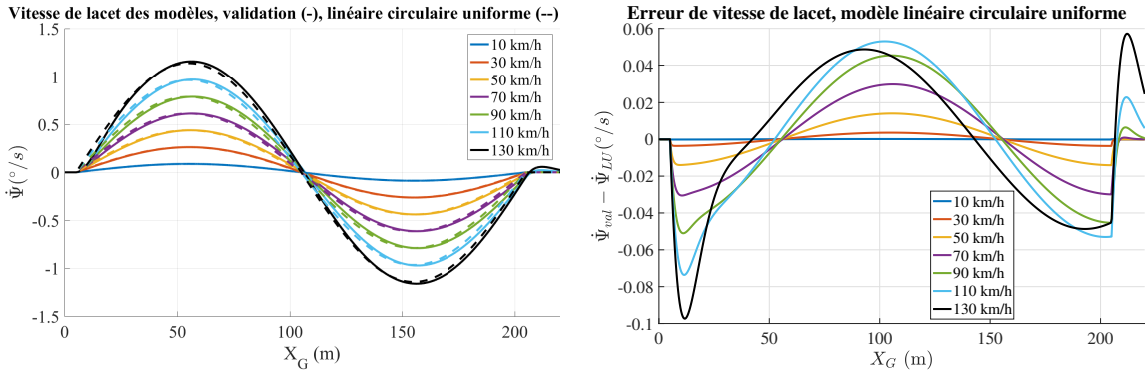
En régime permanent, la position latérale du modèle linéaire en régime circulaire uniforme est conforme à celle du modèle de validation (voir Figure 2.23 (a)). Cependant, il existe un retard dans la dynamique de la position latérale. Ce retard est notamment visible sur la Figure 2.23 (b) représentant l'erreur de position.

La vitesse de lacet du modèle linéaire en régime circulaire uniforme est aussi très proche du modèle de validation avec des amplitudes maximales qui correspondent au modèle non linéaire, Figure 2.24 (a). Le phénomène de déphasage se retrouve aussi sur la vitesse de lacet, Figure 2.24 (b).



(a) (b)

FIGURE 2.23 – Variations de la position latérale des modèles de validation (trait plein) et linéaire uniforme (trait pointillé) (a), et de l'erreur de position latérale du modèle linéaire uniforme par rapport au modèle de validation (b) pour différentes vitesses



(a) (b)

FIGURE 2.24 – Variations de la vitesse de lacet des modèles de validation (trait plein) et linéaire uniforme (trait pointillé) (a), et de l'erreur de vitesse de lacet du modèle linéaire uniforme par rapport au modèle de validation (b) pour différentes vitesses

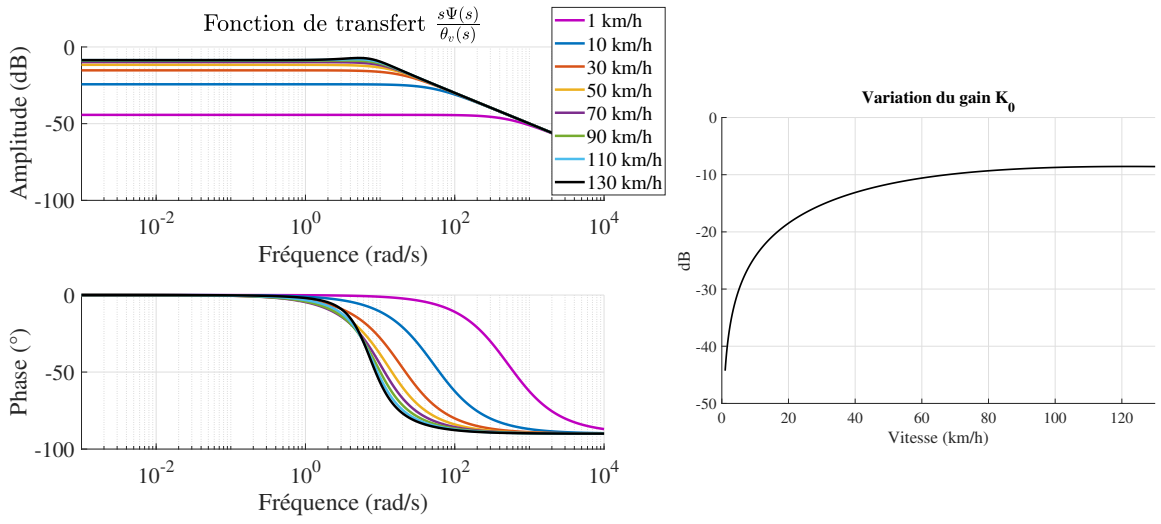
2.2.4 Analyse fréquentielle du modèle linéaire

L'analyse fréquentielle va permettre d'analyser plus finement l'influence de la vitesse longitudinale sur la dynamique latérale. De plus, cette analyse est essentielle pour la conception du régulateur latéral et servira donc de base au Chapitre 3.

Analyse de la fonction de transfert entre θ_v et $\dot{\Psi}$

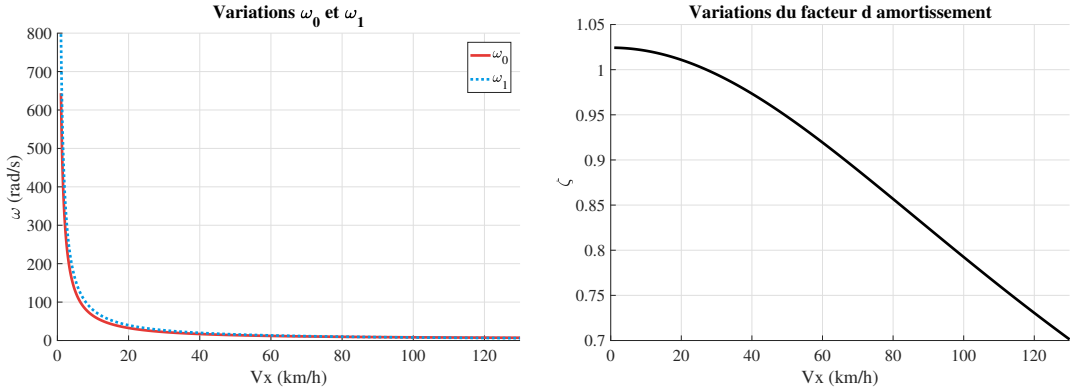
Les diagrammes de Bode de la fonction de transfert $G_L^{(1)}$, décrite par l'équation (2.55), pour différentes vitesses sont tracés sur la Figure 2.25 (a). Sur cette figure, l'influence de la vitesse sur le gain en basse fréquence et sur la fréquence propre du système est bien visible : plus la vitesse augmente et plus le gain augmente et la fréquence de coupure diminue.

Les variations de ces paramètres, exprimés par (2.56), ainsi que celle du facteur d'amortissement ζ_0 en fonction de la vitesse sont tracées sur les Figures 2.25 (b) et 2.26.



(a) (b)

FIGURE 2.25 – Diagrammes de Bode (a) et gain (b) de $G_L^{(1)}(s)$ pour différentes vitesses V_x



(a) (b)

FIGURE 2.26 – Variations des pulsations ω_0 et ω_1 (a) et du facteur d'amortissement (b)

C'est surtout à très basse vitesse, lorsque $V_x < 20$ km/h, que la vitesse a le plus grand impact sur la dynamique latérale et que les variations des paramètres de la fonction de transfert sont les plus importantes.

Analyse de la fonction de transfert entre θ_v et Y_G

La fonction de transfert $G_L^{(2)}(s)$, décrite par l'équation (2.55), est tracée à l'aide de diagrammes de Bode sur la Figure 2.27 pour différentes valeurs de V_{x0}. Ces réponses fréquentielles du modèle linéaire confirment que la vitesse longitudinale influe de manière importante sur la dynamique du véhicule avec environ 78 dB de variation de gain et 138° de variation de phase, avec un maximum atteint à 7 rad/s.

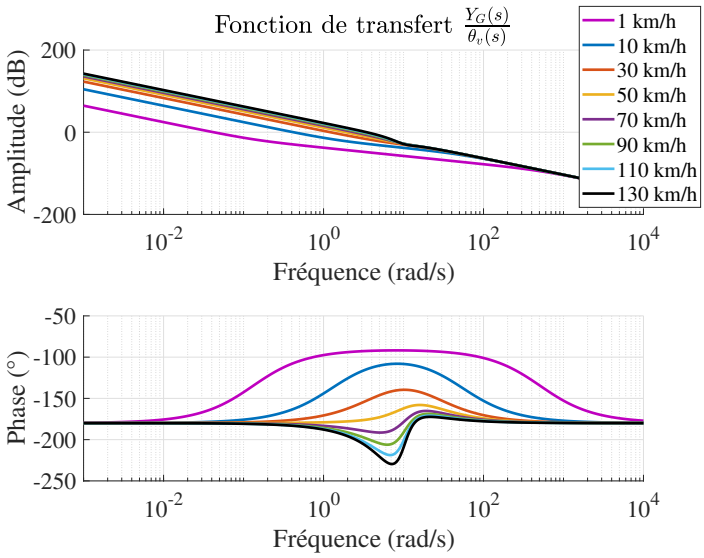
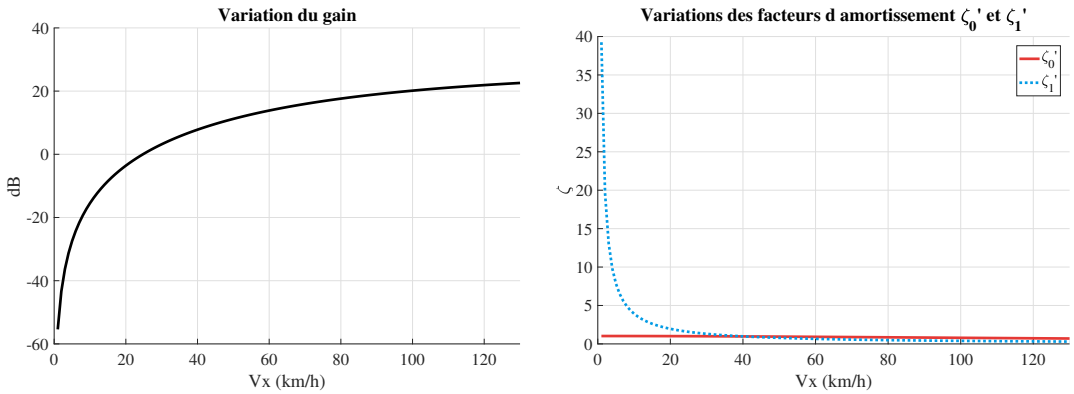


FIGURE 2.27 – Diagrammes de Bode de $G_L^{(2)}(s)$ pour différentes vitesses V_x

La vitesse longitudinale affecte aussi le comportement fréquentiel de $G_L^{(2)}(s)$ avec un comportement à avance de phase à basse vitesse et un comportement à retard de phase à haute vitesse. De plus, il est important de noter que pour une fréquence donnée (par exemple 7 rad/s), et pour un intervalle de vitesse fixe, la variation de phase est plus importante à basse vitesse (environ 92.5° pour une vitesse comprise entre 1 et 65 km/h) qu'à haute vitesse (environ 40° pour une vitesse comprise entre 65 et 130 km/h).



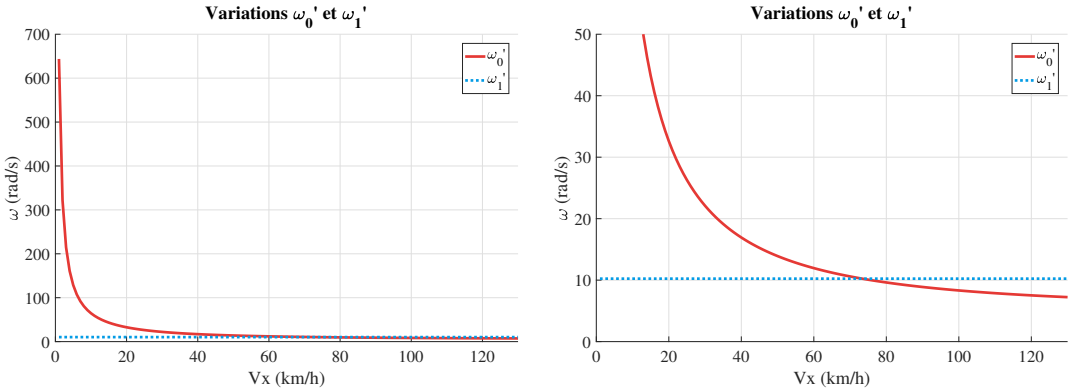
(a) (b)

FIGURE 2.28 – Variation du gain K_0' (a) et des facteurs d'amortissement ζ_0' et ζ_1' (b) en fonction de la vitesse longitudinale

Mis à part ω_1', tous les paramètres de $G_L^{(2)}(s)$, exprimés par (2.57), dépendent de V_{x0}. Les variations de ces paramètres en fonction de la vitesse longitudinale sont tracées sur les Figures 2.28 et 2.29.

On peut remarquer sur la Figure 2.29 que la fréquence ω_0' devient inférieure à ω_1' pour des vitesses supérieures à 65 km/h, ce qui explique le changement de comportement entre un système

à avance de phase à basse vitesse et un système à retard de phase à haute vitesse de la Figure 2.27. Le facteur d'amortissement ζ_0 peut être observé sur la Figure 2.28 (b), elle montre que le système est sous amorti lorsque $V_{x0} > 30$ km/h et amorti lorsque $V_{x0} < 27$ km/h.



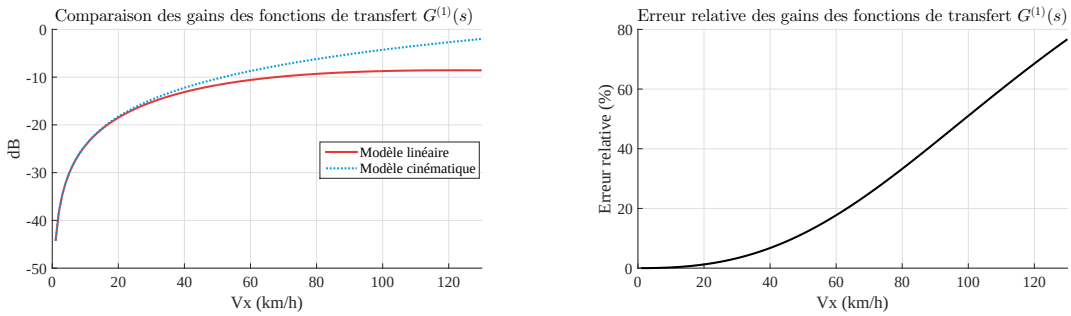
(a) (b)

FIGURE 2.29 – Variations des pulsations ω_0' et ω_1' (a) en fonction de la vitesse longitudinale et un zoom (b)

2.2.5 Comparaison fréquentielle avec les autres modèles simplifiés

Les fonctions de transfert du modèle linéarisé (2.55) sont celles qui représentent le mieux la réalité. Les fonctions de transfert du modèle linéaire en régime circulaire uniforme (2.59) ont les mêmes gains K_0 et K_0' et le même degré. Cependant, comme expliqué précédemment, les fonctions de transfert en régime circulaire uniforme (2.59) ne présentent pas d'évolutions fréquentielles engendrant des déphasages temporels.

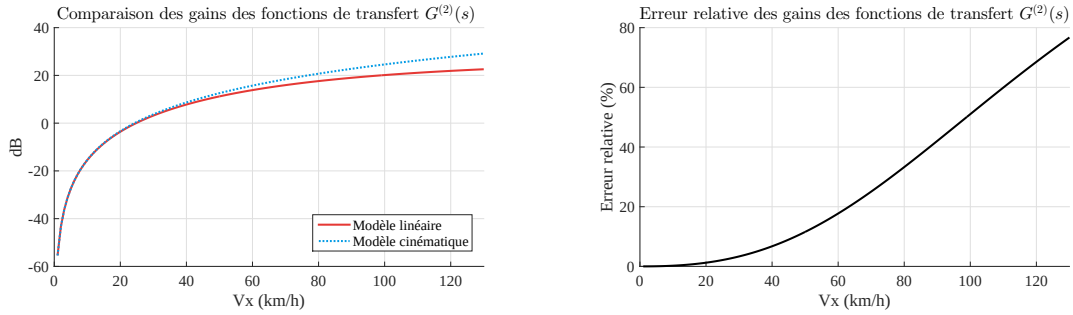
Le modèle cinématique (2.51) garde en représentation fréquentielle, le même degré que le modèle linéaire (2.55), mais pas les mêmes gains. Cette différence de gains s'est ressentie sur les simulations temporelles avec une approximation du modèle géométrique qui se dégrade en même temps que la vitesse augmente.



(a) (b)

FIGURE 2.30 – Variation du gain des fonctions de transfert $G^{(1)}(s)$ en fonction de la vitesse et de l'erreur relative (b)

Afin d'analyser exactement l'influence de la vitesse sur les modèles linéaire et cinématique les variations des gains des fonctions de transfert $G^{(1)}(s)$ et $G^{(2)}(s)$ en fonction de la vitesse longitudinale sont tracées, respectivement, sur les Figures 2.30 (a) et 2.31 (a). Les erreurs relatives des gains des fonctions de transfert cinématique par rapport au modèle linéaire en fonction de la vitesse sont représentés 2.30 (b) et 2.31 (b).



(a) (b)

FIGURE 2.31 – Variation du gain des fonctions de transfert $G^{(2)}(s)$ en fonction de la vitesse et erreur relative (b)

La dualité temps-fréquence fait que l'on observe bien à basse vitesse que les gains des modèles cinématique et linéaire sont très proches, mais que plus la vitesse augmente et plus les modèles diffèrent. Pour les deux fonctions de transfert $G^{(1)}(s)$ et $G^{(2)}(s)$, l'erreur relative est supérieure à 5 % lorsque $V_x > 36$ km/h et supérieur à 10 % lorsque $V_x > 47$ km/h. On peut aussi remarquer qu'à partir de 60 km/h, l'erreur relative des gains est linéaire en fonction de la vitesse.

Remarque. La vitesse longitudinale n'est pas le seul paramètre influençant la dynamique latérale du véhicule. La masse et l'adhérence de la route interviennent aussi dans les paramètres des fonctions de transfert. Ainsi, l'Annexe B analyse l'influence de ces deux paramètres sur la dynamique latérale du véhicule.

2.3 Observateur d'état des systèmes dynamiques : état de l'art

Pour réaliser les algorithmes des systèmes d'aide à la conduite, un certain nombre de variables internes de la dynamique du véhicule sont nécessaires. Cependant, ces variables ne sont pas toujours mesurées par des capteurs ou les mesures peuvent être de mauvaise qualité. Afin d'avoir une bonne estimation des variables de la dynamique du véhicule, des observateurs peuvent être utilisés. Cette section présente un état de l'art des observateurs en fonction du type de modèle dynamique utilisé et la section 2.4 présente deux observateurs développés pour l'estimation de la dynamique latérale du véhicule et plus précisément de la vitesse de lacet.

L'estimation d'état d'un système dynamique consiste à calculer les variables internes non mesurables du système à l'aide d'un modèle dynamique. Comparé à un estimateur, un observateur utilise une ou plusieurs informations mesurées par des capteurs pour corriger l'estimation. Un estimateur se calcule donc en boucle ouverte et un observateur en boucle fermée, comme le montre la Figure 2.32.

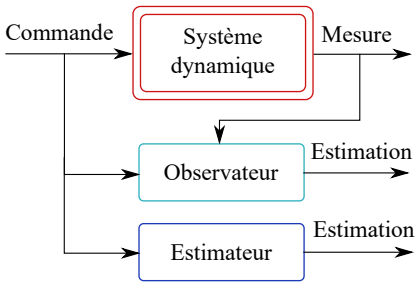


FIGURE 2.32 – Schéma bloc d'un observateur et d'un estimateur

Pour pouvoir estimer une variable d'état d'un système, il faut que celui-ci soit observable. Un système est dit observable si, à partir de la connaissance de l'entrée et de la sortie à l'instant t, il est possible de calculer l'état initial du système.

Nous nous intéressons ici aux observateurs d'état des systèmes dynamiques. Les observateurs ont été beaucoup étudiés et peuvent être décomposés en deux classes : les observateurs de système à modèle linéaire et les observateurs de système à modèle non linéaire. En effet, le choix de l'observateur se fait en fonction du modèle décrivant l'évolution dynamique du procédé.

Dans le modèle non linéaire de la dynamique latérale présenté en début de chapitre au paragraphe 2.1.1, plusieurs variables internes de la dynamique du véhicule ne sont pas directement mesurables par des capteurs. C'est notamment le cas de l'angle de dérive du centre de gravité, des angles de dérive des pneumatiques ou des efforts des pneumatiques.

Pour avoir une estimation des variables de la dynamique latérale du véhicule, de nombreuses méthodes ont été utilisées dans la littérature. Ces méthodes diffèrent en fonction du type d'observateur, du modèle de la dynamique latérale considéré, des variables observées et des capteurs utilisés. L'Annexe C présente différents types d'observateurs en fonction du modèle dynamique du système.

2.3.1 Les méthodes

Toutes les méthodes présentées en Annexe C ont été utilisées pour l'estimation des variables de la dynamique latérale, à commencer par les observateurs de Luenberger (section C.1). Dans [33] et [100], une matrice de gain sur l'erreur entre l'estimation et la mesure sert à l'observation de la vitesse latérale. Dans [199], trois observateurs de Luenberger sont utilisés : un sur l'accélération longitudinale, un sur l'accélération latérale et un sur la vitesse de lacet, pour aussi estimer la vitesse latérale du véhicule. Dans [198], des inégalités matricielles permettent de calculer les gains des observateurs tout en tenant compte des incertitudes.

Un filtre de Kalman (section C.1) est utilisé pour l'estimation de la vitesse de lacet dans [53]. La version étendue du filtre de Kalman (section C.2) sert de modèle dans [116], [8] et [45], et la version « unscented » ([89]) dans [161] et [45]. En cas de fortes non-linéarités, pour éviter les erreurs d'estimation liées à la linéarisation du filtre de Kalman étendu, le filtre de Kalman « unscented » utilise une version mixte entre les filtres à particule [41] et les filtres de Kalman. [189] fait la comparaison de ces deux modèles pour l'estimation de l'angle de dérive du centre de gravité. Les deux méthodes donnent des résultats similaires, avec une précision légèrement supérieure pour le filtre de Kalman « unscented ».

Il existe enfin une autre méthode, qui permet de se passer de modèle : les réseaux de neurones

[125, 49]. À partir de mesures, le réseau va être entraîné de manière à représenter la dynamique latérale du véhicule. Dit autrement, le réseau permet une identification de la dynamique latérale sans modèle a priori. Un réseau de neurones pouvant représenter le comportement de n'importe quelle fonction non linéaire, il pourra donc, s'il est bien entraîné, être un modèle représentatif de la dynamique latérale avec moins de complexité calculatoire qu'un modèle non linéaire. Le problème des réseaux de neurones est qu'ils sont sensibles aux incertitudes paramétriques du véhicule et aux variations externes de l'environnement s'ils n'ont pas été entraînés dans ces situations. La base de données doit donc être exhaustive.

Les réseaux de neurones ont aussi été utilisés en combinaison de filtres de Kalman [12, 163] pour améliorer la qualité des estimations. Dans [47] et [28], des inférences de logique floue sont ajoutées au filtre de Kalman étendu dans ce même objectif.

2.3.2 Les modèles

En fonction de la méthode utilisée et des variables à observer, les modèles utilisés pour l'observation de la dynamique latérale ne sont pas les mêmes. Normalement, et comme nous l'avons vu au début du chapitre, la modélisation de la dynamique latérale est non linéaire. Cependant, pour des raisons de complexité calculatoire et de difficulté de modélisation des pneumatiques, des modèles simplifiés peuvent être utilisés. Dans l'ensemble, il y a deux grandes familles de modèles : les modèles dynamiques et les modèles cinématiques. Dans [115], [199] et [161], des modèles dynamiques non linéaires sont utilisés et dans [103] le modèle non linéaire dynamique est comparé au modèle cinématique. Plusieurs papiers [116, 53, 137], utilisent à la fois un modèle dynamique et un modèle cinématique. Ainsi, dans [53], le modèle cinématique est utilisé comme capteur virtuel et le modèle dynamique pour la synthèse de l'observateur de Kalman. Par contre, dans [137], les modèles cinématique et dynamique sont utilisés séparément puis une synthèse est effectuée pour l'estimation de la vitesse latérale.

La complexité des modèles varie aussi énormément selon les publications. Des modèles linéarisés sont utilisés dans [33] et [170] alors que [189] utilise un modèle horizontal et vertical prenant en compte le roulis. Dans [45], chacune des 4 roues du véhicule est modélisée. Les modèles linéaires fonctionnent pour un point de fonctionnement spécifique et ne sont donc pas adaptés lorsque l'angle volant devient trop grand par exemple. Ainsi, un observateur adaptatif dépendant du frottement pneumatique/sol est utilisé dans [68].

Le panel est donc très large et dépend de l'utilisation qui en est faite.

2.3.3 Les variables observées

Globalement, il existe des recherches pour l'estimation de toutes les variables internes de la dynamique latérale du véhicule. L'observation de l'angle de dérive du centre de gravité fait l'objet de nombreux papiers [116, 125, 115, 189, 198, 170, 32] car n'est pas mesurable directement. Les vitesses latérale et de lacet sont aussi estimées dans [199, 137, 33] pour l'une et [53, 33] pour l'autre, car les mesures des capteurs sont souvent bruitées ou préfiltrées par les calculateurs qui mettent l'information à disposition, ce qui les rend difficilement exploitables. Enfin, il y a de nombreuses recherches sur l'estimation des variables liées aux pneumatiques. En effet, ces variables ne sont pas mesurées par les capteurs de série des véhicules et sont complexes à estimer du fait de la non-linéarité des pneumatiques. De plus, toutes ces variables sont étroitement liées à l'adhérence de la route qui n'est pas mesurable. L'observation des efforts des pneumatiques est étudiée dans [161, 8, 155, 45], les angles de dérive des pneumatiques dans [83] et [45] et les

coefficients de rigidité de dérive dans [8].

2.3.4 Les capteurs supplémentaires

Dans l'ensemble, les méthodes présentées jusqu'ici utilisent les mêmes capteurs déjà présents sur les véhicules. Les capteurs de vitesses de rotation des roues servent à la mesure de la vitesse longitudinale, les ECU de l'ABS et/ou l'ESP permettent de récupérer une mesure de la vitesse de lacet. Généralement, les laboratoires menant ces recherches disposent de prototypes équipés en plus d'une centrale inertielle [137, 161] permettant de mesurer les accélérations et les vitesses de rotation des trois axes au centre de gravité.

Des GPS peuvent aussi être utilisés pour améliorer les observateurs, [177] et [194]. Les données GPS sont alors utilisées pour résoudre les problèmes d'écart et d'inexactitude. Le problème de ces méthodes est la fréquence d'échantillonnage à laquelle les données GPS sont reçues et le coût du capteur.

Dans tous les cas, il convient de trouver le bon compromis entre coût et précision de l'observateur en fonction de l'usage souhaité.

2.4 Estimation de la dynamique latérale par approche multimodèle

De nombreux capteurs sont déjà présents à bord des véhicules de dernière génération pour réaliser les fonctions d'aide à la conduite comme vu dans le Chapitre 1. Pour réaliser les fonctions autonomes, les algorithmes utilisent les données de ces capteurs. Cependant, un véhicule en mode autonome doit à tout instant connaître l'état dans lequel il se trouve. Or, certains capteurs fournissent une mesure trop bruitée ou pas assez précise des variables nécessaires pour les fonctions autonomes de niveau 3 et plus. C'est le cas notamment de la vitesse de lacet. La vitesse de lacet est mesurée par un gyroscope et alimente la fonction ESP permettant de savoir si le véhicule part en dérive. La détection de la perte de trajectoire intervient pour de grandes valeurs de vitesse de lacet et la mesure n'a donc pas besoin d'être très précise autour de 0. Un filtrage, par l'ECU de l'ESP, fait que la valeur de la vitesse de lacet présente une zone morte autour de 0. Pour les fonctions autonomes, la vitesse de lacet peut servir pour la localisation de la voiture et doit donc être connue à chaque instant et dans chaque état du véhicule. Or les dynamiques de lacet en mode autonome peuvent être proches de 0. La mesure de vitesse de lacet utilisée pour la fonction ESP, et transmise au reste des calculateurs, n'est donc pas adaptée pour cette fonction et doit être estimée plus précisément.

Nous avons vu dans la section 2.1 de ce chapitre plusieurs manières de modéliser la dynamique latérale du véhicule. Parmi les modèles exprimés, le modèle linéaire est celui qui présente le meilleur compromis entre simplicité et performance puisque c'est le modèle linéaire qui se rapproche le plus du modèle de validation. Comme présenté dans les sections précédentes C.1, le modèle linéaire de la dynamique latérale est un modèle LPI. En pratique, la vitesse longitudinale du véhicule varie et le modèle est alors LPV. En tenant compte de l'état de l'art précédent, un filtrage de Kalman peut être utilisé directement avec un modèle LPV. Il est cependant possible de réécrire le modèle LPV en une somme de modèles LPI pondérés par un paramètre variant dans le temps, c'est la modélisation Takagi-Sugeno. Cette modélisation permet alors l'utilisation

d'observateur de Luenberger pour l'estimation de la vitesse de lacet.

2.4.1 Le modèle Takagi-Sugeno

L'objectif du modèle de Takagi-Sugeno [173] est de rendre stationnaire une représentation d'état. En considérant le système,

$$\begin{cases} x(t) = f\left(x(t), \mathbf{u}(t)\right) \\ y(t) = h\left(x(t), \mathbf{u}(t)\right), \end{cases} \tag{2.60}$$

il est possible de le réécrire à l'aide de sous-systèmes linéaires représentant le système global en des points de fonctionnement spécifiques tels que :

$$\begin{cases} x(t) = \displaystyle\sum_{i=1}^{r} \nu_i\left(\xi(t)\right)\Big(A_i x(t) + B_i u(t)\Big) \\ y(t) = \displaystyle\sum_{i=1}^{r} \nu_i\left(\xi(t)\right)\Big(C_i x(t) + D_i u(t)\Big), \end{cases} \tag{2.61}$$

avec r le nombre de sous-systèmes, ξ la variable de décision et ν_i les pondérations de chaque sous-système.

Le but est de pouvoir représenter la dynamique latérale du véhicule à toutes les vitesses en utilisant uniquement des systèmes linéaires stationnaires. La vitesse $V_x(t)$ étant un paramètre mesurable, celle-ci est utilisée comme variable de décision. La vitesse longitudinale du véhicule est mesurée par les capteurs de vitesse de roue avec une résolution de 0,01 km/h. Il est à noter cependant qu'en cas de freinage à très basse vitesse, si les roues se bloquent, alors cette mesure n'est plus pertinente. Dans toute la suite du mémoire, il est supposé que les roues ne sont jamais en position de blocage.

Avec $V_x(t)$ comme variable de décision, la représentation d'état $\Sigma_{\dot{\psi}}$ (2.52) se réécrit :

$$\begin{cases} \dot{\underline{x}}(t) = A(V_x(t))\underline{x}(t) + B_{\dot{\Psi}} u(t) \\ y(t) = C_{\dot{\Psi}} \underline{x}(t), \end{cases} \tag{2.62}$$

avec $\underline{x}(t) = \begin{pmatrix} \dot{\psi}(t) \\ v_y(t) \end{pmatrix}$, $u(t) = \theta_v(t)$ et $y(t) = \dot{\Psi}(t)$.

Seule la matrice A est fonction de $V_x(t)$ et par conséquent non stationnaire :

$$\mathbf{A}(V_x(t)) = \begin{bmatrix} -2\dfrac{{L_f}^2 c_{yf} + {L_r}^2 c_{yr}}{I_z V_x(t)} & \dfrac{-2}{I_z V_x(t)}(L_f c_{yf} - L_r c_{yr}) \\ \dfrac{-2}{M_t V_x(t)}(L_f c_{yf} - L_r c_{yr}) - V_x(t) & \dfrac{-2}{M_t V_x(t)}(c_{yf} + c_{yr}) \end{bmatrix}. \tag{2.63}$$

Les matrices $B_{\dot{\Psi}}$ et $C_{\dot{\Psi}}$ s'écrivent :

$$B_{\dot{\Psi}} = \begin{bmatrix} 2\dfrac{c_{yf} L_f}{\lambda I_z} \\ 2\dfrac{c_{yf}}{\lambda M_t} \end{bmatrix} \text{ et } C_{\dot{\Psi}} = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{2.64}$$

Dans (2.63), deux fonctions non stationnaires $f_1(V_x(t))$ et $f_2(V_x(t))$ sont définies :

$$\begin{cases} f_1(V_x(t)) = \dfrac{1}{V_x(t)} \\ f_2(V_x(t)) = V_x(t). \end{cases} \tag{2.65}$$

Si ces fonctions sont continues et bornées, il est possible de les écrire de la manière suivante :

$$\begin{cases} f_1(V_x(t)) = f^1_{min}\alpha^1_1(V_x(t)) + f^1_{max}\alpha^1_2(V_x(t)) \\ f_2(V_x(t)) = f^2_{min}\alpha^2_1(V_x(t)) + f^2_{max}\alpha^2_2(V_x(t)), \end{cases} \tag{2.66}$$

avec α^j_1 et α^j_2 les pondérations locales de la fonction f_j. Comme la vitesse varie entre [1 130] km/h les fonctions f_1 et f_2 respectent les conditions de continuité et de limite.

Les pondérations locales des sous-systèmes peuvent s'exprimer par les valeurs minimales et maximales des fonctions f_1 et f_2 :

$$\begin{cases} \alpha^j_1(V_x(t)) = \dfrac{f^j_{max} - f_j(V_x(t))}{f^j_{max} - f^j_{min}} \\ \alpha^j_2(V_x(t)) = \dfrac{f_j(V_x(t)) - f^j_{min}}{f^j_{max} - f^j_{min}}. \end{cases} \tag{2.67}$$

Le couplage des deux fonctions non stationnaires aboutit à quatre sous-systèmes et donc quatre matrices A_i telles que :

$$\begin{aligned}
\mathbf{A_1} &= \begin{bmatrix} -\dfrac{2}{I_z}f^1_{max}({L_f}^2c_{yf} + {L_r}^2c_{yr}) & \dfrac{-2}{I_z}f^1_{max}(L_fc_{yf} - L_rc_{yr}) \\ \dfrac{-2}{M_t}f^1_{max}(L_fc_{yf} - L_rc_{yr}) - f^2_{max} & \dfrac{-2}{M_t}f^1_{max}(c_{yf} + c_{yr}) \end{bmatrix} \\
\mathbf{A_2} &= \begin{bmatrix} -\dfrac{2}{I_z}f^1_{max}({L_f}^2c_{yf} + {L_r}^2c_{yr}) & \dfrac{-2}{I_z}f^1_{max}(L_fc_{yf} - L_rc_{yr}) \\ \dfrac{-2}{M_t}f^1_{max}(L_fc_{yf} - L_rc_{yr}) - f^2_{min} & \dfrac{-2}{M_t}f^1_{max}(c_{yf} + c_{yr}) \end{bmatrix} \\
\mathbf{A_3} &= \begin{bmatrix} -\dfrac{2}{I_z}f^1_{min}({L_f}^2c_{yf} + {L_r}^2c_{yr}) & \dfrac{-2}{I_z}f^1_{min}(L_fc_{yf} - L_rc_{yr}) \\ \dfrac{-2}{M_t}f^1_{min}(L_fc_{yf} - L_rc_{yr}) - f^2_{max} & \dfrac{-2}{M_t}f^1_{min}(c_{yf} + c_{yr}) \end{bmatrix} \\
\mathbf{A_4} &= \begin{bmatrix} -\dfrac{2}{I_z}f^1_{min}({L_f}^2c_{yf} + {L_r}^2c_{yr}) & \dfrac{-2}{I_z}f^1_{min}(L_fc_{yf} - L_rc_{yr}) \\ \dfrac{-2}{M_t}f^1_{min}(L_fc_{yf} - L_rc_{yr}) - f^2_{min} & \dfrac{-2}{M_t}f^1_{min}(c_{yf} + c_{yr}) \end{bmatrix}.
\end{aligned} \tag{2.68}$$

Chaque sous-système est pondéré par une pondération globale ν_i :

$$\begin{aligned}
\nu_1(V_x(t)) &= \alpha^1_2(V_x(t))\alpha^2_2(V_x(t)) \\
\nu_2(V_x(t)) &= \alpha^1_2(V_x(t))\alpha^2_1(V_x(t)) \\
\nu_3(V_x(t)) &= \alpha^1_1(V_x(t))\alpha^2_2(V_x(t)) \\
\nu_4(V_x(t)) &= \alpha^1_1(V_x(t))\alpha^2_1(V_x(t)),
\end{aligned} \tag{2.69}$$

et donc (2.52) devient :

$$\begin{cases} \dot{x}(t) = \sum_{i=1}^{4} \nu_i \left(V_x(t)\right) \Big(A_i x(t) + Bu(t)\Big) \\ y(t) = C_{\dot{\Psi}} x(t). \end{cases} \tag{2.70}$$

2.4.2 Observabilité du modèle Takagi-Sugeno

La matrice d'observabilité d'un système modélisé par une représentation d'état linéaire (C.1) est définie comme :

$$\mathcal{O}(C, A) = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}, \tag{2.71}$$

avec $n \in \mathbb{N}$, l'ordre de A.

Pour que le système soit observable, il faut que :

$$\text{rang}(\mathcal{O}(C, A)) = n. \tag{2.72}$$

Le modèle Takagi-Sugeno défini précédemment est composé de 4 sous-modèles linéaires. Pour que le modèle soit observable, il faut que l'ensemble des matrices d'observabilité $\mathcal{O}(C_i, A_i)$ avec $i \in \{1, 2, 3, 4\}$ soient observables [11].

Numériquement, on a bien :

$$\text{rang}(\mathcal{O}(C_i, A_i)) = 2, \tag{2.73}$$

pour $i \in \{1, 2, 3, 4\}$.

2.4.3 Validation du modèle Takagi-Sugeno

Afin de valider la représentation multimodèle Takagi-Sugeno représentant la dynamique latérale du véhicule décrite plus haut, une comparaison entre le modèle de validation non linéaire, le modèle linéaire au point de fonctionnement $V_{x0} = 90$ km/h et le modèle Takagi-Sugeno est effectuée dans cette partie.

Pour ce faire, le scénario de changement de voie du premier chapitre est repris, *i.e* un déplacement latéral de 3,5 m sur 200 m. La seule différence est que, la vitesse longitudinale du véhicule varie pendant la première moitié du dépassement. Le schéma bloc de la simulation est visible Figure 2.33. Comme indiqué sur la figure, le modèle non linéaire et le modèle Takagi-Sugeno ont pour entrée l'angle volant et la vitesse longitudinale alors que le modèle linéarisé à 90 km/h utilise uniquement l'angle volant pour estimer la vitesse de lacet. Le modèle non linéaire est utilisé comme référence.

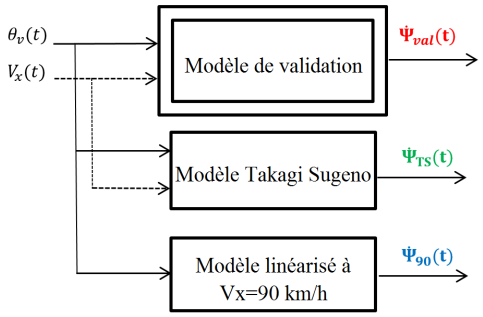


FIGURE 2.33 – Schéma bloc de la simulation de validation

L'angle volant utilisé est le même que celui utilisé dans le chapitre 1 à 90 km/h, Figure 2.4, et est rappelé en Figure 2.34. Au début de la simulation, le véhicule roule à 90 km/h. Cette vitesse augmente jusqu'à 100 km/h lors de la première moitié de la manœuvre de changement de voie. Ces variables sont tracées sur la Figure 2.34.

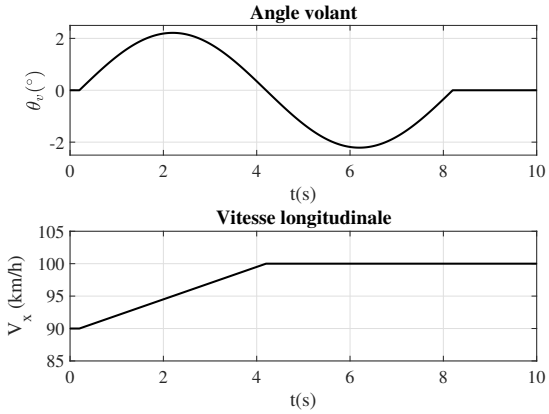
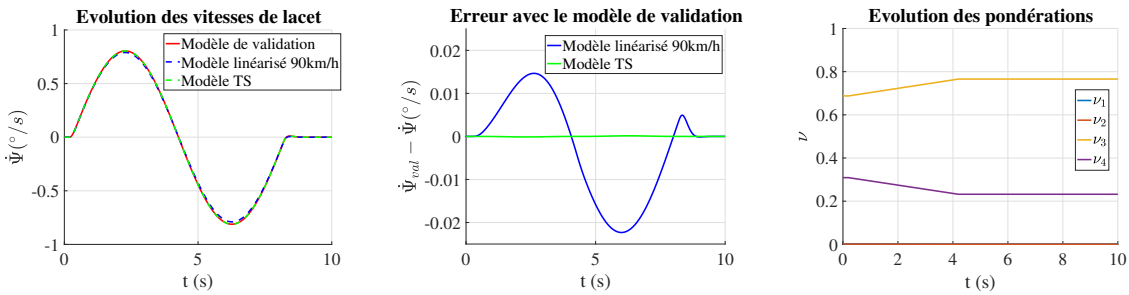


FIGURE 2.34 – Évolution des signaux d'entrée



(a) (b) (c)

FIGURE 2.35 – Évolution des vitesses de lacet des différents modèles (a) des erreurs entre le modèle de validation et les modèles linéaire et Takagi-Sugeno (b) et des pondérations des sous-systèmes du modèle Takagi-Sugeno (c)

La Figure 2.35 montre que le multimodèle Takagi-Sugeno de la dynamique latérale est une bonne représentation du modèle lacet-dérive du véhicule. En effet, comparé au modèle non linéaire de référence, l'erreur de sortie en vitesse de lacet par le modèle Takagi-Sugeno est six fois

plus faible que par le modèle linéarisé à 90 km/h. L'erreur relative maximale est de $2,8\,\%$ pour le modèle linéarisé et de $0,02\,\%$ pour le modèle de Takagi-Sugeno. L'évolution des pondérations globales est tracée sur la Figure 2.35 (c), elle est similaire à celle de la vitesse Figure 2.34.

2.4.4 Problème d'échantillonnage à basse vitesse

À bord du véhicule, les données envoyées par les capteurs aux ordinateurs sont traitées en temps discret à une période de 40 ms.

Théorème de Shannon

Afin d'estimer la vitesse de lacet en temps discret dans le véhicule, le théorème de Shannon doit être vérifié. En effet, pour pouvoir traiter numériquement un signal temporel, ce théorème stipule que le signal doit être échantillonné à une fréquence f_s deux fois supérieure à la plus grande fréquence propre du système f_0. Sachant que pour une fréquence f sa pulsation correspondante est $\omega = 2\pi f$, le théorème est :

$$\omega_s \geq 2\omega_0, \tag{2.74}$$

avec $\omega_s = 2\pi f_s$ la pulsation d'échantillonnage.

La période d'échantillonnage est 0,04 s donnant $\omega_s = 157\,\text{rad/s}$. Le théorème de Shannon est respecté si $\omega_0 < 78,5\,\text{rad/s}$. Cependant, comme vu précédemment, la vitesse influe sur les paramètres de la représentation fréquentielle du modèle lacet dérive dont une de ses fréquences propres est ω_0, Figure 2.26 (a). Sur cette figure, ω_0 augmente fortement lorsque la vitesse longitudinale diminue et devient même supérieure à la valeur limite de 78,5 rad/s dès que $V_x < 10\,\text{km/h}$.

Le fait que le théorème de Shannon ne soit pas respecté à faible vitesse n'est pas un problème pour la représentation linéaire non stationnaire utilisée pour le filtrage de Kalman. En effet, la discrétisation du modèle pour l'utilisation du filtre de Kalman discret introduit un bloqueur d'ordre zéro (C.7). Grâce à ce bloqueur d'ordre zéro, le fait de garder constante la valeur de l'échantillon de l'entrée u_k à l'instant k sur l'intervalle $[k,\, k+1]$ permet d'atténuer les pulsations indésirables supérieures à $\omega_s/2$. Cependant, le modèle de Takagi-Sugeno a besoin d'être optimisé pour être représentatif de la dynamique lacet-dérive à basse vitesse.

Optimisation du modèle Takagi-Sugeno

Pour le modèle Takagi-Sugeno, il est nécessaire de connaître la vitesse de lacet du véhicule à chaque instant, même à basse vitesse. Afin de respecter le théorème de Shannon à basse vitesse, l'idée est d'optimiser le modèle de Takagi-Sugeno en modifiant les pondérations globales. Ces pondérations sont changées de telle sorte à réduire suffisamment la fréquence de coupure et à avoir le nouveau gain du modèle le plus proche possible du gain réel. Pour ce faire, à basse vitesse, lorsque $V_x(t) < 20\,\text{km/h}$, sur les quatre pondérations $\nu_i(V_x(t))$ du multimodèle, trois sont fixées pour leurs valeurs à 20 km/h et la dernière est une fonction linéaire de $V_x(t)$. Cela donne :

$$\begin{aligned} \nu_{1LS} &= \nu_1(20\text{km/h}) \\ \nu_{2LS} &= \nu_2(20\text{km/h}) \\ \nu_{3LS} &= \nu_3(20\text{km/h}) \\ \nu_{4LS}(V_x(t)) &= aV_x(t) + b, \end{aligned} \tag{2.75}$$

avec ν_{iLS} la ième pondération globale à basse vitesse et a et b, les coefficients à trouver pour obtenir un modèle le plus proche de la réalité pouvant être discrétisé.

Avec ces pondérations, les paramètres K_0 et ω_0 de la fonction de transfert à basse vitesse sont :

$$\begin{aligned}
K_{0LS} &= \frac{2c_{yf}c_{yr}L}{\lambda}\frac{\nu_{1LS}+\nu_{2LS}+\nu_{3LS}+aV_x+b}{2c_{yf}c_{yr}L^2\beta_1(V_x)-M_tPDE\beta_2(V_x)},\\
\omega_{0LS}^2 &= \frac{2\beta_1(V_x)}{I_zM_t}\left(2c_{yf}c_{yr}L^2\beta_1(V_x)-M_tPDE\beta_2(V_x)\right)\\
&\text{avec}\\
\beta_1(V_x) &= f_{max}^1(\nu_{1LS}+\nu_{2LS})+f_{min}^1(\nu_{3LS}+aV_x+b)\\
\beta_2(V_x) &= f_{max}^2(\nu_{1LS}+\nu_{3LS})+f_{min}^2(\nu_{2LS}+aV_x+b).
\end{aligned} \tag{2.76}$$

L'objectif est de trouver a et b pour qu'à toute vitesse $V_x < 20$ km/h :

- K_{0LS} soit aussi proche que possible de K_0,
- $\omega_{0LS} \leq 2\omega_s$.

Pour respecter ces conditions, la fonction de coût suivante est définie et doit être minimisée :

$$\begin{aligned}
J(a,b) =&(K_{01}-K_{01LS})^2+(K_{020}-K_{020LS})^2+\\
&(2\omega_s-\omega_{01LS})^2+(2\omega_s-\omega_{020LS})^2,
\end{aligned} \tag{2.77}$$

où K_{0i} et K_{0iLS} correspondent aux gains K_0 et K_{0LS} à la vitesse i et ω_{0iLS} est la pulsation ω_{0LS} à la vitesse i.

Afin de trouver les paramètres a et b adéquats sur toute la plage de basse vitesse, la fonction de coût définie ci-dessus prend en compte les gains et pulsations désirés pour le modèle lacet-dérive aux extrêmes de l'intervalle basses vitesses, *i.e* 1 km/h et 20 km/h.

Les paramètres a et b minimisant la fonction J sont calculés en utilisant le gradient ∇J :

$$\nabla J(a,b) = \left[\frac{\partial J}{\partial a}\ \frac{\partial J}{\partial b}\right]^T. \tag{2.78}$$

Il existe cinq solutions à l'équation :

$$\nabla J(a,b) = [0\ 0]^T, \tag{2.79}$$

signifiant que la paire (a,b) est un optimum de J.

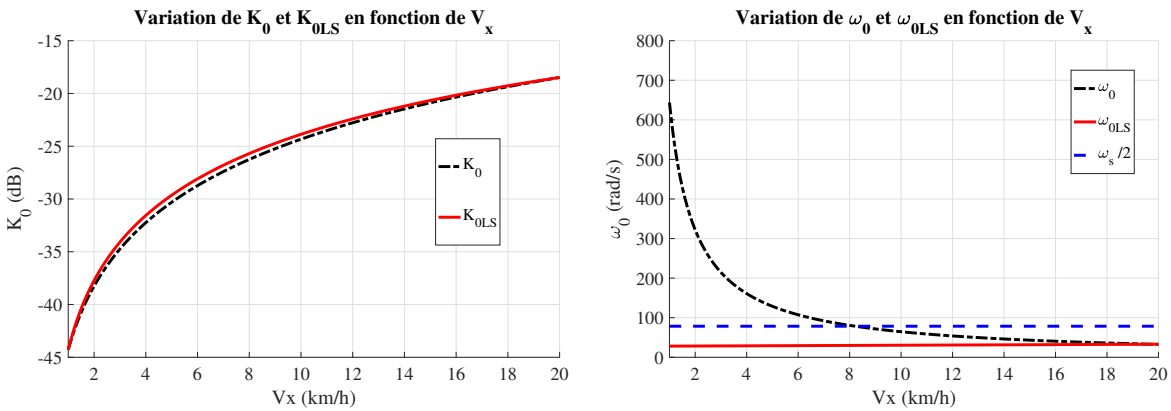
Pour savoir si ces optimums sont des maximums ou des minimums, la matrice hessienne $H(J)$ est calculée :

$$H(J) = \begin{bmatrix} \dfrac{\partial^2 J}{\partial^2 a} & \dfrac{\partial^2 J}{\partial a \partial b} \\ \dfrac{\partial^2 J}{\partial b \partial a} & \dfrac{\partial^2 J}{\partial^2 b} \end{bmatrix} = \begin{bmatrix} r & v \\ v & w \end{bmatrix}. \tag{2.80}$$

Suivant les valeurs de r, v et w, la paire (a, b) solution de (2.79) est :

- un minimum, si $rw - v^2 > 0$ et $r > 0$,
- un maximum, si $rw - v^2 > 0$ et $r < 0$,
- aucune conclusion ne peut être faite, sinon.

Sur les cinq solutions de (2.79), trois respectent les conditions $rw - v^2 > 0$ et $r > 0$. Le minimum global de la fonction est calculé numériquement, donnant la paire $(a; b) = (0,1834\,;\ -0,1897)$.



(a) (b)

FIGURE 2.36 – Variation des gains K_0 et K_{0LS} (a) et des pulsations ω_0 et ω_{0LS} (b) en fonction de la vitesse longitudinale V_{x0}

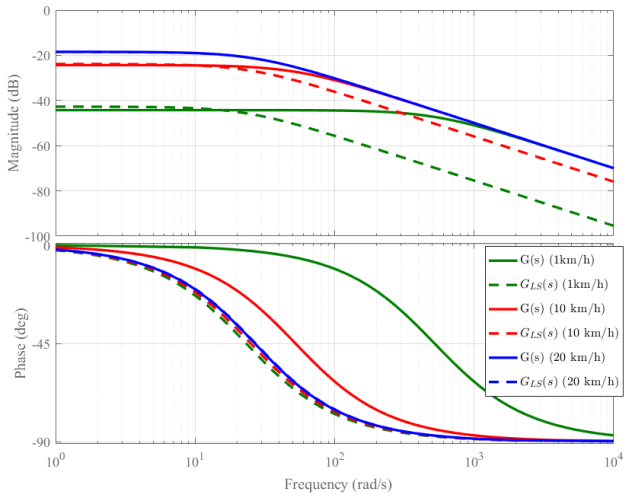


FIGURE 2.37 – Comparaison des diagrammes de Bode de $G(s)$ et $G_{LS}(s)$

La fonction de transfert du système (2.70) avec $\nu_i = \nu_{iLS}$ est écrite $G_{LS}(s)$. Les variations de gain et de pulsation de $G(s)$ et $G_{LS}(s)$ à basse vitesse sont comparées sur la Figure 2.36. Comme il peut être observé sur ces deux figures, le gain du multimodèle optimisé à basse vitesse est très proche du modèle réel avec une erreur inférieure à 8 %. Quant à la pulsation, ω_{0LS}, cette dernière est bien inférieure à $\omega_s/2$ sur tout l'intervalle [1; 20] km/h, signifiant que cette représentation de

la dynamique latérale peut maintenant être discrétisée et utilisée dans les ordinateurs à bord du véhicule. Les diagrammes de Bode des deux fonctions de transfert sont tracés sur la Figure 2.37 pour 1 km/h, 10 km/h et 20 km/h.

2.5 Validation et résultats pour l'estimation de la vitesse de lacet

Plusieurs observateurs sont comparés en utilisant des variables enregistrées sur le prototype : un observateur de Luenberger avec le modèle linéaire utilisé en un point de fonctionnement donné, un filtre de Kalman comme défini au paragraphe C.1 et plusieurs observateurs de Luenberger avec une représentation multimodèle Takagi-Sugeno.

Il est à préciser que les conditions d'observabilité pour les observateurs de Luenberger et le filtre de Kalman sont respectées. En effet, pour toutes les valeurs de $V_x(t)$ comprises entre 1 km/h et 130 km/h avec un échantillonnage de 0,01 km/h, la matrice d'observabilité de (2.62) est de rang 2.

Les capteurs de vitesse de lacet qui équipent actuellement les véhicules de série ont été conçus pour les fonctions telles que l'ESP et présentent des mesures insatisfaisantes pour une utilisation dans les fonctions autonomes. En effet, après traitement de la mesure de la vitesse de lacet par l'ECU de l'ESP, l'information de ces capteurs transmise aux autres calculateurs comporte une zone morte pour de faibles valeurs de vitesse de lacet. Pour pallier ce manque d'information autour de zéro, il faut soit demander au fournisseur de l'ESP de ne plus traiter l'information émise sur le réseau du véhicule mais cela risque d'avoir un impact sur les autres consommateurs de l'information, soit avoir recours à des observateurs.

Les informations sur l'état du véhicule mesurées par les capteurs et utilisées dans la simulation de ce paragraphe sont : la vitesse longitudinale V_x, l'angle volant θ_v, la vitesse de lacet envoyée par l'ESP $\dot{\psi}_s$ et celle mesurée par la centrale inertielle $\dot{\psi}_{IU}$, utilisée comme valeur de référence. Les observateurs linéaires avec modèle linéarisé à 90 km/h, le filtrage de Kalman et le multimodèle Takagi-Sugeno ont pour sorties respectives $\dot{\psi}_1(t)$, $\dot{\psi}_2(t)$ et $\dot{\psi}_3(t)$. Le schéma de la simulation est présenté Figure 2.38.

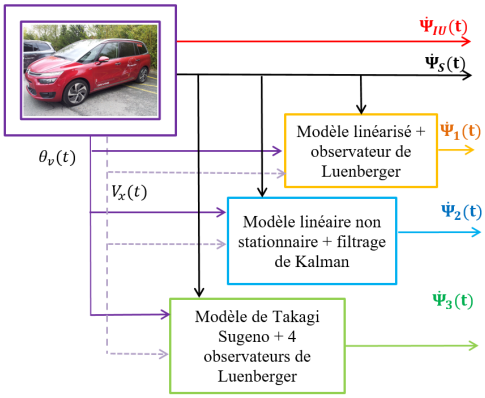
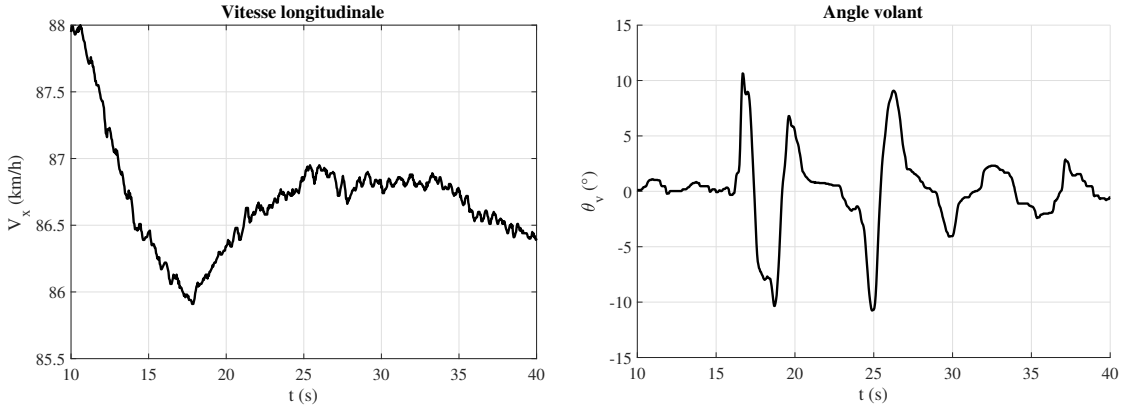


FIGURE 2.38 – Schématisation de la simulation

Deux simulations ont été faites, une première avec un enregistrement à haute vitesse et une deuxième avec un enregistrement basse vitesse.

Simulation haute vitesse

Les variables d'entrée des observateurs, $V_x(t)$ et $\theta_v(t)$, sont tracées Figure 2.39 (a) et (b) respectivement.



(a) (b)

FIGURE 2.39 – Évolution de la vitesse longitudinale $V_x(t)$ (a) et de l'angle volant $\theta_v(t)$ (b) en fonction du temps pour l'enregistrement haute vitesse

Pour l'enregistrement grande vitesse, celle-ci est proche de 90 km/h. Le modèle stationnaire linéarisé est donc à son point de fonctionnement et cela permet une bonne comparaison entre les trois observateurs. En effet, à cette vitesse, tous les modèles devraient être représentatifs du modèle réel. Les sorties des observateurs, à savoir les estimations de vitesse de lacet, ainsi que les variables mesurées sont tracées Figure 2.40.

Les mesures de vitesses sont tracées en rouge pour la centrale inertielle ($\dot{\psi}_{IU}(t)$) et en noir pour l'information ESP ($\dot{\psi}_S(t)$). Comme indiqué précédemment, le signal ESP ne renvoie aucune information pour la vitesse de lacet autour de $0°/s$.

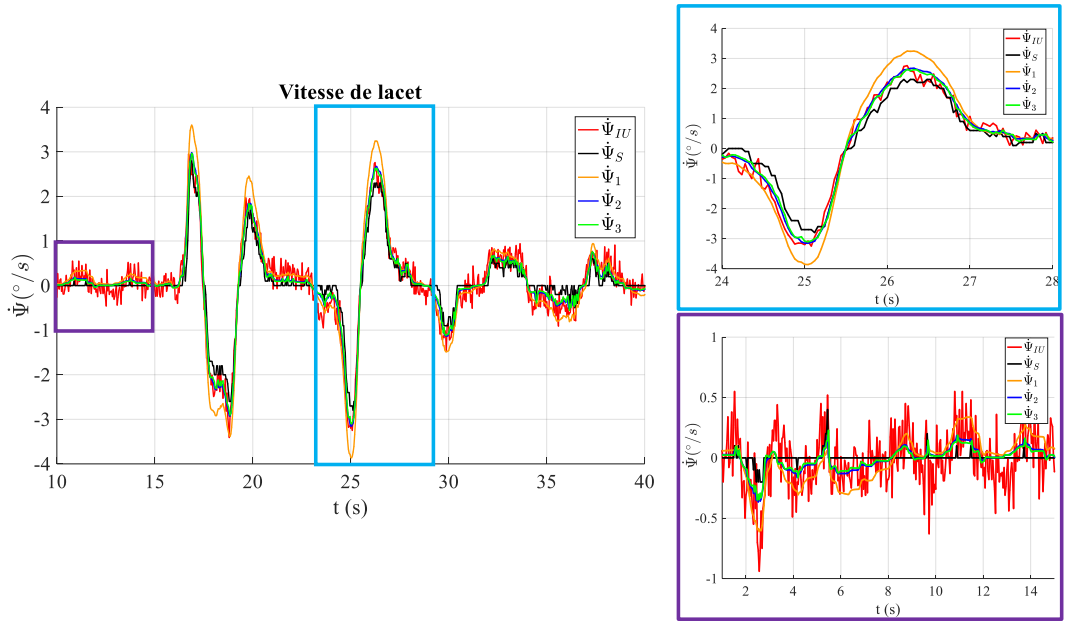


FIGURE 2.40 – Évolution des vitesses de lacet mesurées et estimées en fonction du temps pour l'enregistrement à haute vitesse (zoom sur les cadres violet et bleu à droite)

Pour comparer numériquement les performances de chaque observateur, la norme de la racine carrée de l'erreur quadratique moyenne (NRMSE pour « Normalised Root Mean Square Error ») est calculée :

$$\text{NRMSE} = \frac{\sqrt{\sum_{i=1}^{N_e}(\dot{\Psi}_{IU}(i) - \dot{\Psi}_k(i))^2/N}}{|\max(\dot{\Psi}_{IU}) - \min(\dot{\Psi}_{IU})|}, \tag{2.81}$$

avec N_e le nombre d'échantillons de l'enregistrement et $k \in \{1, 2, 3\}$ l'indice de l'observateur tel que sur la Figure 2.38.

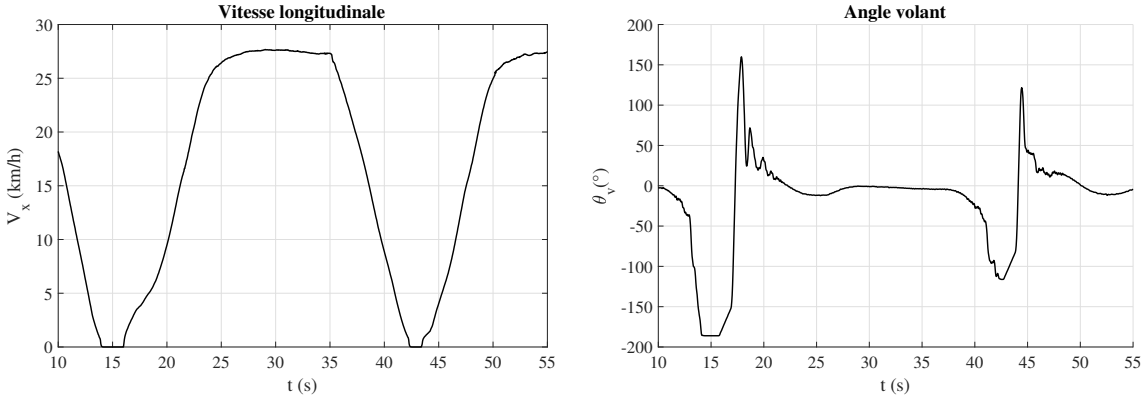
Ainsi, pour l'enregistrement à haute vitesse, l'observateur de Luenberger au point de fonctionnement à 90 km/h présente une erreur quadratique normalisée de 4,5 %, le filtre de Kalman 3,1 % et le modèle Takagi-Sugeno avec les quatre observateurs de Luenberger 3,2 %.

Lorsque l'angle volant est petit, $\theta_v(t) < 2°$, au début de l'enregistrement ; tous les observateurs renvoient une bonne estimation de la vitesse de lacet avec une NRMSE<3,2 % alors que la mesure utilisée en entrée de ces observateurs est fausse. Cependant, pour des valeurs d'angle volant plus importantes, le modèle linéarisé à 90 km/h avec un observateur de Luenberger ne parvient pas à représenter correctement la dynamique latérale. Le filtre de Kalman et les quatre observateurs de Luenberger couplés au modèle lacet-dérive Takagi-Sugeno donnent quasiment la même estimation de la vitesse de lacet.

Simulation basse vitesse

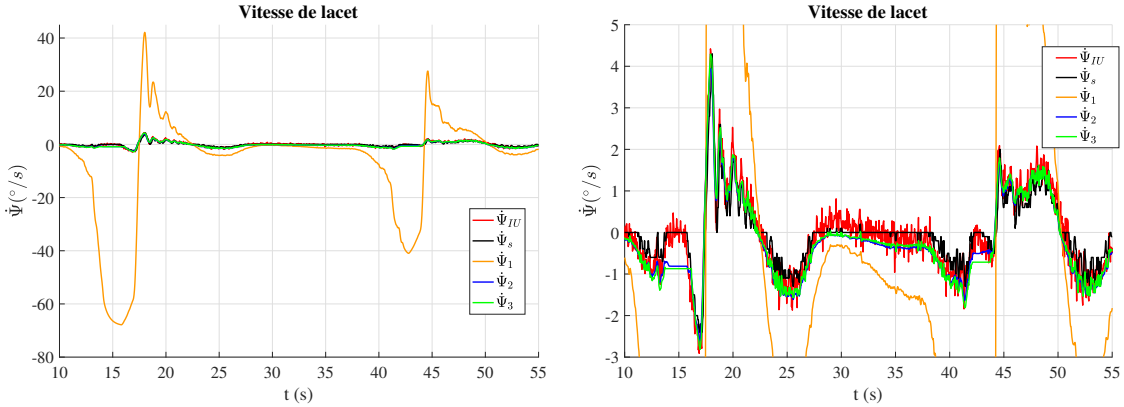
Pour la simulation à basse vitesse, un enregistrement dans le cas d'un embouteillage a été choisi. Le premier objectif est de vérifier si le modèle linéaire à 90 km/h peut tout de même estimer la vitesse de lacet pour une vitesse très éloignée de son point de fonctionnement. Le deuxième est de confirmer que le bloqueur d'ordre 0 introduit par la discrétisation du modèle est capable d'éliminer les fréquences du modèle supérieures à $\omega_s/2$ à basse vitesse. Le dernier est de valider l'optimisation du modèle Takagi-Sugeno pour la basse vitesse $V_x \leq 20$ km/h.

Les variables d'entrée des observateurs $V_x(t)$ et $\theta_v(t)$ pour cette simulation basse vitesse sont tracées respectivement Figure.2.41 (a) et (b). Pour cette simulation, il y a deux régions autour de 15 s et 43 s où $V_x < 10$ km/h et où donc potentiellement, les modèles peuvent ne pas représenter la réalité. Comme la vitesse est plus petite, l'angle volant nécessaire pour effectuer un virage ou un changement de voie est beaucoup plus élevé, Figure. 2.41 (b).



(a) (b)

FIGURE 2.41 – Évolution de la vitesse longitudinale $V_x(t)$ (a) et de l'angle volant $\theta_v(t)$ (b) en fonction du temps pour l'enregistrement basse vitesse



(a) (b)

FIGURE 2.42 – Évolution des vitesses de lacet mesurées et estimées en fonction du temps pour l'enregistrement à basse vitesse (a) et un zoom (b)

Les sorties des observateurs et les mesures de vitesse de lacet des capteurs sont tracées sur la Figure 2.42. Comme attendu, le modèle linéarisé à 90 km/h n'est pas du tout représentatif lorsque la vitesse s'éloigne des 90 km/h avec une NRMSE de 217 %. Cependant, comme pour la simulation à haute vitesse, les deux autres combinaisons modèle/observateur permettent une bonne représentation de la vitesse de lacet. La NRMSE pour le filtrage de Kalman est de 4,8 % et pour le modèle de Takagi-Sugeno avec les observateurs de Luenberger de 4,9 %. Cela veut donc dire que la discrétisation pour le filtrage de Kalman, et l'optimisation pour le modèle Takagi-Sugeno, sont de bonnes solutions pour résoudre le problème du théorème de Shannon à basse vitesse.

Bilan

Afin d'analyser le comportement du véhicule et de développer les fonctions de guidage latéral et de planification de trajectoire, plusieurs modélisations de la dynamique latérale sont présentées dans ce chapitre. Un modèle de validation non linéaire sert de référence et est comparé à trois modèles, à savoir : un modèle dynamique linéarisé ; un modèle linéaire en régime stationnaire ; un modèle cinématique. Les comparaisons sont effectuées à la fois dans le domaine temporel, avec des simulations de changement de voie, et dans le domaine fréquentiel. Il apparaît alors que le modèle linéarisé est le plus représentatif.

Les fonctions nécessaires à la conduite autonome reposent sur l'utilisation des variables de la dynamique latérale. Or les mesures disponibles de ces variables ne sont pas toujours fiables. Le chapitre compare alors plusieurs observateurs pour l'estimation de la vitesse de lacet : un modèle linéarisé à 90 km/h avec un observateur de Luenberger, un filtre de Kalman et un multimodèle Takagi-Sugeno avec un observateur de Luenberger par sous-modèle.

Ces deux dernières méthodes ont des résultats très similaires en simulation. La grande différence réside dans la manière de traiter le modèle. Le filtre de Kalman nécessite une représentation d'état avec une matrice A_k inversible à chaque instant. Le modèle Takagi-Sugeno quant à lui a besoin de moins de calculs embarqués dans le véhicule, mais dans notre cas requiert une optimisation importante du modèle et une analyse préliminaire de la dynamique de lacet.

Chapitre 3

Guidage latéral du véhicule

Sommaire

3.1 État de l'art : guidage latéral du véhicule **88**
3.2 La conception de régulateurs PID et CRONE dans le domaine fréquentiel **91**
3.2.1 Le PID 91
3.2.2 La commande CRONE 92
3.3 Régulation latérale par PID **97**
3.4 Régulation latérale par CRONE de 3ème génération **101**
3.5 Approche multirégulateur **106**
3.6 Régulation latérale par multi-PID **108**
3.7 Théorie et apport du point de visée **113**
3.7.1 Définition du point de visée 113
3.7.2 Influence du point de visée sur la dynamique latérale 114
3.8 Régulation latérale avec point de visée par CRONE de 3ème génération **116**
3.9 Régulation latérale avec point de visée par multi-PID **119**
3.10 Validation et résultats sur prototype **124**
3.10.1 Intégration du régulateur dans le prototype ADN1 124
3.10.2 Résultats sur route ouverte : maintien dans la voie 124
3.10.3 Résultats sur route ouverte : changement de voie 126
3.10.4 Bilan : avantages, inconvénients et perspectives 128

Osez, le progrès est à ce prix.

Victor Hugo

La conduite autonome de niveau 3 et plus nécessite un contrôle complet de la direction et du groupe motopropulseur. La fonction d'autonomie doit ainsi être capable de donner les bonnes entrées en angle volant et en vitesse afin de réaliser les manœuvres nécessaires dans les situations de vie couvertes par la fonction.

Ce chapitre se concentre sur la régulation de l'angle volant pour le guidage latéral du véhicule. D'abord, un état de l'art sur les méthodes de régulation pour le guidage latéral dans le cadre du véhicule autonome est présenté. Puis, la méthode de conception de régulateur dans le domaine fréquentiel est présentée. Ensuite, plusieurs types de régulateurs, PID, CRONE et multi-PID sont calculés sur la base d'un même cahier des charges et comparés sur des simulations de changement de voie pour des vitesses longitudinales différentes. Une méthode de multirégulation est élaborée pour réduire la sensibilité de la régulation à la variation de vitesse qui, comme vu dans le chapitre 2, a un impact très important sur la dynamique latérale du véhicule.

Une étude est ensuite menée sur l'influence de la consigne sur la régulation latérale, et notamment l'apport d'un point de visée à l'avant du véhicule. De nouveaux régulateurs, CRONE et multi-PID, sont alors calculés avec une référence devant le véhicule et comparés avec les mêmes simulations que précédemment.

Enfin, le multi-PID présentant les meilleurs résultats en simulation est implanté dans les calculateurs de ADN1, testé et validé dans les cas de maintien dans la voie et de changement de voie.

3.1 État de l'art : guidage latéral du véhicule

Les premiers prototypes de véhicules autonomes ayant vu le jour dans les années 1980, il existe une littérature importante sur les méthodes de régulation de la direction pour le suivi de trajectoire sans l'action du conducteur. Cette régulation consiste à déterminer l'angle volant en fonction de l'état actuel dans lequel se trouve le véhicule et l'état dans lequel il devrait se trouver pour suivre la trajectoire désirée.

PID

La régulation la plus simple, à savoir un simple gain sur l'erreur de sortie, est utilisée dans [22]. Dans cette étude, le gain, dépendant de la vitesse longitudinale, est déterminé par un ensemble de tests essai/erreur afin d'avoir le comportement désiré en conduite autonome pour du maintien dans la voie et du suivi de véhicule.

Autre régulation très populaire dans le domaine du contrôle, le PID (Proportionnel Intégral Dérivé) [6] est aussi largement utilisé dans le cadre de la conduite autonome. En effet, ce type de régulateur a largement fait ses preuves dans l'industrie pour son rapport simplicité/efficacité de conception. Le PID combine les actions proportionnelle, intégrale et dérivée sur le signal d'erreur de sortie pour calculer la commande à appliquer à l'actionneur. Ces trois actions combinées permettent de régler la stabilité, la précision et la rapidité du régulateur. Dans [147], un PID est utilisé à bas niveau en combinaison d'un algorithme de logique floue à haut niveau pour reproduire la conduite humaine. Le régulateur est ensuite testé sur un circuit de type urbain avec des virages à 90°, il est ainsi calibré pour fonctionner entre 5 et 18 km/h. Plusieurs PID peuvent aussi être combinés. Dans [1], un PIDD sur l'offset de position latérale est associé en cascade à un P (simple gain) sur l'erreur d'angle de lacet. Ceci améliore les performances en position par rapport aux méthodes utilisant uniquement l'erreur en position en entrée du régulateur. Dans [123], deux boucles de régulation imbriquées sont utilisées. La première boucle régule l'erreur de

vitesse de lacet avec un PI et la deuxième l'erreur de position avec un PID. Cette étude utilise un modèle véhicule linéarisé avec une représentation des pneumatiques de Pacejka.

Au signal de commande calculé par le régulateur à partir de l'erreur de sortie (feedback) peut être associé un signal de commande calculé à partir d'une perturbation externe (feedforward) afin d'avoir une action anticipatrice. Ainsi, dans [36], l'angle volant est calculé à partir d'un PID régulant l'erreur de position et d'un gain par feedforward sur la courbure de la trajectoire pour les environnements non structurés du DARPA Challenge.

Les études dans les situations de vie urbaines ne sont pas nombreuses. Pour la régulation de type PID, [146] utilise deux gains sur l'erreur de position et l'erreur d'angle de lacet pour faire circuler une voiture dans un rond-point en simulation.

La commande H_∞

La commande H_∞ [46] résout un problème de rejet de perturbation pour la conception d'une commande optimale au sens mathématique, minimisant les effets de la perturbation sur le comportement du système. La loi de commande est issue d'une minimisation de la norme de la matrice de transfert du système augmenté.

Pour la régulation de l'angle volant en conduite autonome, [136] utilise la commande H_∞ pour prendre en compte la variation de paramètres tels que la vitesse longitudinale et l'adhérence de la route ou de perturbations comme les rafales de vent. La solution est ensuite validée pour des cas de changement de voie sur autoroute. [187] ne considère pas des voitures, mais des poids lourds. [84] utilise des algorithmes génétiques et des inégalités matricielles pour résoudre le problème d'optimisation de la commande H_∞ avec le modèle non linéaire du véhicule. La commande est ensuite validée en simulation dans des cas de changement de voie.

La commande H_∞ est aussi utilisée pour l'assistance à la conduite [152]. Dans cette étude, le régulateur apporte du couple en plus de celui fourni par le conducteur pour permettre au véhicule de rester dans la voie et éviter les franchissements de ligne. La méthode est validée pour des vitesses entre 36 km/h et 72 km/h.

Dans [25], la commande H_∞ est comparée à des commandes adaptatives, PID et logique floue sur du suivi de voie en simulation. Les variations de la vitesse longitudinale, l'adhérence de la route et les perturbations extérieures comme des rafales de vent sont ajoutées aux simulations pour comparer les régulateurs. Il est à noter que ces paramètres restent fixes pendant la simulation. Il en ressort que le PID est la solution avec la plus grande erreur en sortie. H_∞ et le contrôleur flou ont des performances équivalentes, mais inférieures à celles du régulateur adaptatif.

La commande par mode glissant

La commande par mode glissant [51] est une commande développée dans le but de répondre aux incertitudes et variations des paramètres du procédé, mais aussi aux perturbations externes. Elle s'applique aux systèmes linéaires et non linéaires. Le principe de cette régulation est de forcer la dynamique du système à atteindre un état tout en suivant une trajectoire sur une surface de glissement. Cette surface de glissement est créée à partir de commutation de fonctions des variables d'état du système.

L'étude [73] se base sur de la régulation par mode glissant pour améliorer les performances d'un ESP. Dans [77], la commande par mode glissant régule la vitesse de lacet pour faire du changement de voie à vitesse constante pour des vitesses entre 15 km/h et 110 km/h. [172] se base sur une commande par mode glissant d'ordre supérieur pour éviter les phénomènes d'oscillation liés aux commutations introduites par cette méthode de régulation. L'erreur en position est utilisée pour calculer l'angle volant. La commande, validée par des expérimentations sur un circuit avec vitesse longitudinale variable entre 18 et 90 km/h, est robuste aux incertitudes paramétriques et aux non-linéarités du modèle du véhicule.

La commande prédictive

La commande prédictive, plus communément appelée MPC pour « Model Predictive Control », se base sur un modèle d'évolution dynamique du procédé pour anticiper ses futurs états. Cette connaissance des futurs états est utilisée pour calculer la consigne optimale à appliquer à l'actionneur. L'avantage de la commande prédictive est qu'elle résout à chaque instant un problème d'optimisation tenant compte des limitations des actionneurs et conduisant ainsi à une commande plus souple. Cependant, en cas de modèle complexe, le problème d'optimisation peut conduire à des temps de calcul conséquents.

Dans [56], plusieurs MPC utilisant en entrée les erreurs d'angle de lacet, de vitesse de lacet et de position sont comparés en fonction du modèle interne. Un modèle non linéaire et un modèle linéaire sont testés pour faire du changement de voie avec les paramètres d'adhérence et de vitesse pouvant varier. La complexité calculatoire avec le MPC à modèle non linéaire est beaucoup plus importante qu'avec le modèle linéaire. Ce même type de comparaison est mené dans [10], là aussi montrant des temps de calcul supérieurs de plusieurs ordres pour le modèle non linéaire. Avec [196], une commande MPC est développée pour la navigation et l'évitement d'obstacles dans un environnement non structuré. Le problème de coût minimise l'erreur de position et la consommation d'énergie. La commande prédictive est utilisée en intersection dans [102]. La méthode présentée utilise aussi le MPC comme générateur de trajectoire. Enfin dans [7], une régulation simultanée des dynamiques longitudinales et latérales par MPC est développée. La méthode est validée en simulation sur un échangeur en sortie d'autoroute.

Les réseaux de neurones

Avec l'amélioration de la puissance de calcul des ordinateurs et des cartes graphiques, d'autres méthodes de régulation commencent à émerger pour la conduite autonome : l'apprentissage automatique par réseaux de neurones.

Les réseaux de neurones [71] sont des structures permettant de reproduire le comportement entrée/sortie de n'importe quel modèle non linéaire. Ils sont notamment très utilisés pour le traitement d'images. Ainsi, la société de construction de carte graphique NVIDIA [14] a développé un algorithme end-to-end qui, à partir d'une simple image d'une caméra frontale, détermine directement l'angle volant à appliquer pour rester dans une voie. L'entraînement du réseau a nécessité un peu moins de 100 heures d'enregistrement vidéo.

Les méthodes d'apprentissage par renforcement sont aussi utilisées pour la conduite autonome. Le principe de ces méthodes est d'apprendre des lois de comportement par essai/erreur en attribuant des récompenses si le système a un « bon » comportement et des pénalités s'il commet des erreurs. Dans [138], la méthode est utilisée sur un circuit pour du suivi de voie. Le principal inconvénient de cette méthode est la définition des fonctions de récompenses qui vont

conditionner le comportement du véhicule.

L'inconvénient des méthodes par réseaux de neurones est qu'il n'est pour l'instant pas possible de déterminer si les modèles entraînés sont sûrs. Il est compliqué d'assurer la « safety » du véhicule avec de telles méthodes.

3.2 La conception de régulateurs PID et CRONE dans le domaine fréquentiel

Dans ce chapitre, les régulateurs de type PID et CRONE sont largement utilisés. Le calcul de ces régulateurs dans le domaine fréquentiel est présenté ici et utilise les outils de l'Annexe A. Cette annexe fait aussi état des méthodes utilisées pour l'analyse fréquentielle du système bouclé.

3.2.1 Le PID

Le régulateur Proportionel-Intégrateur-Dérivateur, ou PID, est le régulateur le plus utilisé dans l'industrie en raison de sa simplicité et son efficacité. Un régulateur PID est composé de 3 cellules : proportionnelle, intégrale, dérivée. La paramétrisation de ces trois cellules permet d'influencer les performances recherchées de la boucle fermée : rapidité, précision, stabilité. Augmenter la partie P augmente la précision et la rapidité, mais diminue la stabilité. L'augmentation de la partie intégrale I diminue la stabilité et la rapidité, mais augmente la précision. La partie I a comme propriété fondamentale de rendre l'erreur statique nulle et est donc quasi systématiquement présente. Enfin, la cellule D augmente la stabilité et diminue la rapidité. En pratique, la cellule D peut aussi augmenter le bruit de mesure et la forme PI (sans dérivée) du régulateur PID est souvent préférée.

À partir d'un cahier des charges donnant la rapidité du système bouclé désirée par l'intermédiaire de la fréquence au gain unité, ω_u et la stabilité désirée par la marge de phase M_Φ, les équations pour trouver les paramètres du régulateur sont décrites ci-dessous.

Le PID est de la forme :

$$C(s) = C_0 \left(\frac{1 + s/\omega_i}{s/\omega_i} \right) \left(\frac{1 + s/\omega_1}{1 + s/\omega_2} \right), \tag{3.1}$$

où C_0 est un gain, ω_i est la fréquence transitionnelle définie usuellement par $\omega_i = \omega_u/10$, ω_1 et ω_2 sont les pulsations transitionnelles du régulateur à avance ou retard de phase. Sachant que la fonction de transfert en boucle ouverte, pour la régulation du procédé $H(s)$, s'écrit :

$$\beta(j\omega) = C(j\omega)H(j\omega), \tag{3.2}$$

les pulsations ω_1 et ω_2 sont calculées de sorte à respecter la marge de phase M_Φ désirée dont l'expression est donnée par :

$$M_\Phi = \arg\Big(\beta(j\omega_u)\Big) + \pi, \tag{3.3}$$

avec

$$\arg\Big(\beta(j\omega_u)\Big) = \arg\Big(C(j\omega_u)H(j\omega_u)\Big) = \arg\Big(C(j\omega_u)\Big) + \arg\Big(H(j\omega_u)\Big). \tag{3.4}$$

L'avance de phase ϕ_m apportée par la cellule à avance de phase du PID à la fréquence centrale $\omega_m = \sqrt{\omega_1\omega_2}$ peut s'exprimer en fonction de la marge de phase par la relation :

$$\phi_m = M_\Phi - \frac{\pi}{2} - \arg\Big(H(j\omega_u)\Big) - \operatorname{atan}\Big(\frac{\omega_u}{\omega_i}\Big), \tag{3.5}$$

avec $\omega_u = \omega_m$.

Sachant que

$$\alpha = \frac{1+\sin\phi_m}{1-\sin\phi_m}, \tag{3.6}$$

ω_1 et ω_2 s'écrivent :

$$\omega_1 = \omega_u\sqrt{\alpha}, \quad \omega_2 = \omega_u/\sqrt{\alpha}. \tag{3.7}$$

Enfin, le gain C_0 est obtenu à partir de la relation :

$$\Big|\beta(j\omega_u)\Big| = \Big|C(j\omega_u)H(j\omega_u)\Big| = 1. \tag{3.8}$$

3.2.2 La commande CRONE

La commande CRONE (Commande Robuste d'Ordre Non Entier) est un régulateur robuste pour les systèmes à paramètres incertains ou perturbés. La synthèse du régulateur se fait exclusivement dans le domaine fréquentiel en déterminant la fonction de transfert de la boucle ouverte grâce à des dérivées ou intégrations d'ordre non entier. Depuis le début de la commande CRONE dans les années 1980, trois générations de commande CRONE ont été développées [140, 142, 141].

Commande CRONE : première génération

L'avantage de la commande CRONE par rapport à d'autres méthodes de régulation est que les incertitudes et/ou perturbations sont prises en compte sans distinction de nature, qu'elles soient liées à la structure du système ou totalement extérieures. En effet, l'idée derrière la commande CRONE est que, pour rendre robuste le degré de stabilité de la commande face aux variations paramétriques du procédé, il faut réduire les variations de marge de phase.

Ainsi, la première génération de commande CRONE est un régulateur à phase constante autour de la fréquence au gain unité en boucle ouverte ω_u sur un intervalle de fréquence $[\omega_a, \omega_b]$. Par conséquent, si autour de ω_u la phase du procédé est constante et la variation paramétrique du procédé n'influence pas la phase, le régulateur CRONE assure la robustesse de la marge de phase. Ce régulateur s'écrit à partir d'une cellule dérivée d'ordre non entier m :

$$C_{1GF}(s) = C_0\left(\frac{1+\dfrac{s}{\omega_L}}{1+\dfrac{s}{\omega_H}}\right)^m, \tag{3.9}$$

avec $C_0, \omega_L, \omega_H, m \in \mathbb{R}^+$ et $\omega_L < \omega_a, \omega_H > \omega_b$.

Pour l'intégration numérique d'une commande CRONE, il convient de rationaliser la forme fractionnaire (3.9) par une distribution récursive de pôles et de zéros réels :

$$C_{1GR}(s) = C_0\prod_{i=1}^{N}\frac{1+\dfrac{s}{\omega_i'}}{1+\dfrac{s}{\omega_i}}, \tag{3.10}$$

avec $N \in \mathbb{N}$ le nombre de pôles et zéros, $\omega'_i \in \mathbb{R}$ les zéros et $\omega_i \in \mathbb{R}$ les pôles.

Le calcul des pôles et des zéros utilise les relations [142] :

$$\frac{\omega'_{i+1}}{\omega'_i} = \frac{\omega_{i+1}}{\omega_i} = \alpha\eta > 1, \tag{3.11}$$

$$\frac{\omega_i}{\omega'_i} = \alpha \text{ et } \frac{\omega'_{i+1}}{\omega_i} = \eta, \tag{3.12}$$

$$\alpha\eta = \left(\frac{\omega_H}{\omega_L}\right)^{1/N}, \tag{3.13}$$

$$\alpha = (\alpha\eta)^n \text{ et } \eta = (\alpha\eta)^{1-n}, \tag{3.14}$$

$$\omega'_1 = \omega_L \eta^{1/2} \text{ et } \omega_N = \omega_H \eta^{-1/2}. \tag{3.15}$$

Généralement, les paramètres ω_i et ω'_i peuvent être calculés en fixant une valeur pour N qui assure $\alpha\eta \approx 5$.

La Figure 3.1 présente de manière schématique, les diagrammes de Bode asymptotiques des réponses fréquentielles des régulateurs CRONE fractionnaire et rationnel de première génération. La succession de zéros ω'_i et de pôles ω_i conduit à une pente de $m20$ dB/décade et une phase constante de $n\pi/2$ entre ω_a et ω_b.

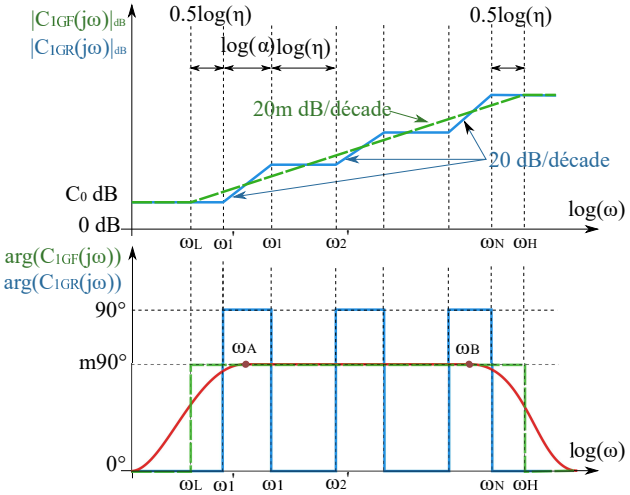


FIGURE 3.1 – Diagrammes de Bode des régulateurs CRONE rationnel et fractionnaire de première génération d'ordre m

Pour garantir une erreur statique nulle en cas de perturbations extérieures et limiter l'effort apporté par la commande, un intégrateur d'ordre m_i et un filtre passe-bas d'ordre m_h peuvent être ajoutés :

$$C_{1GF}(s) = C_0 \left(\frac{\omega_I}{s} + 1\right)^{m_i} \left(\frac{1 + \frac{s}{\omega_L}}{1 + \frac{s}{\omega_H}}\right)^m \frac{1}{\left(1 + \frac{s}{\omega_F}\right)^{m_h}}, \tag{3.16}$$

avec $m_i, m_f \in \mathbb{N}$ et $\omega_I, \omega_F \in \mathbb{R}$.

Commande CRONE : deuxième génération

Le problème de la première génération de la commande CRONE est que, compte tenu des exigences des actionneurs et du cahier des charges, il n'est pas toujours possible d'avoir une phase constante à la fréquence au gain unité ω_u désirée. La commande CRONE de deuxième génération est adaptée pour les procédés dont les variations paramétriques ont une influence uniquement sur le gain autour ω_u et dont la phase est non constante autour de cette même fréquence. Dans le cas de ces procédés, c'est le régulateur qui va devoir compenser la variation de la phase autour de ω_u.

L'idée de la seconde génération de la commande CRONE est alors de directement mettre en forme la fonction de transfert en boucle ouverte (« loop shaping ») puis d'en déduire le régulateur à partir de la fonction de transfert du système qui est connue. La boucle ouverte calculée par la commande CRONE de deuxième génération présente un gabarit vertical dans le diagramme de Nichols, assurant la robustesse aux variations de gain, de la marge de phase et des facteurs de résonance et d'amortissement. La fonction de transfert de la boucle ouverte $\beta(s)$ entre ω_a et ω_b est définie par un intégrateur non entier d'ordre $n \in \mathbb{R}$:

$$\beta_{2G}(s) = \left(\frac{\omega_u}{s}\right)^n. \tag{3.17}$$

Le lieu de la fonction de transfert est tracé sur la Figure 3.2.

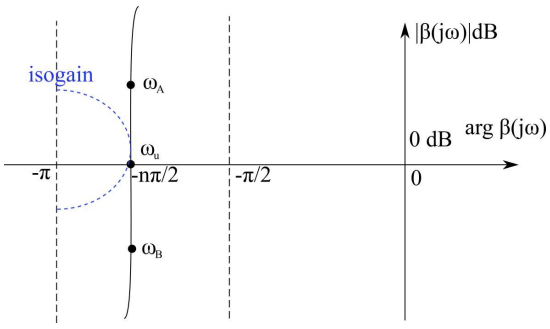


FIGURE 3.2 – Diagramme de Black-Nichols de la boucle ouverte pour une commande CRONE de deuxième génération

Comme pour la commande de première génération, un intégrateur d'ordre n_l et un filtre passe-bas d'ordre n_h sont rajoutés à la boucle ouverte pour garantir précision et rejet de perturbations. Ainsi, la fonction de transfert de la boucle ouverte s'écrit :

$$\beta_{2G}(s) = \beta_0 \left(\frac{\omega_L}{s} + 1\right)^{n_l} \left(\frac{1 + \dfrac{s}{\omega_H}}{1 + \dfrac{s}{\omega_L}}\right)^n \frac{1}{\left(1 + \dfrac{s}{\omega_H}\right)^{n_h}}, \tag{3.18}$$

avec ω_L et ω_H les fréquences transitionnelles basse et haute fréquence au voisinage de ω_u, β_0 le gain de la boucle ouverte, et n l'ordre de la commande compris entre 1 et 2.

Pour avoir le comportement asymptotique non entier sur tout l'intervalle $[\omega_a, \omega_b]$, il faut laisser une décade entre ω_L et ω_a et entre ω_H et ω_b [140] :

$$\begin{cases} \omega_L = \omega_a/10 \\ \omega_H = 10\omega_b. \end{cases} \tag{3.19}$$

D'autre part, pour respecter la robustesse du degré de stabilité, il faut imposer $\omega_a \leq \omega_u \leq \omega_b$. Les diagrammes de Bode asymptotique de la boucle ouverte sont tracés sur la Figure 3.3

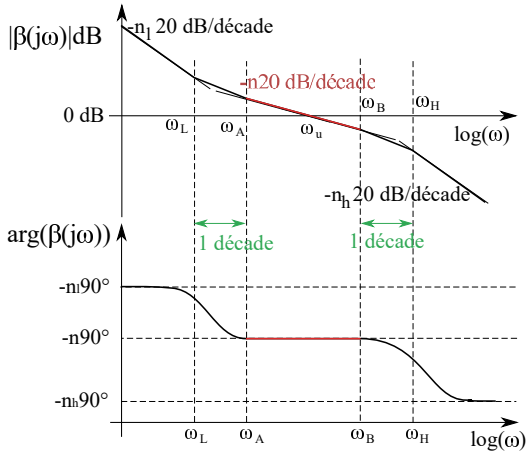


FIGURE 3.3 – Diagramme de Bode de la boucle ouverte de la commande CRONE de deuxième génération

Commande CRONE : troisième génération

Dans les cas où les variations paramétriques du procédé entraînent plus qu'un déplacement vertical du lieu de la fonction de transfert en boucle ouverte dans le diagramme de Black-Nichols, la commande CRONE de deuxième génération ne suffit plus. Des domaines d'incertitude existent donc autour du gabarit vertical et il s'agit de trouver une méthode pour les limiter. La troisième génération de la commande CRONE propose d'incliner la droite du gabarit en utilisant une fonction de transfert en boucle ouverte qui est la partie réelle d'un intégrateur d'ordre non entier complexe, et non plus réel comme dans la deuxième génération. La nouvelle fonction de transfert, avec un gabarit généralisé, s'écrit donc :

$$\beta_{3G}(s) = \mathrm{Re}_{/i}\left(\frac{\omega_u}{s}\right)^n, \tag{3.20}$$

avec $n = a + ib \in \mathbb{C}$.

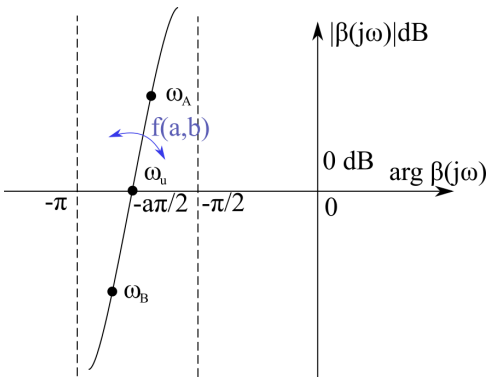


FIGURE 3.4 – Gabarit généralisé d'une commande CRONE de troisième génération dans le diagramme de Black

La partie réelle de l'ordre non entier complexe détermine la position en phase du gabarit dans le lieu de Black et la partie imaginaire détermine son inclinaison par rapport à la verticale, Figure 3.4.

Le fait de pouvoir incliner le gabarit généralisé donne une infinité de solutions dans le choix du gabarit pour un même contour à isogain Q_{r0} à tangenter par le lieu de la boucle ouverte. Le principe de la commande CRONE de troisième génération est d'optimiser ce gabarit généralisé de manière à limiter au maximum les variations du facteur de résonance Q_r suivant les incertitudes paramétriques du procédé ou des perturbations. Comme pour la commande CRONE de deuxième génération, la fonction de transfert d'ordre non entier de la boucle ouverte est définie sur une certaine plage de fréquence et la boucle ouverte complète s'écrit :

$$\beta_{3G} = \beta_L(s)\beta_m(s)\beta_H(s), \tag{3.21}$$

avec $\beta_m(s)$ un ensemble de modèles généralisés tel que :

$$\beta_m(s) = \prod_{k=-N^-}^{N^+} C_k^{\mathrm{sign}(b_k)} \left(\alpha_k \frac{1+\dfrac{s}{\omega_{k+1}}}{1+\dfrac{s}{\omega_k}} \right)^{a_k} \left(\mathrm{Re}_{/i} \left\{ \left(\alpha_k \frac{1+\dfrac{s}{\omega_{k+1}}}{1+\dfrac{s}{\omega_k}} \right)^{ib_k} \right\} \right)^{-q_k \mathrm{sign}(b_k)}, \tag{3.22}$$

avec

$$\alpha_k = \left(\frac{\omega_{k+1}}{\omega_k} \right)^{1/2} \text{ pour } k \neq 0 \text{ et } \alpha_0 = \left(\frac{1+\left(\dfrac{\omega_r}{\omega_0}\right)^2}{1+\left(\dfrac{\omega_r}{\omega_1}\right)^2} \right)^{1/2}, \tag{3.23}$$

et

$$\beta_L(s) = C_L \left(\frac{\omega_{N^-}}{s} + 1 \right)^{n_l} \text{ et } \beta_H(s) = C_H \left(\frac{s}{\omega_{N^+}} + 1 \right)^{-n_h}, \tag{3.24}$$

sachant $N^+, N^-, q_k \in \mathbb{N}$; $\omega_r, \omega_k, \alpha_k, C_k, C_L, C_H, a_k, b_k \in \mathbb{R}$.

Dans la mesure où l'on cherche à minimiser les variations du facteur d'amortissement, Figure 3.5, le critère permettant de trouver les paramètres de la boucle ouverte limitant ces variations s'écrit :

$$J = (Q_{r0} - Q_{rmin})^2 + (Q_{rmax} - Q_{r0})^2, \tag{3.25}$$

avec Q_{r0} le facteur de résonance désiré pour les paramètres nominaux du procédé et Q_{rmin} et Q_{rmax} les facteurs de résonance minimum et maximum tangentés par les domaines d'incertitude.

En plus du critère sur le facteur de résonance, des critères sur les limites hautes et basses des fonctions de sensibilité sont ajoutés de sorte que :

$$\begin{aligned} \min|T(j\omega)| &\geq T_{min} \\ \max|T(j\omega)| &\leq T_{max} \\ \max|S(j\omega)| &\leq S_{max} \\ \max|CS(j\omega)| &\leq CS_{max} \\ \max|GS(j\omega)| &\leq GS_{max}. \end{aligned} \tag{3.26}$$

En pratique, une commande CRONE de troisième génération est générée à partir d'une toolbox sur Matlab. Cette toolbox permet de visualiser les domaines d'incertitude et comment

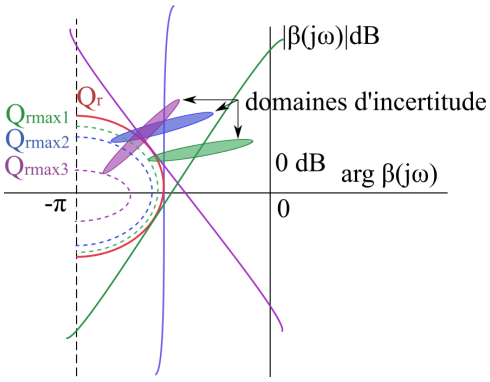


FIGURE 3.5 – Influence du gabarit généralisé sur les domaines d'incertitude

ceux-ci bougent en fonction des paramètres de la boucle ouverte. Une fois que tous les critères de la boucle ouverte sont respectés, un placement graphique de couples pôle/zéro permet d'estimer la forme rationnelle de la boucle ouverte et donc du régulateur. Ce placement graphique permet de définir les paramètres initiaux utilisés ensuite lors d'une procédure d'optimisation effectuée par la toolbox de manière à ce que la forme fractionnaire de la boucle ouverte se rapproche le plus possible de la forme rationnelle.

3.3 Régulation latérale par PID

Dans cette section, un PID classique, largement employé dans l'industrie, est utilisé pour réguler la position latérale $Y_G(t)$ du véhicule. La commande utilisée est l'angle volant $\theta_v(t)$, appliqué par la DAE (Direction Assistée Électrique). Dans toute la suite de l'étude, on suppose que la dynamique de la boucle de commande interne de la DAE est suffisamment rapide pour suivre précisément la commande en angle volant désirée pour le guidage latéral. Le schéma de synthèse pour le calcul du régulateur est présenté en Figure 3.6.

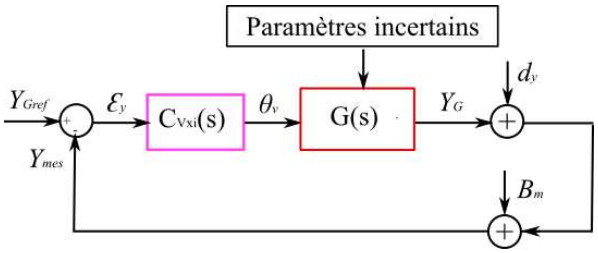


FIGURE 3.6 – Schéma de synthèse pour le calcul de la régulation latérale par PID simple

Sur la Figure 3.6, $Y_{mes}(t)$ est la position latérale $Y_G(t)$ mesurée, $B_m(t)$ représente le bruit de mesure et $d_y(t)$ est l'équivalent en déplacement d'une perturbation en sortie résultant d'une rafale de vent $f_{0y}(t)$. $G(s)$ est la fonction de transfert $Y_G(s)/\Theta_v(s)$ et $C_{Vxi}(s)$ est le PID calculé au point de fonctionnement avec une vitesse longitudinale V_{xi}. Le véhicule, et donc sa représentation fréquentielle, est soumis à des paramètres incertains pouvant varier. Ces paramètres incluent : la vitesse longitudinale, l'adhérence de la route, la masse du véhicule et tous les paramètres qui dépendent de cette dernière.

Le PID est calculé pour une vitesse nominale $V_x = 90$ km/h. Pour ce régulateur, et pour tous

ceux qui sont calculés par la suite, le cahier des charges retenu correspondant aux spécificités des actionneurs des prototypes est le suivant :

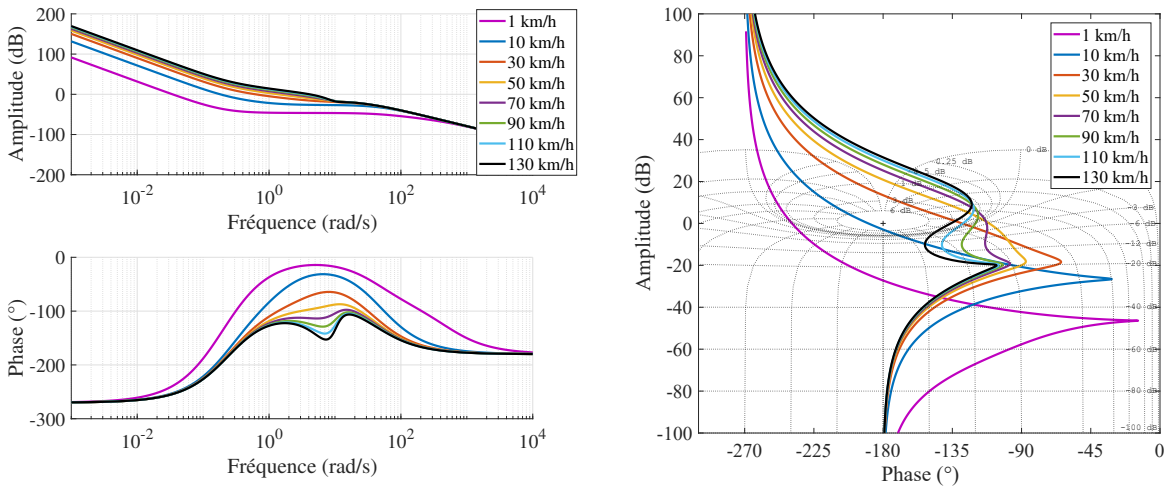
- Pulsation au gain unité $\omega_u = 3$ rad/s
- Marge de phase $M_\Phi = 60°$.

En utilisant la fonction de transfert du modèle linéaire $G_L^{(2)}(s)$ décrite au paragraphe 2.1.4 et en utilisant les équations (3.2) à (3.8) pour le calcul du PID. $C_{V_{xi}}$ est de la forme (3.1) avec les valeurs numériques de la Table 3.1.

$$\begin{aligned} C_0 &= \quad 0{,}0752 \text{ rad/m} \\ \omega_i &= \quad 0{,}3 \text{ rad/s} \\ \omega_1 &= \quad 0{,}2036 \text{ rad/s} \\ \omega_2 &= \quad 44{,}20 \text{ rad/s} \end{aligned}$$

TABLE 3.1 – Valeurs numériques des paramètres du PID

Les diagrammes de Bode et de Nichols de la boucle ouverte sont tracés sur les Figures 3.7 (a) et (b).



(a) (b)

FIGURE 3.7 – Réponses fréquentielles de la boucle ouverte dans les diagrammes de Bode (a) et de Black-Nichols (b) pour le PID simple

Sur le diagramme de Bode en amplitude de la boucle ouverte, une grande variation de la pulsation au gain unité peut être observée, celle-ci variant entre $0,035$ rad/s à 1 km/h et $4,11$ rad/s pour 130 km/h. Le système bouclé risque donc de ne pas être assez rapide à basse vitesse. Le diagramme de Nichols montre même que le système est instable pour des vitesses longitudinales inférieures à 10 km/h. De plus, pour les vitesses supérieures à 90km/h, plus V_x augmente et plus les valeurs des contours d'amplitude tangentés par les courbes sont importantes provoquant ainsi une diminution du degré de stabilité.

Les fonctions de sensibilité, décrites en Annexe A, sont tracées sur la Figure 3.8.

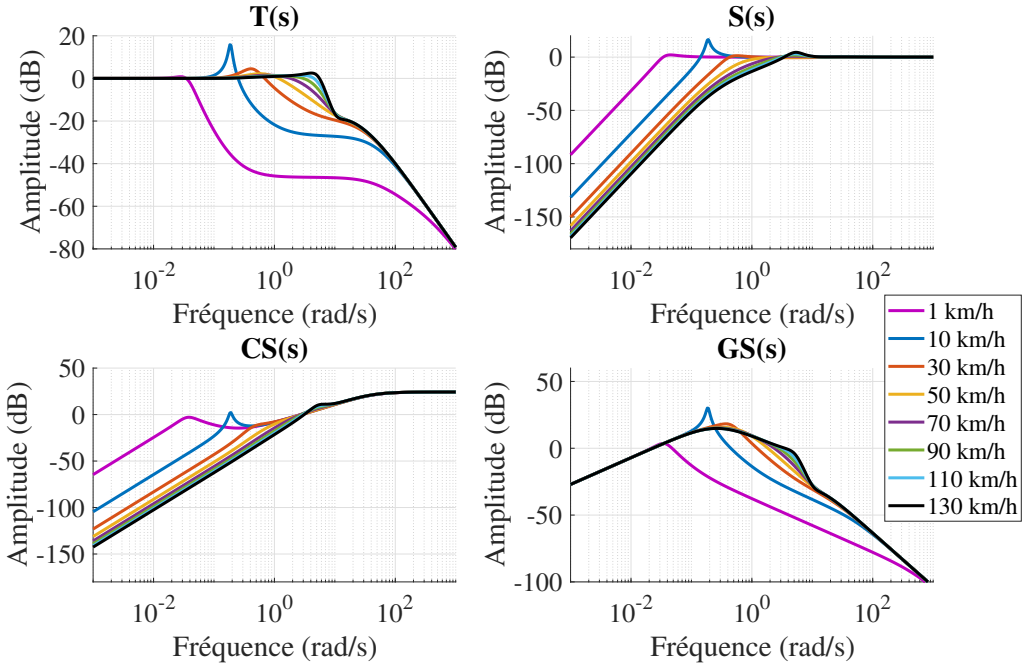


FIGURE 3.8 – Fonctions de sensibilité pour le PID simple

Le tracé de la fonction de sensibilité complémentaire $T(s)$ indique qu'il existe un facteur de résonance plus important à basse vitesse. Plus la vitesse diminue et plus le système bouclé est sensible à l'entrée ($CS(s)$) mais moins sensible aux perturbations en entrée $GS(s)$.

Le comportement du PID est ensuite analysé dans le domaine temporel. En entrée, la trajectoire de référence Y_{Gref} est celle d'un changement de voie, débutant à 1 s et durant 5 s pour ainsi atteindre $Y_{Gref} = 3,5$ m à $t = 6$ s. Ce changement de voie est effectué pour des vitesses longitudinales fixes entre 10 km/h et 130 km/h. Pour toutes les simulations temporelles de ce chapitre, le modèle véhicule utilisé pour le système bouclé est le modèle de validation non linéaire.

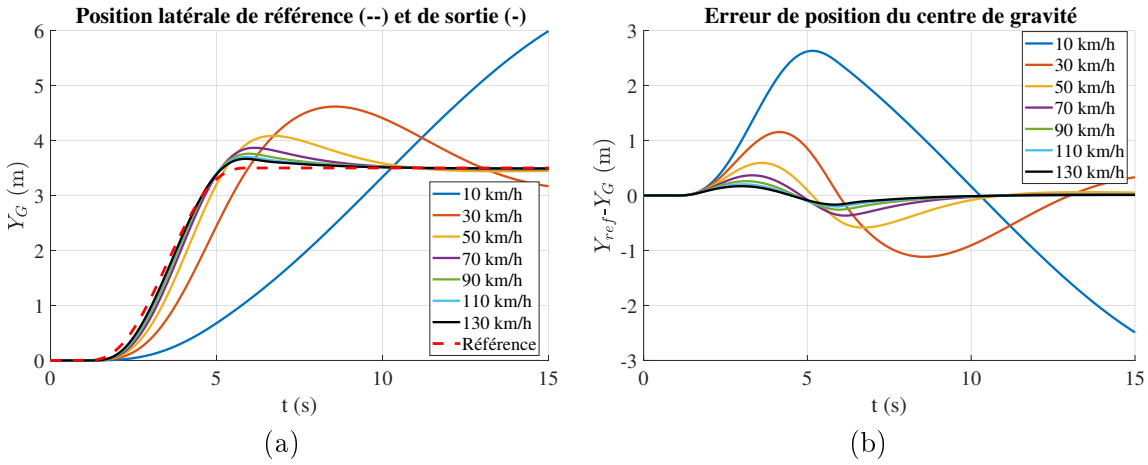
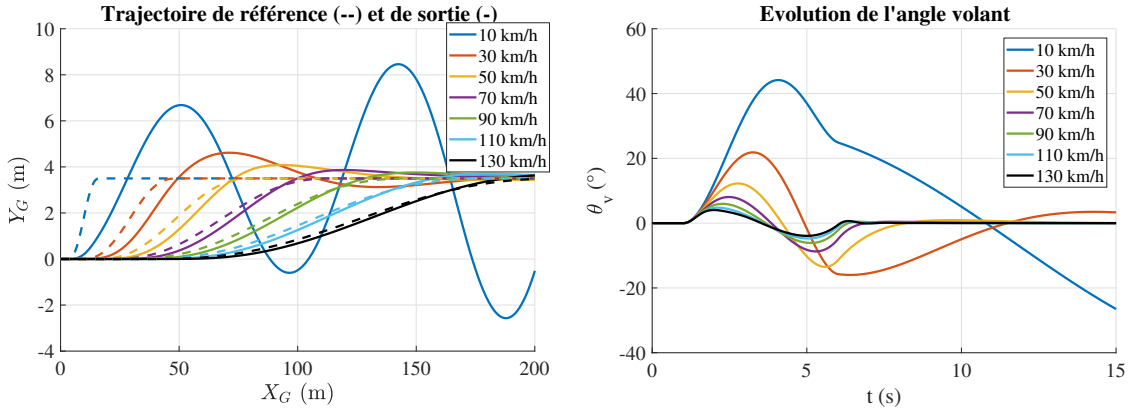


FIGURE 3.9 – Évolution de la position latérale du centre de gravité $Y_G(t)$ (a) et l'erreur de position par rapport à la référence $Y_{Gref}(t)$ (b)

Comme observé sur les figures fréquentielles, le comportement du système bouclé est plus lent à basse vitesse et instable à très basse vitesse, Figure 3.9 (a). Plus la vitesse augmente et plus l'erreur en position diminue, Figure 3.9 (b), avec une erreur relative maximale de 16 % pour $V_x = 50$ km/h et de 4,6 % pour $V_x = 130$ km/h.

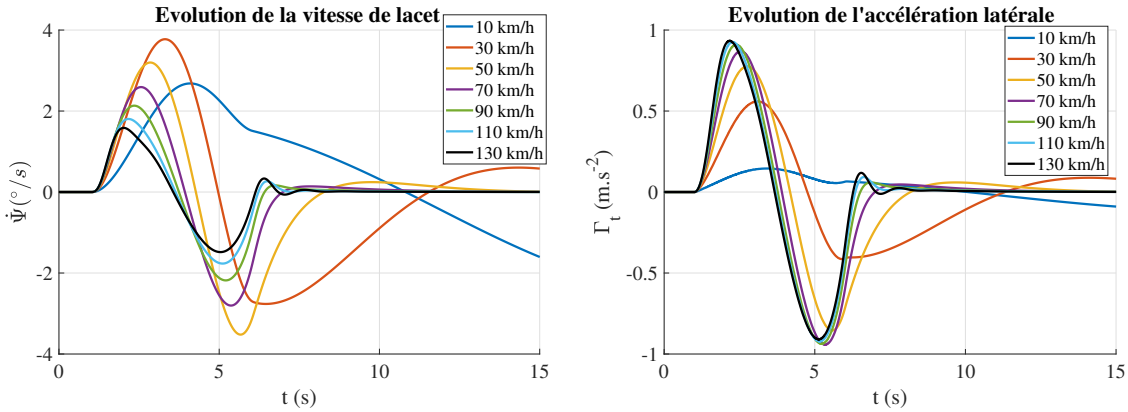
Les trajectoires des changements de voie, c'est-à-dire la position latérale en fonction de la position longitudinale, sont tracées sur la Figure 3.10 (a), avec la position latérale de référence en pointillés et la position latérale de sortie en trait plein. Cette figure permet de mieux apprécier la latence et le dépassement du changement de voie à basse vitesse. La sortie du régulateur, qui est ici la commande en angle volant envoyé à la direction est tracée sur la Figure 3.10 (b).



(a) (b)

FIGURE 3.10 – Trajectoires de références et trajectoires réelles du véhicule pour les différentes vitesses (a) et évolution de l'angle volant $\theta_(t)v$(b)

Le fait que la valeur des contours d'amplitude tangentés par la boucle ouverte avec le PID augmente après 90 km/h s'évalue ici sur la vitesse de lacet et l'accélération latérale. En effet, la diminution du degré de stabilité évoquée précédemment sur les diagrammes de Nichols se traduit ici par des oscillations à la fin de la manœuvre de dépassement, Figure 3.11 (a) et (b). L'accélération latérale étant inférieure à 1 m.s^{-2}, Figure 3.11 (b), pour toutes les simulations avec le PID, celles-ci sont bien représentatives de changements de voie réels.



(a) (b)

FIGURE 3.11 – Évolution de la vitesse de lacet $\dot{\Psi}(t)$ (a) et de l'accélération latérale $\Gamma_t(t)$ pour les différentes vitesses (b)

Remarque. Peu importe la vitesse choisie pour le point de fonctionnement ou la marge de phase utilisée pour le réglage du PID, il y aura toujours des cas d'instabilité. Lorsque le point de fonctionnement est à vitesse faible, il existe aussi des problèmes d'instabilité. En effet, étant donné la variation de phase de $G(s)$ à ω_u, un PID réglé à basse vitesse a tendance à apporter du retard de phase à ω_u alors que le procédé à haute vitesse a besoin que le régulateur apporte de l'avance de phase à la fréquence au gain unité désirée pour pouvoir répondre au cahier des charges.

La méthode de base étant inappropriée pour le cas d'étude présenté ici, il faut trouver une autre méthode de régulation permettant d'assurer la stabilité du système bouclé, quelle que soit la vitesse longitudinale V_x du véhicule.

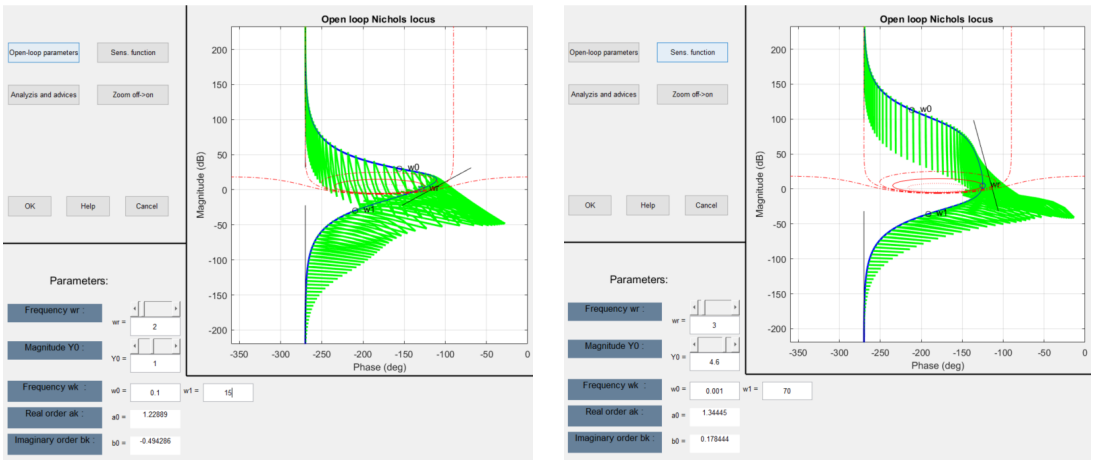
3.4 Régulation latérale par CRONE de 3ème génération

À la section 3.2.2, nous avons présenté la régulation fractionnaire CRONE. La troisième génération de la commande CRONE est adaptée à la régulation de procédés dont les variations paramétriques entraînent des variations de phase et de gain à la fréquence au gain unité désirée. La variation de la vitesse longitudinale ayant un impact important sur le procédé considéré $G_L^{(2)}(s)$, Figure 2.27, la troisième génération de la commande CRONE est donc appropriée pour la régulation de la dynamique latérale du véhicule.

La commande CRONE de troisième génération est générée à l'aide de la CRONE Control Toolbox sur Matlab/Simulink. En entrée, la toolbox reçoit le procédé avec plusieurs jeux de paramètres, ici c'est la fonction de transfert $G_L^{(2)}(s)$ pour les vitesses tracées sur la Figure 2.27. Sont aussi indiqués le procédé nominal, $V_{x0} = 90$ km/h et les extrémaux, $V_{x0} = 1$ km/h et $V_{x0} = 130$ km/h.

Il faut ensuite, à l'aide de l'interface graphique de la toolbox, choisir les paramètres de la fonction de transfert de la boucle ouverte du procédé nominal pour éloigner les domaines d'incertitude du point critique et respecter le cahier des charges. La Figure 3.12 (a) montre un exemple de paramétrage de la boucle ouverte. La réponse fréquentielle en rouge représente le procédé nominal à 90 km/h et les zones vertes représentent les domaines d'incertitudes. La Figure 3.12 (b) est le paramétrage optimal.

Une fois la boucle ouverte calibrée, il faut déterminer, à partir de la forme fractionnaire de la boucle ouverte, sa forme rationnelle. Cette étape se fait aussi à partir d'une interface graphique de la toolbox pour positionner des pôles et des zéros afin de faire correspondre une fonction de transfert rationnelle à celle fractionnaire, Figure 3.13.



(a) (b)

FIGURE 3.12 – Exemples de paramétrages de la boucle ouverte de la commande CRONE de troisième génération dans la toolbox CRONE

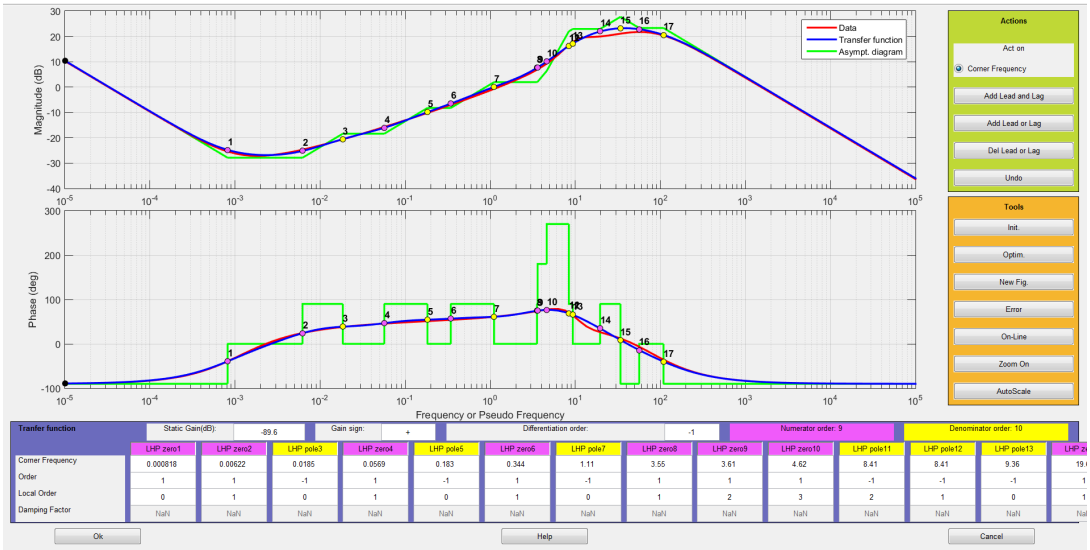


FIGURE 3.13 – Placement de pôles et de zéros dans la toolbox CRONE pour trouver la fonction de transfert rationnelle de la boucle ouverte

Finalement, la commande CRONE de troisième génération pour le contrôle latéral du véhicule est de la forme :

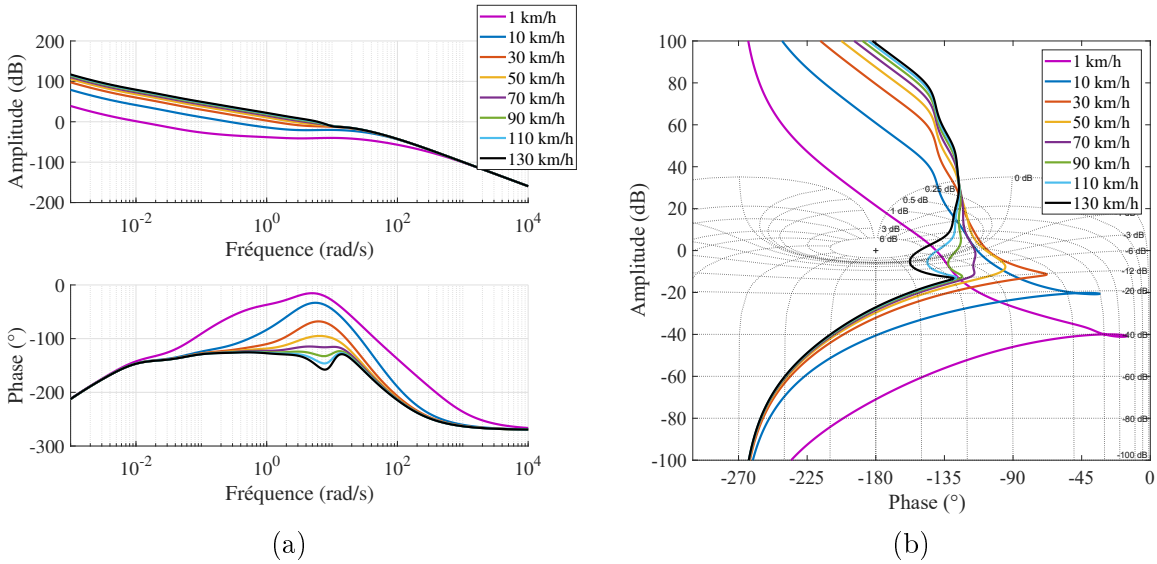
$$\text{CRONE3G}(s) = \frac{C_0}{s} \prod_{k=1}^{9} \left(\frac{1 + s/\omega_k}{1 + s/\omega'_k} \right), \tag{3.27}$$

dont les valeurs numériques figurent dans la Table 3.2. Comparé au PID, l'ordre du régulateur est beaucoup plus grand, passant de l'ordre 2 à l'ordre 10.

Il est maintenant possible d'analyser la boucle ouverte dans le domaine fréquentiel à partir des tracés des réponses fréquentielles dans les diagrammes de Bode, Figure 3.14 (a), et le diagramme de Black-Nichols 3.14 (b).

C_0 ($s^{-1}.m^{-1}$)	$3.311\ 10^{-5}$								
k	1	2	3	4	5	6	7	8	9
ω_k (rad/s)	$8{,}18\ 10^{-4}$	$6{,}22\ 10^{-3}$	0,057	0,344	3,55	3,61	4,62	19,6	56,5
ω'_k (rad/s)	0,0185	0,183	1,11	8,41	8,41	9,36	34	34	109

TABLE 3.2 – Valeurs numériques des paramètres du régulateur CRONE de troisième génération



(a) (b)

FIGURE 3.14 – Réponses fréquentielles de la boucle ouverte dans les diagrammes de Bode (a) et de Black-Nichols (b) pour la commande CRONE de 3ème génération

Le fait de pouvoir orienter les domaines d'incertitude, et de les visualiser à l'aide de la toolbox, permet de trouver des paramètres de boucle ouverte garantissant la stabilité du système régulé à toutes les vitesses. Cependant, la commande CRONE introduit une variation de la fréquence au gain unité en fonction de la vitesse encore plus grande qu'avec le PID : à 1 km/h, $\omega_u = 0,011$ rad/s et à 130 km/h, $\omega_u = 6,03$ rad/s. Les contours d'amplitude de l'abaque de Nichols tangentés par les réponses fréquentielles de la boucle ouverte sont aussi plus grands à grande vitesse avec le CRONE qu'avec le PID. Le système bouclé présente donc un plus petit degré de stabilité à grande vitesse avec le CRONE qu'avec le PID mais est globalement plus précis avec la commande CRONE.

Toujours dans le domaine fréquentiel, les fonctions de sensibilités sont tracées Figure 3.15.

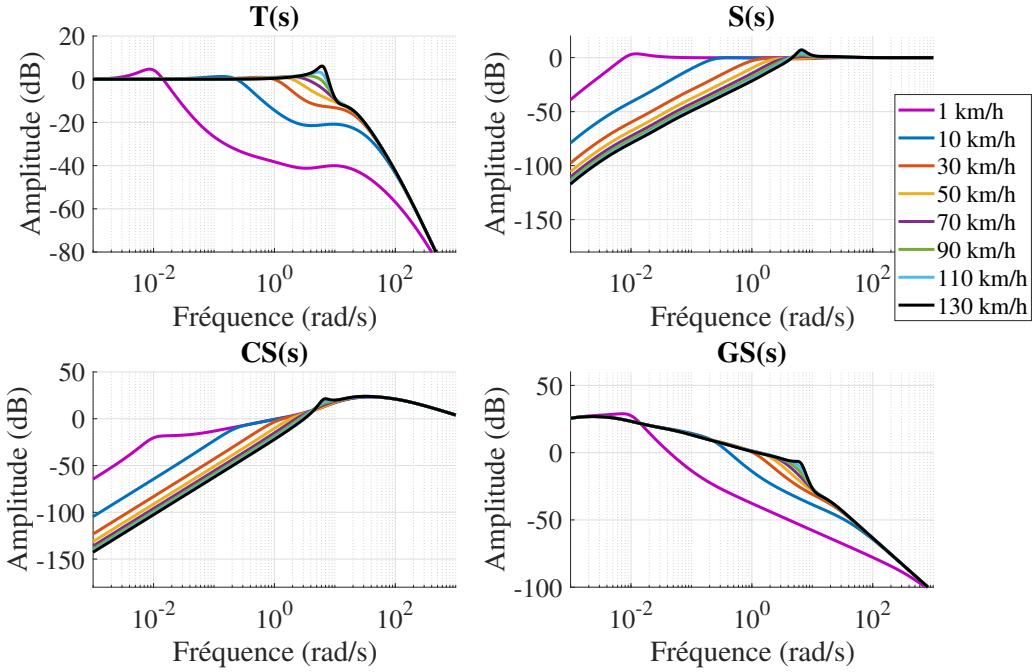


FIGURE 3.15 – Fonctions de sensibilité pour la commande CRONE de 3ème génération

Par rapport au PID, les résonances de la fonction de sensibilité complémentaire $T(s)$ avec le CRONE entre 10 km/h et 30 km/h disparaissent. Cependant, la résonance augmente à haute vitesse et très basse vitesse. Les fonctions de sensibilité $S(s)$ et $GS(s)$ indiquent quant à elles que la régulation avec la commande CRONE est plus sensible qu'avec le PID à basse fréquence et que le PID rejette mieux les perturbations en entrée, toujours à basse fréquence.

Pour comparer le PID et la commande CRONE dans le domaine temporel, la même simulation de changement de voie qu'avec le PID est utilisée.

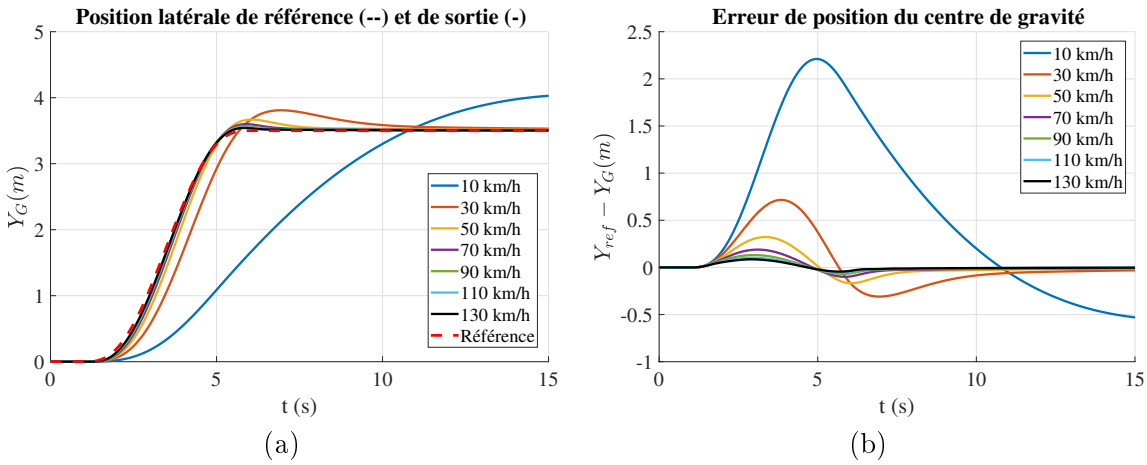
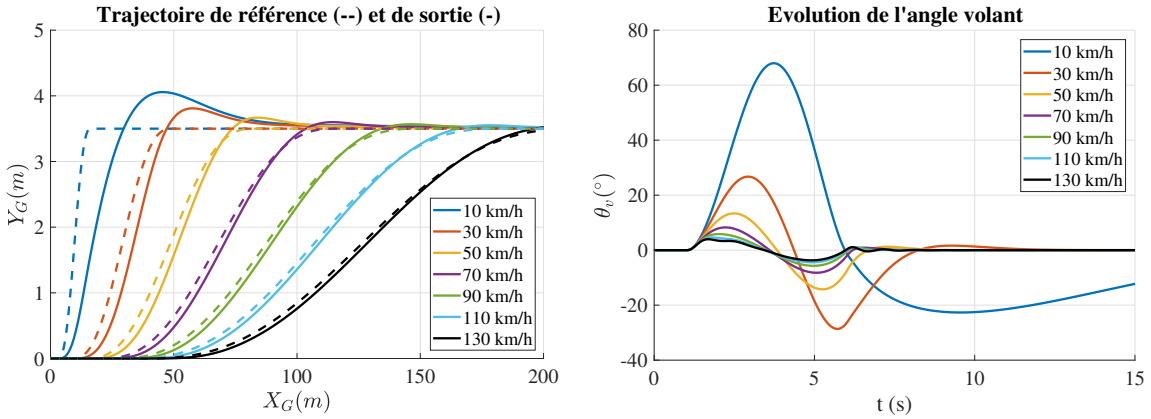


FIGURE 3.16 – Évolution de la position latérale du centre de gravité $Y_G(t)$ (a) et l'erreur de position par rapport à la référence $Y_{Gref}(t)$ (b)

Comme observé sur les figures des réponses fréquentielles de la boucle ouverte, la commande

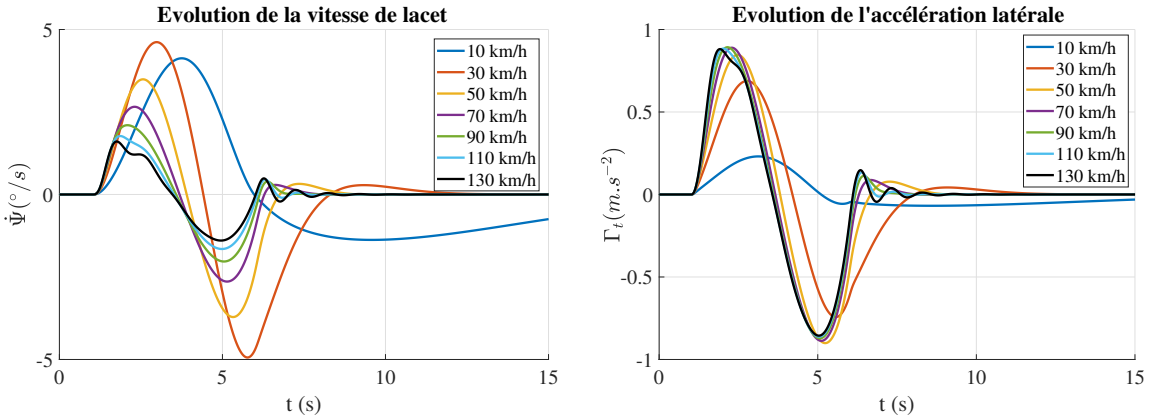
CRONE garantit la stabilité à toutes les vitesses V_{x0}. Par rapport au PID, l'erreur en position est plus faible pour toutes les vitesses (Figure 3.16 (b)). Par contre, comme pour le PID, à basse vitesse le système est trop lent, conduisant à des dépassements supérieurs à 50 cm. Plus la vitesse augmente et plus ces dépassements sont faibles.



(a) (b)

FIGURE 3.17 – Trajectoires de références et trajectoires réelles du véhicule pour les différentes vitesses (a) et évolution de l'angle volant $\theta_v(t)$ (b)

Le fait que la boucle ouverte tangente des contours d'amplitude plus grands à haute vitesse dans le domaine fréquentiel conduit à des oscillations de l'angle volant, Figure 3.17 (b), de la vitesse de lacet, Figure 3.18 (a), et de l'accélération latérale, Figure 3.18 (b), dans le domaine temporel.



(a) (b)

FIGURE 3.18 – Évolution de la vitesse de lacet $\dot{\Psi}(t)$ (a) et de l'accélération latérale $\Gamma_t(t)$ pour les différentes vitesses (b)

La commande CRONE est plus adaptée que le PID pour la régulation de la dynamique latérale. Elle garantit la stabilité pour toute vitesse V_{x0}. Cependant, la grande variation du gain et de la phase de procédé par rapport à la vitesse à la fréquence au gain unité ne permet pas de trouver des paramètres de la boucle ouverte de la commande CRONE de 3ème génération respectant le cahier des charges à toutes les vitesses.

La vitesse étant une variable mesurable, c'est d'ailleurs une des mesures les plus fiables à bord du véhicule, il apparaît opportun de développer une méthode de régulation tenant compte de la mesure de la vitesse longitudinale à chaque instant.

3.5 Approche multirégulateur

Lorsque le procédé à contrôler est complexe et que sa dynamique varie au cours du temps, un PID seul peut ne pas garantir les performances désirées et respecter le cahier des charges dans certaines configurations du système.

Une vue d'ensemble des méthodes est présentée dans [113]. Les problèmes de ces méthodes sont de trouver les points de fonctionnement où linéariser les modèles pour faire le calcul des régulateurs et de trouver comment combiner les régulateurs calculés. Dans la plupart des études, comme dans [40], les points de fonctionnement sont choisis manuellement. Il existe une méthode émergente, appelée « gap metric » en anglais, qui permet de calculer les points de fonctionnement [174]. Comme indiqué dans [48], les deux méthodes d'interpolation des régulateurs sont des switchs ou des poids avec des fonctions gaussiennes, trapézoïdales ou bayésiennes. Dans [44] par exemple, les gains des composantes P, I et D du régulateur sont calculés utilisant des résultats expérimentaux pour 3 vitesses, puis interpolés.

D'autres recherches ont déjà utilisé le principe de multirégulation pour le contrôle de la direction. Dans [9], une commande par retour d'état avec multirégulateur est présentée. Chaque régulateur est calculé par une minimisation de fonction de Lyapunov quadratique en utilisant des inégalités matricielles. Un switch sélectionne le régulateur en fonction d'une estimation de l'angle de dérive des pneumatiques. Les switchs peuvent être évités grâce à une approche Takagi-Sugeno pour le calcul du régulateur [121]. Les sous-systèmes sont fonction de la rigidité de dérive des pneumatiques et les régulateurs sont aussi calculés avec des inégalités matricielles. Cependant, ces recherches sur la multirégulation ne traitent pas de la variation de la vitesse longitudinale. Même si des contraintes peuvent être ajoutées aux problèmes d'optimisation présentés, il survient souvent des problèmes de résolution numérique. Et même si une solution est trouvée, la complexité du problème rend la compréhension des effets de chaque paramètre du régulateur difficile.

La méthode présentée ici utilise plusieurs régulateurs, pas spécifiquement de type PID, calculés pour des points de fonctionnement différents et pondérés par des variables d'activation. Dans la littérature, il n'existe pas de méthode explicite pour choisir les points de fonctionnement de telles structures de régulation. Dans la plupart des cas, ces points sont choisis de manière empirique.

Pour cette méthode, le procédé à réguler s'écrit avec une représentation d'état de type :

$$\begin{cases} \dot{\underline{x}}(t) = A(\xi(t))\underline{x}(t) + B(\xi(t))u(t) \\ y(t) = C\underline{x}(t), \end{cases} \tag{3.28}$$

avec $\underline{x}(t)$ le vecteur d'état, $u(t)$ la commande en entrée, $y(t)$ la sortie et $A(\xi(t))$, $B(\xi(t))$, C sont les matrices d'état. Ces deux premières matrices dépendent d'un paramètre $\xi(t)$ variant dans le temps et mesurable conduisant à un système LPV.

Le paramètre ξ est considéré borné et pouvant varier sur un intervalle $[\xi_{min}, \xi_{max}]$. L'objectif du multirégulateur est de réguler le système sur toute cette plage de variation. ξ est par

conséquent utilisé pour le choix des points de fonctionnement et des fonctions de pondération de chaque régulateur.

Le multirégulateur pondéré s'écrit

$$C(s) = \sum_{i=1}^{N_{op}} w_i(\xi) C_i(s), \tag{3.29}$$

avec N_{op} le nombre de points de fonctionnement et w_i les fonctions de pondération. Pour la régulation d'un procédé modélisé par une fonction de transfert $G(s)$, le schéma bloc de la multirégulation se présente alors comme sur la Figure 3.19

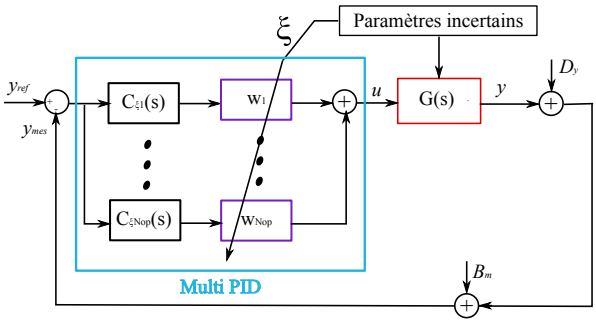


FIGURE 3.19 – Schéma bloc de la multirégulation

La difficulté concernant le choix des points de fonctionnement est de déterminer :

- N_{op} : le nombre de points de fonctionnement,
- ξ_i : la valeur du paramètre variant au i^{eme} point de fonctionnement, avec $i \in \mathbb{N}$ tel que $i \in [\![1; N_{op}]\!]$.

Une méthode intuitive pour choisir les valeurs de ξ_i est de diviser l'intervalle $[\xi_{min}, \xi_{max}]$ en $N_{op} - 1$ sous-intervalles de même taille de sorte que $\forall i \in [\![1; N_{op} - 1]\!]$:

$$\xi_{i+1} - \xi_i = \text{Var}_\xi, \tag{3.30}$$

avec Var_ξ la variation de ξ entre deux points de fonctionnement.

Cependant, ξ n'a pas forcément une influence linéaire sur la dynamique du système et cette méthode ne donne pas N_{op} qui doit toujours être choisi manuellement.

Pour trouver les points de fonctionnement, la méthode développée dans cette thèse utilise la variation de la phase du procédé à la fréquence au gain unité en fonction de ξ. De cette manière, N_{op} et ξ_i sont choisis de sorte qu'entre deux points de fonctionnement consécutifs ξ_i et ξ_{i+1} la phase du procédé $G(s)$ varie de Var_ϕ :

$$\left| \arg\left(G(j\omega_u)|_{\xi_{i+1}} \right) - \arg\left(G(j\omega_u)|_{\xi_i} \right) \right| = \text{Var}_\phi. \tag{3.31}$$

La paramétrisation de Var_ϕ au lieu de N_{op} et Var_ξ permet de trouver un nombre de points de fonctionnement avec une distribution plus optimale. Var_ϕ est choisi en faisant un compromis

entre robustesse, performance et nombre de sous systèmes. La valeur empirique choisie dans le cas pratique de la régulation latérale est $\mathrm{Var}_\phi = 15°$.

Une fois les ξ_i identifiés, les régulateurs $C_i(s)$ sont calculés en chaque point de fonctionnement.

Comme énoncé précédemment, chaque régulateur est activé en utilisant des pondérations. Afin de conserver la continuité entre les activations des régulateurs ξ_i, des fonctions sigmoïdes sont utilisées comme fonctions d'activation des régulateurs. Une sigmoïde $f_\kappa(x)$, avec $x \in \mathbb{R}$, $\kappa \in \mathbb{R}$, s'écrit :

$$f_\kappa(x) = \frac{1}{1 + e^{-\kappa x}}. \tag{3.32}$$

Les pondérations w_i des régulateurs correspondant au point de fonctionnement à ξ_i sont calculées avec la relation :

$$\begin{aligned} w_1(\xi) &= 1 - f_{\kappa_1}(\xi) \\ w_i(\xi) &= 1 - f_{\kappa_i}(\xi) + f_{\kappa_{i-1}}(\xi), \text{ Pour } i > 1 \text{ et } i < N_{op} \\ w_{N_{op}}(\xi) &= f_{\kappa_{N_{op}-1}}(\xi) \end{aligned} \tag{3.33}$$

sachant $1 - f_{\kappa_i}(\xi)$ une sigmoïde sur l'intervalle $[\xi_i;\ \xi i + 1]$ et $f_{\kappa_{i-1}}(\xi)$ une sigmoïde sur l'intervalle $[\xi_{i-1};\ \xi_i]$.

L'Annexe D compare différents choix de positionnement des points de fonctionnement pour le mulirégulateur. Il est ainsi montré qu'un positionnement en fonction de Var_ϕ garantit un meilleur degré de stabilité.

3.6 Régulation latérale par multi-PID

La multirégulation par PID consiste à utiliser plusieurs PID et à les pondérer par une variable d'activation. Comme l'objectif est de pouvoir réguler la position latérale du véhicule à toutes les vitesses, la variable d'activation qui sert à pondérer chaque PID est V_{x0}. La structure de cette régulation est présentée Figure 3.20.

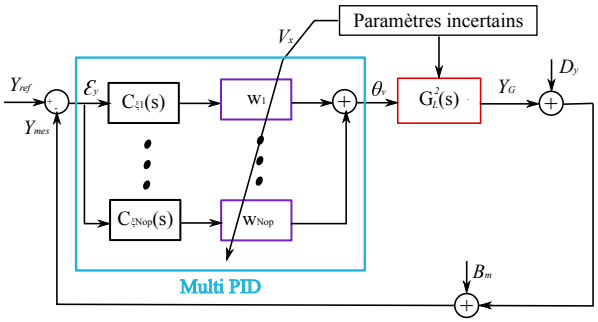


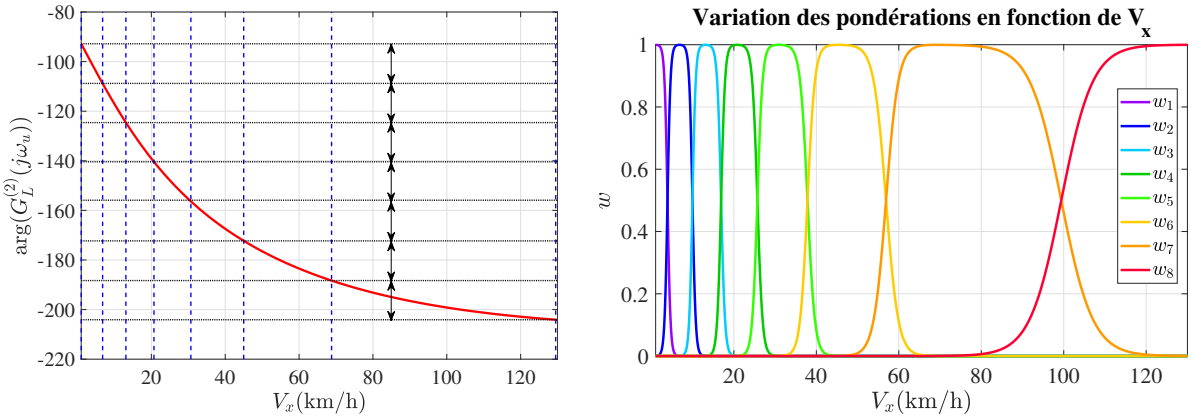
FIGURE 3.20 – Schéma de synthèse pour le calcul de la régulation latérale par multi-PID

Pour le i^{eme} point de fonctionnement ξ_i, le régulateur associé est noté $C_{\xi_i}(s)$. Il y a N points de fonctionnement, chacun pondéré par w_i qui dépend de la variable d'activation $V_x(t)$.

La méthode de multirégulation décrite dans le paragraphe précédent est utilisée pour calculer les points de fonctionnement du nouveau régulateur. Pour pouvoir comparer efficacement les

régulateurs entre eux, le cahier des charges reste inchangé : $\omega_u = 3$ rad/s et $M_\Phi = 60°$.

Le choix des points de fonctionnement se fait en fonction de la variation de la phase de la fonction de transfert $G_L^{(2)}(s) = \dfrac{Y_G(s)}{\theta_v(s)}$ à ω_u. Ainsi, entre deux points de fonctionnement, la variation de phase du procédé est identique et $\simeq 15°$. La Figure 3.21 (a) représente la variation de l'argument de $G_L^{(2)}(s)$ à ω_u et le positionnement des points de fonctionnement. Il en résulte 8 points de fonctionnement aux vitesses $\{1\,;\,6{,}8\,;\,13{,}1\,;\,20{,}7\,;\,30{,}7\,;\,45\,;\,68{,}8\,;\,130\}$ km/h, avec une variation de phase $\mathrm{Var}_\phi = 15,8°$. Les fonctions de pondération w_i des régulateurs en fonction de la vitesse sont tracées Figure 3.21 (b).



(a) (b)

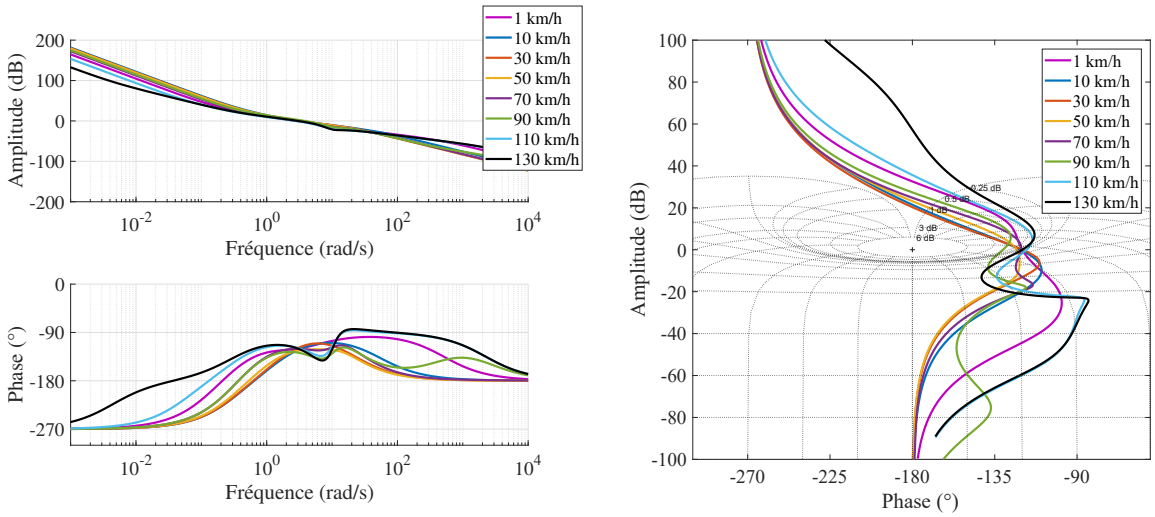
FIGURE 3.21 – Variation de la phase de $G_L^{(2)}(s)$ (a) et des pondérations de chaque PID (b) en fonction de la vitesse longitudinale

Les PID sont calculés de la même manière que précédemment et les valeurs numériques pour chaque PID du multirégulateur sont synthétisées dans la Table 3.3. À chaque instant, deux PID sont activés en même temps. Comme les cellules intégrales sont identiques pour chaque PID, le régulateur est d'ordre 3.

i	1	2	3	4	5	6	7	8
V_{x0} (km/h)	1	6,8	13,1	20,7	30,7	45	68,8	130
C_0 (rad/m)	336,26	35,96	13,16	5,54	2,28	0,84	0,22	0,001
ω_i (rad/s)	0,3	0,3	0,3	0,3	0,3	0,3	0,3	0,3
ω_1 (rad/s)	4,40	3,30	2,50	1,87	1,33	0,86	0,42	0,004
ω_2 (rad/s)	2,04	2,73	3,59	4,81	6,75	10,45	21,37	2230,7
κ_i	9,93	9,14	7,58	5,76	4,03	2,42	0,94	

TABLE 3.3 – Valeurs numériques des paramètres du multi-PID

Une fois le régulateur calculé, il est ensuite possible de faire l'analyse de la boucle ouverte dans le domaine fréquentiel.



(a) (b)

FIGURE 3.22 – Réponses fréquentielles de la boucle ouverte dans les diagrammes de Bode (a) et de Black-Nichols (b) pour le multi-PID

Comparé au simple PID calculé à 90 km/h, le multi-PID permet d'assurer la stabilité du système bouclé sur tout l'intervalle des vitesses avec une meilleure robustesse du degré de stabilité et de précision du suivi de consigne comme le montre le diagramme de Black-Nichols de la boucle ouverte Figure 3.22 (b). De plus, le fait d'utiliser plusieurs PID en parallèle garantit un meilleur respect du cahier des charges sur la rapidité de la boucle ouverte ω_u, qui varie pour les vitesses tracées, entre 2,7 et 3,8 rad/s.

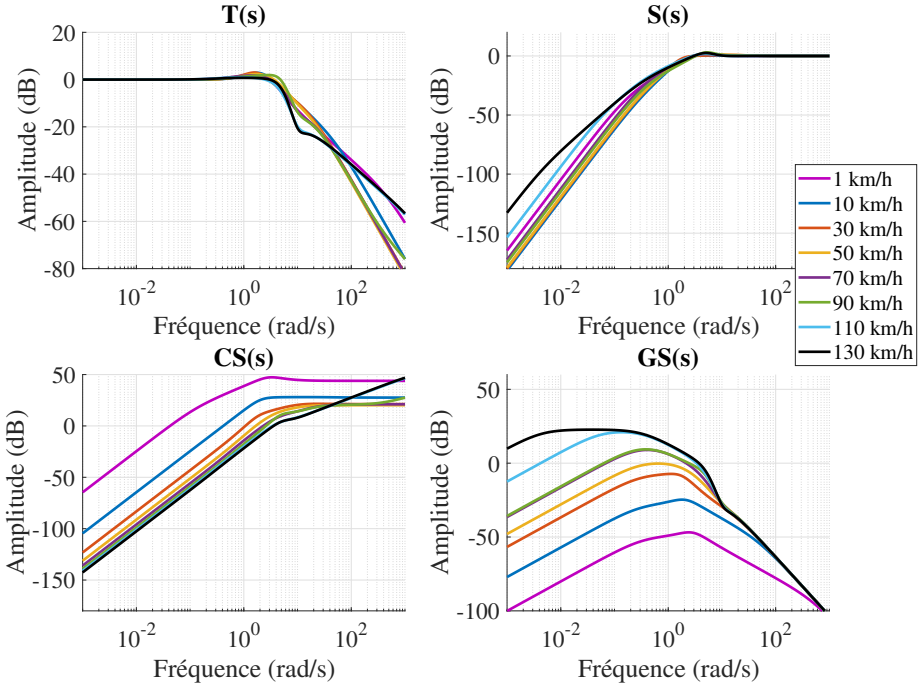
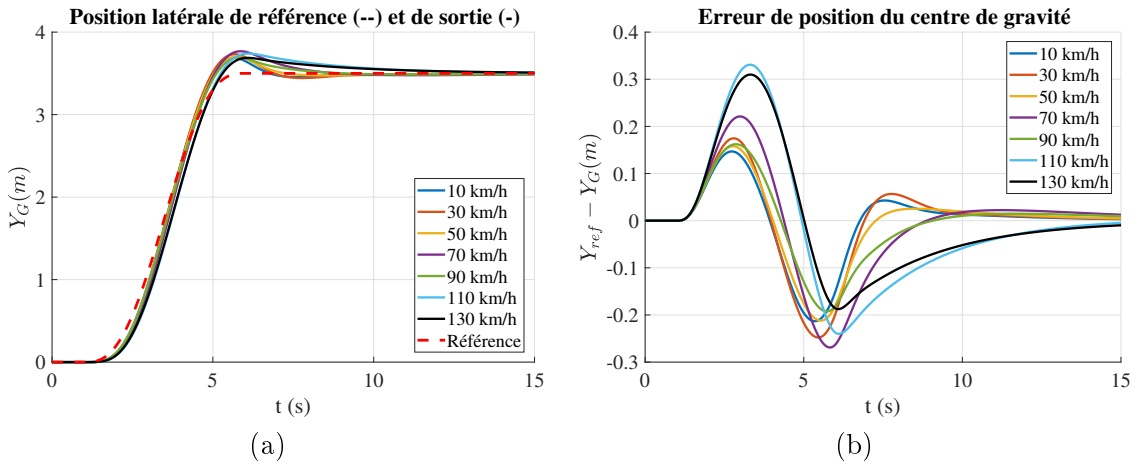


FIGURE 3.23 – Fonctions de sensibilité pour le multi-PID

Les fonctions de sensibilité Figure 3.23 montrent que, par rapport à la commande CRONE, le multi-PID permet une diminution du facteur de résonance avec une augmentation de la bande passante par l'analyse de $T(s)$. L'analyse de $S(s)$ montre une désensibilisation du système bouclé, celle de $CS(s)$ une augmentation de la sensibilité en entrée à basse vitesse et celle de $GS(s)$ une réjection plus importante des perturbations en entrée. Par rapport au PID simple, la régulation multi-PID est moins sensible aux perturbations en entrée pour des vitesses inférieures à 50 km/h mais plus sensible pour des vitesses supérieures à 70 km/h.

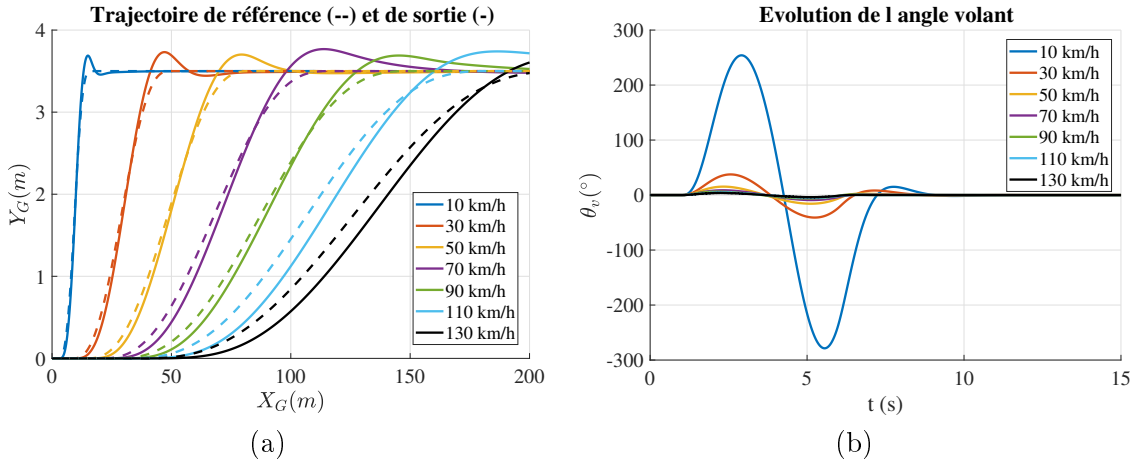
Les entrées des simulations temporelles sont identiques aux simulations précédentes.



(a) (b)

FIGURE 3.24 – Évolution de la position latérale du centre de gravité $Y_G(t)$ (a) et l'erreur de position par rapport à la référence $Y_{Gref}(t)$ (b)

Le multirégulateur améliore la robustesse du degré de stabilité. En effet, le dépassement est quasi identique pour toutes les vitesses, Figure 3.24.



(a) (b)

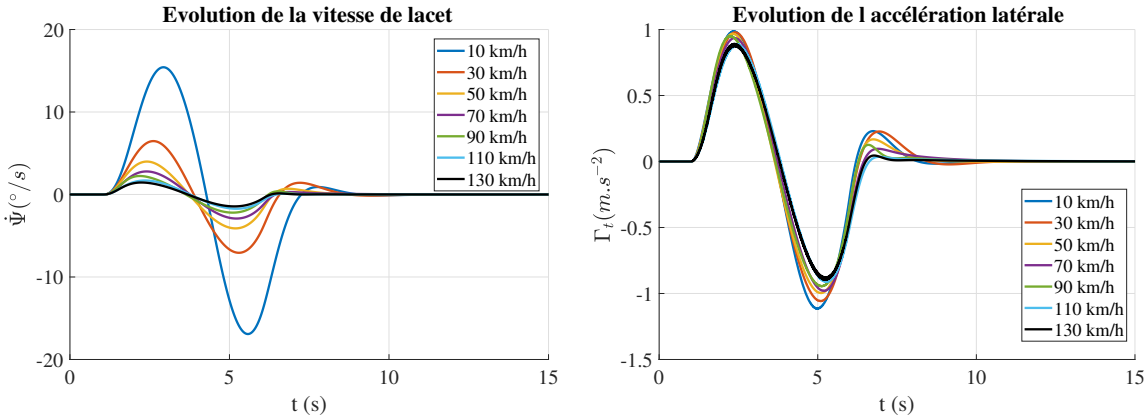
FIGURE 3.25 – Trajectoires de références et trajectoires réelles du véhicule pour les différentes vitesses (a) et évolution de l'angle volant $\theta_v(t)$ (b)

Les erreurs maximales en positions pour les différents régulateurs et les différentes vitesses sont indiquées dans la Table 3.4. La régulation par multi-PI présente une erreur de position maximale supérieure aux autres régulateurs à haute vitesse. Cependant, contrairement aux régulations CRONE et PID simple, le multi-PID permet un meilleur suivi de trajectoire à basse

vitesse dès le début du changement de voie, Figure 3.25 (a), ceci grâce à une fréquence au gain unité plus grande et respectant le cahier des charges.

V_{x0} (km/h)	10	30	50	70	90	110	130
Erreur max avec PID (m)	∞	1,15	0,59	0,37	0,26	0,20	0,17
Erreur max avec CRONE (m)	2,21	0,72	0,32	**0,19**	**0,13**	**0,10**	**0,08**
Erreur max avec multi-PID (m)	**0,21**	**0,24**	**0,21**	0,27	0,19	0,33	0,31

TABLE 3.4 – Erreurs maximales de sortie en position pour chaque régulateur pour les différentes vitesses V_{x0}



(a) (b)

FIGURE 3.26 – Évolution de la vitesse de lacet $\dot{\Psi}(t)$ (a) et de l'accélération latérale $\Gamma_t(t)$ pour les différentes vitesses (b)

Les variables de vitesse de lacet et d'accélération latérale présentent aussi moins d'oscillations avec le multi-PID qu'avec les autres régulateurs, Figure 3.26.

Le multi-PID calculé garantit la stabilité locale, à vitesse longitudinale constante, mais la stabilité globale n'a pas été prouvée. La stabilité globale pourrait être prouvée avec les équations de Lyapunov en formalisant le problème par un multimodèle Takagi-Sugeno. Cependant, cette méthode est dans notre cas trop restrictive et aucune fonction de Lyapunov garantissant la stabilité globale n'a pu être trouvée. On peut aussi noter le rôle important des sigmoïdes sur la stabilité. Il a été constaté a posteriori que l'utilisation de sigmoïdes à la place de droite permettait, en garantissant la continuité, d'améliorer la stabilité autour des points de fonctionnement.

Le multi-PID est globalement plus performant que le PID simple et la commande CRONE de 3ème génération, car il améliore le degré de stabilité du système bouclé. Cette méthode et le fait d'avoir une variable de décision mesurable permettent d'améliorer le compromis entre robustesse, aux variations de cette variable, et performance globale de la régulation. En pratique, la régulation du positionnement du véhicule se fait rarement par rapport au centre de gravité, car tout comme un conducteur humain regarde au loin pour diriger le véhicule, les capteurs perçoivent l'environnement devant le véhicule et non dessous. Ainsi, un point devant le véhicule, appelé point de visée, est utilisé pour réguler la position.

La section suivante analyse l'influence du point de visée sur la dynamique latérale et montre pourquoi il est plus intéressant de réguler la position du véhicule sur un point de visée plutôt qu'au centre de gravité.

3.7 Théorie et apport du point de visée

Outre les différentes stratégies de régulation de la dynamique latérale par rapport au type de régulateur, les stratégies peuvent se différencier en fonction de la consigne. Dans la littérature, il existe deux principales méthodes : une méthode par régulation au centre de gravité utilisée dans les paragraphes précédents [70], et une méthode utilisant un point de visée devant le véhicule pour déterminer la consigne latérale [24].

3.7.1 Définition du point de visée

Les équations décrivant la dynamique latérale sont définies section 2.1.1 et leurs linéarisations section 2.1.2. Le point de visée est le point sur la trajectoire de référence à une distance l_s devant le véhicule, Figure 3.27. La dynamique de la position latérale de ce point dans le repère absolu, appelé $Y_L(t)$, est décrite par :

$$\dot{Y_L}(t) = v_y(t) + V_{x0}\psi(t) + l_s\dot{\psi}(t), \tag{3.34}$$

à vitesse longitudinale constante V_{x0}.

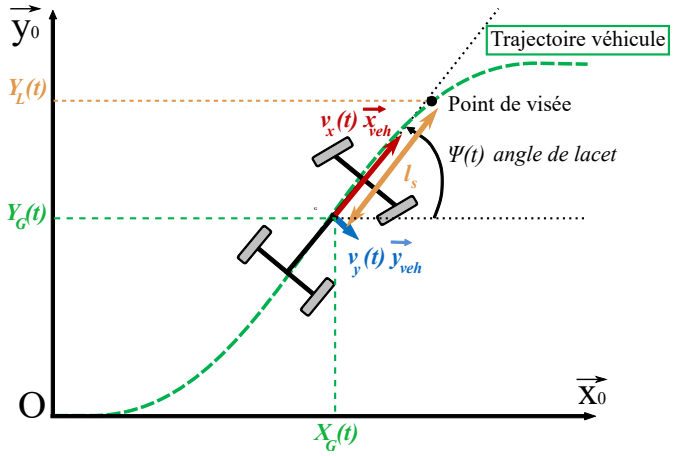


FIGURE 3.27 – Schéma du point de visée

La représentation d'état de la dynamique latérale du véhicule incluant le point de visée s'écrit :

$$\begin{cases} \dot{\underline{x}}(t) = A_{PV}\underline{x}(t) + B_{PV}u(t) \\ y(t) = C_{PV}\underline{x}(t), \end{cases} \tag{3.35}$$

avec le vecteur d'état $\underline{x}(t) = \Big(\psi(t),\ \dot{\psi}(t),\ V_y(t),\ Y_L(t)\Big)^T$, la commande $u(t) = \theta_v(t)$ et la sortie $y(t) = Y_L(t)$.

Les matrices de la représentation d'état (3.35) sont :

$$A_{PV} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\dfrac{2({L_f}^2 c_{yf} + {L_r}^2 c_{yr})}{I_z V_{x0}} & -\dfrac{2}{I_z V_{x0}}(L_f c_{yf} - L_r c_{yr}) & 0 \\ 0 & -\dfrac{2}{M_t V_{x0}}(L_f c_{yf} - L_r c_{yr}) - V_{x0} & -\dfrac{2(c_{yf} + c_{yr})}{M_t V_{x0}} & 0 \\ V_{x0} & l_s & 1 & 0 \end{bmatrix},$$

$$B_{PV} = \begin{bmatrix} 0 \\ \dfrac{2c_{yf}L_f}{\lambda I_z} \\ \dfrac{2c_{yf}}{\lambda M_t} \\ 0 \end{bmatrix} \text{ et } C_{PV} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.36}$$

3.7.2 Influence du point de visée sur la dynamique latérale

La fonction de transfert $G_{PV}(s)$ utilisée pour le calcul du régulateur est :

$$G_{PV}(s) = \frac{Y_L(s)}{\Theta_v(s)} = C_{PV}[sI - A_{PV}]^{-1} B_{PV}. \tag{3.37}$$

$G_{PV}(s)$ est une fonction de transfert d'ordre 4 avec un double intégrateur telle que :

$$G_{PV}(s) = \frac{K_{0PV}}{s^2} \frac{1 + 2\zeta_{1PV}(s/\omega_{1PV}) + (s/\omega_{1PV})^2}{1 + 2\zeta_{0PV}(s/\omega_{0PV}) + (s/\omega_{0PV})^2}, \tag{3.38}$$

avec K_{0PV} le gain, ζ_{0PV} et ζ_{1PV} les facteurs d'amortissement, ω_{0PV} et ω_{1PV} les fréquences transitionnelles. Chacun de ces paramètres peut s'exprimer en fonction des paramètres physiques du véhicule, tels que :

$$\begin{aligned} K_{0PV} &= \frac{2c_{yf}c_{yr}V_{x0}^2 L}{\lambda(2c_{yf}c_{yr}L^2 - M_t V_{x0}^2 PDE)}, \\ \zeta_{0PV} &= \frac{M_t(L_f^2 c_{yf} + L_r^2 c_{yr}) + I_z(c_{yf} + c_{yr})}{\sqrt{2I_z M_t(2c_{yf}c_{yr}L^2 - M_t V_{x0}^2 PDE)}}, \\ \omega_{0PV} &= \sqrt{2}\sqrt{\frac{2c_{yf}c_{yr}L^2 - M_t V_{x0}^2 PDE}{I_z M_t V_{x0}^2}}, \\ \zeta_{1PV} &= \frac{L_r + l_s}{V_{x0}}\sqrt{\frac{c_{yr}L}{2(I_z + L_f M_t l_s)}}, \\ \omega_{1PV} &= \sqrt{2\frac{c_{yr}L}{I_z + L_f M_t l_s}}. \end{aligned} \tag{3.39}$$

Les paramètres cruciaux pour la génération de la référence et le calcul du régulateur sont la distance l_s et la vitesse longitudinale V_{x0}. En effet, nous l'avons vu précédemment, la vitesse

longitudinale a un impact important sur la dynamique latérale et varie en fonction du temps. Tous les paramètres de $G_{PV}(s)$ décrits par (3.39) dépendent de la vitesse V_{x0}, sauf ω_{1PV}. Par contre, seuls les numérateurs de ω_{1PV} et ζ_{1PV} contiennent la distance l_s dans leurs expressions.

Il y a deux manières de définir l_s : la première est de fixer la valeur de la distance directement, et l'autre est de la faire varier en fonction de la vitesse longitudinale telle que :

$$l_s = V_{x0}\mathrm{d}T, \tag{3.40}$$

avec $\mathrm{d}T$ une constante qui définit le temps pour atteindre le point de visée à la vitesse longitudinale actuelle V_{x0}.

Pour analyser l'influence de l_s et V_{x0} sur la dynamique latérale du véhicule, les diagrammes de Bode de la réponse fréquentielle de $G_{PV}(s)$ sont tracés Figure 3.28.

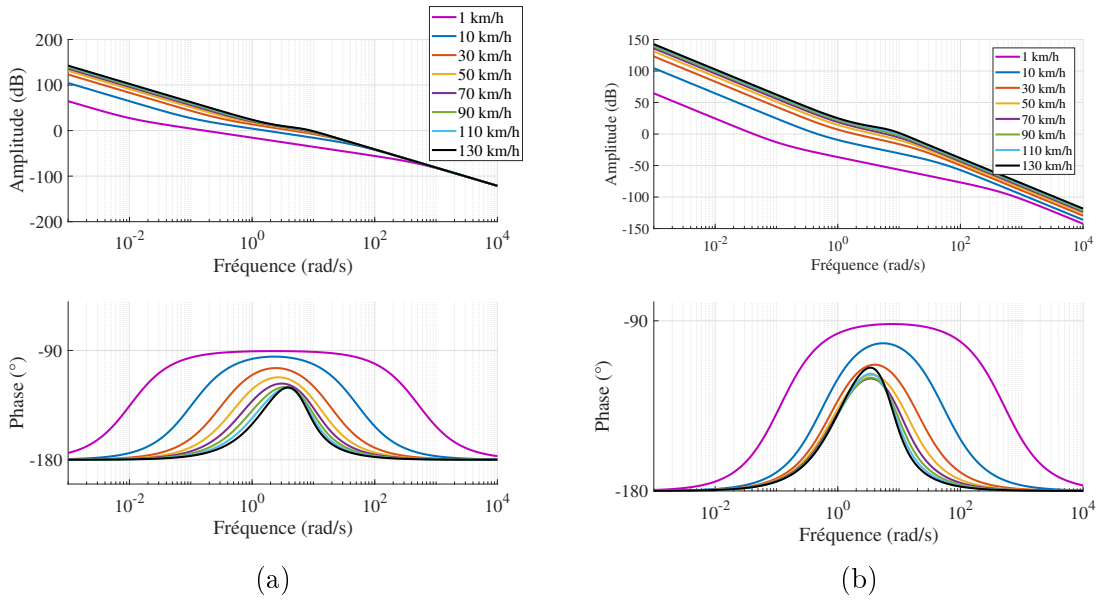
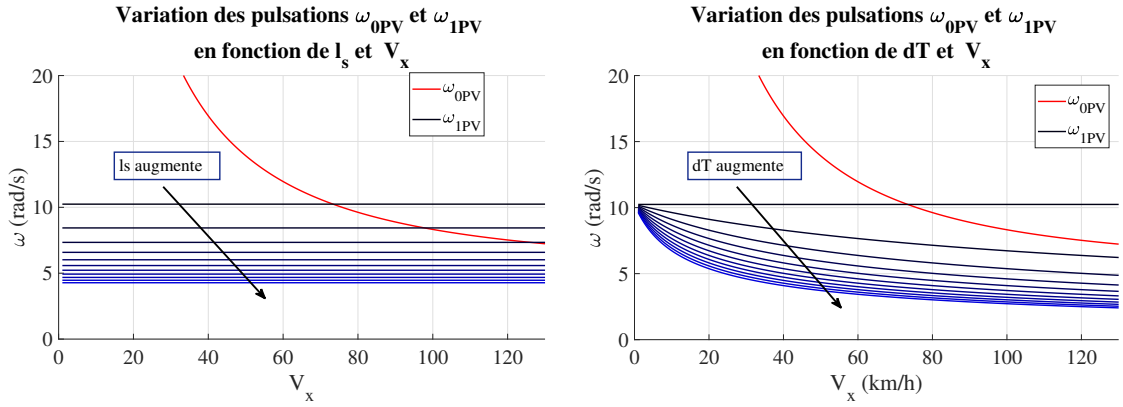


FIGURE 3.28 – Diagrammes de Bode de $G_{PV}(s)$ pour $l_s = 25$ m (a) et $l_s = \mathrm{d}TV_{x0}$ avec $\mathrm{d}T = 1$ s (b)

Ces figures montrent que le fait d'utiliser un point de fonctionnement comme référence pour la régulation de la dynamique latérale aide à la stabilisation du système. Comme nous l'avons vu sur la Figure 2.27, $G_{PV}(s)$ présente une variation de phase importante suivant la vitesse longitudinale du véhicule lorsque $l_s = 0$ m. L'ajout d'un point de visée réduit considérablement la variation de cette phase. Par exemple, à $\omega = 3$ rad/s, la variation de phase est de 109° pour $l_s = 0$ m, 26° pour $l_s = 25$ m (Figure 3.28 (a)), et 28° pour $l_s = \mathrm{d}TV_{x0}$ avec $\mathrm{d}T = 1$ s (Figure 3.28 (b)).

Le phénomène d'avance ou de retard de phase est lié aux fréquences transitionnelles ω_{0PV} et ω_{1PV}. Lorsque $\omega_{0PV} < \omega_{1PV}$ le procédé est à avance de phase, sinon, il est à retard de phase. Pour avoir une estimation plus précise de comment le point de visée influe sur les pulsations propres, Figure 3.29 trace ces pulsations en fonction de la vitesse longitudinale pour l_s entre 0 m et 10 m échantillonnée de 1 m, et pour $\mathrm{d}T$ entre 0 s et 1 s échantillonné à 0,1 s.



(a) (b)

FIGURE 3.29 – Pulsations ω_{0PV} et ω_{1PV} en fonction de la vitesse longitudinale et du point de visée

Ces figures montrent qu'il suffit que l_s soit supérieure à 3 m pour avoir $\omega_{0PV} < \omega_{1PV}$ et stabiliser le système.

Cette analyse fréquentielle est une autre approche que [29] pour mettre en évidence le fait qu'il est plus facile de concevoir un régulateur pour le guidage latéral par une approche avec un point de visée comme référence plutôt que le centre de gravité.

3.8 Régulation latérale avec point de visée par CRONE de 3ème génération

Les variations de gain et de phase du procédé étant plus faibles au point de visée qu'au centre de gravité, la conception d'un régulateur CRONE de 3ème génération devient plus facile.

Nous considérons ainsi le procédé avec un point de visée à une distance de $l_s = V_x \mathrm{d}T$, avec $\mathrm{d}T = 1$ s, devant le véhicule. Cette valeur est adaptée, car à des distances plus grandes les marquages au sol repérés par la caméra sont moins fiables.

En considérant le procédé avec point de visée et toujours en utilisant la toolbox CRONE pour respecter le même cahier des charges, la commande CRONE calculée est d'ordre 8 avec un intégrateur pur :

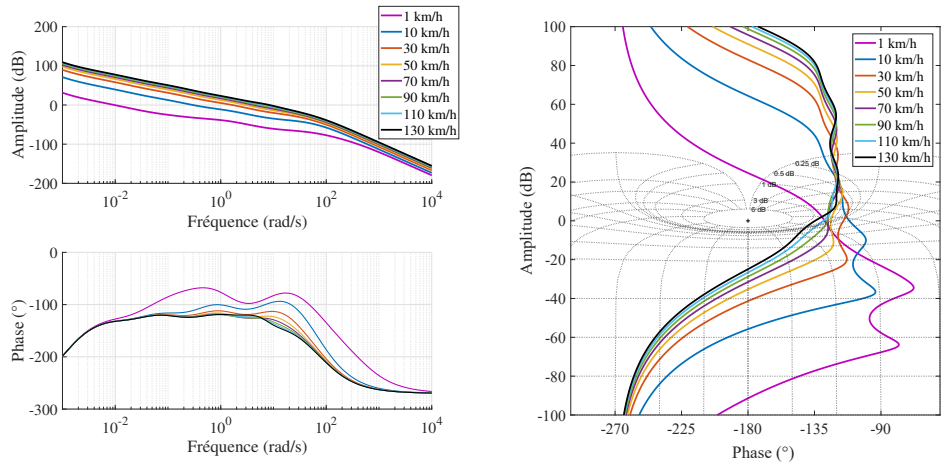
$$\mathrm{CRONE3G_{PV}}(s) = \frac{C_{0PV}}{s} \sum_{k=1}^{7} \left(\frac{1 + s/\omega_k}{1 + s/\omega'_k} \right), \tag{3.41}$$

dont les valeurs numériques figurent dans la Table 3.5.

C_{0PV} ($\mathrm{s}^{-1}.\mathrm{m}^{-1}$)	1,318 10^{-5}						
k	1	2	3	4	5	6	7
ω_k (rad/s)	1,22 10^{-3}	1,22 10^{-3}	4,52 10^{-3}	0,028	0,40	5,3	10,6
ω'_k (rad/s)	3,26 10^{-3}	0,012	0,157	1,44	1,44	36,2	96,8

TABLE 3.5 – Valeurs numériques des paramètres du régulateur CRONE de troisième génération pour le procédé avec point de visée

Les réponses fréquentielles en boucle ouverte sont tracées sur les Figures 3.30.



(a) (b)

FIGURE 3.30 – Réponses fréquentielles de la boucle ouverte dans les diagrammes de Bode (a) et de Black-Nichols (b) pour le CRONE de 3ème génération avec point de visée

Par rapport à la commande CRONE précédente au centre de gravité du véhicule, les variations de phase de la boucle ouverte sont beaucoup plus faibles autour de la fréquence au gain unité et les contours d'amplitude tangents aux réponses fréquentielles dans l'abaque de Nichols sont aussi plus faibles ce qui améliore la stabilité du système. La variation de la fréquence au gain unité en fonction de la vitesse reste cependant importante avec des valeurs entre $9,6\,10^{-3}$ rad/s à 1 km/h et $8,33$ rad/s à 130 km/h.

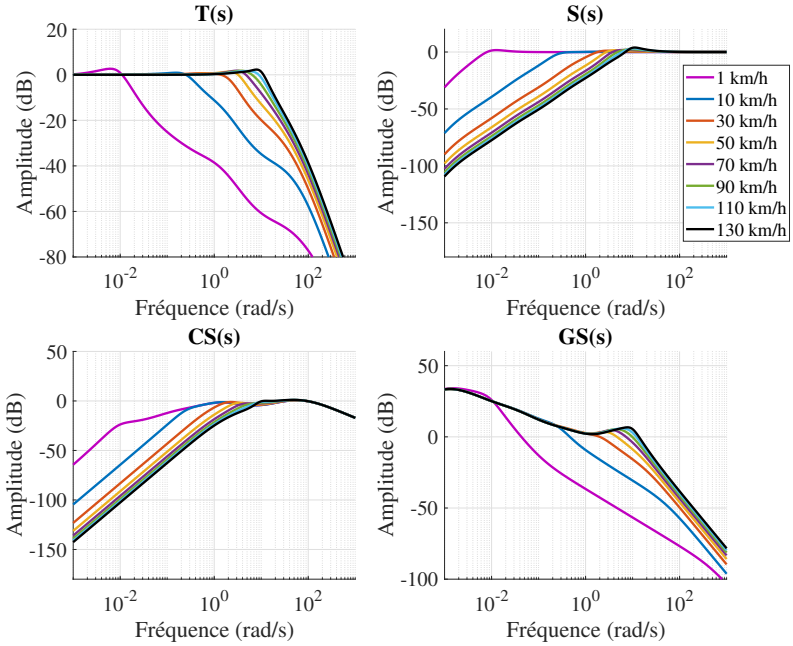
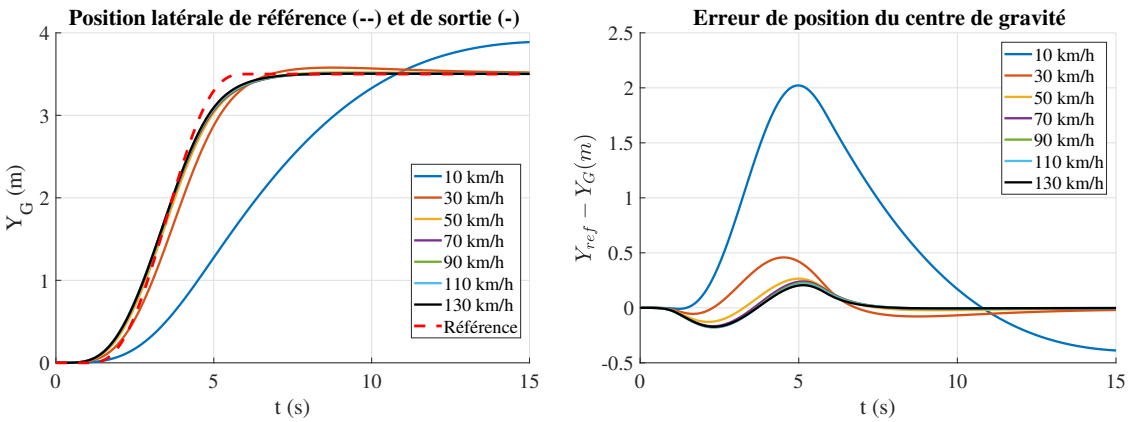


FIGURE 3.31 – Fonctions de sensibilité pour le CRONE de 3ème génération avec point de visée

Concernant les fonctions de sensibilité, Figure 3.31, celles-ci montrent une diminution du facteur de résonance et une diminution de la sensibilité en entrée à haute fréquence par rapport au système bouclé avec la commande CRONE calculée pour la régulation latérale au centre de gravité.

Le fait d'avoir une consigne avec point de visée améliore considérablement les performances de suivi de trajectoire dans le domaine temporel. En effet, par rapport aux simulations au centre de gravité, il n'y a quasiment plus de dépassement Figure 3.32 (a) pour les vitesses supérieures à 30 km/h. Avec la régulation CRONE au centre de gravité, le premier dépassement à 50 km/h est de 17 cm et avec la commande CRONE au point de visée, le dépassement est de 2 cm. Cependant, le dépassement à 10 km/h reste très important, 39 cm, car la fréquence au gain unité est trop faible à cette vitesse.

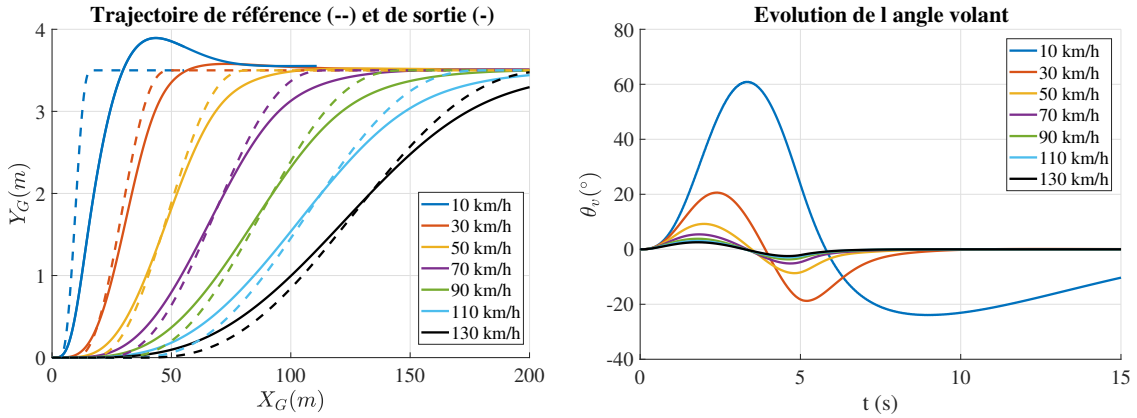


(a) (b)

FIGURE 3.32 – Évolution de la position latérale du centre de gravité $Y_G(t)$ (a) et l'erreur de position par rapport à la référence $Y_{Gref}(t)$ (b)

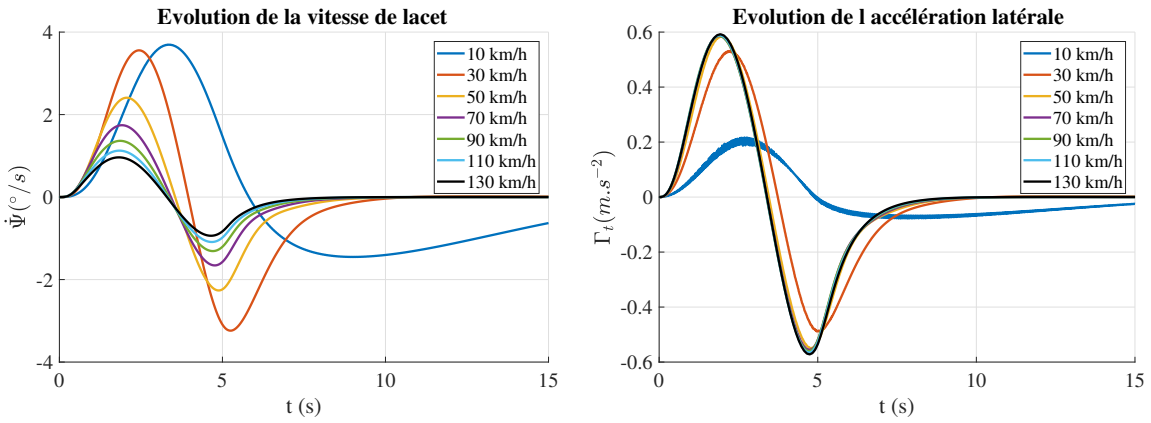
Avoir un point de consigne devant le véhicule fait que l'erreur de sortie est non nulle à un instant t plus faible qu'avec une consigne au centre de gravité. L'angle volant commence à tourner avant et le véhicule est en « avance » par rapport à la trajectoire de référence (Figure 3.33 (a)), expliquant la diminution du dépassement.

Autre avantage d'avoir un point de visée pour la régulation latérale, l'anticipation du mouvement latéral diminue l'accélération latérale, Figure 3.34 (b). Comparée aux simulations au centre de gravité, l'accélération latérale pour les vitesses supérieures à 30 km/h est d'environ 0,6 m.s^{-2} alors qu'elle était de 1 m.s^{-2}.



(a) (b)

FIGURE 3.33 – Trajectoires de références et trajectoires réelles du véhicule pour les différentes vitesses (a) et évolution de l'angle volant $\theta_v(t)$ (b)



(a) (b)

FIGURE 3.34 – Évolution de la vitesse de lacet $\dot{\Psi}(t)$ (a) et de l'accélération latérale $\Gamma_t(t)$ (b) pour les différentes vitesses

3.9 Régulation latérale avec point de visée par multi-PID

La commande CRONE de troisième génération pour le procédé avec point de visée permet d'améliorer les performances du changement de voie en termes de précision et de stabilité. Cependant, les variations de la fréquence au gain unité de la boucle ouverte restent trop importantes par rapport au cahier des charges, engendrant des erreurs de position jusqu'à 2 m à 10 km/h.

La régulation par multi-PID au centre de gravité permet d'avoir une fréquence au gain unité qui reste proche de 3 rad/s mais engendre des dépassements. Le fait d'utiliser cette méthode de régulation sur le procédé avec point de visée peut donc permettre de réduire ces dépassements mais aussi de réduire le nombre de PID utilisés, car la variation de la phase du procédé à ω_u est plus faible.

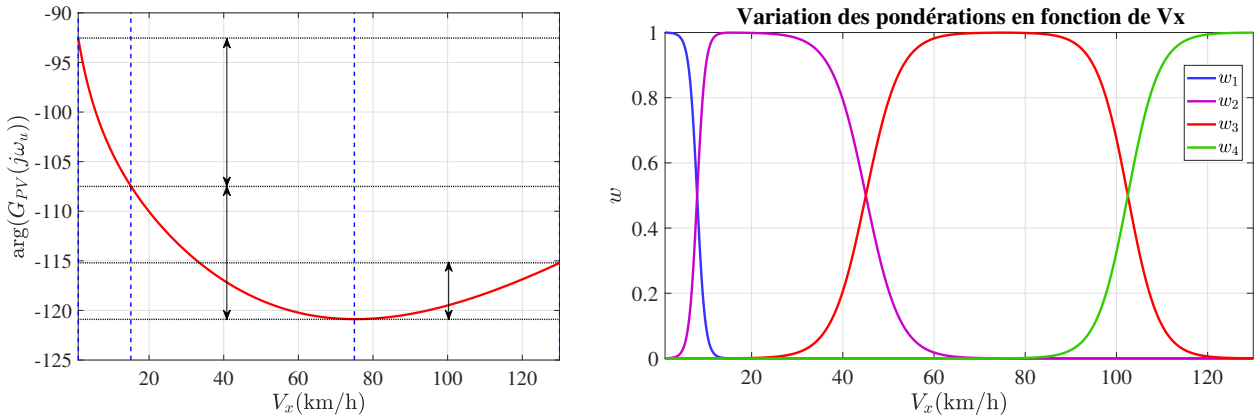
En effet, le fait de garantir une variation de phase d'au maximum 15° entre deux points de fonctionnement conduit à calculer des PID aux vitesses {1 ; 15,1 ; 75 ; 130} km/h Figure 3.35 (a), soit deux fois moins que pour le multi-PID en considérant le procédé au centre de gravité.

Les pondérations de chaque PID sont des sigmoïdes, tracées sur la Figure 3.35 (b).

Les valeurs numériques du régulateur sont répertoriées dans la Table 3.6.

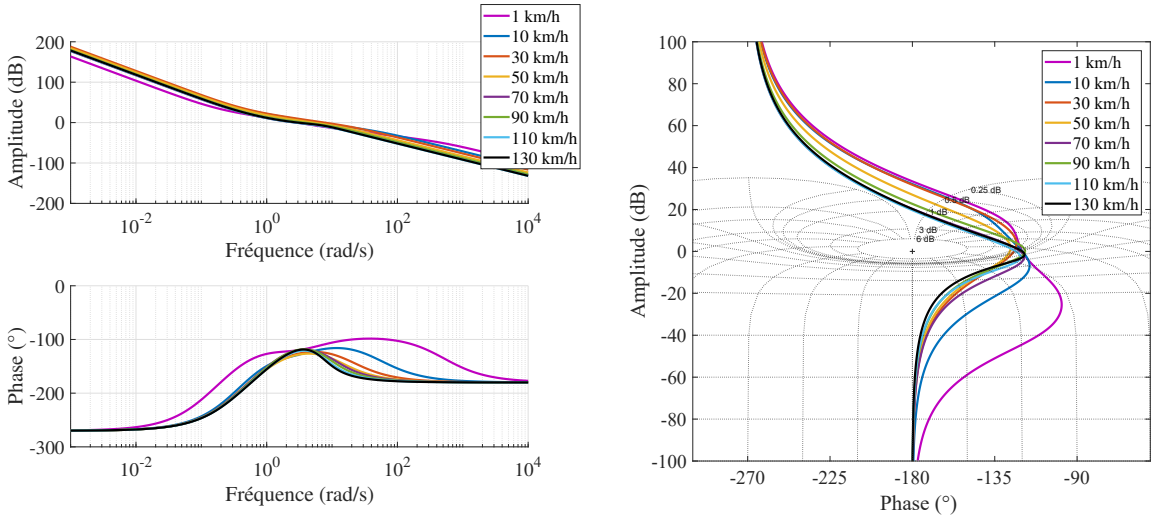
i	1	2	3	4
V_{x0} (km/h)	1	15,1	75	130
C_0 (rad/m)	299,12	5,80	0,36	0,21
ω_i (rad/s)	0,3	0,3	0,3	0,3
ω_1 (rad/s)	4,42	3,38	2,67	2,95
ω_2 (rad/s)	2,03	2,66	3,37	3,05
κ_i	3,05	4,09	0,96	

TABLE 3.6 – Valeurs numériques des paramètres du multi-PID avec point de visée



(a) (b)

FIGURE 3.35 – Variation de la phase de $G_{PV}(s)$ (a) et des pondérations de chaque PID (b) en fonction de la vitesse longitudinale



(a) (b)

FIGURE 3.36 – Réponses fréquentielles de la boucle ouverte dans les diagrammes de Bode (a) et de Black-Nichols (b) pour le multi-PID avec point de visée

Avec la régulation par multi-PID, la fréquence au gain unité de la boucle ouverte reste proche du cahier des charges avec des valeurs entre 2,8 rad/s et 6,7 rad/s, Figure 3.36 (a). Par contre, les contours d'amplitude tangents aux réponses fréquentielles sont compris entre 1,5 dB et 3 dB, plus importants donc que ceux du CRONE.

Au niveau de fonctions de sensibilité, les régulations par multi-PID au centre de gravité et avec point de visée sont équivalentes. Par contre, en comparant les fonctions de sensibilité des régulations CRONE et multi-PID avec point de visée, il peut être noté que la sensibilité en entrée est plus importante avec le multi-PID et que la sensibilité aux perturbations en entrée est plus faible avec ce même régulateur impliquant un rejet des perturbations plus important.

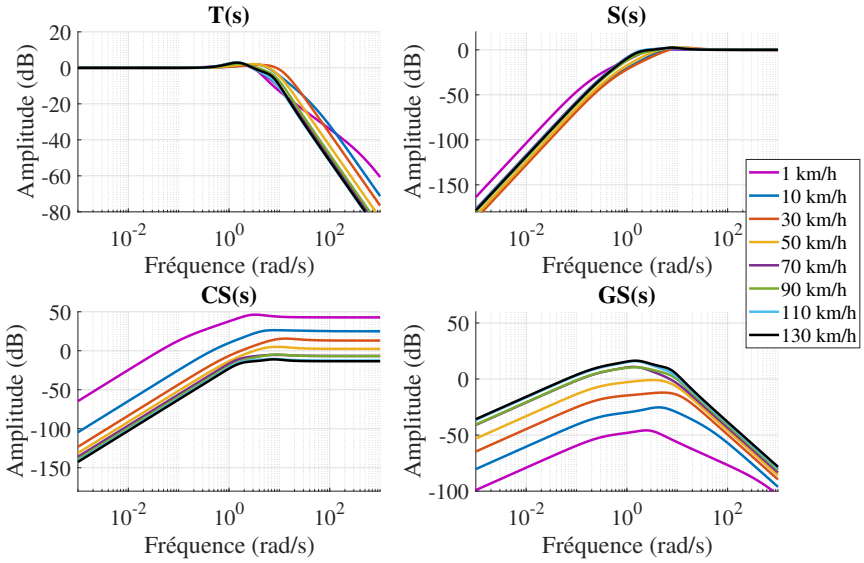
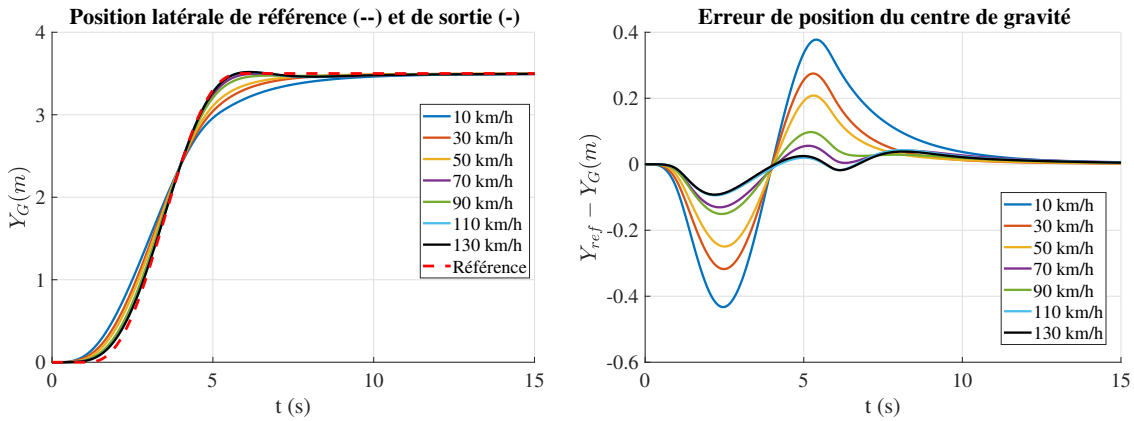


FIGURE 3.37 – Fonctions de sensibilité pour le multi-PID avec point de visée

Les simulations temporelles, toujours avec les mêmes références en entrée, montrent des résultats très concluants avec le régulateur multi-PID avec point de visée. La conduite est beaucoup plus fluide, car quasiment sans dépassement : 1 cm à 110 km/h et 130 km/h, Figure 3.38 (a).



(a) (b)

FIGURE 3.38 – Évolution de la position latérale du centre de gravité $Y_G(t)$ (a) et l'erreur de position par rapport à la référence $Y_{Gref}(t)$ (b)

La Table 3.7 récapitule les valeurs des premiers dépassements pour toutes les simulations temporelles avec les différents régulateurs. Le fait de changer la référence permet de diminuer l'amplitude du premier dépassement d'un facteur compris entre 1,5, à basse vitesse, et 10, à haute vitesse, pour les commandes de type CRONE. Pour les régulateurs de type multi-PID, les dépassements disparaissent même complètement.

V_{x0} (km/h)	10	30	50	70	90	110	130
Multi-PID (m)	0,18	0,23	0,2	0,26	0,19	0,24	0,19
Multi-PID (%)	5,14	6,57	5,71	7,43	5,43	6,86	5,43
CRONE (m)	0,56	0,31	0,17	0.10	0,07	0,05	0,04
CRONE (%)	16	8,86	4,86	2,86	2,00	1,43	1,14
CRONE avec PV (m)	0,39	0,08	0,02	0,01	0,007	**0,006**	**0,005**
CRONE avec PV (%)	11,14	2,29	0,57	0,29	0,20	**0,17**	**0,14**
Multi-PID avec PV (m)	**0**	**0**	**0**	**0**	**0**	0,01	0,01
Multi-PID avec PV (%)	**0**	**0**	**0**	**0**	**0**	0,29	0,29

TABLE 3.7 – Valeurs des premiers dépassements pour les simulations temporelles aux différentes vitesses de chaque régulateur

Les Tables 3.8 et 3.9 recensent quant à elles les erreurs de position maximales au centre de gravité pour les simulations avec référence au point de visée et les erreurs moyennes pour toutes les simulations entre 0 s et 15 s.

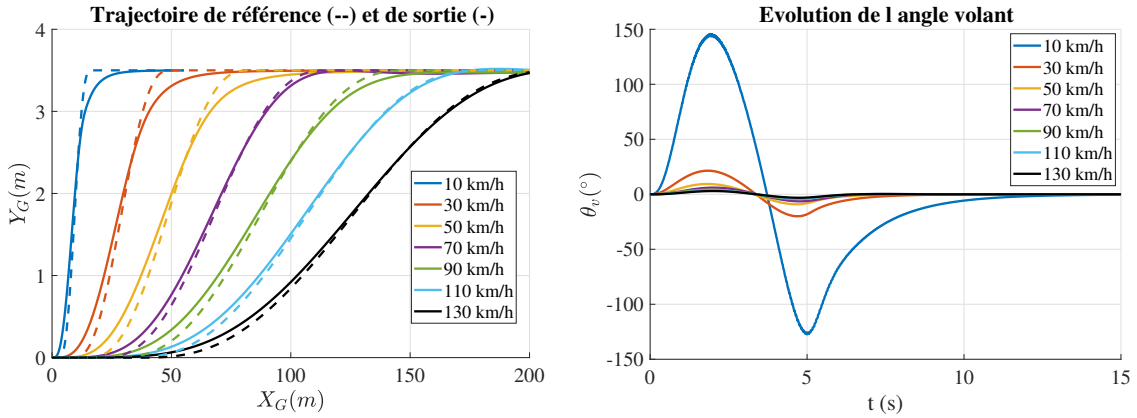
V_{x0} (km/h)	10	30	50	70	90	110	130
Erreur max CRONE avec PV (m)	2,02	0,46	0,26	0,24	0,23	0,22	0,2
Erreur max multi-PID avec PV (m)	**0,43**	**0,31**	**0,24**	**0,13**	**0,15**	**0,09**	**0,09**

TABLE 3.8 – Erreurs maximales en position pour les simulations aux différentes vitesses avec un point de visée comme référence

Avec la stratégie de multirégulateur proposée ici, les erreurs maximales en position sont plus faibles qu'avec la stratégie CRONE, Table 3.8. Cependant, en considérant l'erreur moyenne en position de la simulation entre 0 s et 15 s, ce sont les méthodes par référence au centre de gravité qui permettent en moyenne un meilleur suivi de trajectoire, le multi-PID pour la basse vitesse et la commande CRONE pour la haute vitesse, Table 3.9. Ceci s'explique par l'« anticipation » procurée par le point de visée. Au global, en prenant en compte toutes les simulations, le régulateur de type multi-PID avec le point de visée comme consigne latérale est la meilleure solution pour le suivi de trajectoire avec un meilleur compromis entre précision, premier dépassement et complexité calculatoire.

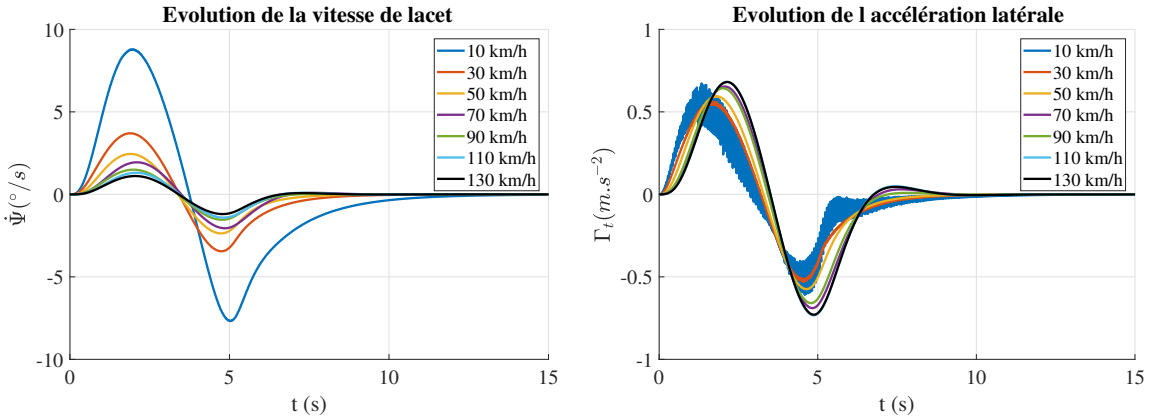
V_{x0} (km/h)	10	30	50	70	90	110	130	Toutes
Er. moy multi-PID (m)	**0,048**	**0,058**	**0,051**	0,072	0,052	0,102	0,091	0,068
Er. moy CRONE (m)	0,759	0,190	0,077	0,044	**0,031**	**0,024**	**0,020**	0,163
Erreur moy CRONE avec PV (m)	0,664	0,107	0,059	0,055	0,054	0,051	0,049	0,148
Erreur moy multi-PID avec PV (m)	0,131	0,086	0,066	**0,034**	0,039	0,025	0,024	**0,058**

TABLE 3.9 – Erreurs moyennes en position latérale entre 0 s et 15 s pour les simulations aux différentes vitesses



(a) (b)

FIGURE 3.39 – Trajectoires de références et trajectoires réelles du véhicule pour les différentes vitesses (a) et évolution de l'angle volant $\theta_v(t)$ (b)



(a) (b)

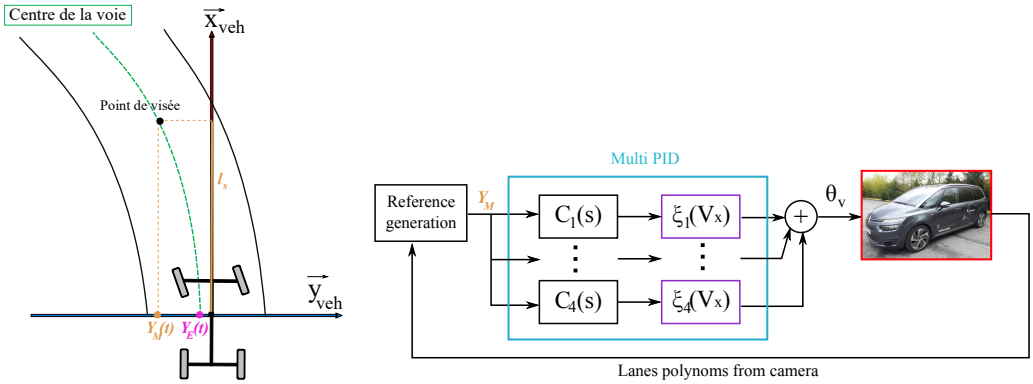
FIGURE 3.40 – Évolution de la vitesse de lacet $\dot{\Psi}(t)$ (a) et de l'accélération latérale $\Gamma_t(t)$ pour les différentes vitesses (b)

Pour résumer, la multirégulation semble être la solution la plus adaptée au problème de régulation de la dynamique latérale du véhicule pour le suivi de trajectoire. En effet, celle-ci, en s'appuyant sur des variables connues telles que la vitesse longitudinale permet d'améliorer le degré de stabilité du système en fonction de la variation de la vitesse. En ajoutant une stratégie de régulation par point de visée, le contrôle de la trajectoire du véhicule devient plus simple du fait de la diminution de la sensibilité du procédé à la vitesse longitudinale. Enfin, la combinaison de la régulation multi-PID avec point de visée permet une conduite plus souple en supprimant les dépassements lors des changements de voie, le tout avec une erreur moyenne en position inférieure à 6 cm en simulation.

3.10 Validation et résultats sur prototype

3.10.1 Intégration du régulateur dans le prototype ADN1

Le multi-PID avec point de visée développé dans le paragraphe précédent a été implanté dans le prototype ADN1, section 1.5.2. Le prototype est équipé d'une fonction ACC qui gère la vitesse longitudinale du véhicule en fonction du trafic et des limitations de vitesse. Pour ces essais sur prototype, le régulateur latéral a pour objectif de maintenir le véhicule au centre de la voie.



(a) (b)

FIGURE 3.41 – Point de visée dans le repère véhicule (a) et schéma bloc de commande dans le prototype (b)

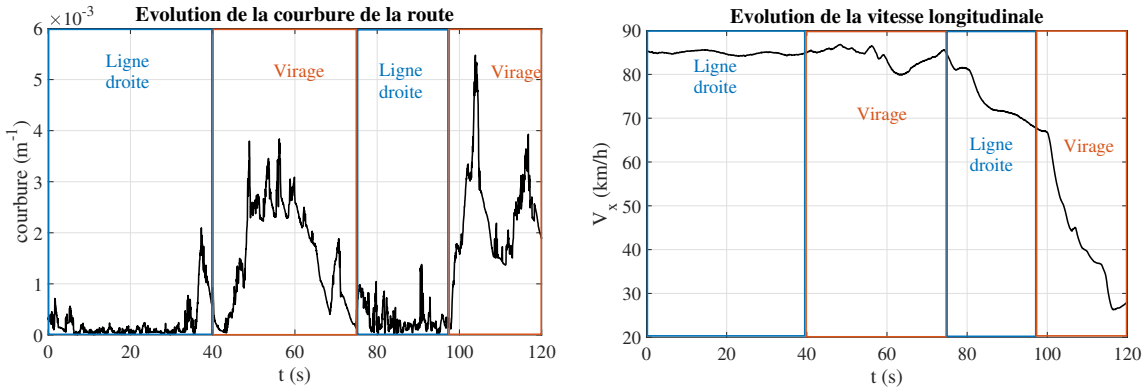
La caméra frontale d'ADN1 fournit une interpolation linéaire de la géométrie des deux lignes à gauche et des deux lignes à droite les plus proches du véhicule. En utilisant ces interpolations, qui sont des polynômes d'ordre 3, il est facile de positionner le point de visée à la distance l_s en face du véhicule, Figure 3.41 (a). Ces polynômes servent aussi à calculer la position du véhicule par rapport au centre de la voie $Y_E(t)$.

À bord des prototypes, les capteurs extéroceptifs mesurent les positions des objets de l'environnement dans le repère de l'ego véhicule. Cependant, la position $Y_L(t)$ définit le point de visée dans le repère absolu. Ainsi, la distance $Y_M(t)$ représentant la distance latérale entre le centre de gravité du véhicule et le point de visée, Figure 3.41 (a), est directement utilisée en entrée du régulateur comme erreur de position. Le schéma bloc de contrôle est représenté Figure 3.41 (b).

3.10.2 Résultats sur route ouverte : maintien dans la voie

Les résultats présentés ici ont été obtenus sur une route ouverte à voies à chaussée séparée sur la N118 dans des conditions réelles de trafic.

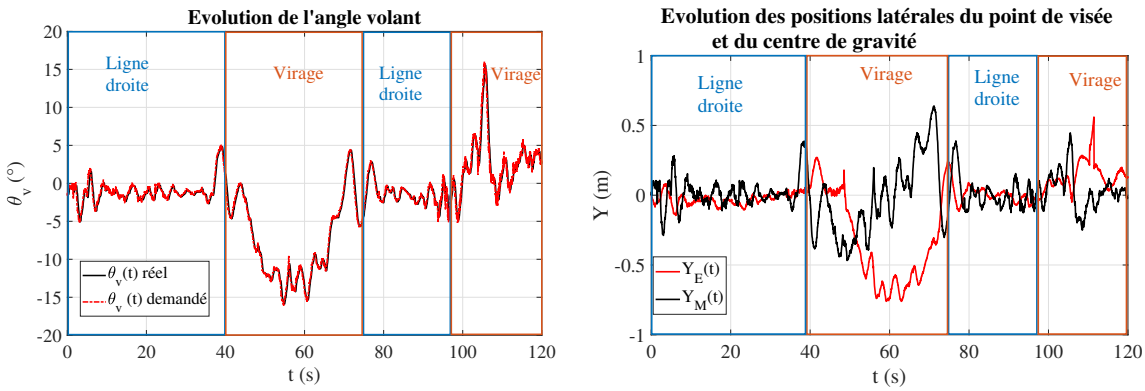
Le profil de la route peut être observé sur la Figure 3.42 (a) avec la courbure de la route, mesurée par le traitement d'image. Il y a quatre phases de roulage sur cet enregistrement : deux phases où le véhicule roule en ligne droite, en bleu, et deux phases où le véhicule est dans un virage, en rouge. La vitesse du véhicule varie entre 26 km/h et 86 km/h avec une vitesse quasi constante sur les deux premières phases et une décélération sur les deux dernières, Figure 3.42 (b). Cet enregistrement est intéressant, car il montre quatre combinaisons de variation de vitesse et de courbure de la route qui ont toutes un impact important sur le calcul de l'angle volant par le régulateur à appliquer à la DAE et donc sur la dynamique du véhicule.



(a) (b)

FIGURE 3.42 – Évolutions de la courbure de la route (a) et de la vitesse longitudinale $V_x(t)$ (b)

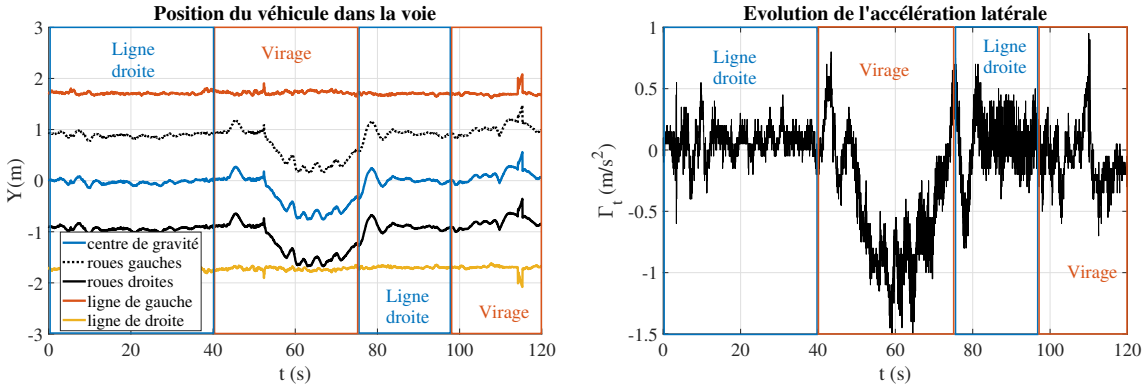
$Y_E(t)$, la distance latérale entre le centre de gravité du véhicule et le centre de la voie, est tracée sur la Figure 3.43 (b) en rouge et $Y_M(t)$, la distance latérale du centre de gravité au point de visée, est tracée sur cette même figure en noir. Pendant les phases de ligne droite, les deux courbes restent proches de 0 avec le véhicule qui dérive de moins de 10 cm par rapport au centre de la voie. En moyenne, sur l'ensemble de l'enregistrement, le véhicule reste à -7,6 cm du centre de la voie avec un maximum de 76 cm et un écart type de 25 cm. Cet écart s'explique par la dynamique de correction assez lente pour des raisons de confort. À haute vitesse, cela ce traduit par un déport latéral plus grand.



(a) (b)

FIGURE 3.43 – Évolutions de l'angle volant $\theta_v(t)$ (a) des positions latérales du point de visée $Y_M(t)$ et du centre de gravité par rapport au centre de la voie $Y_E(t)$

Dans les virages, $Y_M(t)$ augmente. Lors de la première phase, à haute vitesse, le véhicule s'écarte de 70 cm à droite dans la ligne de circulation (les valeurs négatives de $Y_E(t)$ correspondent à la droite du centre de la voie). Cela veut dire que le véhicule est proche de la ligne de droite mais toujours dans la voie comme le montre la Figure 3.44 (a). L'angle volant appliqué, Figure 3.43 (a), est lisse et permet une conduite confortable. Dans la deuxième phase de virage, le véhicule parvient à rester dans sa voie alors qu'un freinage est nécessaire à cause du trafic. Aux vitesses plus faibles dans le virage, l'erreur de positionnement au centre de la voie est moins importante que lors du premier virage à 80 km/h.



(a) (b)

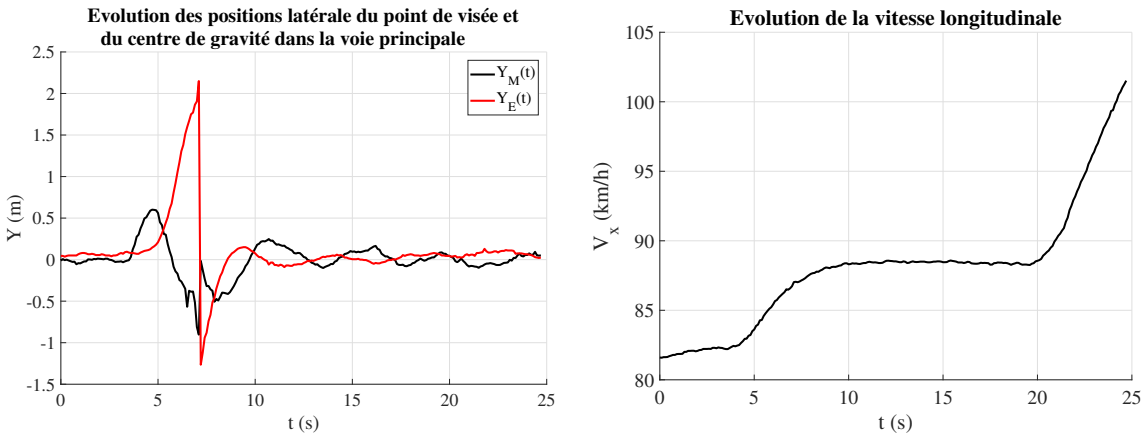
FIGURE 3.44 – Évolutions de la position du véhicule dans la voie (a) et de l'accélération latérale (b)

Le fait que le véhicule se rapproche de la ligne n'est pas problématique d'un point de vue performance. Il est plus important d'assurer une conduite confortable avec des niveaux d'accélération inférieurs à $2\,\mathrm{m.s}^{-2}$ comme c'est le cas ici, voir Figure 3.44 (b).

3.10.3 Résultats sur route ouverte : changement de voie

En plus de valider le régulateur sur des essais en maintien dans la voie, la méthode de multirégulation développée a été validée sur prototype en changement de voie. Comme pour les résultats présentés précédemment, les résultats présentés ici ont été obtenus dans des conditions de trafic réelles, sur l'autoroute A86.

Lors de ce changement de voie, l'ego véhicule a une vitesse longitudinale d'environ 85 km/h, voir Figure 3.45 (b). Le changement de voie est initié à $t = 3,5$ s, lorsque la position latérale du point de visée $Y_M(t)$ commence à augmenter, voir Figure 3.45 (a). À $t = 7$ s, le véhicule franchit la ligne de gauche, d'où le pic de $Y_E(t)$ à cet instant. Pour mieux visualiser le déplacement de l'ego véhicule et son changement de voie, la Figure 3.46 trace l'évolution de la position latérale du véhicule par rapport aux lignes.



(a) (b)

FIGURE 3.45 – Évolutions des positions latérales du centre de gravité et du point de visée (a) et de la vitesse longitudinale (b) pour l'exemple du changement de voie

Comme dans les simulations précédentes, le dépassement dure environ 5 s et le véhicule s'est déplacé latéralement de 3,5 m. Il est alors possible de comparer les variables d'angle volant et d'accélération latérale mesurées pour évaluer la correspondance de la simulation à la réalité.

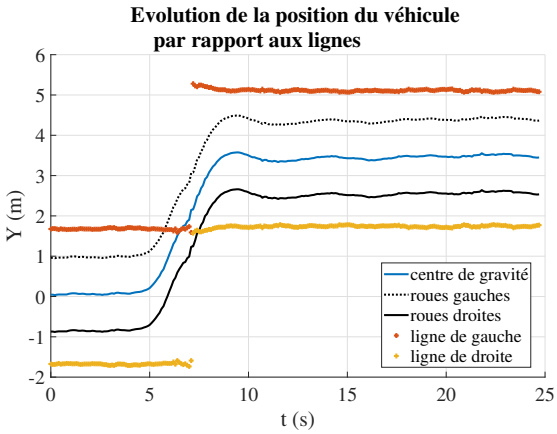
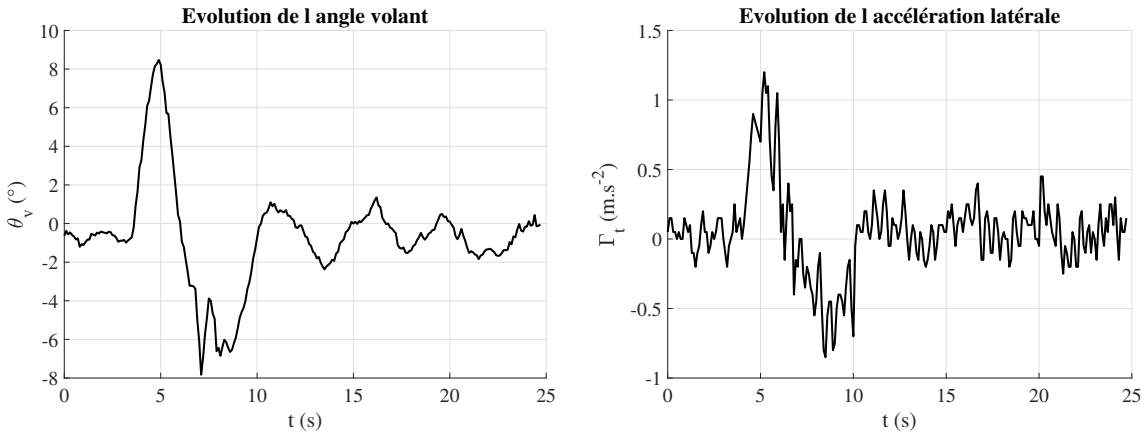


FIGURE 3.46 – Position latérale du véhicule dans les voies

Dans les simulations précédentes, avec point de visée et multi-PID, pour un déplacement latéral de 3,5 m en 5 secondes à 80 km/h, l'angle volant a une amplitude de 5°. Celle de l'accélération latérale est de 0,67 m.s^{-2}. En réalité, sur le prototype, l'angle volant varie entre +8° et −6°, voir Figure 3.47 (a). Ces valeurs sont plus élevées de 60 % et 20 % par rapport à la simulation. Concernant l'accélération latérale, sur cet enregistrement, sa valeur est comprise entre 1 m.s^{-2} et -0,6 m.s^{-2} (Figure 3.47 (b)). Les erreurs relatives aux amplitudes maximum sont donc de 49% et 10%.



(a) (b)

FIGURE 3.47 – Évolutions de l'angle volant (a) et de l'accélération latérale (b)

Le multirégulateur intégré dans le prototype permet de maintenir le véhicule dans sa voie et d'effectuer des changements de voie de manière autonome, et ce à toutes les vitesses. Les essais sur prototypes ont montré que la méthode était stable et respecte le cahier des charges PSA en termes de rapidité de la commande, et donc de confort.

3.10.4 Bilan : avantages, inconvénients et perspectives

Nous avons développé dans ce chapitre une méthode de multirégulation pour le guidage latéral du véhicule en mode autonome. Cette méthode a été comparée à des régulateurs robustes de type CRONE. Il a d'abord été mis en avant que la consigne du régulateur jouait un rôle important dans le développement du régulateur. En effet, la dynamique latérale du véhicule est très sensible à la vitesse longitudinale, mais cette sensibilité est maximale au centre de gravité du véhicule et diminue lorsque l'on considère un point de visée devant le véhicule. En simulation, le multi-PID respecte mieux le cahier des charges en rapidité et précision que le régulateur CRONE de 3ème génération.

Le multi-PID développé a été implanté dans le prototype ADN1 du Groupe PSA. Les résultats sont concluants puisque le régulateur permet de réaliser les scénarios de maintien dans la voie et de changement de voie sur toute la plage de vitesse. En termes de performance, l'angle volant appliqué à la direction permet une conduite confortable, avec des accélérations latérales faibles. Cette dynamique assez lente, privilégiée sur les autoroutes et voies à chaussées séparées est valide à haute vitesse, car les rayons de courbure sont grands. Lors de virage, le véhicule peut donc se rapprocher de la ligne.

En l'état, le problème de la méthode de multirégulation développée ici est stable localement mais la stabilité globale n'a pas été prouvée. Nous avons essayé de formuler le problème en boucle fermée dans le cadre du formalisme multimodèle Takagi-Sugeno, puis de prouver la stabilité à partir de ce modèle. Cependant, cette méthode reste trop restrictive pour prouver la stabilité de notre système et d'autres outils sont étudiés afin de prouver la stabilité du système bouclé.

La méthode proposée pourrait être enrichie en n'utilisant pas uniquement des PID mais d'autres types de régulateurs comme des régulateurs CRONE et H_∞. Cela permettrait d'introduire un degré de robustesse aux variations paramétriques dans le régulateur. Une autre variante consisterait à utiliser différents types de régulateurs dans le multirégulateur et ainsi avoir une somme de PID, PI ou autre. En effet, en revenant au cas du guidage latéral, un PI est suffisant aux basses vitesses, car il n'y a pas besoin de rajouter de phase. Le PID est utile pour les hautes vitesses.

Enfin, la méthode a été validée en environnement autoroutier mais doit aussi être testée en environnement urbain, notamment en intersection et en rond-point qui sont des scénarios avec de fortes dynamiques latérales.

Chapitre 4

Contrainte de l'environnement et prédiction de trajectoire des véhicules

Sommaire

4.1 L'influence de l'environnement sur le calcul de la trajectoire : la conscience de situation . . . 132
4.1.1 Le concept de la conscience de situation . . . 132
4.1.2 État de l'art : détection d'intention . . . 133
4.1.3 État de l'art : prédiction de trajectoire . . . 134
4.1.4 Introduction à l'apprentissage automatique . . . 136
4.1.5 Objectifs et méthodes . . . 139
4.2 Les réseaux de neurones : fonctionnement et apprentissage . . . 140
4.2.1 Fonctionnement d'un neurone . . . 140
4.2.2 Propagation de l'information dans le réseau . . . 141
4.2.3 Entraînement du réseau : la rétropropagation . . . 143
4.2.4 Méthodologie de construction du réseau . . . 146
4.3 Détection d'intention : changement de voie sur autoroute . . . 148
4.3.1 Description du problème . . . 148
4.3.2 Détection d'intention : méthode à base de règles . . . 149
4.3.3 Méthode par réseau de neurones . . . 154
4.3.4 Résultats . . . 158
4.4 Prédiction de trajectoire : le cas du changement de voie . . . 160
4.4.1 Description du problème . . . 160
4.4.2 Le réseau Long-Short Term Memory . . . 160
4.4.3 Structures de LSTM utilisées . . . 162
4.4.4 Résultats . . . 162
4.4.5 Perspectives pour la prédiction de trajectoire en environnement urbain 164

Celui qui ne prévoit pas les choses lointaines s'expose à des malheurs prochains.

Confucius

Afin de calculer la trajectoire à suivre par le véhicule et d'appliquer les consignes correspondantes en angle volant et vitesse longitudinale aux actionneurs, il est crucial de prendre en compte les objets dynamiques qui circulent autour. En effet, leur comportement impacte directement celui de l'ego véhicule qui doit s'adapter en permanence à son environnement, c'est la conscience de situation.

Ce chapitre se focalise donc sur la prédiction d'intention, ou de manœuvre, des véhicules ainsi que la prédiction de leur trajectoire. Le principe de conscience de situation est présenté dans un premier temps avec un état de l'art de la prédiction d'intention et de trajectoire pour les autres véhicules de l'environnement. Beaucoup des méthodes de l'état de l'art utilisent l'apprentissage automatique pour identifier les comportements des conducteurs à partir d'une base de données. Ces méthodes, utilisées par la suite, sont donc présentées plus en détail. Ensuite, deux méthodes sont utilisées et comparées pour la détection de changement de voie : une méthode probabiliste à base de règles et une méthode par apprentissage automatique avec des réseaux de neurones. Le scénario du changement de voie n'est pas exclusif au cas urbain, pour des raisons de base de données beaucoup plus importante sur voie à chaussées séparées, l'étude s'est portée sur un scénario applicable sur autoroute et en ville. Enfin, à partir d'une base de données identique, une méthode de prédiction de trajectoire à base de réseau de neurones récurrents est analysée afin de déterminer quels sont les paramètres pertinents qui influent sur la prédiction de trajectoire des véhicules de l'environnement.

4.1 L'influence de l'environnement sur le calcul de la trajectoire : la conscience de situation

4.1.1 Le concept de la conscience de situation

Avant de décider ce qu'il va faire et agir, un conducteur en conduite manuelle perçoit son environnement et l'analyse : c'est la conscience de situation. Le concept de la conscience de situation a été introduit par Endsley dans les années 1990 [54]. Il l'a défini comme : « La perception des éléments de l'environnement dans un volume de temps et d'espace, la compréhension de leur signification et la projection de leur état dans le futur proche ».

Originellement développée pour les pilotes d'avion, l'analogie avec d'autres secteurs, dont la conduite automobile a été faite par la suite. Un conducteur, avec une conscience de la situation « forte », a ainsi une bonne vision de son environnement et sait comment s'interfacer dans celui-ci pour accomplir sa mission. À l'inverse, un conducteur avec une conscience de situation « faible » est dans une situation qui lui semble complexe, il est perdu dans l'environnement.

Le modèle de conscience de situation de Endsley se décompose en trois parties, comme le montre la Figure 4.1. Ce modèle intègre la relation entre la conscience de situation et la prise de décision, ce qui le rend intéressant dans le cadre de la conduite autonome. Les concepts d'expertise et de mémoire sont aussi ajoutés. En effet, à environnement identique, la conscience de situation peut varier en fonction du conducteur. Endsley fait donc l'hypothèse que la conscience de situation est influencée par l'expérience et l'entraînement.

La conscience de situation est donc la capacité à comprendre ce qui est en train de se passer et à anticiper ce qui va se passer autour du véhicule. Le modèle d'Endsley est dit « linéaire » [26] et décompose la conscience de situation en trois modules positionnés en série. Le premier module est celui de la perception des éléments de l'environnement. Par exemple, le conducteur perçoit les voitures autour de lui, les panneaux, les marquages au sol, les piétons, etc. À cette étape, aucune interprétation n'est faite, mais un conducteur expérimenté arrivera mieux à distinguer et organiser les éléments pertinents de l'environnement qu'un conducteur novice. La deuxième

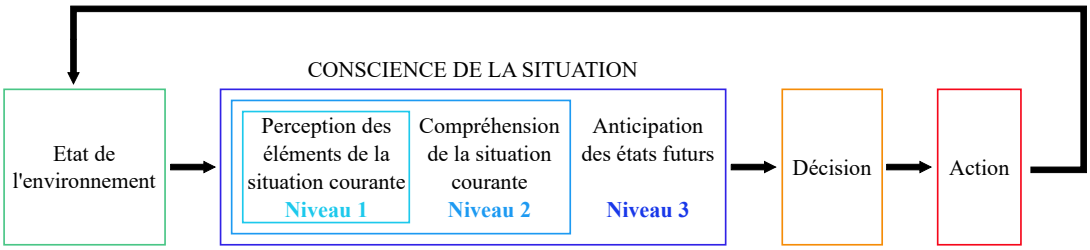


FIGURE 4.1 – Schéma du concept de la conscience de situation [26]

phase de la conscience de la situation consiste à intégrer ces éléments entre eux pour comprendre et interpréter la situation. Enfin, la dernière étape est l'anticipation des futurs états des objets de l'environnement pour décider de la meilleure action à prendre. Par exemple, si un conducteur roule derrière une voiture et que des piétons s'apprêtent à traverser un peu plus loin, le conducteur va anticiper le fait que la voiture de devant va freiner puis s'arrêter devant le passage piéton. Le conducteur va donc se préparer à freiner. Là aussi, en fonction de l'expérience du conducteur, les scènes ne sont pas interprétées de la même manière.

Outre le modèle d'Endsley, il existe d'autres modèles de conscience de la situation [26]. Ce concept peut être utilisé dans le cadre de la conduite autonome pour la partie analyse de l'environnement. Il permet ainsi de découper le problème d'analyse de manière simple et logique.

Dans cette thèse, les problèmes de compréhension (niveau 2) et d'anticipation (niveau 3) sont traités pour la conduite autonome. La partie perception (niveau 1) étant réalisée essentiellement par les fournisseurs de capteurs, elle représente une donnée d'entrée. Dans un premier temps, ces problèmes sont traités sur des voies à chaussées séparées pour des questions de base de données. En effet, pour l'instant les prototypes du Groupe PSA ont essentiellement roulé sur autoroute, il y a très peu d'enregistrements en milieu urbain. De plus, il apparaît plus judicieux de tester et valider les méthodes dans des environnements « simples », puis de les adapter pour les situations de vie plus complexes.

Ainsi, l'objectif de ce chapitre est, dans un premier temps, de détecter les intentions des véhicules autour de l'ego véhicule sur autoroute, et dans un second temps de déterminer leurs trajectoires futures, toujours sur autoroute. Une ouverture est faite sur le traitement de ces problématiques en environnement urbain dans le cas du rond-point.

4.1.2 État de l'art : détection d'intention

Le changement de voie est la seule manœuvre latérale possible d'un véhicule sur autoroute. Cela représente à la fois des risques de collision [130], mais aussi des perturbations sur la fluidité du trafic. Dans le cadre de la conduite autonome sur autoroute, l'ego véhicule doit donc être capable d'analyser son environnement et de détecter si un véhicule est en train de changer de voie. Idéalement, la détection doit intervenir avant que le véhicule ait effectivement changé de voie afin d'anticiper le freinage, ou la décélération, et ainsi de limiter le risque de collision. Cette anticipation permet aussi de rendre la conduite plus confortable et moins stressante pour l'utilisateur.

Vu que cette question est critique pour la sécurité du système et des usagers, des recherches ont déjà été menées sur le sujet. Le sujet est assez large et ne traite pas seulement des changements de voie, mais plus généralement des comportements conducteurs des autres véhicules

de l'environnement. La majorité des études peut être séparée en deux catégories : les méthodes probabilistes et les méthodes par intelligence artificielle.

Plusieurs recherches ont étudié le problème d'interprétation du comportement conducteur avec les méthodes probabilistes. Dans [95], les auteurs utilisent des réseaux bayésiens pour détecter les changements de voie. Dans [179], la théorie de Dempster-Shafer est utilisée pour la prédiction de manœuvre sur autoroute. Ces manœuvres sont constituées du changement de voie, dépassement et insertion. [16] et [118] estiment, avec des modèles de Markov cachés (HMM pour « Hidden Markov Model »), quelles manœuvres sont en train de réaliser les véhicules. [114] utilise aussi des HMM, mais pour détecter les changements de voie de l'ego véhicule afin de modéliser le comportement conducteur pour la sécurité des systèmes d'aide à la conduite actifs. La différence entre ces trois méthodes réside dans le fait que la méthode par Dempster-Shefer est basée sur des mesures de croyance de l'état de l'environnement, alors que les réseaux bayésiens utilisent des probabilités conditionnelles et les modèles de Markov cachés des distributions de probabilités sur des séquences d'observation. L'avantage des méthodes probabilistes est qu'elles sont faciles à comprendre et à interpréter. Cependant, il est souvent difficile de transcrire le problème en distribution de probabilités.

Une autre solution du problème de détection de changement de voie des véhicules autour de l'ego est l'utilisation de méthodes dites à base d'intelligence artificielle. Ainsi, un réseau de neurones est utilisé dans [200] et des machines à vecteurs de support dans [5]. Les réseaux de neurones sont intéressants car ils n'ont pas besoin de modèle et peuvent représenter n'importe quel modèle non linéaire. Par contre, la phase d'apprentissage, c'est-à-dire l'identification des paramètres du réseau, nécessite une base de données importante pour modéliser le problème correctement et l'interprétation des résultats peut être complexe, car le modèle est difficile à interpréter.

Les autres méthodes incluent la logique floue [13] ou les méthodes au cas par cas [66], pouvant être étendues avec de nouveaux cas rencontrés lors de la conduite.

4.1.3 État de l'art : prédiction de trajectoire

Comme le suggère le concept de conscience de situation, le fait d'anticiper et de prévoir les mouvements des objets de l'environnement va aider le conducteur à décider comment agir sur celui-ci. De la même manière, en fonction des méthodes de génération de trajectoire utilisées, le fait de prédire la trajectoire future des exo véhicules, c'est à dire des véhicules autres que l'ego véhicule, dans sa globalité est une information qui peut être essentielle pour la conduite autonome.

D'après l'état de l'art sur la prédiction de trajectoire et le calcul des risques de collision [111], trois méthodes principales sont utilisées : une méthode par extrapolation de l'état actuel de la dynamique de l'objet [85] et [126], une méthode par classification de la manœuvre, [139] et [178], et une dernière méthode par interaction entre les usagers, [91] et [110]. Ces méthodes sont illustrées sur la Figure 4.2 et numérotées respectivement de (a) à (c).

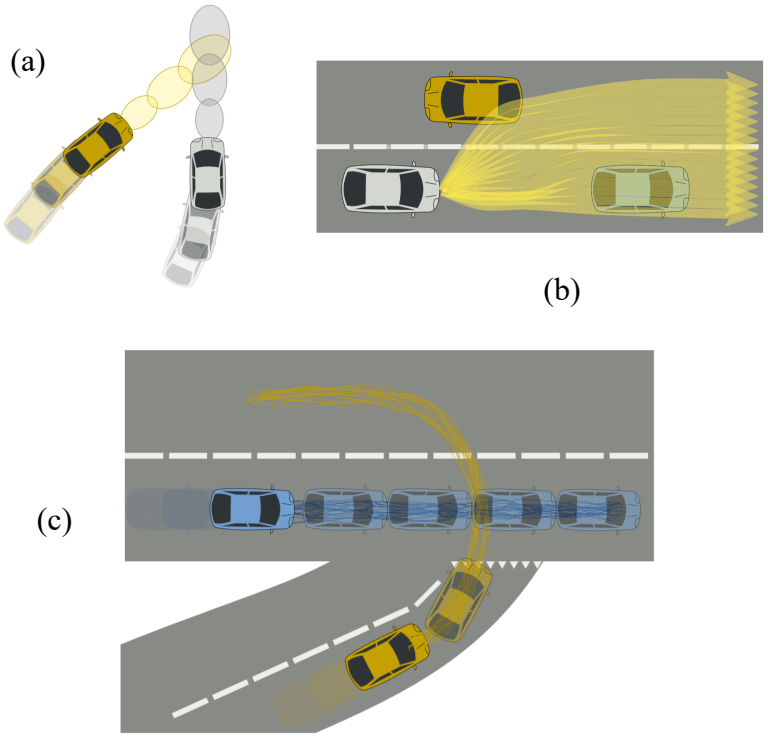


FIGURE 4.2 – Illustrations issues de [111] des méthodes de prédiction de trajectoire par modèle dynamique (a), détection de manœuvre (b) et interaction entre véhicules (c)

Plusieurs recherches font aussi le choix de combiner les méthodes avec modèle dynamique et détection de manœuvre [191], [81] et [120]. En effet, la première méthode n'est efficace que sur le court terme alors que la deuxième l'est plutôt sur le long terme. L'idée est donc d'avoir une démarche permettant d'avoir une meilleure précision de la trajectoire entre le début et la fin de la manœuvre. [191] établit une telle démarche dans le but de prédire la trajectoire de l'ego véhicule. Dans [81], les manœuvres de changement de voie et de maintien dans la voie sont considérées. La manœuvre est détectée avec les derniers échantillons de position du véhicule et leurs positions par rapport au centre de la ligne. La prédiction de trajectoire est ensuite une fonction pondérée de deux trajectoires, un polynôme d'ordre 5 dépendant de la manœuvre et une trajectoire basée sur un modèle à lacet et accélération constante.

Les modèles physiques manquent de précision car les modèles utilisés sont forcément simplifiés du fait que les capteurs ne permettent pas d'avoir une connaissance suffisante des états physiques des exo véhicules comme le cap, le lacet ou accélération. C'est pourquoi [120] suppose que les véhicules puissent communiquer ce type d'information entre eux pour avoir des prédictions plus proches de la réalité.

En environnement urbain, [79] compare un modèle cinématique de prédiction de trajectoire avec un filtre particulaire. Un filtre particulaire va estimer l'état caché d'un système dynamique bruité, modélisé par une chaîne de Markov. La deuxième méthode se révèle beaucoup plus satisfaisante surtout aux horizons de temps élevés. [144] utilise une technique de graphe d'exploration RRT (« Rapidly-exploring Random Tree ») pour trouver un chemin admissible que pourrait suivre le véhicule dans l'environnement et un modèle de mélange gaussien. La méthode présente des limites lorsque le graphe se trompe dans la manœuvre que va effectuer le véhicule, les erreurs de prédiction sont alors importantes.

Enfin, les méthodes d'apprentissage profond ont aussi été utilisées pour la prédiction de trajectoire avec des réseaux de neurones de type LSTM (« Long Short Term Memory »). Ces réseaux présentent une mémoire interne leur permettant d'analyser des séquences. Dans [101], le réseau utilise en entrée l'historique des positions et vitesses des exo véhicules et les variables de l'ego véhicule telles que l'angle de lacet et la vitesse longitudinale. La future position de l'exo véhicule dans une grille d'occupation prédéfinie est alors prédite à un horizon entre 0,5 s et 2 s. La méthode montre une erreur moyenne absolue de 87 cm à 2 s. Dans [4], le LSTM inclut aussi comme caractéristiques d'entrée le type du véhicule et le temps de collision aux différents objets. La précision latérale est de 70 cm à 10 s dans le futur.

Le LSTM est une approche séduisante car il peut, à partir de l'analyse d'une séquence en entrée, renvoyer une séquence en sortie, et donc une trajectoire. Cependant, dans les études menées par [101] et [4], [101] renvoie la position future du véhicule dans une grille et non pas sa trajectoire, et [4] utilise les données publiques de Next Generation Simulation (NGSIM). Sur ces données, les positions des véhicules sont très bruitées, car obtenues par des analyses vidéos de caméras positionnées sur les infrastructures. Les données d'entraînement du réseau ne sont donc pas représentatives de ce que « voit » l'ego véhicule par ses capteurs.

4.1.4 Introduction à l'apprentissage automatique

Comme il a pu être observé au travers de l'état de l'art de la prédiction de manœuvre et la prédiction de trajectoire, les techniques d'apprentissage automatique sont très utilisées dans ces domaines. De manière générale, dans le secteur du véhicule autonome, ce type d'algorithme est très répandu pour tout ce qui concerne la partie perception de l'environnement et interprétation. En effet, les algorithmes d'apprentissage profond par exemple sont très performants pour l'analyse d'images, que ce soit pour la classification [104] ou pour la détection d'objet dans l'image [158].

L'apprentissage automatique, plus connu par son expression anglaise « machine learning » , est un secteur d'étude de l'intelligence artificielle, de l'informatique et des mathématiques pour la modélisation numérique de problèmes sans l'écriture de règles spécifiques. L'idée derrière l'apprentissage automatique est de pouvoir concevoir des modèles directement à partir de données d'échantillonnages. En effet, les problématiques de traitement d'image telle que la reconnaissance de forme, ou de traduction automatique sont difficiles à mettre en forme algorithmiquement à la main. L'apprentissage automatique permet, à partir d'une analyse statistique d'une base de données, de construire un modèle reproduisant le comportement désiré.

À partir de cette base, il y a trois grandes familles d'apprentissage de modèle : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. Pour un apprentissage supervisé, la sortie de chaque élément de la base de données est connue. Le modèle généré doit être capable de prédire la bonne sortie pour une nouvelle entrée. Suivant que la valeur cible à prédire est continue ou discrète, le problème d'apprentissage est un problème de régression ou de classification. L'apprentissage non supervisé n'a quant à lui pas de sortie prédéterminée. L'algorithme d'apprentissage doit alors être capable d'identifier la structure des données à travers leurs similarités. Enfin, l'apprentissage par renforcement conçoit un modèle maximisant une fonction de récompense en fonction de l'impact de ce modèle sur l'environnement [171]. Par exemple, si on veut apprendre la fonction ACC par renforcement, une fonction de récompense pourrait se baser sur le temps inter-véhicule (TIV) entre l'ego véhicule et le véhicule de devant. Ainsi, si le

TIV est inférieur à une valeur désirée, la fonction de récompense est faible, le comportement est pénalisé. À l'inverse, si le véhicule respecte les distances de sécurité, la fonction de récompense est importante et ce type de comportement est privilégié par la suite.

Outre les différents types d'apprentissages, il existe plusieurs algorithmes pour définir les modèles en fonction d'une base de données. Les plus connus sont les arbres de décision, les machines à vecteurs de support et les réseaux de neurones.

Arbres de décision et forêts aléatoires

Les arbres de décision sont des outils d'aide à la décision se basant sur des modèles de graphes. Les décisions sont produites par une série de tests à chaque nœud du graphe. Pour des problèmes de classification, les arbres de décisions permettent de répartir l'espace de données en sous-espaces homogènes en fonction des attributs des éléments de la base de données. La Figure 4.3 (a) montre un exemple d'arbre de décision. L'avantage de cette méthode pour la classification est que les arbres de décision sont facilement compréhensibles et peuvent utiliser des données symboliques. Des algorithmes d'apprentissage automatique, notamment CART [20] et ID3 [150], existent pour déterminer les liens et règles entre les attributs qui peuvent être difficilement identifiables à partir de la base de données brute. Par contre, les algorithmes d'apprentissages sont très sensibles au bruit et peuvent avoir du mal à représenter certaines fonctions, comme la fonction XOR par exemple. De plus, les arbres de décision peuvent induire des problèmes de surapprentissage si l'arbre de décision est trop profond et que plus de branches que nécessaire ont été créées. Pour éviter que l'arbre ne grandisse trop, des techniques d'« élagage » existent, [122] mais sont souvent complexes.

Les forêts aléatoires [19], ou forêts d'arbres de décision, ont été développées dans le but de pallier les problèmes de sur-apprentissage et de sensibilité des arbres de décision. Le principe des forêts aléatoires est d'utiliser plusieurs arbres de décision en parallèle. La sortie est alors celle choisie par le plus grand nombre d'arbres. Les arbres sont créés de manière aléatoire afin d'être le moins corrélés possible. Ainsi, chaque arbre est défini à partir d'un sous-ensemble (dont les données sont réutilisables pour les autres arbres) de la base de données d'entraînement et n'utilise qu'une partie des attributs de chaque entrée. Le principe des forêts aléatoires est présenté sur la Figure 4.3 (b). Contrairement aux arbres de décision simples, les forêts aléatoires sont plus précises, plus robustes et présentent moins de risque de sur-apprentissage. En contrepartie, les forêts aléatoires perdent en lisibilité par rapport aux arbres de décision et sont donc moins facilement interprétables.

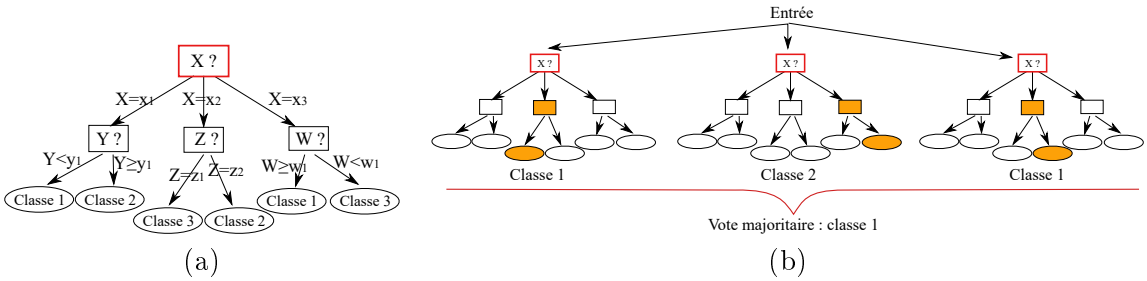


FIGURE 4.3 – Schémas d'un arbre de décision (a) et d'une forêt aléatoire (b)

Machines à vecteurs de support

Les machines à vecteurs de support [35], ou SVM pour « Support Vector Machine », sont des méthodes de classification de données. Les vecteurs de support peuvent être vus comme des « frontières » entre les classes. Le but des SVM est d'optimiser la marge de ces vecteurs entre les classes. Il y a deux étapes dans la détermination des vecteurs de support. La première est de rendre linéaire la séparation entre les classes afin de pouvoir définir ces vecteurs, comme indiqué sur la Figure 4.4 (a). Pour cela, la dimension de l'espace de représentation des données est augmentée jusqu'à obtenir une séparation linéaire. La deuxième étape consiste à maximiser la marge entre le vecteur et les échantillons les plus proches, comme illustré par la Figure 4.4 (b).

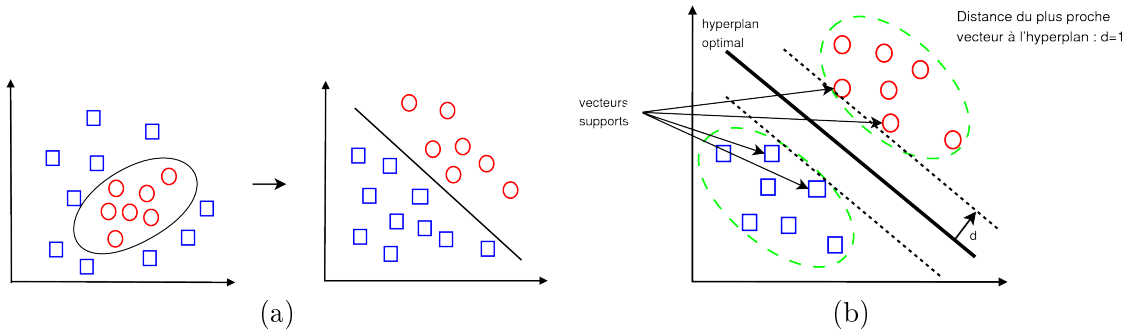


FIGURE 4.4 – Linéarisation de la séparation des classes pour le SVM (a) et calcul de la marge (b) [Source :[156]]

Les SVM sont reconnus pour leurs précisions. De plus, ils sont très efficaces sur de petites bases d'entraînement. Cependant, les SVM présentent un temps de calcul considérable pour les grandes bases de données et voient leurs efficacités réduites si les données sont bruitées et présentent des cas aberrants.

Réseaux de neurones

Les réseaux de neurones sont des modèles universels, ils sont capables de représenter n'importe quelle fonction continue. Ils peuvent être utilisés à la fois pour la classification et la régression. La théorie des réseaux de neurones puise son concept dans les relations neuronales biologiques. Dans le cerveau, chaque neurone peut créer, envoyer et recevoir des signaux électrochimiques. Un neurone émet une impulsion si le signal transmis dépasse un seuil de déclenchement. Mis bout à bout, les neurones sont capables d'effectuer des tâches cognitives complexes, d'où l'idée de se baser sur un tel concept pour l'apprentissage automatique. Dans un réseau de neurones artificiel, chaque neurone reçoit des entrées pondérées par les paramètres du réseau. Ces entrées sont ensuite combinées et évaluées à partir de la fonction d'activation du neurone qui va ensuite transmettre cette information au reste du réseau. L'apprentissage d'un réseau de neurones consiste à trouver les paramètres pondérant chaque lien entre les neurones pour reproduire correctement le fonctionnement entrée/sortie de la base de données.

Il y a trois grandes familles de réseaux de neurones : les perceptrons multicouches MLP (« Multilayer Perceptron »), les réseaux de neurones convolutifs CNN (« Convolutional Neural Network ») et les réseaux de neurones récurrents RNN (« Recurrent Neural Network »), [64]. Les MLP sont la représentation classique des réseaux de neurones. Ils possèdent plusieurs couches de neurones, voir Figure 4.5 (a). Lorsque le réseau possède plus de trois couches internes, ou couches cachées, on parle alors d'apprentissage profond ou « deep learning ». Les MLP servent plutôt pour

les problèmes de classification et de régression avec des données d'entrée sous forme de tableaux numériques. Les CNN sont inspirés du cortex visuel et sont très efficaces pour le traitement et la reconnaissance d'image. La première étape de traitement de l'image par CNN est la convolution. En utilisant une ou plusieurs matrices de convolution successivement sur l'image, ou une partie de l'image, les caractéristiques de l'image peuvent être identifiées. Ensuite, un réseau de neurones de type MLP est utilisé en sortie pour déterminer, à partir de ces caractéristiques, quelle est la classe de l'image. Le fonctionnement d'un CNN est visible sur la Figure 4.5 (b). Enfin, les RNN sont des réseaux de neurones présentant des récurrences. Contrairement au MLP, il existe donc des cycles entre les neurones dans ce type de structure, voir Figure 4.5 (c). Cet aspect des RNN les rend très efficaces pour l'analyse de séquences telles des phrases ou des enregistrements vocaux.

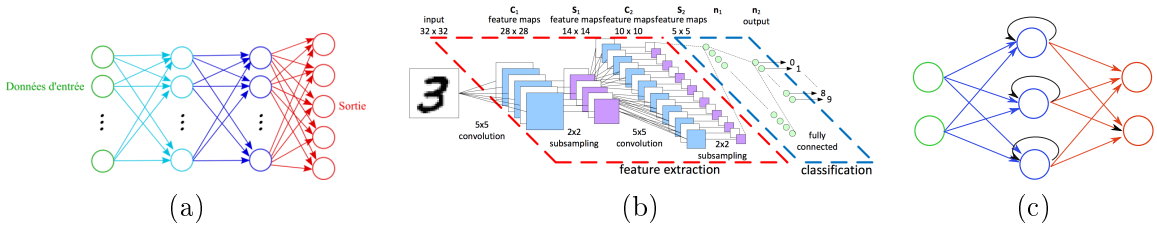


FIGURE 4.5 – Exemples de structures pour des réseaux de neurones MLP (a), CNN (b) et RNN (c)

L'avantage des réseaux de neurones est leur aptitude à reproduire n'importe quel comportement non linéaire. De plus, en fonction du problème à traiter (traitement d'image, traduction automatique, classification, etc.), plusieurs structures types existent déjà dans la littérature en fonction du problème à traiter. Contrairement aux arbres de décision, les réseaux de neurones sont efficaces avec des données bruitées ou incomplètes. De plus, comme les réseaux de neurones sont de plus en plus utilisés, de nombreux outils pour leur exploitation ont été développés ces dernières années. L'inconvénient majeur des réseaux de neurones, qui est souvent mis en avant, est leur effet « boîte noire », car il est souvent difficile de trouver des justifications aux solutions données par le réseau. Cela peut donc compliquer l'interprétation des résultats. Un autre inconvénient est la définition de l'architecture du réseau. Il n'existe pas encore de méthode pour déterminer le nombre de neurones et le nombre de couches nécessaires dans le réseau en fonction d'un problème.

4.1.5 Objectifs et méthodes

L'objectif de ce chapitre est la détection de manœuvres et la prédiction de trajectoires des véhicules en environnement urbain. Par rapport à l'autoroute, les situations de vie sont plus complexes, que ce soit au niveau de la structure de la route ou au niveau des règles de circulation et interactions avec les autres usagers. En effet, sur autoroute, tous les véhicules circulent dans le même sens et sur des routes peu courbées. En ville cependant, les véhicules se croisent dans les intersections et les ronds-points, et peuvent se dépasser sur des routes à double sens. Beaucoup de règles du Code de la route viennent aussi s'ajouter en ville : les stops et cédez le passage aux intersections, les priorités à droite, les passages piétons, les flèches de marquage au sol, etc. Enfin, d'autres usagers apparaissent en ville alors qu'ils ne sont pas du tout présents sur les routes à grandes vitesses : les piétons, les vélos ou encore les scooters.

L'ego véhicule doit donc être capable de gérer beaucoup plus de situations complexes en ville que sur autoroute. Et pour les gérer, il doit être capable d'anticiper correctement le mouvement

des autres usagers. Étant donné la complexité de ces situations, les méthodes d'apprentissage artificiel, et notamment les réseaux de neurones, sont une approche séduisante. En effet, ces situations peuvent être difficiles à modéliser et interpréter par des méthodes classiques. On peut penser qu'avec une base de données importante, les réseaux de neurones soient capables de représenter ces scénarios correctement et d'anticiper les futures positions des autres véhicules.

Pour pouvoir représenter correctement les comportements des exo véhicules, il faut une base de données conséquente. Cependant, les prototypes du Groupe PSA, au moment de cette étude, n'ont pas effectué suffisamment de roulage en ville pour utiliser des réseaux de neurones. Afin de définir une stratégie de détection d'intention et de prédiction de trajectoire, nous avons donc utilisé une base de données créée à partir de roulages sur autoroute. Cela va permettre de confronter plusieurs méthodes et de les valider pour ensuite les adapter en environnement urbain. Pour avoir un scénario transposable de l'autoroute à l'urbain, les stratégies développées se concentrent alors sur les manœuvres de changement de voie.

Pour la détection d'intention, une méthode classique à base de règles probabilistes et de prédiction de trajectoire est comparée à un réseau de neurones MLP pour déterminer si un véhicule de l'environnement est en train de changer de voie. Pour la prédiction de trajectoire, plusieurs structures de réseaux de neurones de type RNN sont comparées.

4.2 Les réseaux de neurones : fonctionnement et apprentissage

La deuxième méthode de détection de changement de voie utilise des réseaux de neurones. Au début du chapitre, nous avons expliqué de manière macroscopique le principe de fonctionnement d'un réseau de neurones. Ici, nous allons détailler comment une sortie d'un réseau de neurones est calculée et comment celui-ci est entraîné pour un problème de classification.

4.2.1 Fonctionnement d'un neurone

Un neurone seul a pour entrée un vecteur de données $d = [d_1 \ d_2 \ \cdots \ d_n]^T$. Ces données sont pondérées par un vecteur de paramètres $\Theta = [\theta_0 \ \theta_1 \ \cdots \ \theta_n]^T$. La somme pondérée des paramètres du neurone et des données d'entrée est utilisée pour calculer la sortie h_Θ grâce à une fonction d'activation. Les fonctions d'activation sont généralement la fonction d'Heaviside, la tangente hyperbolique ou la sigmoïde. Nous utilisons ici cette dernière.

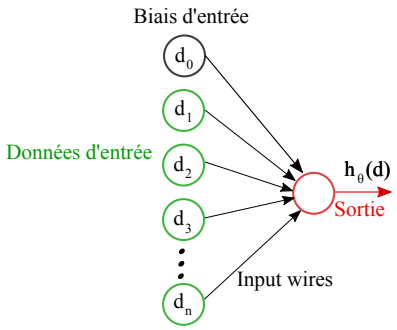


FIGURE 4.6 – Représentation d'un neurone

Pour le réseau de neurones précédent, représenté sur la Figure 4.6, et avec une entrée $d =$

$[d_1\ d_2\ \cdots\ d_n]^T$, la sortie se calcule par :

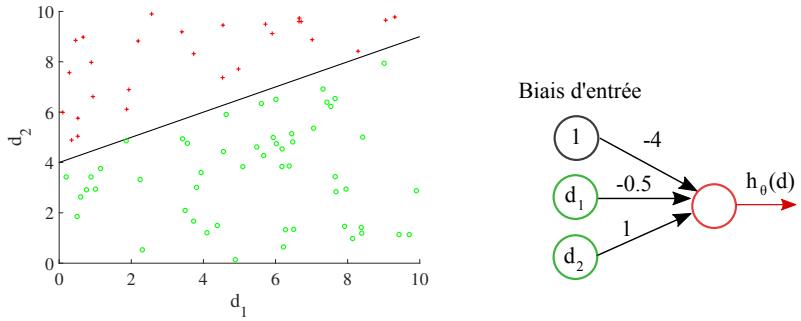
$$h_\Theta(d_b) = \frac{1}{1 + e^{-\Theta^T d_b}}. \tag{4.1}$$

avec d_b l'entrée d avec un biais $d_0 = 1$ telle que :

$$d_b = \begin{bmatrix} d_0 \\ d \end{bmatrix}. \tag{4.2}$$

L'ajout d'un biais rend le neurone plus flexible. En effet, si les poids représentent la « forme » du modèle, le biais positionne son zéro. L'intérêt du biais et le fonctionnement d'un neurone pour un problème de classification sont illustrés dans l'exemple suivant.

L'objectif est de déterminer, pour une entrée $d = [d_1\ d2]^T$ si cette classe appartient à la classe $+$ ou la classe **o** dont la répartition est représentée sur la Figure 4.7 (a). Avec un neurone pondéré comme sur la Figure 4.7 (b), il est possible d'identifier la classe de d.



(a) (b)

FIGURE 4.7 – Répartition des classes (a) et représentation du neurone pour la classification (b)

Pour d, la sortie vaut :

$$h_\Theta(d) = \frac{1}{1 + e^{-\Theta^T d_b}}, \tag{4.3}$$

avec $d_b = [1\ d^T]^T$ et

$$\Theta^T d_b = \theta_0 + \theta_1 d_1 + \theta_2 d_2, \tag{4.4}$$

sachant qu'ici $\theta_0 = -4$, $\theta_1 = -0.5$ et $\theta_2 = 1$.

Si $h_\Theta(d) > 0,5$, cela signifie que $\Theta^T d_b > 0$ et que, sur la Figure 4.7 (a), d est au-dessus de la droite noire et donc appartient à la classe $+$. À l'inverse, si $h_\Theta(d) < 0,5$, d appartient à la classe **o**.

4.2.2 Propagation de l'information dans le réseau

En général, un réseau est constitué de plusieurs couches de neurones et chaque couche contient plusieurs neurones. On appelle couche d'entrée la couche du vecteur d'entrée et couche de sortie la couche contenant les neurones calculant les sorties. Les couches intermédiaires constituent la couche cachée du réseau. Pour illustrer comment l'information se propage du vecteur d'entrée à la sortie, prenons l'exemple d'un réseau à n entrées, une seule couche cachée composée de trois

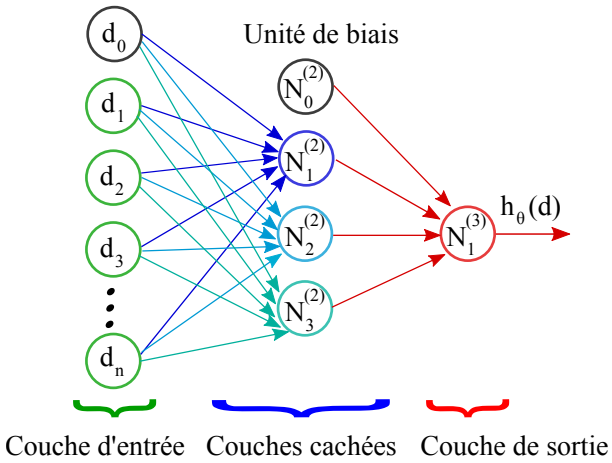


FIGURE 4.8 – Représentation d'un neurone

neurones et une sortie, comme illustré sur la Figure 4.8.

On appelle $N_i^{(j)}$ le $i$ème neurone de la couche j. Ainsi, pour la première couche cachée :

$$\begin{aligned} N_1^{(2)} &= g\left(\theta_{10}^{(1)} d_0 + \theta_{11}^{(1)} d_1 + \theta_{12}^{(1)} d_2 + \cdots + \theta_{1n}^{(1)} d_n\right) \\ N_2^{(2)} &= g\left(\theta_{20}^{(1)} d_0 + \theta_{21}^{(1)} d_1 + \theta_{22}^{(1)} d_2 + \cdots + \theta_{2n}^{(1)} d_n\right) \\ N_3^{(2)} &= g\left(\theta_{30}^{(1)} d_0 + \theta_{31}^{(1)} d_1 + \theta_{32}^{(1)} d_2 + \cdots + \theta_{3n}^{(1)} d_n\right), \end{aligned} \tag{4.5}$$

avec $g(z)$ la fonction d'activation sigmoïde qui s'écrit :

$$g(z) = \frac{1}{1 + \mathrm{e}^{-z}}, \tag{4.6}$$

et $\theta_{ij}^{(l)}$ la pondération de la $j$ème entrée du $i$ème neurone de la couche l à la couche $l+1$. La couche de sortie donne alors :

$$h_\theta(d) = g\left(\theta_{10}^{(2)} N_0^{(2)} + \theta_{11}^{(2)} N_1^{(2)} + \theta_{12}^{(2)} N_2^{(2)} + \theta_{13}^{(2)} N_3^{(2)}\right). \tag{4.7}$$

Le fait d'augmenter la hauteur du réseau en ajoutant des neurones dans chaque couche et d'augmenter la profondeur du réseau en ajoutant des couches permet d'accroitre et de complexifier les relations entre les paramètres d'entrée. Le modèle entrée/sortie du réseau aura donc plus de possibilités combinatoires entre les différentes entrées et permettra donc la représentation de modèles très variés. Ainsi, si un neurone permet la classification d'une fonction linéaire du premier ordre comme vu précédemment, un réseau de neurones avec plusieurs couches cachées et un grand nombre de neurones par couche peut permettre par exemple la classification d'image, voir Figure 4.9. Pour la classification du type d'objet dans l'environnement à partir d'une image, chaque donnée d'entrée correspond à un pixel de l'image et chaque sortie indique si oui ou non, l'image appartient à la classe considérée. Pour les problèmes de classification d'image, les réseaux de neurones convolutifs sont cependant plus performants que les réseaux MLP.

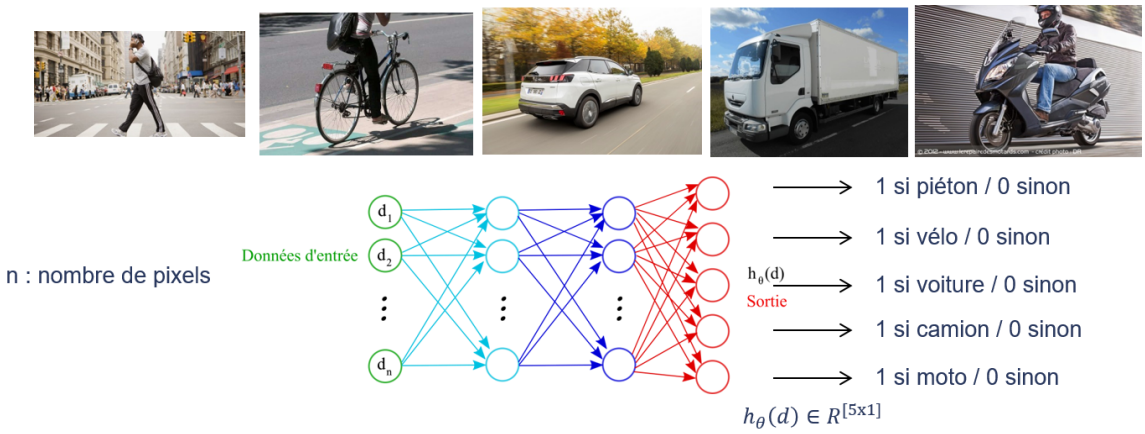


FIGURE 4.9 – Réseau de neurones profond pour la classification d'images

4.2.3 Entraînement du réseau : la rétropropagation

La phase d'apprentissage des paramètres Θ du réseau utilise une méthode appelée rétropropagation [71, 162]. L'objectif de l'apprentissage est de trouver les paramètres du réseau qui minimisent une fonction de coût $J(\Theta)$. Il existe plusieurs fonctions de coût utilisées dans la littérature, nous utiliserons celle de la forme :

$$J(\Theta) = \frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{2}\Big(c^{(i)}[k]\log(\widehat{c}^{(i)}[k])+(1-c^{(i)}[k])\log(1-\widehat{c}^{(i)}[k])\Big), \tag{4.8}$$

avec m le nombre d'exemples dans la base de données, $c^{(i)}[k]$ la k^{eme} ligne de la sortie réelle correspondant au i^{eme} vecteur d'entrée d de la base de données et $\widehat{c}^{(i)}[k]$ la k^{eme} ligne de la sortie calculée par le réseau avec le même vecteur d'entrée d et les paramètres Θ.

Pour pouvoir mettre à jour les paramètres en fonction de l'erreur de sortie, il faut calculer le gradient de la fonction de coût $J(\Theta)$. C'est cette étape qui est effectuée par la méthode de rétropropagation. Intuitivement, la rétropropagation consiste à propager l'erreur de sortie vers l'entrée, conduisant au calcul du gradient. Une fois le gradient calculé, une méthode d'optimisation telle que la descente de gradient est utilisée pour mettre à jour les paramètres Θ du réseau. Dans ce paragraphe, nous allons uniquement expliquer comment calculer le gradient de la fonction de coût par rétropropagation.

Les notations utilisées ici sont identiques à celles de l'exemple précédent. Ainsi la matrice Θ de paramètres est une matrice à trois dimensions composée d'éléments $\theta_{ij}^{(l)}$ correspondant à la pondération de l'entrée j du neurone i de la couche l. La sortie du neurone i de la couche l est notée $N_i^{(l)}$ et $N^{(l)}$ correspond au vecteur de sortie de la couche l. Pour la couche d'entrée, à savoir $l = 1$, on a : $N_i^{(l)} = d_i$. On note $\delta_i^{(l)}$ l'erreur de propagation du noeud i de la couche l et $\delta^{(l)}$ le vecteur d'erreur de la couche l. En prenant l'exemple de la Figure 4.10 avec deux couches cachées, ces paramètres sont calculés par les relations suivantes. Pour la couche de sortie, l'erreur pour la sortie i est :

$$\delta_i^{(4)} = N_i^{(4)} - c[i], \tag{4.9}$$

ou pour toute la couche de sortie :

$$\delta^{(4)} = N^{(4)} - c. \tag{4.10}$$

Ensuite, la propagation de l'erreur dans les autres couches donne récursivement :

$$\begin{aligned} \delta^{(3)} &= \theta^{(3)^T}\delta^{(4)} \circ g'(\theta^{(2)}N^{(2)}) \\ \delta^{(2)} &= \theta^{(2)^T}\delta^{(3)} \circ g'(\theta^{(1)}N^{(1)}), \end{aligned} \tag{4.11}$$

avec $g'(z)$ la dérivée de la fonction d'activation (4.6). L'opérateur $\circ$ indique la multiplication terme à terme des matrices.

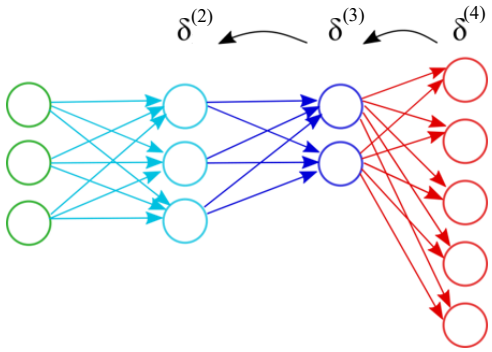


FIGURE 4.10 – Schématisation de la propagation de l'erreur

Le fait que la méthode soit récursive nécessite donc une initialisation des paramètres Θ. Ici, ces valeurs sont choisies aléatoirement, proches de 0 (inférieures à 0,1), afin d'empêcher la symétrisation des paramètres.

Remarque. Dû à l'ajout de biais sur la couche d'entrée et les couches cachées, les vecteurs $\delta^{(2)}$ et $\delta^{(3)}$ calculés avec l'équation (4.11) ont une valeur en trop : la première. Il faut donc l'enlever pour que les multiplications matricielles suivantes soient de la bonne dimension.

Avec m données d'entraînement, l'algorithme utilisant la rétropropagation pour le calcul du gradient de la fonction de coût $\dfrac{\partial J(\Theta)}{\partial \theta_{ij}^{(l)}}$ est le suivant :

Pour k allant de 1 à m **faire**
  Pour l allant de 2 à L **faire**
    Calculer $N^{(l)}$ par propagation avant
  Fin Pour
  Pour l allant de L à 2 **faire**
    Calculer $\delta^{(l)}$ par rétropropagation
  Fin Pour
  Pour l allant de 1 à $L-1$ **faire**
    Pour chaque entrée j et sortie i **faire**
      Mise à jour $\Lambda_{ij}^{(l)} = \Lambda_{ij}^{(l)} + N_j^{(l)}\delta_i^{(l+1)}$
    Fin Pour
  Fin Pour
Fin Pour
Calculer le gradient : $\dfrac{\partial J(\Theta)}{\partial \theta_{ij}^{(l)}} = \dfrac{1}{m}\Lambda_{ij}^{(l)}$

L'obtention du gradient de la fonction de coût en fonction des paramètres permet ensuite la mise à jour de ces derniers par la méthode de descente de gradient, illustrée par la Figure 4.11.

Cette méthode consiste à trouver le paramètre qui minimise une fonction en se déplaçant sur la courbe de celle-ci. Pour cela, l'algorithme se déplace de manière itérative jusqu'à atteindre un minimum. Le déplacement se fait en ajoutant au paramètre une pondération du gradient de la fonction en ce point. Lorsque le gradient du point est nul, l'algorithme s'arrête, un minimum a été trouvé. De manière générale, plusieurs itérations de rétropropagation sont nécessaires pour trouver Θ optimal.

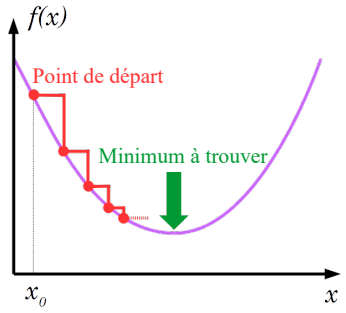


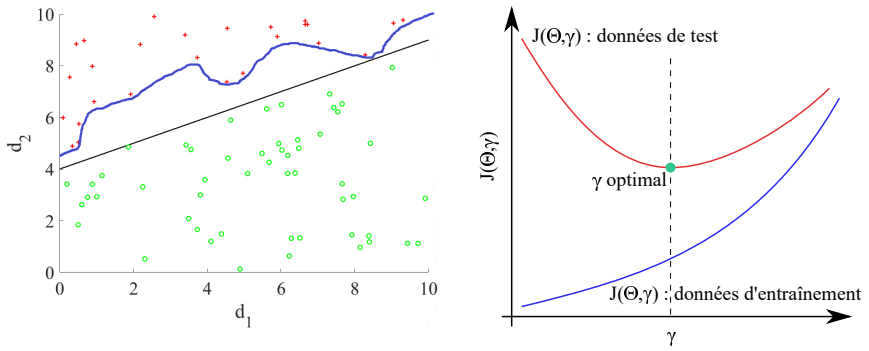
FIGURE 4.11 – Schématisation de la méthode de descente de gradient [Source : adapté de https://www.neural-networks.io/fr/single-layer/gradient-descent.php

Le phénomène de sur-apprentissage

Un problème récurrent dans l'apprentissage des réseaux de neurones est le sur-apprentissage. Ce phénomène apparaît lorsque le réseau est trop spécifique à la base de données utilisée pour l'entraînement. Le réseau est alors très efficace pour les données qu'il connaît, mais fait beaucoup d'erreurs pour les nouvelles entrées. Pour éviter ce problème, un terme de régularisation γ est ajouté dans la fonction de coût :

$$\begin{aligned} J(\Theta_1, \Theta_2, \gamma) = & \frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{2} \left(c^{(i)}[k] \log(\widehat{c}^{(i)}[k]) + (1 - c^{(i)}[k]) \log(1 - \widehat{c}^{(i)}[k]) \right) \\ & + \frac{\gamma}{2m} \sum_{l=1}^{La-1} \sum_{j=1}^{s_l[1]} \sum_{k=1}^{s_l[2]} \Theta_l[j][k], \end{aligned} \tag{4.12}$$

avec $La = 3$ le nombre de couches, $s_l = [p_l \; q_l]$ le vecteur de taille des matrices Θ_l avec p_l le nombre de lignes et q_l le nombre de colonnes. Le paramètre de régularisation γ pénalise les paramètres du réseau d'ordres importants pour simplifier le réseau et améliorer sa généralisation. Ceci peut s'illustrer à l'aide du problème de classification des classes $+$ et o précédent. Par exemple, avec un réseau trop grand et sans régularisation, au lieu de modéliser une fonction linéaire (courbe noire), le réseau pourrait plutôt représenter la courbe bleue de la Figure 4.12 (a). La régularisation permet aussi de limiter le phénomène d'augmentation de l'erreur avec un nombre de neurones trop important.



(a) (b)

FIGURE 4.12 – Illustration du phénomène de sur-apprentissage (a) et variation de la fonction de coût en fonction du paramètre de régularisation

La base de données est divisée en deux : une base d'entraînement réservée pour la rétropropagation et l'apprentissage des paramètres du réseau et une base de test utilisée pour la validation du réseau. La valeur de γ est déterminée de telle sorte qu'elle minimise la fonction de coût (4.8) de la base de test. Les réseaux sont entraînés avec plusieurs valeurs de γ avec la fonction de coût (4.12), puis (4.8) est calculé avec les bases d'entraînement et de test. La Figure 4.12 (b) montre l'allure des variations de la fonction de coût des réseaux entraînés pour les deux bases en fonction de γ.

4.2.4 Méthodologie de construction du réseau

Il n'y a pas de règles précises pour l'entraînement d'un réseau de neurones. C'est l'expérience du concepteur qui va primer dans l'analyse des entrées/sorties et des résultats obtenus. Cependant, même sans règles pour le nombre de couches cachées et de neurones, il est possible de mettre en place une démarche pour la conception et l'entraînement d'un réseau. La méthodologie mise en place ici est illustrée Figure 4.13.

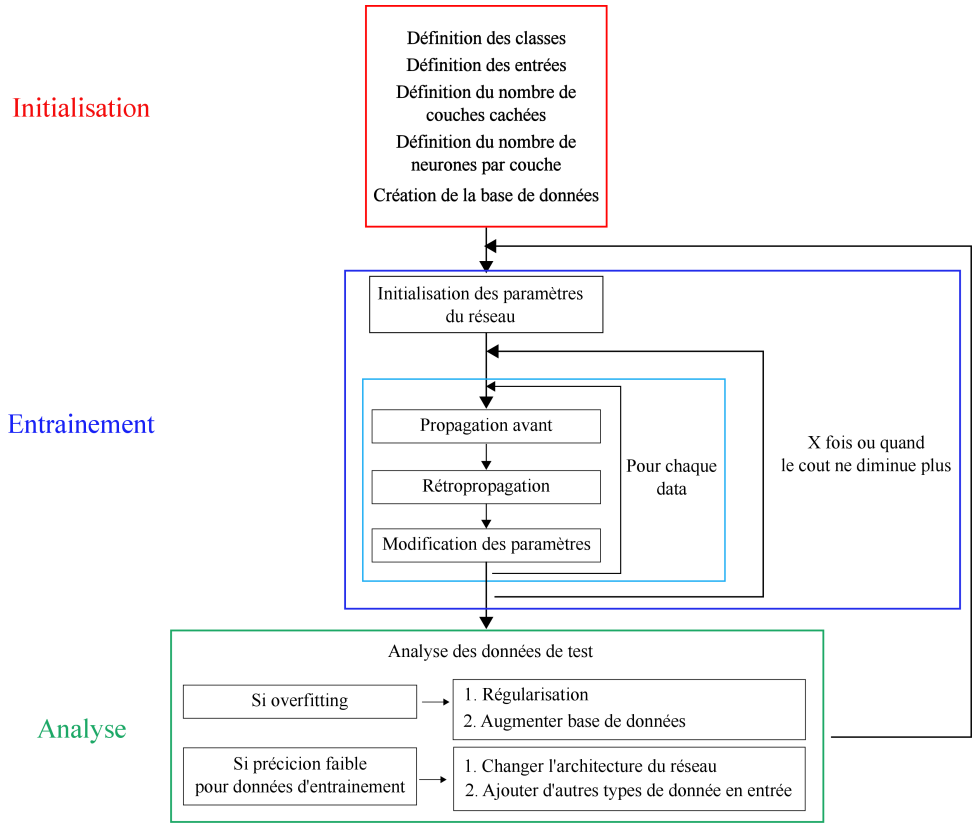


FIGURE 4.13 – Méthodologie pour l'élaboration et l'analyse des résultats d'un réseau de neurones

Cette méthodologie est découpée en trois phases : une phase d'initialisation, une phase d'entraînement et une phase d'analyse du réseau.

L'initialisation passe par la définition des entrées, des sorties et d'une première architecture du réseau, déterminée par le nombre de couches cachées et de neurones par couche.

Pour la phase d'entraînement, les paramètres du réseau sont dans un premier temps initialisés. Puis, pour chaque donnée de la base d'entraînement, les paramètres sont modifiés avec la méthode de rétropropagation. La boucle d'entraînement est répétée un nombre fixe de fois ou s'arrête lorsque le coût ne diminue plus.

Enfin, une phase d'analyse du réseau, avec une base de données de test, vérifie si le réseau a été entraîné correctement. La base de données de test est totalement décorrélée de la base d'entraînement. L'objectif est de vérifier si le réseau est représentatif des nouvelles entrées. Un phénomène souvent rencontré lors d'apprentissage de réseaux de neurones est le sur-apprentissage, ou « overfitting ». Cela signifie que le réseau est devenu trop spécifique à la base d'entraînement, il n'arrive pas à représenter correctement de nouvelles entrées. Ainsi, la précision du réseau sera très élevée pour les données d'entraînement et plus faible pour celles de test. En cas de sur-apprentissage, une première solution est d'ajouter un paramètre de régularisation dans la fonction de coût, une seconde consiste à augmenter la base de données.

Quand les données d'entraînement ont une précision faible en sortie, cela veut dire que la forme du réseau n'est pas adaptée au problème à modéliser. Les interactions entre les paramètres d'entrée sont potentiellement trop simples pour représenter correctement les comportements entrées/sorties. Il faut donc changer l'architecture du réseau pour voir si cela permet d'améliorer la précision. Une autre solution au problème de faible précision de la base d'entraînement est l'ajout d'autres types de données en entrée. En effet, il manque peut-être des informations au réseau pour pouvoir correctement estimer les sorties.

4.3 Détection d'intention : changement de voie sur autoroute

Pour la problématique de détection de changement de voie, deux méthodes sont comparées afin d'identifier le comportement des véhicules autour de l'ego. La première méthode est une méthode à base de règles qui utilise une première étape de prédiction de trajectoire et une deuxième étape de probabilités en fonction de cette prédiction. La deuxième méthode utilise un réseau de neurones dans un problème de classification de comportement. Ces deux méthodes sont comparées avec une base de données enregistrée sur le prototype TJC sur autoroute dans des conditions réelles de conduite.

4.3.1 Description du problème

Les scénarios

L'ego véhicule roule sur une route à chaussées séparées. Ainsi, tous les véhicules de l'environnement sont dans le même sens de circulation. L'objectif de l'étude des deux prochaines sections est de détecter si un véhicule de l'environnement, un exo véhicule, va changer de voie de circulation ou continuer de rouler dans sa voie. Sur autoroute, le fait de pouvoir anticiper le comportement des autres véhicules va permettre de pouvoir mieux gérer la vitesse longitudinale de l'ego véhicule. En anticipant les comportements des véhicules de l'environnement, le freinage ou l'accélération peut être anticipé. La conduite sera alors plus fluide et confortable. Sept cas de changement de voie ont été identifiés suivant la voie initiale du véhicule par rapport à l'ego et son action. Ces cas sont définis dans la Table 4.1 et illustrés Figure 4.14 (a).

Table 4.1 – Scénarios de changement de voie des exos véhicules

Cas	Voie initiale	Action
1	gauche	reste dans la même voie
2	milieu	reste dans la même voie
3	droite	reste dans la même voie
4	gauche	change vers la voie de l'ego
5	droite	change vers la voie de l'ego
6	milieu	change vers la voie à gauche de l'ego
7	milieu	change vers la voie à droite de l'ego

Les données utilisées pour la détection

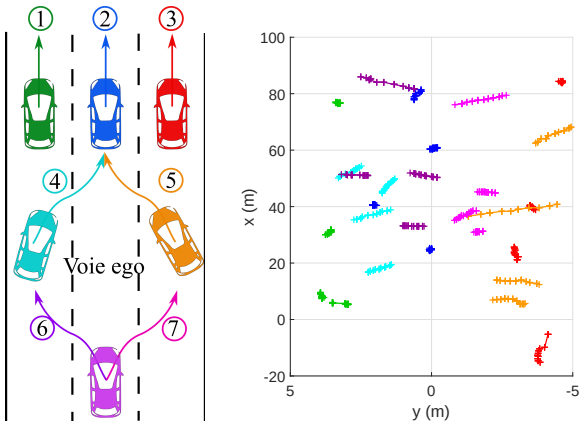
La base de données utilisée pour cette étude est créée à partir d'enregistrements de roulage de TJC sur des routes nationales et autoroutes proches de Paris. Pendant les enregistrements, l'ego véhicule est sur une route à deux ou trois voies avec chaussées séparées. Tous les autres véhicules de l'environnement roulent dans le même sens que l'ego. L'ego véhicule peut être soit à droite, soit au milieu, soit à gauche. Pour ne pas influencer la détection de changement de voie, l'ego véhicule ne change pas de voie dans les échantillons d'enregistrements utilisés. Tous les échantillons de la base de données ont été labellisés manuellement pour correspondre aux scénarios de la Table 4.1, illustrés sur la Figure 4.14 (a). Pour pouvoir réaliser cette étude, une grande quantité de données a été analysée et traitée pour isoler les cas de changement de voie, mais aussi les cas où les véhicules restent dans leur voie. Pour chacun des enregistrements, un découpage temporel est réalisé pour séparer les données considérées comme « passées » qui

servent à la prédiction et les données futures qui servent à la validation.

La position des objets est issue de la fusion de données des différents capteurs de TJC. Les exo véhicules sont positionnés dans le repère de l'ego véhicule. La caméra détecte les lignes et leur représentation est une fonction polynomiale d'ordre 3 de telle sorte que :

$$\begin{aligned} y_R(x) &= C_{0R} + C_{1R}x + C_{2R}x^2 + C_{3R}x^3 \\ y_L(x) &= C_{0L} + C_{1L}x + C_{2L}x^2 + C_{3L}x^3, \end{aligned} \tag{4.13}$$

avec $y_R(x)$ et $y_L(x)$ les positions latérales des lignes à droite et à gauche de l'ego véhicule à une distance x devant celui-ci. C_{iR} et C_{iL} sont les coefficients d'ordre $i \in \{0, 1, 2, 3\}$ des deux polynômes.



(a) (b)

FIGURE 4.14 – Scénarios de changement de voie (a) et exemples d'enregistrements (b)

Pour les scénarios labellisés 4, 5, 6 et 7, correspondants à des changements de voie, les échantillons prélevés des enregistrements sont coupés juste avant que le véhicule ait complètement franchi la ligne. Comme le but est de détecter les changements de voie le plus tôt possible, il faut que les algorithmes soient capables de le faire avant le changement effectif. Il est à noter qu'il n'y a pas que des enregistrements sur des routes droites, mais aussi sur des routes courbées afin d'évaluer la capacité des algorithmes à faire la distinction entre un virage et un changement de voie. Quelques exemples d'échantillons utilisés dans cette étude sont tracés sur la Figure 4.14 (b). Les couleurs des échantillons respectent les couleurs des scénarios de la Figure 4.14 (a).

Pour avoir une comparaison juste entre les méthodes de détection d'intention proposées, les données utilisées en entrée sont identiques. Ainsi, pour chaque objet i autour de l'ego véhicule à l'instant k, ces données sont composées : des coefficients polynomiaux des lignes à droite et à gauche de l'ego véhicule, les 25 derniers échantillons des positions et vitesses longitudinale et latérale de l'objet i relatives à celles de l'ego, *i.e.*, $X_i(k)$, ..., $X_i(k-25)$, $Y_i(k)$, ..., $Y_i(k-25)$, $V_{xi}(k)$, ..., $V_{xi}(k-25)$ et $V_{yi}(k)$, ..., $V_{yi}(k-25)$. Sachant que l'enregistrement des capteurs se fait à une période de 40 ms, l'ensemble positions/vitesses correspond à un historique d'une seconde.

4.3.2 Détection d'intention : méthode à base de règles

La première méthode présentée pour la détection du changement de voie est une méthode à base de règles et de probabilités, utilisant une prédiction de trajectoire à base de modèle

d'évolution dynamique. Des règles, utilisant des probabilités de changement de voie suivant la vitesse latérale de l'objet et sa distance à la ligne, sont définies pour déterminer dans quel scénario se trouve l'objet. Ainsi, plus la vitesse transverse de l'exo véhicule est grande et sa distance à la ligne faible, plus la probabilité que le véhicule soit en train de changer de voie est grande, voir Figure 4.15.

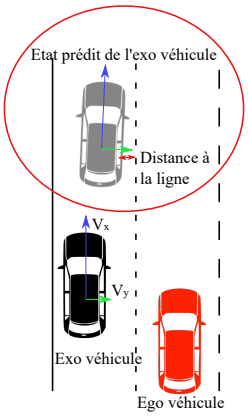


FIGURE 4.15 – Principe de fonctionnement de la méthode probabiliste

Filtrage de Kalman et prédiction

La vitesse latérale des objets estimée par la fusion de données des capteurs est bruitée et n'est pas toujours disponible pour tous les objets de l'environnement. Elle ne peut donc pas être utilisée telle quelle. La formule utilisant le delta de position entre les deux derniers échantillons pour calculer la vitesse latérale V_y :

$$V_{yi}(k) = \frac{Y_i(k) - Y_i(k-1)}{T_e}, \tag{4.14}$$

avec T_e la période d'échantillonnage, donne aussi un résultat très bruité. V_y est donc estimée en utilisant un filtrage de Kalman.

Le filtrage de Kalman, aussi utilisé dans la section 2.5 pour l'estimation de la vitesse de lacet, est capable d'estimer les états d'un système à partir d'une série de mesures et d'un modèle dynamique d'évolution. Comme les seules informations disponibles sur les autres véhicules sont leurs positions relatives à l'ego véhicule et leurs vitesses longitudinales, un modèle d'évolution simple est utilisé. Pour ce modèle, les vitesses longitudinales et latérales des exo véhicules sont considérées constantes de sorte que leurs trajectoires relatives à l'ego s'écrivent :

$$\begin{cases} X_i(k+1) = X_i(k) + T_s V_{xi}(k) - T_s V_{xego}(k) \\ V_{xi}(k+1) = V_{xi}(k) \\ Y_i(k+1) = Y_i(k) + T_s V_{yi}(k) \\ V_{yi}(k+1) = V_{yi}(k), \end{cases} \tag{4.15}$$

avec $X_i(k)$ et $Y_i(k)$ les positions longitudinale et latérale de l'objet i, relatives à celle de l'ego, à l'instant k. $V_{xi}(k)$ et $V_{yi}(k)$ sont les vitesses longitudinale et latérale de l'objet i à l'instant k.

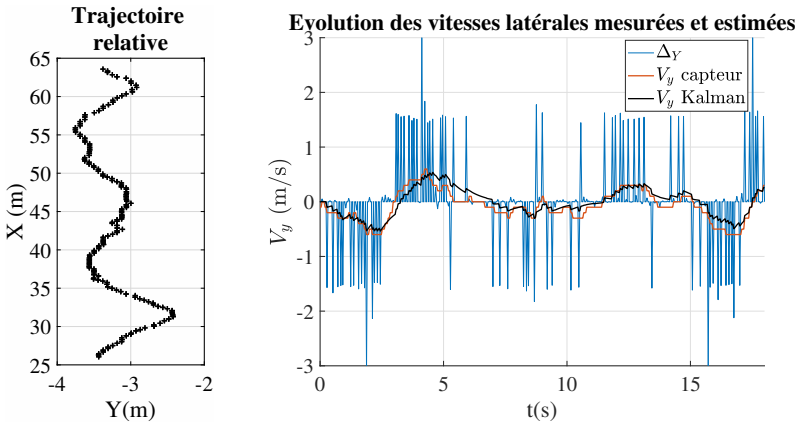
La représentation d'état du modèle défini ci-dessus peut s'écrire :

$$x(k+1) = A_{DI}x(k) + B_{DI}u(k), \tag{4.16}$$

avec le vecteur d'état $x(k) = [X_i(k)\ V_{xi}(k)\ Y_i(k)\ V_{yi}(k)]^T$, l'entrée $u(k) = V_{xego}(k)$ est la vitesse de l'ego véhicule et les matrices de transition d'état s'écrivent :

$$A_{DI} = \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } B_{DI} = \begin{bmatrix} -T_s \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{4.17}$$

En utilisant ce modèle d'évolution dynamique et les équations du filtrage de Kalman, comme définies à la section C.1, une meilleure estimation de la vitesse transverse des objets est obtenue. La Figure 4.16 (b) compare la vitesse estimée par les capteurs (en rouge) et celles calculées par le delta de position (en bleu) et filtrage de Kalman (en noir) pour un véhicule ayant effectué la trajectoire de la Figure 4.16 (a). Il peut être observé que le calcul « simple » de la vitesse latérale n'est pas du tout exploitable. La vitesse latérale donnée par la fusion des capteurs présente par moment des échelons importants de variation de vitesse, notamment entre 5 s et 10 s. Le filtrage permet une estimation plus fluide de la vitesse transverse.



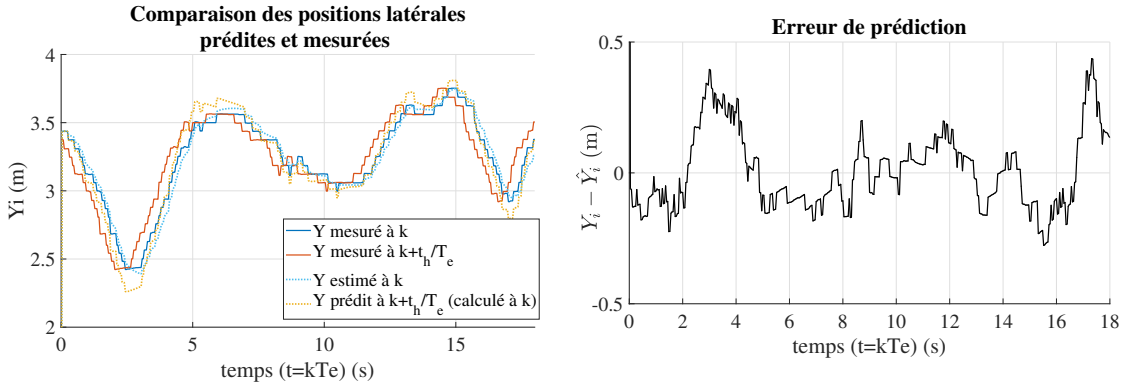
(a) (b)

FIGURE 4.16 – Trajectoire relative mesurée d'un exo véhicule (a) et comparaison de la mesure (rouge) et des estimations de sa vitesse latérale par (4.14) (bleu) et Kalman (noir) (b)

Afin de détecter le changement de voie le plus rapidement possible, une étape de prédiction de trajectoire est ajoutée. L'objectif de cette étape est de déterminer les futures positions des objets de l'environnement à un horizon $t_h = 15T_e = 0,6$ s. La définition de t_h est empirique, elle est choisie pour faire en sorte que la détection se fasse tôt avec le moins de fausse détection de changement de voie possible. En effet, si la vitesse latérale du véhicule est grande, la phase de prédiction peut potentiellement estimer que le véhicule va franchir la ligne de marquage à $t + t_h$ alors que le véhicule est dans un virage. À l'inverse, si t_h est trop faible, la détection va être trop tardive. La position estimée à $t + t_h$ est calculée par :

$$\hat{x}(k + t_h/T_s|k) = A^{t_h/T_e}\hat{x}(k|k). \tag{4.18}$$

Les positions latérales mesurées aux instants t et $t+t_h$ ainsi que leur estimation et prédiction par filtrage de Kalman pour le même exemple que précédemment ont été tracés sur la Figure 4.17 (a). La Figure 4.17 (b) trace ensuite l'erreur de prédiction à l'instant $t + t_h$. En moyenne, la prédiction présente une erreur de 12 cm en position latérale.



(a) (b)

FIGURE 4.17 – Positions latérales estimées et prédites à l'instant t pour l'horizon t_h (a) et erreur de position à l'horizon t_h (b)

Règles de décision

Une fois la vitesse latérale correctement prédite, les règles probabilistes liées à cette dernière et à la distance à la ligne sont ensuite mises en œuvre pour détecter s'il y a un changement de voie. Ainsi, deux probabilités sont définies et utilisées par les règles de détection des changements de voie. Ces deux probabilités, s'écrivant $P(V_y)$ et $P(\Delta_y)$, indiquent respectivement la probabilité de l'objet à changer de voie en fonction de sa vitesse latérale et en fonction de sa position à la ligne la plus proche à l'horizon de prédiction $t + t_h$.

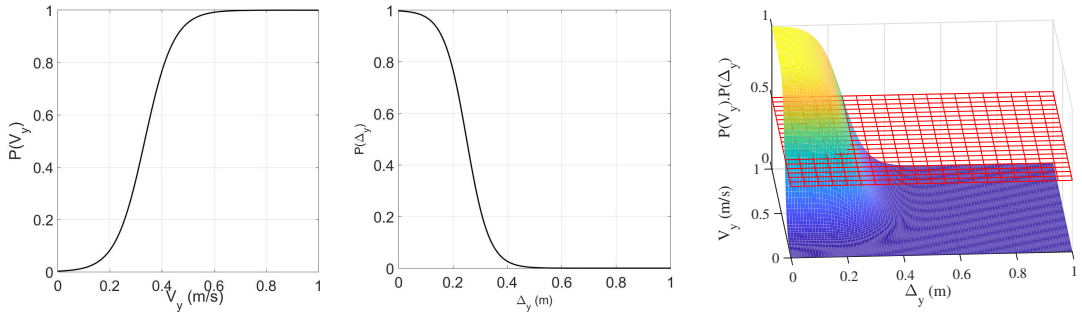
Elles sont calculées en utilisant des fonctions sigmoïdes :

$$P(\chi) = \frac{1}{1 + \mathrm{e}^{-\beta(\chi-\alpha)}}. \tag{4.19}$$

La Table 4.2 récapitule les valeurs de β et α pour les probabilités et la Figure 4.18 montre les courbes des probabilités. Les paramètres des fonctions ont été choisis après analyse des enregistrements. Il a été observé que les changements de voie pouvaient se produire avec des vitesses transverses inférieures à $V_y = 0,4$ m.s^{-1} et que les véhicules pouvaient rouler assez proche des lignes de marquage. Un compromis est fait dans le but de détecter les changements de voie assez tôt tout en évitant les fausses détections.

TABLE 4.2 – Valeurs des paramètres des fonctions de probabilité

$\boldsymbol{P}$	$\boldsymbol{\beta}$	$\boldsymbol{\alpha}$
$P(V_y)$	18	0,33
$P(\Delta_y)$	-24	0,25



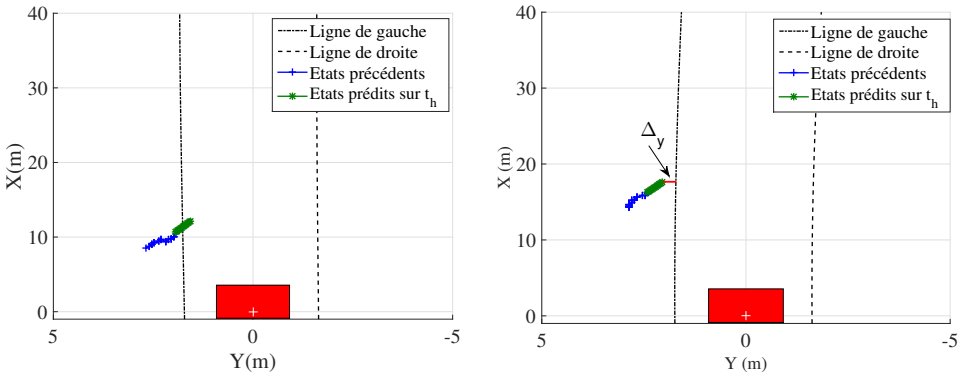
(a) (b) (c)

FIGURE 4.18 – Probabilités de changement de voie en fonction de la vitesse latérale (a), en fonction de la distance à la ligne la plus proche à t_h (b) et combinaison des deux probabilités (c)

Les règles utilisées pour déterminer les intentions des objets sont les suivantes :

Si Entre k et $k + t_h/T_e$ l'objet a franchi une ligne **alors**
  Le véhicule change de voie
Sinon
  Calcul des probabilités $P(V_y)$ et $P(\Delta_y)$
  Calcul de $P(V_y \cap \Delta_y) = P(V_y)P(\Delta_y)$
  Si $P(V_y \cap \Delta_y) > 0,5$ **alors**
    L'objet change de voie
  Sinon
    L'objet reste dans sa voie
  Fin Si
Fin Si

Deux exemples sont donnés afin d'illustrer la méthode.



(a) (b)

FIGURE 4.19 – Exemples avec (a) et sans (b) prédiction de franchissement de voie

La Figure 4.19 (a) illustre un premier exemple. Dans cette situation, l'objet est à gauche de l'ego véhicule. Ce dernier est représenté par le rectangle rouge. L'historique des positions en bleu suggère que l'objet se déplace vers la voie de l'ego véhicule. Les futurs états prédis, en vert sur la figure, montrent qu'à $t + t_h$ le véhicule est supposé avoir changé de voie et ainsi être dans la même voie que celle de l'ego. Un changement de voie est donc détecté.

Un deuxième exemple, voir Figure 4.19 (b), montre un scénario où la prédiction estime que le véhicule sera toujours dans sa voie à $t+t_h$. Les deux probabilités sont calculées. Elles donnent $P(V_y) = 0,9973$ et $P(\Delta_y) = 0,02676$. L'intersection donne $P(V_y \cap \Delta_y) = 0,0275$ et la méthode indique un maintien dans la voie. Dans cet exemple, le véhicule va effectivement changer de voie, mais est encore trop loin de la ligne pour que le changement soit détecté, $P(\Delta_y)$ est trop faible. À l'instant $t+t_h$, le changement de voie ne sera donc pas effectif selon nos règles. Plus d'échantillons sont nécessaires pour que la détection se fasse. En effet, à $t + 0,36$ s, soit 9 échantillons plus tard, $P(\Delta_y) = 0,739$ et $P(V_y) = 0,998$, signifiant que le changement de voie est détecté. La détection de changement de voie est donc plus tardive.

En reprenant l'enregistrement complet, les Figures 4.20 tracent l'évolution des probabilités autour de l'échantillon $k = 0$, correspondant à l'exemple de la base de données, et de la position réelle du véhicule par rapport à la ligne de gauche. Ainsi, cinq échantillons avant l'exemple présenté, la probabilité de changement de voie en fonction de la vitesse latérale $P(V_y) \approx 1$. Celle en fonction de la distance à la ligne commence à augmenter quatre échantillons plus tard, pour une détection effective au bout de neuf échantillons. Ainsi, le changement de voie est détecté plus tard, alors que le véhicule se trouve à une distance de 0,5 m de la ligne.

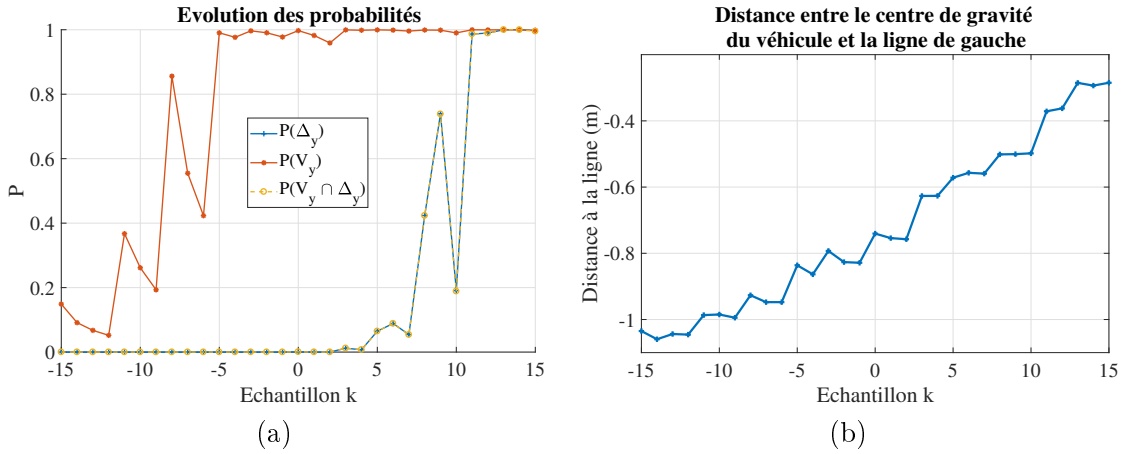


FIGURE 4.20 – Évolutions des probabilités (a) et de la distance entre le véhicule et la ligne (b) autour de l'exemple

4.3.3 Méthode par réseau de neurones

La deuxième méthode de détection de changement de voie utilise des réseaux de neurones. Il s'agit maintenant de construire et d'entraîner un réseau de neurones pour la classification d'intention des exo véhicules tel que défini au paragraphe 4.3.1.

Choix de la structure du réseau de neurones pour la détection d'intention de changement de voie

Un réseau de neurones est composé de trois couches : une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Dans notre cas, et comme indiqué auparavant, les paramètres de la couche d'entrée sont : l'historique des positions et vitesses sur 1 s échantillonnée à 40 ms et les coefficients des polynômes des lignes pour un total de 112 paramètres. La couche cachée du réseau de neurones utilisé ici contient une seule couche de trente neurones. Enfin, parce que les intentions sont définies en sept classes, il y a sept sorties sur la dernière couche du

réseau. Un diagramme Figure 4.21 (a) illustre la structure du réseau. Il n'y a pas de méthode pour déterminer de façon certaine le nombre de couche cachée et de neurones nécessaires pour qu'un réseau de neurones soit représentatif du système à modéliser. Il faut gérer le compromis entre complexité du réseau et précision. Une analyse sur l'influence du nombre de neurones dans notre cas applicatif est présentée par la suite.

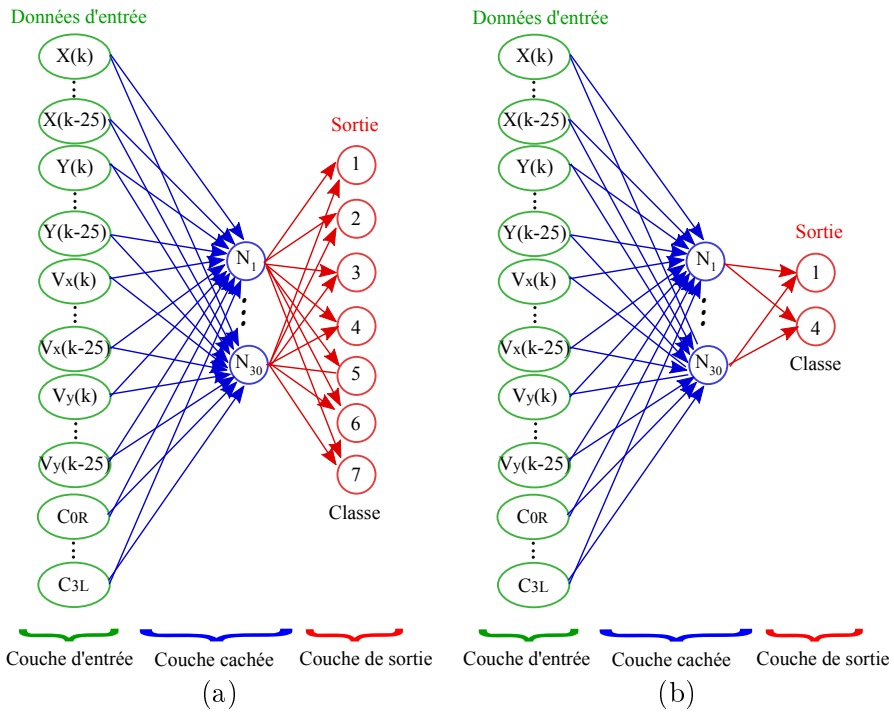


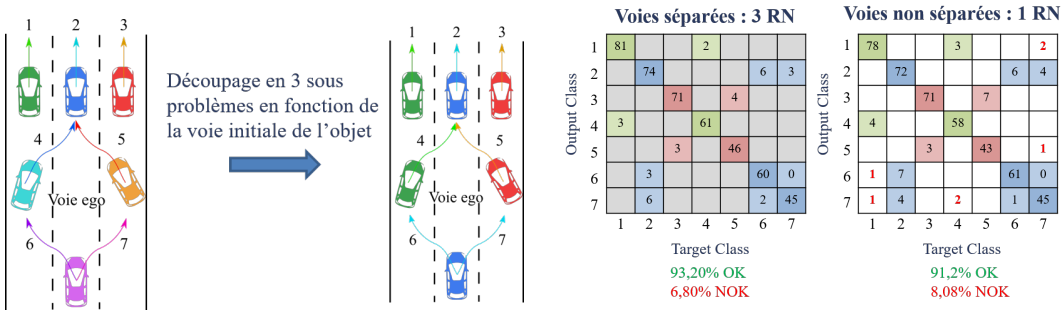
FIGURE 4.21 – Structure des réseaux de neurones

Les matrices des paramètres s'écrivent $\Theta^{(1)}$ pour les paramètres de transition entre la couche d'entrée et la couche cachée, et $\Theta^{(2)}$ pour les paramètres entre la couche cachée et la couche de sortie. La fonction d'activation de chaque neurone est la fonction sigmoïde g (4.6).

À partir d'un vecteur d'entrée $d = [X(k) \ldots C_{3L}]^T$, de la fonction d'activation g et des poids Θ_i ($i \in \{1, 2\}$), les équations de type (4.5) sont utilisées pour calculer la sortie c. c est un vecteur 7x1, chaque ligne correspondant à une classe. La valeur des sorties se trouve entre 0 et 1. La classe correspondant à l'entrée d est celle avec la valeur la plus proche de 1.

Le réseau, ainsi défini, n'a pas la connaissance a priori du positionnement de l'exo véhicule dans les voies. Il peut alors apparaître des résultats aberrants. Par exemple, le réseau peut prédire qu'un exo véhicule se déplace de la voie de gauche vers la voie centrale (classe 4) alors qu'en réalité, il reste dans la voie centrale du début à la fin. Le réseau n'ayant pas d'information de positionnement des objets ou de présence des voies, il peut se tromper. Vu que l'information sur la ligne de départ est connue, ces aberrations peuvent être évitées en ajoutant cette information dans le vecteur d'entrée d, ou complètement supprimées en créant trois sous réseaux : un pour chaque voie de circulation de départ des exos véhicules, voir Figure 4.22 (a).

Pour évaluer l'importance d'une séparation du problème, nous avons entraîné un réseau complet avec les sept classes en sortie et trois sous-réseaux en fonction de la voie de départ des objets. Les matrices de confusion montrant les erreurs des réseaux sont visibles sur la Figure 4.22 (b).



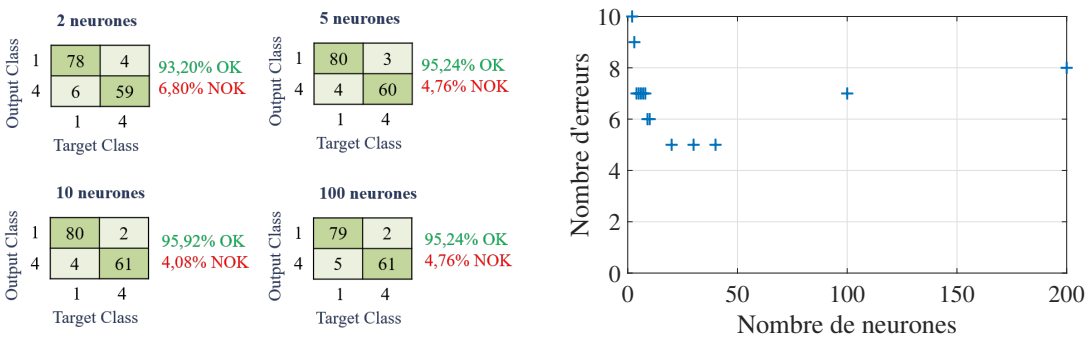
(a) (b)

FIGURE 4.22 – Illustration du découpage du problème (a) et comparaison des matrices de confusions pour les méthodes avec et sans voies séparées (b)

Sur les 470 données, la méthode avec trois réseaux de neurones découplés en fonction de la voie initiale de l'exo véhicule se trompe 32 fois. En utilisant un seul réseau, il y a 46 erreurs, dont 7 qui sont aberrantes, et supprimées avec la technique de séparation de problèmes.

La deuxième option, avec l'entraînement de trois réseaux, est préférée. Chacun de ces réseaux a la même structure de couche d'entrée et cachée. Les réseaux pour les voies de gauche et droite ont deux sorties alors que le réseau de la voie du milieu en a trois. Étant donné que la méthodologie et les résultats sont similaires, le reste de l'étude se focalise sur la voie de gauche. La représentation schématique du réseau pour la voie de gauche est illustrée Figure 4.21 (b). Seules les classes 1 (reste dans la voie de gauche) et 4 (se déplace de la voie de gauche vers la voie du milieu) avec des positions initiales dans la voie de gauche restent.

Pour arriver au nombre de 30 neurones pour la couche cachée, plusieurs essais avec la même base de données d'entraînement ont été effectués pour vérifier quel était le nombre optimal de neurones à conserver. Le but est d'avoir le moins de paramètres possible tout en conservant une bonne modélisation des relations entrée/sortie. Ainsi, la Figure 4.23 (a) montre les matrices de confusion pour les réseaux avec 2, 5, 10 et 100 neurones et la Figure 4.23 (b) trace la variation du nombre d'erreurs en fonction du nombre de neurones dans le réseau. Sur tous les tests, le nombre d'erreurs varie entre 5 et 10 sur un total de 147 données. Lorsque l'on augmente le nombre de neurones entre 2 et 20, le nombre d'erreurs diminue. Puis un palier est atteint entre 20 et 40 neurones et le nombre d'erreurs augmente lorsque trop de neurones sont ajoutés.



(a) (b)

FIGURE 4.23 – Matrices de confusion en fonction du nombre de neurones (a) et variation du nombre d'erreurs en fonction du nombre de neurones dans le réseau (b)

Comme expliqué dans la section 4.3.2 de la méthode par règles, la vitesse latérale peut être calculée de deux manières : en utilisant la relation (4.14) ou en utilisant un filtrage de Kalman. Afin d'analyser l'influence de la vitesse latérale sur le réseau de neurones, deux réseaux sont entraînés et comparés : un avec V_y comme dans (4.14) et un autre avec V_y estimée par filtrage de Kalman.

Entraînement

La phase d'apprentissage des paramètres Θ_1 et Θ_2 du réseau de neurones pour la détection d'intention utilise la méthode de rétropropagation décrite précédemment. La fonction de coût $J(\Theta_1, \Theta_2)$ à minimiser est définie comme :

$$J(\Theta_1, \Theta_2) = \frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{2} \Big(c^{(i)}[k] \log(\widehat{c}^{(i)}[k]) + (1 - c^{(i)}[k]) \log(1 - \widehat{c}^{(i)}[k]) \Big), \tag{4.20}$$

avec m le nombre d'exemples dans la base de données, $c^{(i)}[k]$ la k^{eme} ligne de la sortie réelle correspondant au i^{eme} vecteur d'entrée d de la base de données et $\widehat{c}^{(i)}[k]$ la k^{eme} ligne de la sortie calculée par le réseau avec le même vecteur d'entrée d, Θ_1 et Θ_2.

Pour que l'apprentissage soit efficace, il convient aussi de faire en sorte que toutes les caractéristiques des vecteurs d'entrée soient du même ordre. Une normalisation est donc effectuée pour faire en sorte que toutes les valeurs d'entrée soient comprises entre -3 et 3. Chaque caractéristique F est normalisée par l'équation :

$$F' = 3 \frac{F - \text{mean}(F)}{\max(F) - \min(F)}. \tag{4.21}$$

La base de données est divisée en deux : une base d'entraînement et une base de test. Un terme de régularisation γ est ajouté à la fonction de coût (4.20). La valeur de γ est déterminée de telle sorte qu'elle minimise la fonction de coût (4.20) de la base de test. Les réseaux sont entraînés avec plusieurs valeurs de γ avec la fonction de coût (4.12), puis (4.20) est calculé avec les bases d'entraînement et de test. La Figure 4.24 montre les variations de la fonction de coût des réseaux entraînés pour les deux bases en fonction de γ.

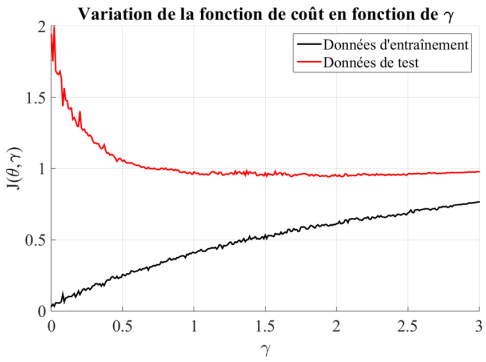


FIGURE 4.24 – Variation du coût en fonction du paramètre de régularisation pour le réseau de neurones avec V_y calculée comme une variation de positions

Sur cette figure, quand γ augmente, le coût des données d'entraînement augmente. Cela veut dire que le réseau devient moins spécifique à ces données. Dans le même temps, le coût de la base

de test commence à diminuer lorsque γ est petit puis augmente légèrement. Au début, le réseau est donc surentraîné et trop proche de la base d'entraînement. Il devient ensuite plus généralisé aux différents enregistrements. Lorsque γ devient trop grand, le réseau n'est plus représentatif du changement de voie, il manque des paramètres. Pour le réseau avec V_y calculée avec (4.14), la valeur de γ minimisant la fonction de coût pour la base de données de test est $\gamma = 1,94$. Celle du réseau avec V_y estimé par filtrage de Kalman est $\gamma = 0,85$.

4.3.4 Résultats

La méthode à base de règles et les deux réseaux de neurones sont comparés dans cette section.

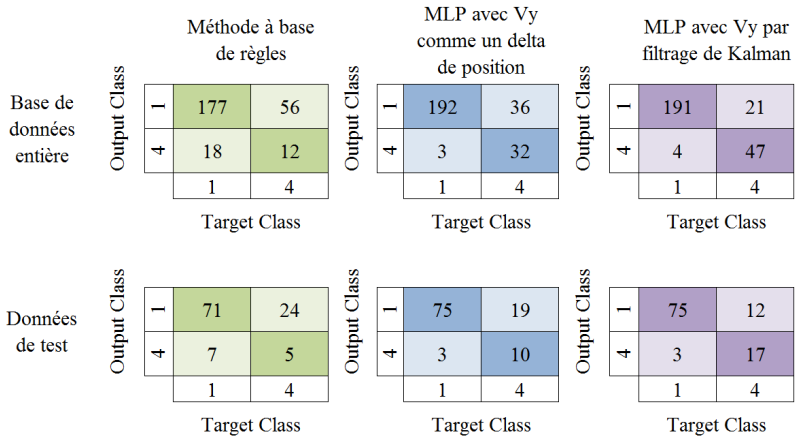


FIGURE 4.25 – Matrices de confusion des trois méthodes pour l'ensemble de la base de données et la base de tests

La Figure 4.25 montre les matrices de confusion de toute la base de données et de la base de données de test pour les trois méthodes. Ces matrices représentent le nombre de fois où les méthodes arrivent ou non à classer correctement les scénarios en fonction du vecteur d'entrée. Par exemple, pour la méthode à base de règles sur toute la base de données, 177 scénarios de classe 1 (maintien dans la voie) sont détectés correctement, mais 18 scénarios de classe 1 sont identifiés comme étant de la classe 4 (changement de voie). Dans ces tableaux, la majorité des erreurs surviennent quand un changement de voie n'est pas détecté : la sortie est 1 au lieu de 4. Peu importe la méthode, il en ressort la même tendance, à savoir que les solutions ont plus tendance à ne pas détecter les changements de voie plutôt que faire de fausses détections. On peut aussi noter que le fait de filtrer la vitesse latérale permet de limiter le nombre d'erreurs.

Cependant, pour la méthode à base de règles, comme indiqué précédemment dans les exemples de fonctionnement de la méthode, plusieurs échantillons peuvent être nécessaires pour classifier la situation comme un changement de voie et non pas un suivi de voie. Sur les cas de changement de voie où la méthode par règles classe la situation en catégorie 1 au lieu de la catégorie 4, dans un enregistrement en temps réel, le changement de voie aurait pu être détecté avec 3 ou 4 échantillons en plus. Cela veut aussi dire que la méthode par réseau de neurones est potentiellement plus rapide à la détection des changements de voie que la méthode probabiliste à base de règles.

Il existe plusieurs méthodes pour décrire et mesurer l'efficacité d'un classifieur [167]. Ici, la précision, le rappel et la F-mesure sont utilisés. La précision représente la proportion de solutions trouvées qui sont pertinentes, c'est à dire la proportion de bonnes détections par rapport aux

fausses détections, et le rappel correspond à la proportion des solutions pertinentes effectivement trouvées, c'est à dire la proportion de bonnes détections par rapport aux non-détections. La F-mesure est une moyenne harmonique de la précision et du rappel.

TABLE 4.3 – Précision, rappel et F-mesure de chaque méthode

	Classe 1			**Classe 4**		
	Précision	Rappel	F-mesure	Précision	Rappel	F-mesure
Méthode à base de règles	75%	91%	82%	42%	17%	24%
Réseau de neurones 1	78%	96%	86%	77%	34%	47%
Réseau de neurones 2	86%	96%	91%	85%	59%	70%

Pour une classe i, la précision de la classe est calculée par la relation :

$$\text{precision}_i = \frac{t_{pi}}{t_{pi} + f_{pi}}, \tag{4.22}$$

le rappel par :

$$\text{rappel}_i = \frac{t_{pi}}{t_{pi} + f_{ni}}, \tag{4.23}$$

et la F-mesure par :

$$\text{F} - \text{mesure}_i = 2\frac{\text{precision}_i * \text{rappel}_i}{\text{precision}_i + \text{rappel}_i}, \tag{4.24}$$

avec t_{pi} le nombre de vrais positifs (sortie du classificateur et vraie sortie dans la classe i), f_{pi} le nombre de faux positifs (sortie du classificateur dans la classe i, mais pas la vraie sortie : fausse détection) et f_{ni} est le nombre de faux négatif (vraie sortie dans la classe i, mais pas la sortie du classificateur : pas de détection).

La Table 4.3 présente les valeurs de ces indicateurs de mesure pour évaluer la pertinence des méthodes de classification proposées. « Réseau de neurones 1 » et « Réseau de neurones 2 » sont respectivement les réseaux de neurones avec V_y calculée comme un delta de position et V_y estimée par filtrage de Kalman. Globalement, le deuxième réseau de neurones présente les meilleurs résultats en termes de précision, rappel et F-mesure. Toutes les méthodes sont moins efficaces pour classer les changements de voie (classe 4) que les maintiens dans la voie (classe 1). En effet, en analysant les rappels, alors que les méthodes détectent toutes plus de 90% des situations en classe 1, seulement 17%, 34% et 59% des situations de changement de voie sont effectivement détectées pour la méthode par règles, le réseau de neurones 1 et le réseau de neurones 2. Les précisions indiquent que pour les deux réseaux de neurones, la même fraction de solutions pertinentes est trouvée pour chaque classe. Pour la méthode à base de règles, la précision est meilleure avec la classe 1 qu'avec la classe 4.

La vitesse latérale V_y a un impact important sur l'entraînement et les résultats du réseau de neurones. Le fait d'avoir une vitesse latérale plus représentative de la réalité aide à classifier correctement les scénarios de changement de voie.

4.4 Prédiction de trajectoire : le cas du changement de voie

4.4.1 Description du problème

L'objectif est maintenant de prédire la trajectoire future des véhicules sur autoroute à un horizon de 1 s. Pour cela, nous allons utiliser un réseau de neurones récurrent appelé LSTM (Long-Short Term Memory), et analyser quelles caractéristiques en entrée sont nécessaires pour une bonne prédiction des positions futures des véhicules. La structure du LSTM peut aussi influencer les résultats et sera donc aussi analysée.

Les données utilisées pour l'entraînement et le test du réseau sont identiques à celles utilisées pour la détection d'intention. Dans un premier temps, uniquement les précédentes positions longitudinales et latérales de la dernière seconde sont utilisées en entrée. Ensuite, les vitesses longitudinales et latérales sont ajoutées pour évaluer leurs influences sur la qualité de la prédiction. Enfin, un deuxième réseau LSTM est ajouté en série en premier pour voir si une structure différente améliore la prédiction. Pour une comparaison équivalente, les données d'entraînement et de test sont identiques à chaque fois.

4.4.2 Le réseau Long-Short Term Memory

Un réseau Long-Short Term Memory est un réseau de neurones récurrent. La structure du réseau n'est plus linéaire, comme les réseaux utilisés précédemment, mais présente des récurrences, des cycles, entre les différents neurones. Les réseaux de neurones récurrents sont utiles pour l'analyse de séquences temporelles et sont donc très populaires pour la reconnaissance vocale ou la traduction automatique. Le fait d'avoir des cycles dans le réseau permet à ce dernier de mémoriser les éléments passés. Le problème d'une telle méthode est la disparition du gradient lors de l'apprentissage par rétropropagation, les paramètres du réseau ne sont alors plus modifiés et mal entraînés.

L'intérêt du LSTM est de pallier cette disparition de gradient. Pour cela, chaque unité du LSTM contient un paramètre lié à sa mémoire. Une unité d'un LSTM peut se schématiser comme sur la Figure 4.26, où d_k représente le vecteur d'entrée de l'unité k, $\hat{c}_k$ le vecteur de sortie, mem_k la mémoire et hid_k l'état caché. Le fonctionnement d'une unité peut se décomposer en trois phases indiquées sur la même figure. La première correspond à la phase d'oubli qui permet une remise à zéro de l'état de mémoire si nécessaire. La deuxième phase est celle de la mise à jour, ou non, de la cellule de mémoire. La dernière est celle de la génération de la sortie. Les équations d'une cellule LSTM sont les suivantes :

$$\begin{aligned}
F_k &= \mathrm{sigm}\left(W_F d_k + U_F hid_{k-1} + b_F\right) \\
I_k &= \mathrm{sigm}\left(W_I d_k + U_I hid_{k-1} + b_I\right) \\
O_k &= \mathrm{sigm}\left(W_O d_k + U_O hid_{k-1} + b_O\right) \\
mem_k &= F_k \circ mem_{k-1} + I_k \circ \tanh\left(W_c d_k + U_c hid_{k-1} + b_c\right) \\
hid_k &= O_k \circ \tanh\left(mem_k\right),
\end{aligned} \tag{4.25}$$

avec F_k le vecteur d'oubli, I_k le vecteur de mise à jour, O_k le vecteur d'envoi de la sortie et W, U et b les matrices et vecteurs de paramètre du réseau.

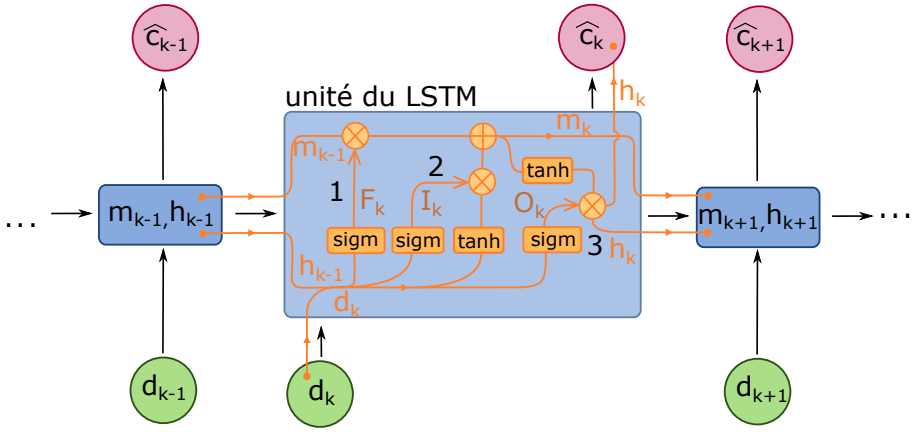


FIGURE 4.26 – Schématisation du fonctionnement d'une unité LSTM [Source : adaptée de https://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_r%C3%A9currents]

En fonction de la séquence en entrée et celle en sortie désirée, le positionnement, et donc la structure du réseau LSTM change. Les principales structures de LSTM sont représentées à la Figure 4.27.

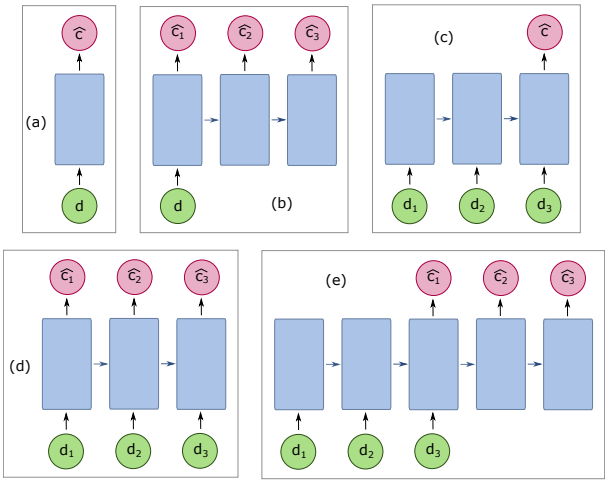


FIGURE 4.27 – Différentes structures de LSTM : (a) « one to one », (b), « one to many », (c) « many to one », (d) « many to many » synchronisé, (e) « many to many » désynchronisé

Sur cette figure, la structure (a) correspond à un réseau classique pour la classification d'image par exemple. La structure (b) avec un vecteur d'entrée et plusieurs vecteurs de sortie peut être utilisée pour la description d'une image par une phrase, avec un mot par sortie. La structure (c) a quant à elle plusieurs entrées, mais une seule sortie. Elle peut donc être utilisée pour la description de phrases, avec un mot par entrée. La sortie pourrait correspondre à vrai ou faux si le but est de répondre à une question par exemple. Les deux dernières structures ont plusieurs entrées et plusieurs sorties. La forme (d) de LSTM où les entrées et sorties sont synchronisées est adaptée pour la classification d'images de vidéo. Enfin, la dernière structure (e) où les entrées et sorties sont désynchronisées est utilisée pour la traduction de phrases. C'est ce type de LSTM qui est pertinent dans notre cas d'étude. En effet, l'objectif est de déterminer, à partir d'un historique de données, les données suivantes.

4.4.3 Structures de LSTM utilisées

Le but de cette section est la prédiction de trajectoire des véhicules de l'environnement. À l'instant k, la sortie est un vecteur $\hat{c}$ des états estimés aux instants $k+1$ jusqu'à $k+25$, avec un échantillonnage de 40 ms. Ce vecteur de sortie contient les futures positions longitudinales X et latérales Y de l'objet dans le repère absolu. L'entrée, est un vecteur d contenant les précédents états du véhicule sur 1 s, soit entre $k-25$ et k.

Trois structures de LSTM sont comparées. Elles sont illustrées sur la Figure 4.28. La sortie est identique pour chacune d'entre elles. La première a pour entrée les positions longitudinales et latérales. Le modèle d'apprentissage est lui constitué d'un LSTM suivi d'une couche cachée complètement connectée de type MLP. Le LSTM dispose de 40 unités car le fait de rajouter des unités ne permet pas une meilleure modélisation dans notre cas. La couche cachée supplémentaire permet à la fois d'avoir une meilleure modélisation, mais aussi de récupérer le bon nombre de sorties.

Par rapport à la première méthode, la deuxième structure n'a plus les mêmes caractéristiques en entrée. Les vitesses longitudinales V_x et latérales V_y, calculées par (4.14), ont été rajoutées pour estimer leurs impacts sur la prédiction de trajectoire.

Enfin, la troisième structure possède la même entrée que la seconde, mais un LSTM a été rajouté. L'idée ici est de rajouter de la profondeur au modèle pour le complexifier et voir s'il est ainsi plus proche du modèle réel.

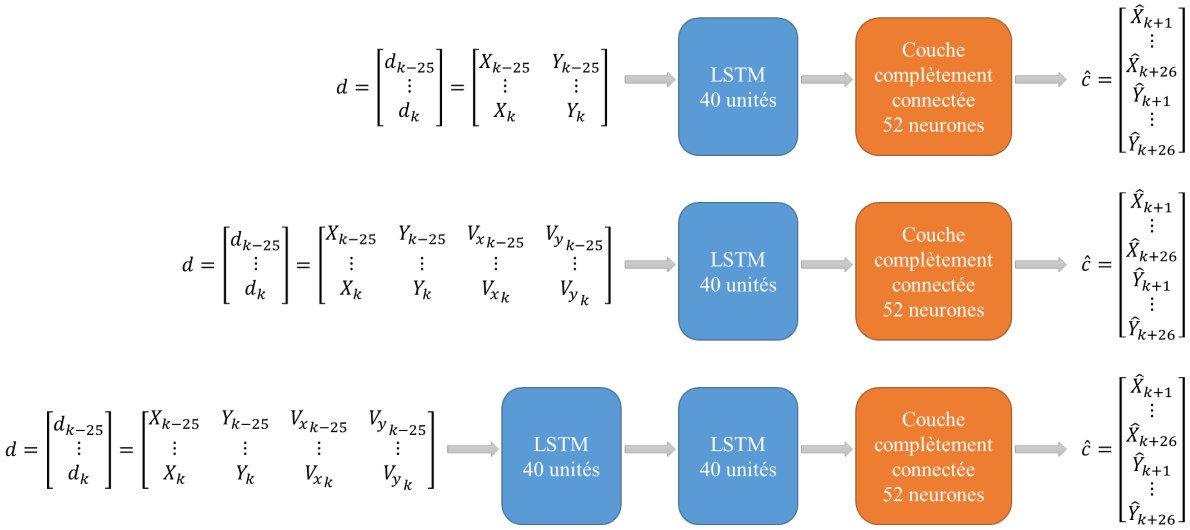
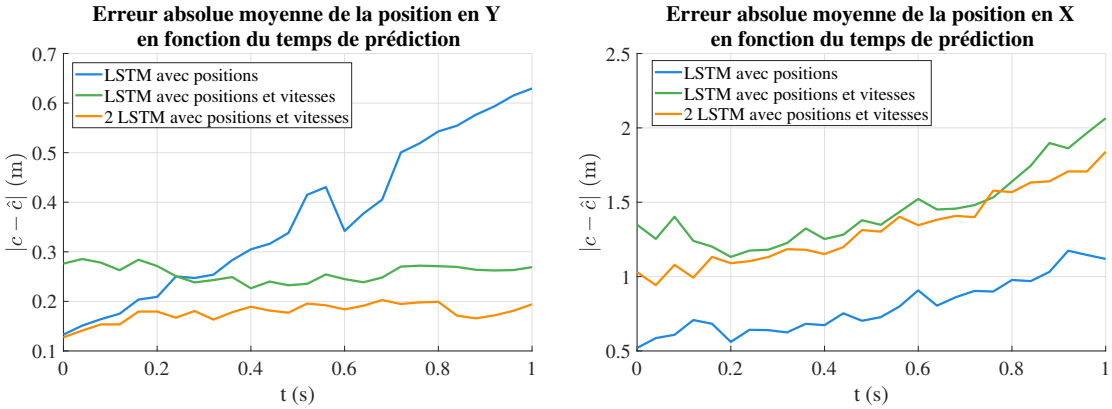


FIGURE 4.28 – Différentes structures de LSTM utilisées pour la prédiction de trajectoire

Pour avoir une comparaison équitable, les modèles sont tous entraînés avec la même base de données. Leurs performances sont donc aussi comparées avec les mêmes enregistrements.

4.4.4 Résultats

Les performances des modèles sur la prédiction de trajectoire des exo véhicules sont évaluées par leurs erreurs de positions absolues. La Figure 4.29 montre l'évolution de ces erreurs en fonction du temps de prédiction.



(a) (b)

FIGURE 4.29 – Moyennes des erreurs absolues en position longitudinale (a) et latérale (b) en fonction de l'horizon de prédiction

La Figure 4.29 (a) montre que le fait d'ajouter les vitesses en entrée améliore considérablement la précision de la prédiction latérale à long terme. En effet, à 1 s, l'erreur de position latérale est en moyenne de 0,6 m pour le modèle sans les vitesses et de 0,27 m pour celui avec. Par contre, le fait d'ajouter les vitesses réduit la précision à cout terme, pour des horizons inférieurs à 0,25 s. Le fait d'ajouter une deuxième LSTM permet une meilleure précision que les deux autres modèles sur tout l'intervalle de temps prédit, avec une erreur moyenne latérale de 20 cm à 1 s.

Paradoxalement, le fait d'ajouter les vitesses en entrée améliore la prédiction de la position latérale, mais dégrade celle de la position longitudinale, voir Figure 4.29 (b). Pour tous les modèles, l'erreur de prédiction de la position longitudinale augmente en fonction de l'horizon de prédiction. À 1 s, l'erreur en position longitudinale est de 1.12 m pour le modèle avec uniquement les positions en entrée, 1,83 m pour le modèle avec les positions et les vitesses en entrée et 2,06 m pour celui avec deux LSTM en série.

Des exemples de bonnes et mauvaises prédictions de trajectoires sont tracés sur les Figures 4.30 (a) et (b) pour des données de la base de test. À noter, ces exemples sont tracés sur la même figure, mais n'ont pas eu lieu en même temps. Il n'y a donc aucune corrélation entre les véhicules tracés sur ces figures.

Dans les cas où la prédiction se passe bien pour tous les modèles, Figure 4.30 (a), le véhicule a très peu de dynamique latérale et un comportement assez linéaire. À l'inverse, au moins un des modèle fait une mauvaise prédiction lorsque le véhicule tourne ou change de voie, Figure 4.30 (b). Les exemples où $X > 0$ m montrent bien les limitations du LSTM utilisant uniquement les positions en entrée. On retrouve bien les résultats précédents sur l'erreur de prédiction latérale qui augmente en fonction du temps de prédiction pour ce modèle. Les deux autres modèles, avec les indications sur la vitesse, arrivent beaucoup mieux à gérer les phases impliquant une dynamique latérale.

Sur deux exemples, le deuxième modèle avec positions et vitesses en entrées du LSTM, la sortie présente un offset latéral conduisant à une mauvaise prédiction. Sur l'exemple où $X \approx -50$ m et $Y \approx 3$ m, les deux LSTM avec la vitesse en entrée conduisent à une moins bonne prédiction que le premier LSTM sans vitesse. Leurs prédictions sont désordonnées et ne ressemblent pas à une trajectoire. Ceci est probablement dû au fait que l'entrée n'est pas très stable.

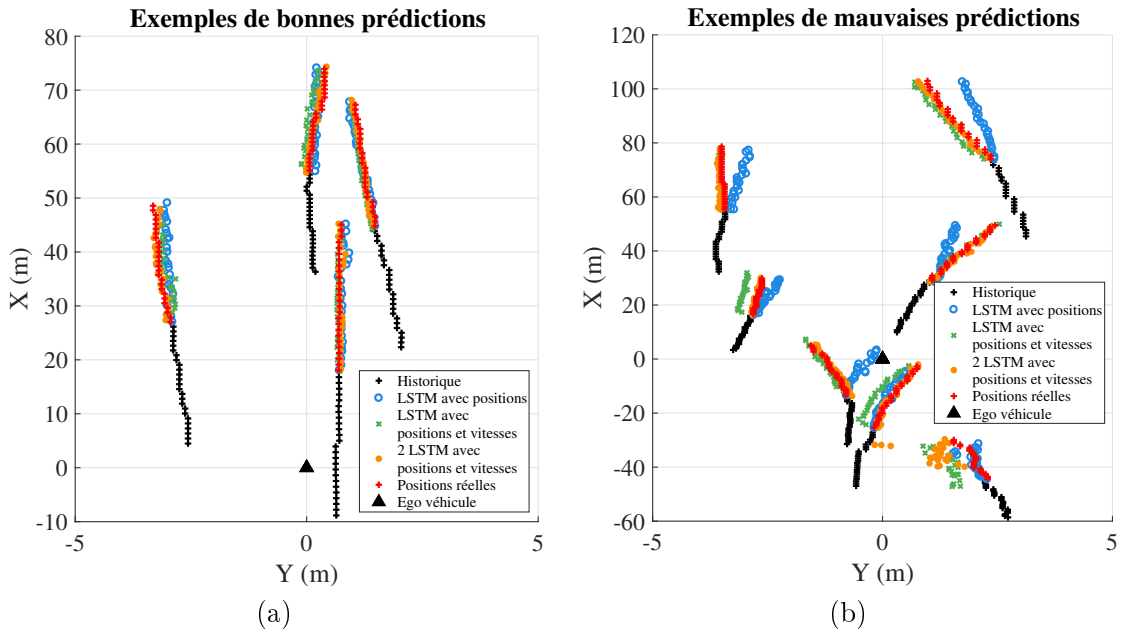


FIGURE 4.30 – Exemples de bonnes (a) et de mauvaises (b) prédictions de trajectoire pour les différents modèles

Les résultats de prédiction de trajectoire avec un réseau d'apprentissage profond de type LSTM présentent de bons résultats, même avec une base de données relativement petite. On pourrait donc s'attendre à une meilleure prédiction avec de nouveaux exemples ajoutés à cette dernière. Contrairement au réseau de neurones pour la prédiction d'intention, la prédiction de trajectoire n'a pas encore été implantée dans le véhicule TJC et doit être validée en temps réel.

À terme, le but est la prédiction de trajectoire en environnement urbain. Lors de cette étude, il n'y avait pas suffisamment de données de roulage pour faire une étude sur la prédiction de trajectoire en intersection ou en rond-point. Cependant, il est déjà possible de reconstituer l'environnement à partir des capteurs présents sur la voiture, d'analyser ce que voient les capteurs dans de telles situations et ainsi de choisir la stratégie adaptée pour la prédiction de trajectoire.

4.4.5 Perspectives pour la prédiction de trajectoire en environnement urbain

Nous avons roulé avec HZTR plusieurs fois autour d'un rond-point, afin d'évaluer si les stratégies de prédiction d'intention et de prédiction de trajectoire peuvent s'appliquer dans ce cas. Des exemples de cet enregistrement sont tracés Figure 4.31.

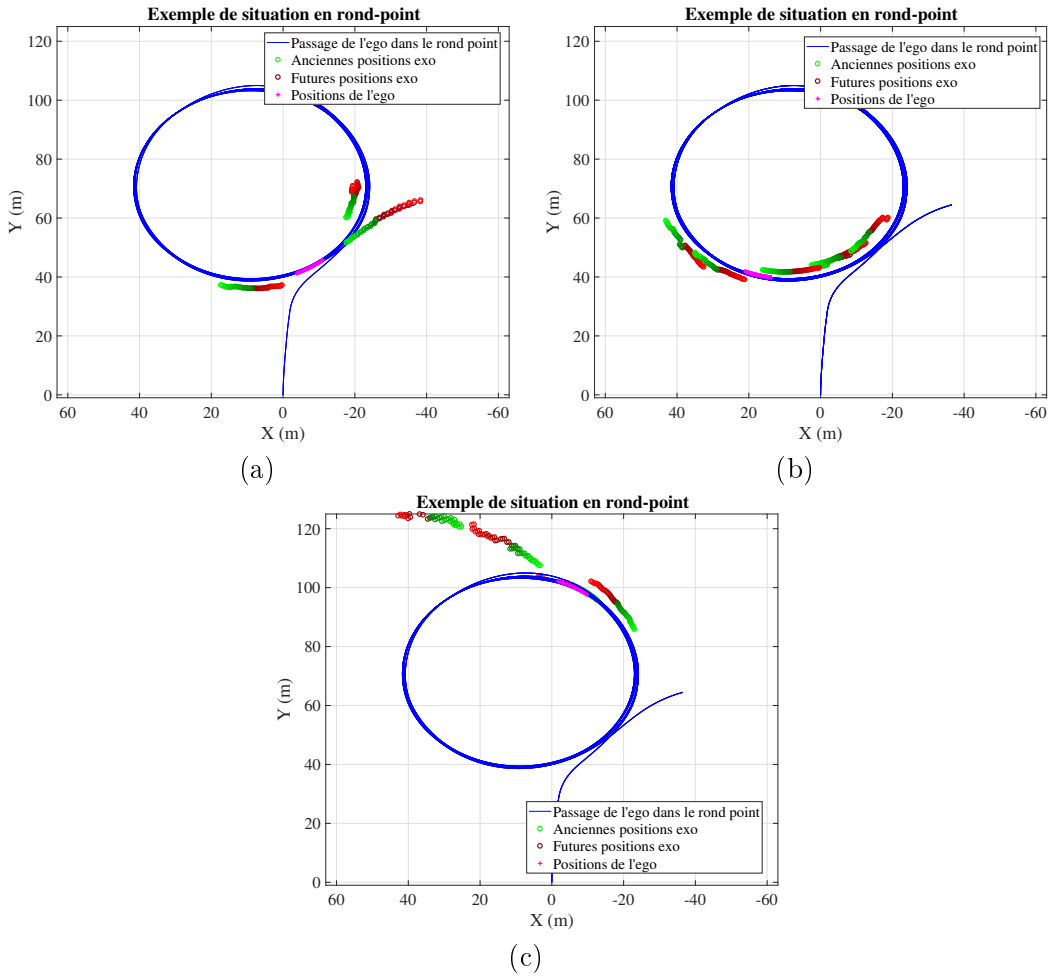


FIGURE 4.31 – Enregistrements dans un rond-point

Sur ces figures, la position de l'ego véhicule est en rose et la trace GPS du chemin effectué pendant l'enregistrement est en bleu. À l'instant où est enregistrée la figure, les trajectoires précédentes des exo véhicules sont tracées en vert. Plus la couleur est claire et plus la position est ancienne avec un maximum de 1 s. Les positions futures sont quant à elles en rouge, ce sont les trajectoires à prédire à l'instant enregistré.

De ces exemples, on peut en tirer que les capteurs arrivent à bien percevoir les véhicules de l'environnement, même si l'orientation des capteurs n'est pas forcément idéale pour percevoir les objets qui entrent et sortent du rond-point. Étant donné les fortes dynamiques longitudinales et latérales dans de telles situations, on peut aisément s'avancer en disant que le simple fait d'utiliser les positions pour la prédiction de trajectoire à base de LSTM ne sera pas suffisant. En effet, cela ne suffisait déjà pas dans l'étude précédente où les mouvements latéraux étaient beaucoup plus faibles.

Contrairement au cas du changement de voie, où les véhicules peuvent se déporter et se rabattre avec assez de liberté vis-à-vis de l'infrastructure, en environnement urbain, les trajectoires latérales des véhicules sont plus complexes et très fortement corrélées à la forme des ronds-points et des intersections. Dès lors, il apparaît important, voire crucial, d'intégrer la géométrie de la route pour la prédiction de trajectoire.

Un autre point à améliorer et à explorer dans le cadre de cette étude est l'interaction entre

les différents usagers. Pour l'instant, nous ne considérons que l'état d'un objet, mais pas ceux des objets qui l'entourent. Intégrer ces éléments dans la prédiction d'intention et de trajectoire permettrait d'avoir une vue d'ensemble de la situation autour de l'ego véhicule.

Les deux points abordés précédemment pourraient être traités simultanément, par exemple à l'aide d'un réseau de neurones convolutif qui prendrait en entrée une image représentant l'infrastructure avec les trajectoires passées de tous les véhicules et en sortie estimerait leurs trajectoires futures.

Chapitre 5

Génération de trajectoire d'un véhicule autonome en environnement urbain

Sommaire

5.1 Planification de trajectoire d'un véhicule autonome : état de l'art 170
5.1.1 En environnement statique 171
5.1.2 Approches locales : des approches réactives à court terme 176
5.1.3 En environnement dynamique 178
5.2 Recherche de chemin en environnement statique 180
5.2.1 A* 181
5.2.2 Recherche de chemin par A* sous contrainte géométrique : exemples applicatifs 186
5.2.3 Dijkstra 191
5.2.4 Cas applicatif : le carrefour giratoire 192
5.2.5 Résultats sur prototype en parking 199
5.3 Génération de trajectoire en environnement dynamique 200
5.3.1 Intégration de la dimension temporelle pour la recherche de trajectoire 200
5.3.2 Application : dépassement d'un vélo 202
5.3.3 Intégration des contraintes dynamiques dans le A* 204
5.3.4 Application : dépassement sur voie à double sens 206
5.3.5 Bilan : avantages, problématiques et perspectives 213

On ne va jamais aussi loin que lorsqu'on ne sait pas où on va.

Christophe Colomb

Du fait d'une infrastructure plus complexe et d'un environnement plus riche, les manœuvres et trajectoires possibles en environnement urbain sont plus nombreuses. La dynamique latérale est davantage sollicitée. Le véhicule autonome doit être capable de se déplacer dans ce type d'environnement tout en tenant compte des autres usagers. Ainsi, le module de planification de trajectoire utilisera les résultats du chapitre précédent sur la détection d'intention et la prédiction de trajectoire des véhicules environnants.

Ce chapitre présente dans un premier temps un état de l'art complet des méthodes de planification de mouvement existantes dans le cas du véhicule autonome. Ensuite, une méthode de recherche de chemin en environnement statique est présentée. Le but étant de choisir en premier lieu une solution efficace en termes de temps de calcul et respectant les contraintes de mouvement du véhicule. La solution se base sur l'algorithme A*, enrichie afin de générer des chemins admissibles par le véhicule. Des points de passage, dont le positionnement est lié à l'infrastructure, sont ajoutés pour améliorer le temps de calcul. La méthode est ensuite appliquée au cas du rond-point en simulation et sur un parking avec les données capteur de HZTR.

Enfin, la méthode est améliorée afin de tenir compte des mouvements dynamiques des autres objets de l'environnement. Ainsi, les futures positions de ces objets sont utilisées pour calculer la trajectoire du véhicule. La méthode est validée en simulation pour des scénarios de dépassement sur une route à double sens de circulation.

5.1 Planification de trajectoire d'un véhicule autonome : état de l'art

Pour pouvoir se déplacer dans l'environnement, le véhicule autonome a besoin de déterminer comment il va se mouvoir dans celui-ci. Ces références du mouvement à suivre vont alors alimenter les boucles de régulation des actionneurs des dynamiques longitudinales et latérales, à savoir le moteur et la direction. En planification de mouvement, un chemin est un ensemble de points par lesquels doit passer la voiture. Une trajectoire contient, en plus du chemin, une dimension temporelle. Chaque point indique à quel temps il est atteint.

La planification de mouvement est à l'origine utilisée en robotique mobile. C'est pourquoi la majorité des méthodes de planification de trajectoire des véhicules autonomes est dérivée des techniques de robotique. La principale difficulté dans l'adaptation de ces méthodes est le passage d'un robot holonome, c'est à dire qui peut se déplacer dans tous les sens, à une voiture non holonome dont les déplacements sont limités par son rayon de courbure. [88] fait un état de l'art complet des méthodes de planification de mouvement pour la robotique. La Figure 5.1 reprend l'ensemble de ces méthodes, en gardant celles qui ont été utilisées dans le cadre de la conduite de voiture autonome.

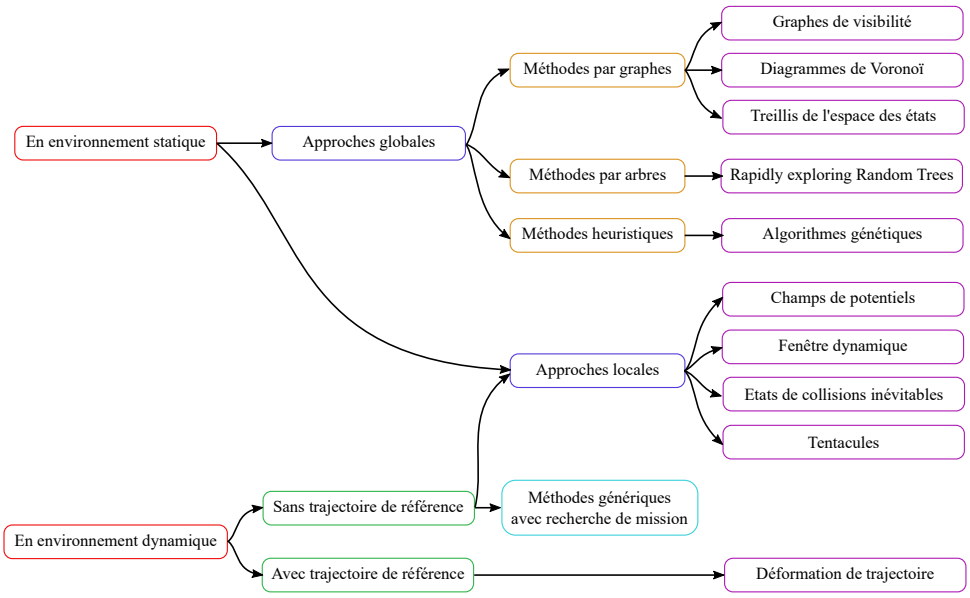


FIGURE 5.1 – Les différentes méthodes de planification de mouvement pour la robotique [Source : adapté de [88]]

Une distinction est faite entre les méthodes considérant un environnement statique et celles considérant l'environnement comme dynamique. Ces deux types de méthodes se recoupent avec les approches locales de planification de trajectoire. Avec ces approches, la trajectoire du véhicule est calculée à chaque instant sur des horizons de temps ou des distances très courtes. Le véhicule est donc réactif à son environnement et adapte sa trajectoire dès qu'il rencontre un objet, qu'il soit statique ou dynamique.

5.1.1 En environnement statique

Dans cette sous-section, nous présentons les approches globales de la planification de mouvement en environnement statique. Ces approches sont dites « délibératives », c'est-à-dire qu'elles déterminent le mouvement du robot, ou de la voiture ici, d'un point de départ à un point d'arrivée dans un environnement supposé connu. Les approches globales de planification de mouvement peuvent être découpées en trois sous-méthodes : les méthodes par graphes, les méthodes par arbres et les méthodes heuristiques.

Méthodes par graphes

Les graphes sont des objets mathématiques composés de nœuds (ou sommets) et de liens (ou arêtes). Deux nœuds sont reliés entre eux par un lien. Ce lien peut être orienté et/ou pondéré, signifiant que le passage d'un nœud à l'autre ne peut se faire que dans un sens et que celui-ci à un coût. Pour la planification de mouvement, les graphes servent à représenter l'environnement dans lequel doit se déplacer l'objet. Ils découpent ainsi l'environnement dans lequel va se déplacer la voiture en sous-espaces. Les nœuds sont donc des points de passage avec des coordonnées cartésiennes et les liens les chemins potentiels. Les liens sont pondérés par un coût représentant par exemple la longueur du chemin pour déterminer un chemin optimal au sens du coût choisi dans le graphe.

Pour planifier le chemin ou la trajectoire que va devoir suivre le véhicule avec un graphe, il faut dans un premier temps créer celui-ci à partir des informations de l'environnement puis calculer le chemin dans celui-ci. Il existe trois types de graphes principalement utilisés dans la

planification de mouvement : les graphes de visibilité, les diagrammes de Voronoï, les treillis de l'espace d'états ou encore des grilles régulières. Pour parcourir ces graphes, des algorithmes tels que Dijkstra ou A* peuvent être utilisés.

Les graphes de visibilité ont comme principe de créer des polygones autour des objets puis de lier les sommets de ces formes géométriques entre eux. Ne sont conservés ensuite que les liens ne coupant aucun des polygones, voir Figure 4.30 (a). Ce sont ces liens qui constituent les chemins possibles du véhicule. Originellement développé dans [134] pour la robotique, un diagramme de visibilité est aussi utilisé dans [135] pour génération de chemin d'un véhicule autonome. Avec cette approche, le diagramme de visibilité sert à calculer un chemin de référence qui sera ensuite optimisé pour ajouter des contraintes de dynamique véhicule. L'inconvénient majeur des graphes de visibilité est que le chemin final défini par la méthode longe forcément les objets. C'est pourquoi les diagrammes de Voronoï sont en pratique plus utilisés.

Les diagrammes de Voronoï sont, comme les graphes de visibilité, créés à partir des polygones décrivant les formes d'objets. Cependant, au lieu de prendre comme chemin admissible les liens entre les sommets de ces polygones, les diagrammes de Voronoï sont constitués des hyperplans équidistants des obstacles les plus proches, comme montré sur la Figure 5.2 (b). La méthode de calcul du diagramme de Voronoï est expliquée dans [108]. Pour le véhicule autonome, [201] découpe l'espace libre d'un parking avec un diagramme de Voronoï pour se déplacer dans celui-ci.

Le principal problème des diagrammes de Voronoï est qu'ils ne tiennent pas compte des contraintes dynamiques du véhicule dans le découpage de l'environnement.

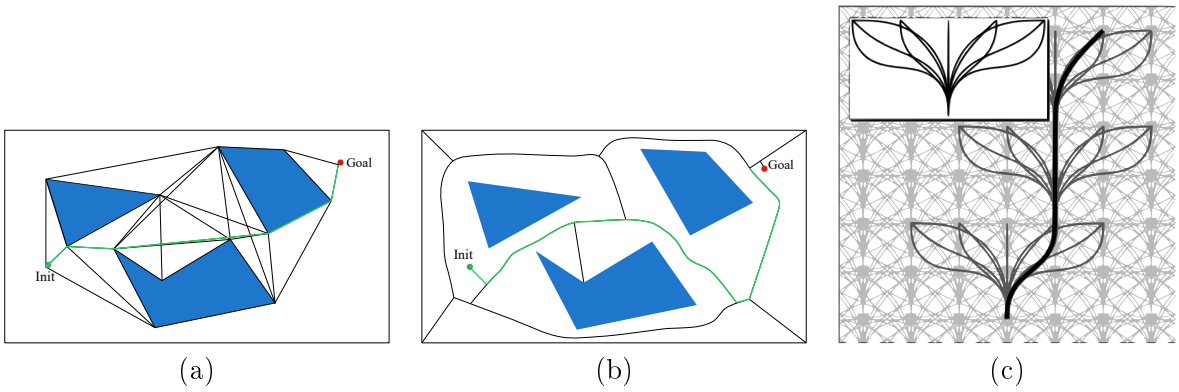


FIGURE 5.2 – Diagramme de visibilité (a), diagramme de Voronoï (b) et treillis de l'espace d'état [6] (c)

Les treillis, ou « lattice » en anglais, de l'espace d'état découpent l'environnement par les états et les chemins admissibles par le véhicule, voir Figure 4.30 (c). L'intérêt des treillis est de pouvoir représenter la manœuvrabilité des robots non holonomes, et donc des voitures. C'est-à-dire, de découper l'environnement uniquement par les états et chemins admissibles par le véhicule respectant ces contraintes de rayon de courbure. Ainsi, [149] applique la méthode des treillis pour le déplacement d'une astromobile (ou « rover »). Les treillis permettent ainsi au robot de se déplacer vers des points spécifiques en effectuant des manœuvres et avec la possibilité de respecter un certain cap à l'arrivée. [124] utilise des treillis pour générer la trajectoire d'un véhicule en mode autonome. Dans cette étude, les treillis respectent la structure de la route et contiennent une dimension temporelle de déplacement du véhicule. Ainsi, la méthode fonctionne aussi dans un environnement dynamique et est utilisée sur des exemples de changement de voie. Le problème de cette méthode est la construction du treillis, et donc du graphe, qui peut s'avérer couteuse en temps de calcul.

La méthode la plus simple pour découper l'environnement est de le découper par une grille régulière, avec des cases de tailles fixes. Dans [127], pour le DARPA Urban Challenge, la recherche de chemin se fait dans une grille régulière avec un algorithme dérivé du A* respectant les contraintes cinématiques du véhicule. [195] utilise aussi une grille régulière. Ici, l'algorithme A* explore la grille case par case, mais s'étend de manière à respecter le rayon de courbure minimum admissible par le véhicule.

Pour définir le chemin à suivre dans ces graphes, il faut utiliser des algorithmes de parcours de graphe. Les algorithmes les plus connus sont ceux de Dijkstra [43] et d'A* [76].

L'algorithme de Dijkstra permet de résoudre le problème du plus court chemin à partir d'un graphe pondéré, orienté ou non. Il choisit les points intermédiaires pour relier le nœud de départ à tous les autres nœuds du graphe avec la somme des liens entre les nœuds la plus faible possible. Pour cela, l'algorithme crée un sous-graphe en explorant tous les nœuds du graphe avec une heuristique de déplacement croissant.

L'algorithme A* est un dérivé du Dijkstra. Le but n'est plus de trouver la solution optimale, celle du chemin le plus court, mais de trouver une solution le plus rapidement possible. La solution trouvée sera optimale si l'heuristique choisie par le concepteur est admissible, c'est-à-dire inférieure à l'heuristique réelle. Le A* est un algorithme itératif qui avance nœud par nœud pour aller du point de départ au point d'arrivée en minimisant une fonction de coût. À chaque itération, le A* visite un nœud du graphe et lui attribue un coût en fonction de l'heuristique choisie, généralement la distance euclidienne au point d'arrivée, et du chemin déjà parcourus. Les nœuds « visitables » sont les nœuds adjacents aux nœuds déjà visités. Le A* s'étend en se déplaçant sur le nœud « visitable » avec l'heuristique la plus faible.

Il existe plusieurs variantes du A*. On peut notamment citer le D* [168] pour « Dynamic A* » qui permet de replanifier le chemin en cas de modification de l'environnement ou le anytime A* [117], aussi noté ARA*, qui donne une contrainte de temps au calcul du chemin. En effet, l'exploration d'un graphe par A* peut s'avérer couteuse en temps de calcul si le graphe est grand et l'heuristique mal calibrée. Le ARA* introduit une limite de temps de calcul. L'algorithme va alors rechercher dans un chemin sous optimal rapidement, puis améliorer ce chemin petit à petit tant qu'il lui reste du temps de calcul.

Méthodes par arbres

Contrairement aux graphes qui sont créés en amont pour découper l'environnement, les arbres s'étendent de manière itérative pour aller au point d'arrivée. Les « Rapidly-exploring Random Trees » [109], ou RRT, constituent la méthode par arbre quasi exclusive utilisée pour la recherche de chemin d'un véhicule autonome. Le principe de la méthode est d'explorer aléatoirement l'environnement pour étendre les branches de l'arbre. L'algorithme se déroule en trois étapes. La première définit une configuration aléatoire dans l'environnement libre N_{rand}. La deuxième sélectionne le nœud de l'arbre le plus proche de cette configuration N_{near}. Enfin, l'arbre s'étend d'une distance d à partir de N_{near} en direction de N_{rand} pour créer un nouveau nœud de l'arbre N_{new}. Ce processus d'exploration est illustré sur la Figure 5.3 (a).

Utilisés tels quels, les RRT ne sont pas applicables pour les comportements non holonomes des voitures. Cependant, de nombreuses recherches utilisent des RRT en contraignant l'extension des branches de l'arbre par le comportement dynamique du véhicule. On peut par exemple citer [106] qui, pour le DARPA Urban Challenge, a développé CL-RRT « Closed Loop Rapidly-exploring Random Tree ». Les nœuds des branches créées par le RRT sont utilisés comme référence dans

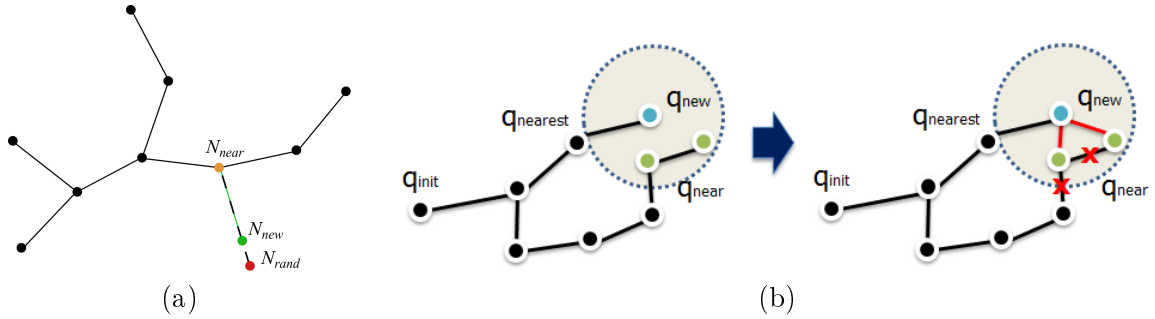


FIGURE 5.3 – Illustrations des RRT (a) et RRT* (b)

une boucle de régulation « fictive ». La boucle de régulation va simuler le comportement, la trajectoire, qu'effectuerait le véhicule pour arriver au nouveau nœud. C'est cette trajectoire qui sera utilisée comme branche dans l'arbre.

Une dérivée du RRT, le RRT* [94] est utilisé dans [87]. Derrière cette dérivée, une fonction de coût est ajoutée pour optimiser l'expansion des nœuds. Lorsqu'une configuration aléatoire est choisie, ce n'est pas forcement le nœud le plus proche en distance qui sera le parent du nouveau nœud. L'algorithme va considérer les nœuds les plus proches dans un certain rayon et l'arbre va s'étendre à partir de celui minimisant une fonction de coût sur toute la longueur de l'arbre, depuis la racine. Après avoir choisi le « bon » parent, le RRT* peut éventuellement « rebrancher » les liens déjà existants à proximité du nouveau nœud si cela permet de réduire le coût au global, voir Figure 5.3 (b). Les chemins seront ainsi plus directs. [87] utilise le principe de RRT* et un modèle bicyclette pour avoir des liens de l'arbre respectant les contraintes de la dynamique véhicule.

Les RRT sont faciles à mettre en œuvre et l'exploration rapide de l'environnement va forcément converger vers une solution si celle-ci est atteignable. La méthode classique RRT ne prend pas en compte le coût pour arriver à la solution ce qui peut conduire à des chemins longs et anguleux. Les RRT* résolvent ce problème en s'améliorant au fur et à mesure que le nombre de nœuds augmente dans l'arbre. Les RRT présentent cependant des problèmes de convergence lorsque l'arbre se rapproche du point d'arrivée ou lorsqu'il faut passer par des passages étroits. Enfin, les RRT sont des processus aléatoires, ils sont donc peu prévisibles ce qui peut poser problème pour du calcul embarqué.

Méthodes heuristiques

Le problème de planification de mouvement est généralement un problème sous contraintes. Pour respecter ces contraintes, une solution est de formuler un problème d'optimisation dans le but de les minimiser. Les problèmes sont souvent non-convexes, c'est-à-dire qu'ils présentent un optimum global et plusieurs optimums locaux. Pour les résoudre, il faut des méta heuristiques qui sont des algorithmes d'optimisation de problèmes complexes. Ces algorithmes explorent l'espace des solutions pour essayer de converger vers la meilleure solution suivant l'heuristique choisie. Les méta heuristiques sont souvent inspirées de phénomènes naturels : optimisation par essaims particulaires, colonies de fourmis, algorithmes génétiques.

De manière générale, ces algorithmes peuvent être utilisés en robotique mobile et beaucoup de travaux sont menés avec ces méthodes pour la planification de trajectoire d'UAV « Unmaned Aerial Vehicle ». En pratique, ils sont très peu utilisés pour les problèmes de conduite autonome. On peut citer [55] qui présente une méthode de planification de trajectoire par algorithme

génétique pour les robots non holonomes. [157] combine les algorithmes génétiques, qui vont déterminer les meilleurs points intermédiaires à atteindre par le véhicule, et les champs de potentiels, qui vont permettre de s'éloigner des objets. L'algorithme génétique utilise trois étapes successives, inspirées de la théorie de l'évolution de Darwin, qu'il va répéter jusqu'à atteindre le minimum global. La première de ces étapes est la sélection des individus. Parmi un ensemble de solutions possibles, l'algorithme sélectionne les individus ayant le plus grand potentiel. Ensuite, de nouveaux individus sont créés par croisement à partir des individus choisis. Les nouveaux individus auront donc un mélange des caractéristiques des individus sélectionnés. Enfin, la dernière étape est une étape de mutation qui va altérer aléatoirement les caractéristiques de certains individus.

L'avantage des algorithmes génétiques est qu'ils sont robustes à l'initialisation. De plus, ils sont capables de trouver un minimum local ou de s'en rapprocher. Par contre, pour atteindre ce minimum, un nombre important d'évaluations est nécessaire et l'algorithme peut mettre du temps à converger.

L'optimisation par essaims particulaires, aussi notée PSO pour « Particle Swarm Optimization » s'inspire, elle, des déplacements des groupes d'oiseaux et leurs interactions. Les oiseaux migrateurs doivent parcourir de longues distances, leur formation en « V » leur permet d'optimiser l'énergie dépensée en utilisant l'aspiration procurée par la formation. En contrepartie, le groupe va moins vite qu'un oiseau seul. L'algorithme PSO utilise un groupe de particules, des solutions potentielles au problème d'optimisation, qui vont se déplacer dans l'espace des solutions jusqu'à atteindre l'optimum. Le déplacement d'une particule se fait à partir de trois paramètres : la vitesse actuelle de la particule, sa meilleure solution et la meilleure solution obtenue dans son voisinage. Trois types de comportements peuvent caractériser une particule : égoïste (elle se déplace suivant sa propre vitesse), conservateur (elle revient à sa meilleure position) ou panurgien (elle va suivre la meilleure particule). Les PSO sont utilisées dans [74] pour calculer le cap et la vitesse optimale d'un chemin d'un véhicule autonome.

Les PSO sont faciles à coder, ont relativement peu de paramètres à ajuster et sont capables de converger rapidement. Cependant, le choix des paramètres peut être compliqué et des phénomènes de convergence prématurée (l'algorithme peut rester bloquer dans un minimum local) peuvent apparaitre.

Parmi les méthodes bio-inspirées, celles de déplacement en « essaim » sont aussi utilisées dans le cadre de la conduite autonome coopérative. C'est le cas des bancs de poisson [129] qui permettent de reproduire les interactions entre véhicules, et donc de générer une trajectoire en fonction de ces interactions. Ainsi, la modélisation d'un banc de poissons se base sur le principe de trois comportements : l'attraction (les individus restent proches pour former un groupe), la répulsion (les individus s'éloignent pour éviter les collisions) et l'imitation (les individus adoptent la même vitesse et la même direction de déplacement).

Globalement, les optimisations par méta heuristiques ont l'avantage d'être discrètes, c'est-à-dire qu'elles n'ont pas besoin de calculer le gradient des fonctions à optimiser qui peut s'avérer être une tâche complexe. De plus, ces méthodes sont assez intuitives à comprendre et facilement implémentables, car inspirées de phénomènes naturels. Cependant, le choix des paramètres qui vont conditionner l'exploration peut être difficile et le temps de calcul pour arriver à une solution optimale est souvent très élevé ce qui peut poser des problèmes pour la programmation en temps réel d'un véhicule autonome.

5.1.2 Approches locales : des approches réactives à court terme

Contrairement aux approches globales qui calculent l'intégralité du chemin ou de la trajectoire que devra suivre le véhicule, les approches locales calculent à chaque pas de temps la commande à envoyer aux actionneurs pour se déplacer dans l'environnement. Ces approches sont aussi dites « réactives » et fonctionnent pour des horizons de temps très courts.

Champs de potentiels

Les champs de potentiels [99] partent du principe que le véhicule se déplace dans un champ électromagnétique qui va conditionner son déplacement. Ce champ est constitué de champs attractifs, l'emplacement à atteindre par le véhicule, et de champs répulsifs, les objets à éviter. Le chemin, ou la trajectoire, résultant fait en sorte que le véhicule se déplace vers le champ attractif tout en s'éloignant des champs répulsifs. La méthode se réduit finalement à un problème d'optimisation.

[190] développe une méthode de planification de trajectoire par champs de potentiels sur autoroute. Plusieurs champs de potentiels répulsifs sont définis, ils concernent les autres voitures de l'environnement, les lignes des voies pour rester au milieu de celles-ci, la route et la vitesse cible pour rester proche de la vitesse désirée. Dans cette étude, la voiture est capable de changer de voie pour dépasser des véhicules plus lents ou de freiner si aucun dépassement n'est possible à cause d'un trafic important. Cependant, dans cette étude, le véhicule est modélisé par un point omnidirectionnel et non pas par un modèle cinématique ou bicyclette.

[154] considère la dynamique du véhicule en calculant la trajectoire du véhicule par un MPC qui inclut dans sa fonction objectif les champs de potentiels des objets pour l'évitement d'obstacles.

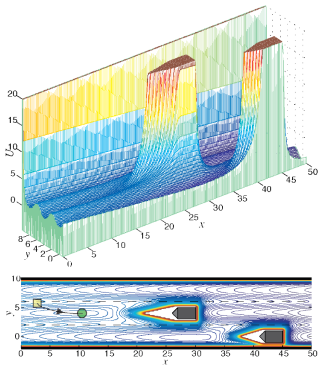


FIGURE 5.4 – Illustration des champs de potentiel de [190]

La planification de trajectoire par champs de potentiels est relativement facile à mettre en œuvre et à interpréter. De plus, il n'est pas nécessaire de connaitre précisément la dimension des objets pour s'en éloigner. Cependant, les champs de potentiels sont sujets aux minimaux locaux pouvant entraîner une replanification. De plus, de grands échantillons de temps de calcul de la trajectoire peuvent amener à un comportement oscillatoire et des collisions. La méthode présente aussi des limitations dans les environnements contraints avec beaucoup d'obstacles ou étroits.

Fenêtre dynamique

Les fenêtres dynamiques [57] sont des approches réactives permettant de contrôler le robot directement en vitesse linéaire et vitesse angulaire. Ces commandes sont choisies parmi les vitesses respectant les contraintes cinématiques et dynamiques du robot et garantissant la sécurité, c'est à dire permettant au robot de s'arrêter avant d'entrer en collision. Parmi ces vitesses admissibles, les vitesses finalement appliquées sont celles maximisant une fonction de coût pour que le robot se dirige vers un point cible sans entrer en collision. La méthode a été adaptée pour les environnements dynamiques dans [21].

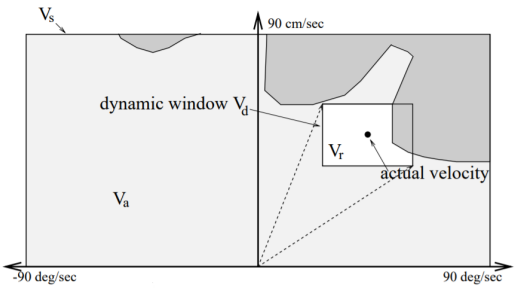


FIGURE 5.5 – Illustration de la fenêtre dynamique de [57]

[38] utilise la méthode de la fenêtre dynamique et l'a implantée sur un prototype. L'approche réactive permet ainsi de calculer directement l'angle volant et la vitesse. Dans cette étude, le véhicule se déplace dans un environnement non structuré. Les mêmes auteurs ont étendu leur méthode pour des environnements dynamiques et structurés pour faire du suivi de voie avec évitement d'objets [39].

L'inconvénient de cette méthode est qu'elle ne « regarde » que les objets atteignables dans sa fenêtre de commande et ne considère donc pas tous les objets. Certaines informations peuvent donc être omises, il n'y a pas de vision globale de l'évolution de l'environnement.

Tentacules

La méthode des tentacules [186] consiste à calculer plusieurs trajectoires admissibles par le véhicule sur un horizon de temps assez court, voir Figure 5.6. Ces trajectoires possibles sont ensuite évaluées par un critère de sécurité et de confort pour choisir la meilleure trajectoire à suivre à l'instant donné. Les tentacules sont calculés dans le repère du centre de gravité du véhicule et peuvent être des trajectoires en arc de cercle [186] ou en clothoïdes [80].

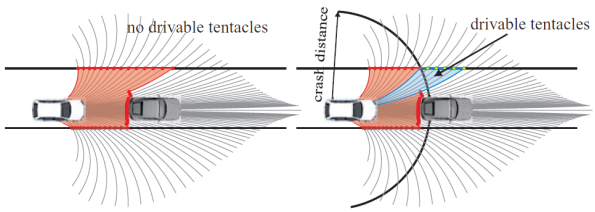


FIGURE 5.6 – Illustration de la méthode des tentacules [Source : [186]]

Cette méthode est par exemple utilisée dans [3] et [131] pour la conduite autonome d'un véhicule dans les cas de suivi de voie et de changement de voie. Le principe de génération de

trajectoire par tentacules permet de générer une trajectoire sûre et admissible. Le véhicule peut ainsi se déplacer dans un environnement incertain. Cependant, il apparaît que la méthode peut présenter un temps de calcul important la rendant complexe à intégrer en temps réel.

États de collisions inévitables

Les états de collisions inévitables ont été introduits dans [58]. Un système est dans un état de collision inévitable si, quelle que soit la séquence de contrôle appliquée, il existe un temps t pour lequel le système va entrer en collision. Pour déterminer si un état est un état de collision inévitable, il faut prendre en considération la dynamique du véhicule ainsi que celles des véhicules de l'environnement et l'estimation de leurs futures positions. En théorie, cette détermination nécessite de vérifier que toutes les futures trajectoires possibles sont bien sans collision. En pratique [145] réduit l'espace de recherche des états de collisions inévitables.

Dans [86], les états de collisions inévitables sont utilisés comme heuristique d'un algorithme A* pour rechercher les bonnes commandes à appliquer en vitesse en environnement urbain. La méthode présentée ne gère cependant pas la dynamique latérale, le véhicule reste dans sa voie.

La méthode de planification de trajectoire utilisant les états de collisions inévitables présente l'avantage d'intégrer directement la dynamique du véhicule, ceux des objets et de garantir la sécurité du véhicule. Cependant, la méthode est très couteuse en temps de calcul et dépend fortement de la fiabilité de la prédiction de trajectoire des autres objets de l'environnement.

5.1.3 En environnement dynamique

Méthodes de déformation de trajectoire

Dans le cas où le véhicule possède déjà une trajectoire de référence, calculée par une des approches globales par exemple, plusieurs études ont été menées pour déformer la trajectoire précalculée en cas d'évènement inattendu au moment du calcul de la trajectoire. Le but étant d'éviter la replanification totale pour gagner en temps de calcul et en réactivité.

La déformation de chemin par bande élastique ou « elastic band » [151] est une de ces méthodes de déformation de trajectoire. À partir des configurations du robot le long du chemin, des « bulles » sont construites et représentes des états atteignables et sans collision, Figure 5.7. Lors du déplacement du robot, deux forces sont appliquées sur les bulles : des forces répulsives en cas d'apparition de nouveaux objets et des forces de contraction qui tendent le chemin si les objets s'écartent. La méthode a été adaptée pour les propriétés non holonomes d'une voiture dans [98].

Les premières études ne prenaient pas en compte le temps de déplacement du robot le long du chemin. C'est pourquoi dans [97], la dimension temporelle est ajoutée dans le choix du positionnement des points de repère le long de la trajectoire. Ainsi, en plus d'influer sur la dynamique latérale, le véhicule va pouvoir modifier sa dynamique longitudinale. [42] utilise en plus les futures positions des objets dynamiques, en supposant leurs vitesses longitudinale et latérale constantes, et la dynamique du système pour calculer les futures positions du véhicule.

Dans [107], une méthode de déformation de chemin utilisant des champs de potentiel est mise en place pour des robots non holonomes. Elle est étendue dans [15] pour la déformation de trajectoire d'une voiture. Avec cette méthode, plusieurs trajectoires déformées sont calculées et

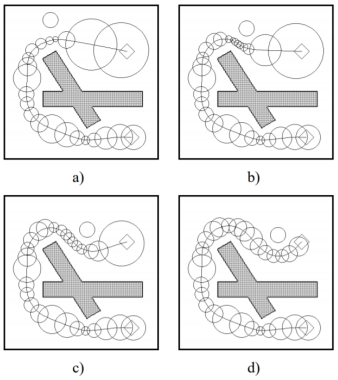


FIGURE 5.7 – Illustration des bandes élastiques de [151]

testées pour respecter toutes les conditions d'optimisation. Cela conduit, d'après les auteurs, à un temps de calcul élevé. De plus, il n'y a pas de garantie de solution.

Méthodes génériques avec recherche de mission

Dans le cadre du véhicule autonome, la planification de trajectoire est à la fois influencée par l'infrastructure de la route, mais aussi par les objets de l'environnement. Ainsi, [96] résume la planification de trajectoire d'un véhicule autonome par un enchainement de trois niveaux de planification, voir Figure 5.8.

La première est celle de la recherche de chemin dans l'environnement par une approche globale ou locale. Elle est couplée avec la recherche de manœuvre à effectuer par le véhicule. Le but de cette étape étant de trouver l'action de haut niveau à effectuer pour respecter le Code de la route et conserver la sécurité du véhicule et de ses passagers. Les missions sont donc du type : changer de voie, s'arrêter, freiner, s'insérer, etc. Enfin, la trajectoire finale est générée.

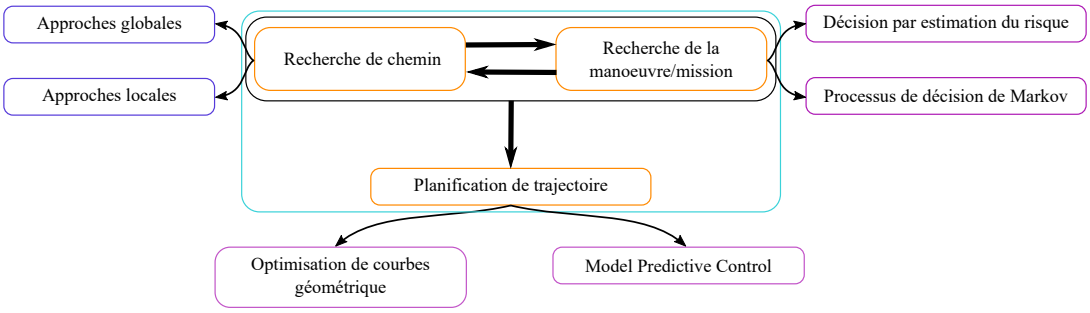


FIGURE 5.8 – La planification de trajectoire en environnement dynamique avec planification de mission

Nous avons vu précédemment les approches globales et locales pour la recherche de chemin. Nous allons donc expliquer ici les principales méthodes de recherche de mission et de définition de trajectoire.

Les approches de recherche de mission peuvent se découper en deux catégories : les méthodes de décision par estimation du risque et celles par processus de décision de Markov.

Dans [60], une fonction d'utilité est calculée pour toutes les manœuvres disponibles. La manœuvre respectant le mieux les objectifs de sécurité est choisie. La liste des manœuvres doit donc être exhaustive. [37] utilise la même méthode de détermination de manœuvre dans un environnement à haute vitesse et en estimant le comportement des véhicules autour. [63] étudie la conduite coopérative sur autoroute, si la manœuvre du conducteur est différente de la manœuvre la plus sûre, le système peut intervenir sur les actionneurs. Il y a neuf manœuvres disponibles qui sont les combinaisons de : aller tout droit, changer de voie à gauche, changer de voie à droite et rester à la même vitesse, accélérer, freiner. La manœuvre choisie est celle présentant le risque de collision le plus faible. Dans [92], pour la conduite autonome sans marquage au sol, neufs manœuvres différentes sont définies et des règles sont établies pour savoir laquelle appliquer.

Les processus de décision de Markov sont des raisonnements probabilistes maximisant une fonction de récompense [82]. Le but est de trouver l'action ayant la plus grande probabilité d'avoir un gain important. Dans [17], un processus de décision de Markov sert à déterminer la meilleure action à prendre parmi un ensemble de combinaisons d'accélérations et vitesses latérales pour effectuer des changements de voies ou faire de l'évitement d'objets. En pratique, et pour la conduite autonome, les états des objets ne sont pas entièrement connus à cause du bruit des capteurs ou de l'occlusion. C'est pourquoi la plupart des études sur la définition de manœuvre pour le véhicule autonome utilisant les processus de Markov préfèrent les processus de décision de Markov partiellement observables [90] permettant de modéliser l'incertitude de la connaissance de l'environnement. Dans [181] cette méthode est utilisée pour faire du changement de voie en environnement urbain et [18] l'utilise pour décider de l'insertion du véhicule dans des intersections en T. Les processus de décision de Markov partiellement observables ont pour avantages de modéliser l'incertitude des actions, mais aussi de la connaissance de l'environnement. Cependant, le fait de rechercher la meilleure action à prendre en considérant tous les états de croyance est extrêmement couteux en temps de calcul et est le principal désavantage de telles méthodes.

Une fois la mission et le chemin associé déterminés, la trajectoire que doit suivre le véhicule est calculée. Deux grandes méthodes ici aussi sont utilisées dans la littérature : l'optimisation de courbes géométriques et le MPC « Model Predictive Control ».

Dans le cas de l'optimisation de courbes géométriques, les trajectoires générées par le module de planification de trajectoire peuvent être de différentes natures. On y trouve les courbes de Béziers [75], [30], les polynômes [63], [192] ou les trajectoires reproduisant la conduite humaine [69, 105, 78]. Dans tous les cas, l'objectif est de calculer la trajectoire respectant les contraintes dynamiques du véhicule et garantissant sécurité et confort aux passagers.

Le calcul de la trajectoire par MPC combine planification de trajectoire et régulation. La méthode tire profit de la connaissance du modèle dynamique utilisé par la commande prédictive pour définir la trajectoire à suivre. [135] utilise un chemin de référence calculée à l'aide d'un diagramme de visibilité puis utilise un MPC pour calculer une trajectoire admissible à partir du chemin. Dans [65], les contraintes de l'environnement sont ajoutées directement au problème d'optimisation du MPC. Cependant, contrairement aux autres méthodes, ici la trajectoire se calcule sur un horizon de temps de 2,5 s. L'horizon ne peut pas être trop grand pour que le problème d'optimisation puisse être résolu dans un temps raisonnable.

5.2 Recherche de chemin en environnement statique

Pour un véhicule autonome de niveau 3 et plus, la première étape est de pouvoir se déplacer dans un environnement statique, sans autres objets dans l'environnement. C'est cette problématique qui est traitée dans cette section. Il s'agira ensuite de faire évoluer la méthode développée

afin de prendre en compte les objets dynamiques. Dans un premier temps, nous allons donc nous focaliser sur la détermination du chemin à suivre par le véhicule.

Suite à l'état de l'art présenté précédemment, une méthode ne nécessitant pas trop de temps de calcul et facilement adaptable est la méthode basée sur une grille régulière avec recherche de chemin par A*. L'avantage de cette méthode est qu'elle ne nécessite pas la construction d'une grille complexe. De plus, la recherche du chemin dans une telle grille permet d'y inclure les règles du Code de la route. Enfin, le fait de choisir une approche globale plutôt qu'une approche réactive permet d'être certain qu'une solution est disponible ou non. Le choix du A* comme algorithme d'exploration est justifié par la volonté de trouver un chemin rapidement. En effet, le temps de calcul est crucial pour le bon fonctionnement du véhicule en mode autonome.

5.2.1 A*

Principe et algorithme

L'algorithme du A* est un algorithme de recherche de type « best first », c'est à dire qui parcours un graphe de manière à explorer les nœuds avec un coût faible, ou respectant le mieux les règles fixées, en premier. Contrairement au Dijkstra qui va explorer tous les liens pour déterminer le chemin optimal, le A* va explorer l'environnement itérativement en utilisant une heuristique qui va le guider jusqu'à l'arrivée. Le Dijkstra donne tous les chemins optimaux en partant d'un nœud alors que le A* a un objectif. Pour se déplacer dans le graphe, l'algorithme va attribuer aux nœuds voisins s des nœuds qu'il visite une fonction de coût. Cette fonction de coût est composée de deux éléments : un premier représentant la distance parcourue depuis le nœud de départ s_S et un deuxième qui est l'heuristique entre le nœud et le point d'arrivée s_G représentant la distance restante à parcourir. On peut ainsi écrire :

$$f(s) = g(s) + h(s), \tag{5.1}$$

avec $f(s)$ la fonction de coût attribuée au nœud s, $g(s)$ la distance parcourue de s_S à s, et $h(s)$ l'heuristique entre s et s_G.

Algorithmiquement, le A* fabrique deux listes répertoriant les nœuds explorés. Chaque nœud est associé à ses coordonnées, son coût et les coordonnées du nœud parent. Chaque nœud visité est positionné dans la « liste fermée » et chaque nœud visitable dans la « liste ouverte ». Au début de l'algorithme, la liste fermée est vide et la liste ouverte contient le nœud de départ. Pour chaque itération, l'algorithme choisit dans la liste ouverte le nœud avec le coût le plus faible. Si ce nœud n'est pas le nœud d'arrivée, les fonctions de coût de ses voisins sont calculées et ils sont alors ajoutés à la liste ouverte. Le nœud exploré est ensuite inclus dans la liste fermée. Une nouvelle itération commence tant que le nœud de la liste ouverte choisi n'est pas le nœud d'arrivée ou que la liste ouverte est vide. Si le nœud d'arrivée s_G est atteint, le chemin est reconstruit à partir de la liste fermée en remontant les parents de chaque nœud à partir du nœud d'arrivée.

On note G le graphe dans lequel doit se déplacer le A*, s_S le point de départ, s_G le point d'arrivée, g la fonction calculant la distance parcourue entre deux points, h la fonction heuristique estimant la distance restante à parcourir pour aller au point d'arriver et $f = g + h$ la fonction de coût à minimiser. Sachant ces notations, le pseudo-code de l'algorithme A* s'écrit :

Algorithme 1 Fonction A*

Entrée : s_S, s_G, G
Sortie : *chemin*

Création de la file ClosedList
Création de la file OpenList
Ajout de s_S dans OpenList
Tant que OpenList n'est pas vide **faire**
  Chercher le nœud s dans OpenList avec le coût $s.f$ le plus faible
  Si $s = s_G$ **alors**
    Ajouter s à la ClosedList
    chemin=ReconstruireChemin(ClosedList,s_G,s_S)
    Retourner *chemin*
  Fin Si
  Pour chaque voisin v de s dans G **faire**
    Calculer la fonction g : $v.g \leftarrow s.g + g(s, v)$
    Calculer la fonction h : $v.h \leftarrow h(v, s_G)$
    Calculer la fonction f : $v.f \leftarrow v.g + v.h$
    Si v n'existe pas dans l'OpenList ou la ClosedList ou a un coût supérieur **alors**
      Ajouter v à l'OpenList
    Fin Si
  Fin Pour
  Ajouter s à la ClosedList
Fin Tant que
Terminer le programme : chemin non trouvé

Dans cet algorithme, la fonction « ReconstruireChemin », va récupérer les points dans la ClosedList qui ont permis d'arriver à s_G.

Pour représenter de manière visuelle l'exploration de l'environnement par le A*, la Figure 5.9 présente un exemple. Dans cet exemple, le A* doit trouver un chemin dans une grille de 6x6. Ici, le A* peut se déplacer dans les huit directions de la grille : haut gauche, haut, haut droite, gauche, droite, bas gauche, bas, bas droite. La fonction heuristique $h(s)$ est la distance euclidienne entre s et s_G, la fonction $g(s)$ vaut 1 si le déplacement se fait en ligne et $\sqrt{2}$ si diagonal.

La Figure 5.9 montre pour chaque itération (lecture de gauche à droite) comment les coûts des nœuds sont calculés et comment ils sont ajoutés aux listes ouverte et fermée. Les coûts de chaque nœud sont indiqués dans les grilles. Pour calculer le chemin optimal allant de s_S à s_G, l'algorithme a besoin de 10 itérations et s'est étendu sur 18 nœuds, en comptant s_G.

Grille d'occupation

s_S : nœud de départ
s_G : nœud d'arrivée

Itération 1

Voisins de s_S : v_1, v_2 et v_3

Calcul de $f(v_1)$, $f(v_2)$ et $f(v_3)$:
- $f(v_1) = g(v_1) + h(v_1) = 1 + \sqrt{4^2 + 5^2} = 7{,}40$
- $f(v_2) = g(v_2) + h(v_2) = \sqrt{2} + \sqrt{4^2 + 4^2} = 7{,}07$
- $f(v_3) = g(v_3) + h(v_3) = 1 + \sqrt{5^2 + 4^2} = 7{,}40$

Mettre v_1, v_2 et v_3 dans la liste ouverte

Mettre s_S dans la liste fermée

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Itération 2

Nœud avec le coût le plus faible dans la liste ouverte : v_2

Voisins de v_2: s_S, v_1, v_3, v_4, v_5 et v_6

Calcul de $f(v_3)$, $f(v_4)$ et $f(s_S)$:
- $f(s_S) = g(s_S) + h(s_S) = 2 * \sqrt{2} + \sqrt{5^2 + 5^2} = 9{,}90$
- $f(v_1) = \sqrt{2} + 1 + \sqrt{4^2 + 5^2} = 8{,}81$
- $f(v_3) = \sqrt{2} + 1 + \sqrt{5^2 + 4^2} = 8{,}81$
- $f(v_4) = 2 * \sqrt{2} + \sqrt{3^2 + 5^2} = 8{,}66$
- $f(v_5) = \sqrt{2} + 1 + \sqrt{3^2 + 4^2} = 7{,}41$
- $f(v_6) = \sqrt{2} + 1 + \sqrt{5^2 + 3^2} = 8{,}24$

s_S, v_1, v_3 et v_4 déjà dans la liste ouverte avec des coûts plus faibles : ne rien faire
Mettre v_5 et v_6 dans la liste ouverte

Mettre v_2 dans la liste fermée

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Itérations 3, 4 et 5

Itération 3 : v_1 est mis dans la liste fermée
Itération 4 : v_3 est mis dans la liste fermée

Itération 5 : v_5 nœud de la liste ouverte avec coût le plus faible

Voisins de v_5: v_1, v_2, v_4, v_7 et v_8

Calcul des coûts :
- $f(v_7) = \sqrt{2} + 2 + \sqrt{2^2 + 4^2} = 7{,}89$
- $f(v_8) = 2 * \sqrt{2} + 1 + \sqrt{2^2 + 3^2} = 7{,}43$
- v_1, v_2 et v_4 ont des coûts plus importants

v_1, v_2 et v_4 déjà dans une liste avec des coûts plus faibles : ne rien faire
Mettre v_7 et v_8 dans la liste ouverte

Mettre v_5 dans la liste fermée

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Itération 6

v_8 nœud de la liste ouverte avec coût le plus faible

Voisins de v_8: v_5, v_7, v_9, v_{10}, v_{11}, v_{12} et v_{13}

Calcul des coûts :
- $f(v_9) = 9{,}36$
- $f(v_{10}) = 7{,}99$
- $f(v_{11}) = 7{,}48$
- $f(v_{12}) = 7{,}66$
- $f(v_{13}) = 8{,}85$
- v_7 et v_5 ont des coûts plus importants

v_7 et v_5 déjà dans une liste avec des coûts plus faibles : ne rien faire
Mettre v_9, v_{10}, v_{11}, v_{12} et v_{13} dans la liste ouverte
Mettre v_8 dans la liste fermée

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Itération 7

v_{11} nœud de la liste ouverte avec coût le plus faible

Voisins de v_{11}: v_8, v_{10}, v_{10}, v_{14}, v_{15} et v_{16}

Calcul des coûts :
- $f(v_{14}) = 7{,}99$
- $f(v_{15}) = 7{,}48$
- $f(v_{16}) = 7{,}66$
- v_8, v_{10} et v_{12} ont des coûts plus importants

v_8, v_{10} et v_{12} déjà dans une liste avec des coûts plus faibles : ne rien faire
Mettre v_{14}, v_{15} et v_{16} dans la liste ouverte

Mettre v_{11} dans la liste fermée

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Itérations 8 et 9

Itération 8 : v_{12} est mis dans la liste fermée

Itération 9 : v_{15} nœud de la liste ouverte avec coût le plus faible

Voisins de v_{15}: v_{11}, v_{14}, v_{17} et s_G

Calcul des coûts :
- $f(v_{17}) = 9{,}07$
- $f(s_G) = 7{,}66$
- v_{11} et v_{14} ont des coûts plus importants

v_{11} et v_{14} déjà dans une liste avec des coûts plus faibles : ne rien faire
Mettre v_{17} et s_G dans la liste ouverte

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Itération 10

s_G est le nœud de la liste ouverte avec le coût le plus faible
s_G est le nœud d'arrivée → fin des itérations
s_G est mis dans la liste fermée

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

Reconstitution du chemin

Reconstitution du chemin avec la liste fermée.

A partir des parent de s_G, remonter itérativement chaque parent jusqu'à s_S.

Nœuds dans la liste fermée
Nœuds dans la liste ouverte

FIGURE 5.9 – Exemple d'exploration d'une grille par A*

L'algorithme A* possède la propriété de complétude, c'est-à-dire que l'algorithme est sûr de trouver une solution si elle existe. De plus, une autre propriété intéressante de cet algorithme est son admissibilité. Cela signifie que si la fonction heuristique h choisie est inférieure à l'heuristique optimale, l'algorithme trouvera forcément le chemin le plus court. Avec l'utilisation d'une liste fermée dans l'algorithme, pour être admissible, le A* doit en plus être monotone (ou consistent). Cela veut dire que pour tout couple de nœuds (s_1, s_2), l'inégalité $h(s_1) \leq d(s_1, s_2) + h(s_2)$ doit être respectée, avec $d(s_1, s_2)$ le coût entre les deux nœuds. Sous ces conditions, le A* est optimal.

Adaptation à la navigation d'un véhicule non holonome

L'algorithme A*, comme présenté ci-dessus, ne permet pas de trouver un chemin admissible par le véhicule étant donné son caractère non holonome. En reprenant les travaux de [195], il devient cependant possible, en contraignant l'expansion du A*, de calculer un chemin respectant le rayon de courbure minimum de la voiture. Normalement, pour l'exploration d'une grille régulière avec un axe longitudinal X, et un axe latéral Y, chaque noeud s exploré et placé dans la liste fermée possède les attributs suivants :

- position longitudinale $s.X$
- position latérale $s.Y$
- heuristique au point d'arrivée $s.h$
- coût du déplacement pour arriver à s, $s.g$
- coût total du nœud $s.f$
- position longitudinale du nœud parent $s.X_p$
- position latérale du nœud parent $s.Y_p$.

Pour le KSA* « Kinematics and shape-aware A* » de [195], sont en plus ajoutés :

- sens $s.sens$
- orientation $s.cap$.

La distinction entre marche avant et marche arrière va ainsi permettre l'exécution de manœuvres et l'orientation va servir à respecter le rayon de courbure minimum.

Comme la recherche de chemin se fait dans une grille régulière, il y a huit orientations possibles, représentées sur la Figure 5.11, qui correspondent à des caps de multiples de 45°.

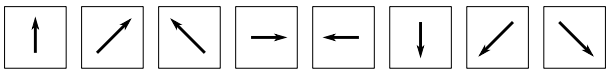


FIGURE 5.10 – Orientations possibles de chaque nœud

Ensuite, en fonction de l'orientation du nœud, six voisins sont visitables. Chacun se verra alors attribuer une orientation et un sens en fonction du nœud parent. La Figure 5.11 montre les voisins atteignables ainsi que leurs orientations et sens pour les deux premières orientations. Le principe est le même pour les autres orientations, moyennant une rotation.

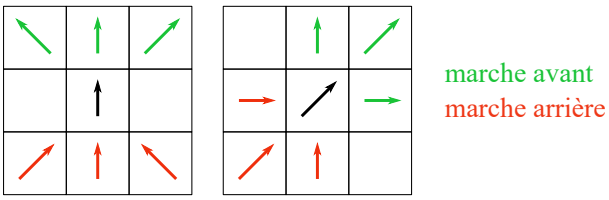


FIGURE 5.11 – Exemples d'expansion dans la grille en fonction de l'orientation

Enfin, pour respecter le rayon de courbure, il faut que l'exploration des noeuds conserve la même orientation pendant un nombre de pas noté n_p. En effet, si le A* cherche à faire un virage à droite, et ainsi passer d'un cas de 0° à 90°, il faut qu'il reste plusieurs fois avec l'orientation à 45° pour respecter un rayon de courbure minimum. Par exemple, la Figure 5.12 illustre le déplacement dans la grille pour effectuer un cercle avec $n_p = 4$. Pour une grille de 1x1, le rayon R_L du cercle tangent aux mouvements horizontaux est calculé par :

$$R_L = \frac{3n_p}{2}, \tag{5.2}$$

et R_D, le rayon du cercle tangent aux mouvements diagonaux :

$$R_D = n_p\sqrt{2}. \tag{5.3}$$

Le rayon R_L est légèrement plus grand que R_D et sera donc utilisé pour calculer n_p nécessaire dans le cas de notre véhicule pour être sûr de pouvoir suivre le chemin.

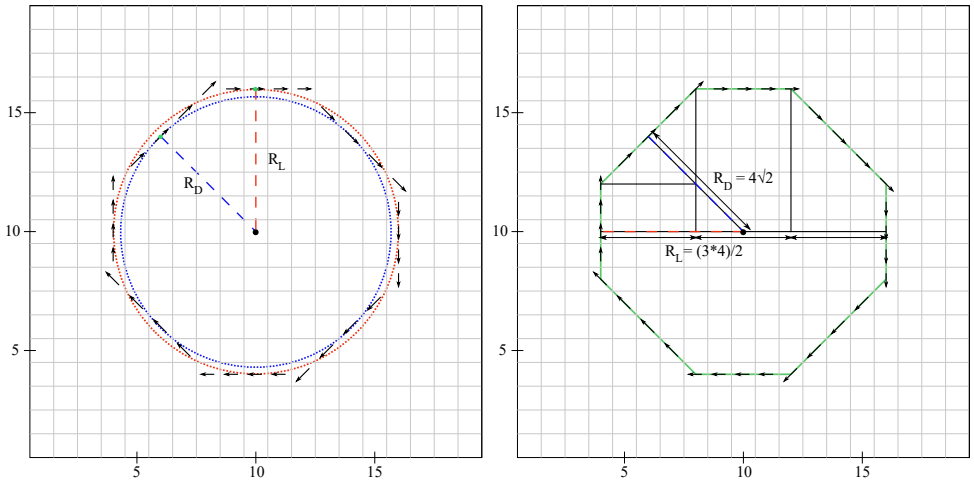
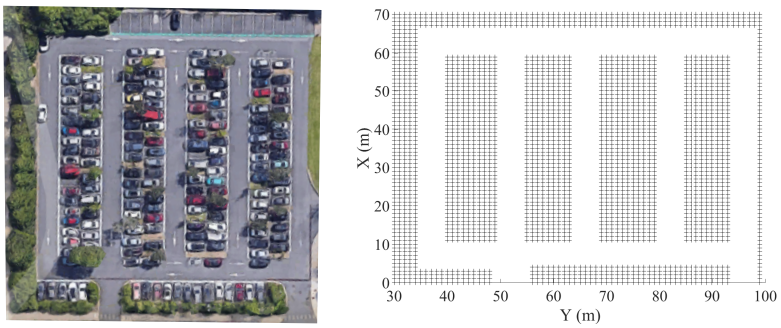


FIGURE 5.12 – Illustration des rayons de courbure R_L et R_D

En termes d'exploration algorithmique, étant donné que plusieurs attributs ont été ajoutés à chaque nœud, les conditions d'ajout à la liste ouverte sont modifiées. En effet, maintenant, une même position dans la grille peut avoir un sens ou une orientation différente en fonction de comment cette position a été atteinte. Ainsi, pour ne pas limiter les déplacements, un nœud voisin v est ajouté à la liste ouverte s'il n'existe pas déjà dans la liste ouverte ou la liste fermée un nœud ayant la même position X et Y, le même sens et la même orientation ayant un coût inférieur.

5.2.2 Recherche de chemin par A* sous contrainte géométrique : exemples applicatifs

Pour tester cet algorithme de recherche de chemin, prenons l'exemple de la circulation dans un parking. Un parking du site de Vélizy du Groupe PSA, voir Figure 5.13 (a), est représenté de manière schématique par une grille d'occupation de 1 m x 1 m sur la Figure 5.13 (b). Même si des sens de circulation sont visibles sur la première figure, on considère ici que le véhicule peut se déplacer dans toutes les allées librement.



(a) (b)

FIGURE 5.13 – Vue satellite du parking du site PSA de Vélizy (a) et représentation sous Matlab

Afin d'explorer l'environnement, le A* utilise deux fonctions. Une fonction g modélisant la distance parcourue et une fonction heuristique h représentant la distance restante au point d'arrivée. Ici, la fonction h est la distance euclidienne. Ainsi, pour deux points s_1 et s_2 de la grille :

$$h(s_1, s_2) = \sqrt{(s_1.X - s_2.X)^2 + (s_1.Y - s_2.Y)^2}. \tag{5.4}$$

La fonction g est définie pour représenter la distance parcourue tout en pénalisant les changements d'orientation. En effet, étant donné que les déplacements en diagonale vont permettre au A* de se rapprocher beaucoup plus rapidement du point d'arrivée, utiliser uniquement la distance parcourue pour la fonction g va entraîner beaucoup de changements d'orientation. Pour pénaliser ces changements d'orientation, la distance à ajouter pour calculer la valeur de la fonction g au point s va dépendre de son orientation et de l'orientation de son parent. Si l'orientation est identique, une distance $d_1 = \sqrt{2}$ est ajoutée. Sinon c'est une distance $d_2 = 2$. Le choix de ces valeurs n'est pas anodin, il permet de conserver la propriété monotone de l'heuristique. En effet, pour n'importe quel nœud s de la grille et son parent s_p, l'inégalité

$$h(s_p, s_G) \leq d_i + h(s, s_G), \tag{5.5}$$

avec $i \in \{1; 2\}$, sera forcément respectée.

La Figure 5.14, compare le chemin trouvé avec le A* pour deux fonctions g différentes avec un exemple simple. Les points de départ et d'arrivée sont marqués par des triangles vert △ et rouge △ indiquant leurs orientations. Les points en violet sont les points de la trajectoire calculée. Les orientations de 45°, −45°, 135° et −135° sont marquées par des croix, les autres par des triangles pointant vers la bonne direction. Sur la figure de gauche, le coût du chemin g est calculé comme étant la vraie distance euclidienne parcourue. Un phénomène d'oscillation apparaît alors, lié aux changements répétés d'orientation. Sur la figure de droite, avec la contrainte sur le changement d'orientation, le chemin généré est plus « naturel » .

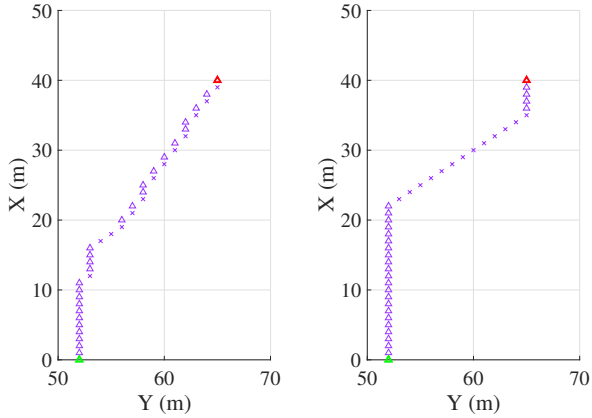


FIGURE 5.14 – Chemin trouvé avec g comme la vraie distance parcourue (a) et avec une contrainte en fonction du changement d'orientation (b)

Pour voir quelle trajectoire le A* calcule et pour évaluer ses performances en termes de nombre de points explorés et de temps de calcul, plusieurs scénarios sont définis et représentés sur la Figure 5.15. Dans tous les cas, la position de départ et l'orientation du véhicule sont identiques. Le premier scénario est un cas simple : le véhicule doit aller tout droit. Ceci va permettre d'analyser comment le A* explore l'environnement dans une grille de grande dimension. Pour le deuxième scénario, le véhicule doit se diriger dans la quatrième allée. Le troisième cas est plus complexe, car le véhicule doit arriver avec une orientation inverse. Enfin, pour le dernier cas, le véhicule va devoir se garer en haut à gauche du parking. Pour ces exemples, $n_p = 3$. Il est à noter que pour toutes les simulations suivantes, l'encombrement du véhicule est pris en compte.

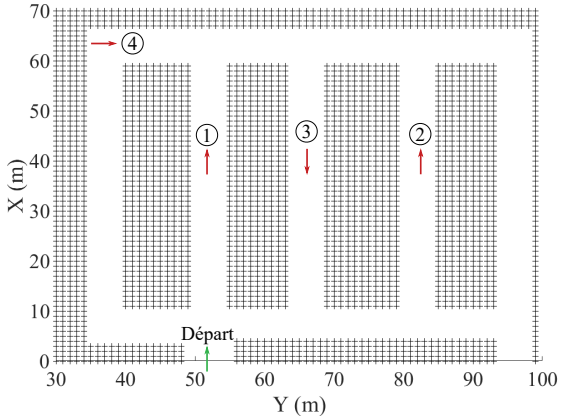


FIGURE 5.15 – Les scénarios testés pour l'évaluation du A*

Cas 1

Le premier cas est le cas le plus simple pouvant être rencontré dans un problème de planification : le véhicule doit aller tout droit en marche avant.

La Figure 5.16 montre le résultat de la recherche de chemin pour deux configurations différentes du A*. Sur cette figure, ainsi que sur les prochaines, le chemin final est tracé en violet. Les cercles cyan représentent les nœuds de la grille qui ont été explorés. Il est à noter que deux

nœuds explorés peuvent avoir la même position, mais pas la même orientation. Par conséquent, un cercle cyan représente potentiellement plusieurs nœuds de la liste fermée.

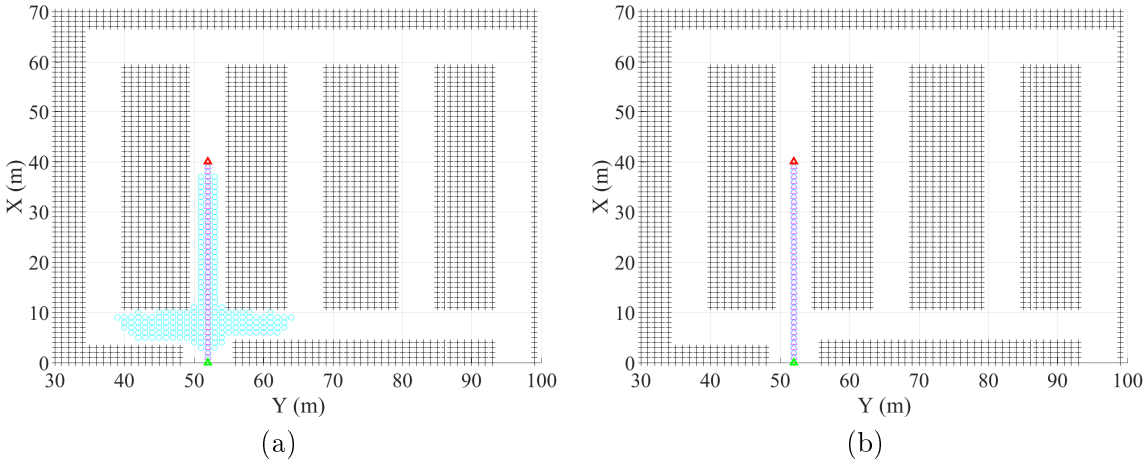


FIGURE 5.16 – Cas 1 avec $d_1 = \sqrt{2}$ et $d_2 = 2$ (a), et $d_1 = 1$ et $d_2 = \sqrt{2}$ (b)

Les paramètres de la première configuration, Figure 5.16 (a), sont $d_1 = \sqrt{2}$ et $d_2 = 2$, comme indiqué précédemment. Pour un parcours de 40 m longitudinalement, la planification de chemin dans cette configuration explore 470 nœuds pour un temps de calcul complet de 0,63 s. Le tableau 5.1, à la fin des exemples, récapitule les configurations et résultats de chaque scénario. Pour un chemin simple, mais dans une grille étendue, le A* explore donc largement la grille d'occupation avec dix fois plus de points que le chemin final. Étant donné que ce cas est le plus simple possible, on peut supposer que pour des cas plus complexes, avec des virages ou des inversions de sens, le temps de calcul sera trop important. Le temps de calcul de la trajectoire est critique dans le cadre de la conduite autonome. Il faut que le véhicule soit capable de définir son chemin ou sa trajectoire rapidement pour ne pas se retrouver immobilisé et ne pas gêner les autres véhicules. Un moyen de réduire le temps de calcul du A* est de diminuer la valeur des paramètres d_1 et d_2. Ainsi l'heuristique h aura un poids plus important dans la fonction de coût que le chemin parcouru g et le A* ira plus rapidement vers le point cible. Cependant, avec cette modification, le A* le sera plus monotone et le chemin final ne sera pas forcément optimal. Il y a donc un dilemme entre temps de calcul et optimalité du chemin à évaluer et définir.

L'exemple 2 du premier cas, Figure 5.16 (b), utilise les paramètres $d_1 = 1$ et $d_2 = \sqrt{2}$. Le principe de pénaliser le changement d'orientation est donc conservé, mais limité pour favoriser une recherche plus rapide vers le point désiré. Avec cette configuration, le A* explore ainsi exactement le même nombre de points que le chemin final. Il ne se disperse plus dans l'environnement comme précédemment. Les paramètres d_1 et d_2 choisis ici sont ceux les plus proches des valeurs précédentes avec une exploration minimale de l'environnement. Le but étant de garder le même ordre de grandeur des paramètres précédents qui garantissaient un chemin optimal.

Cas 2

Dans le deuxième scénario, le véhicule doit se déplacer dans la quatrième allée. Le but ici est d'analyser à quel point le véhicule explore l'environnement pour se déplacer dans un environnement complexe. Pour arriver à son but, l'algorithme va explorer 837 points différents pour un temps de calcul total de 1,41 s. Ici, l'algorithme va explorer la moitié des deuxième et

troisième allées, car dans un premier temps, ce sont celles qui vont rapprocher le véhicule le plus rapidement du point d'arrivée, voir Figure 5.17. Cependant, étant donné le nombre important de points occupés, l'algorithme va s'étendre horizontalement pour atteindre la quatrième allée. Une fois celle-ci atteinte, le chemin restant est facilement calculé.

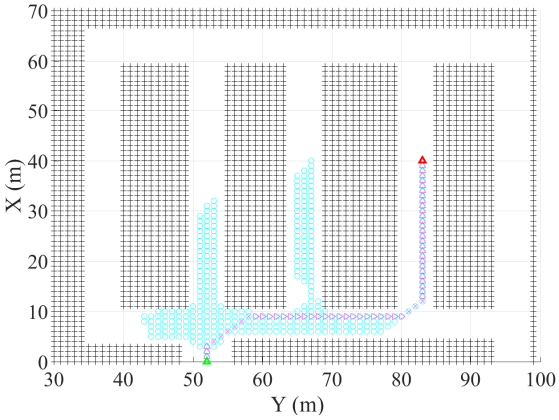


FIGURE 5.17 – Cas 2 avec $d_1 = 1$ et $d_2 = \sqrt{2}$

Cas 3

Le troisième scénario est plus complexe : le véhicule doit arriver dans la troisième allée dans l'orientation opposée de son orientation de départ. Le véhicule doit donc trouver un chemin faisant le tour de la deuxième rangée ou effectuer une manœuvre dans l'allée. Pour voir si l'algorithme arrive facilement à faire le tour du parking, la première simulation inhibe la marche arrière pour éviter que le véhicule fasse une manœuvre. Ainsi, sur la Figure 5.18 (a), l'algorithme cherche dans toutes les allées avant de trouver la position finale pour un total de 2365 points explorés et 7,21 s de temps de calcul. Le temps de calcul devient donc important et n'est pas admissible.

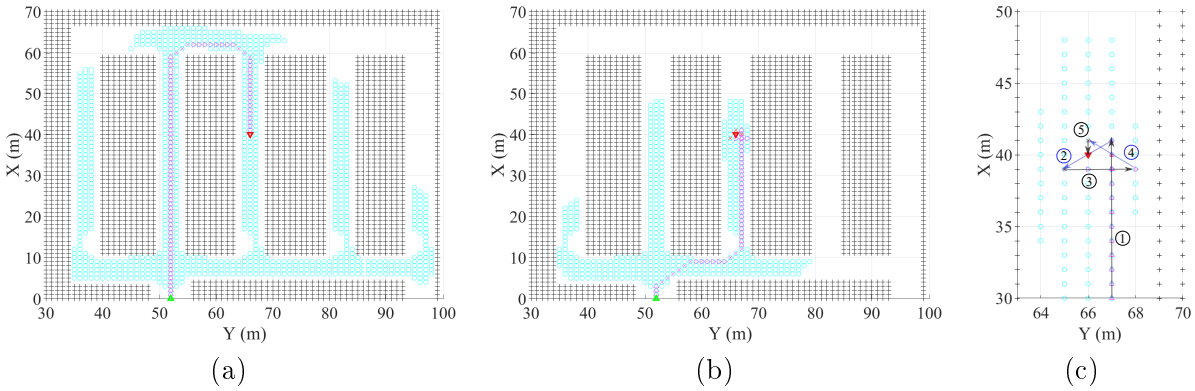


FIGURE 5.18 – Cas 3 avec $d_1 = 1$ et $d_2 = \sqrt{2}$, sans marche arrière (a), avec marche arrière (b) et zoom autour de la manœuvre de (b) (c)

La Figure 5.18 (b) montre le résultat de la recherche de chemin, mais avec la marche arrière autorisée. En faisant demi-tour dans l'allée, l'algorithme a ainsi exploré deux fois moins de points. Le détail de la manœuvre est zoomé sur la Figure 5.18 (c). Les flèches noires indiquent

une marche avant et les flèches bleues une marche arrière. Les flèches sont numérotées dans l'ordre d'application.

Cas 4

Dans le dernier scénario, le A* doit calculer un chemin pour que le véhicule effectue une manœuvre : qu'il se gare en marche arrière. En observant les points explorés sur la Figure 5.18, on aperçoit que l'algorithme essaie d'abord d'atteindre le point d'arrivée en passant par la deuxième allée en allant tout droit. Cependant, en passant par cette allée, le véhicule devra faire une manoeuvre pour arriver à la position désirée. Le nombre de changements nécessaires pour changer d'orientation devient alors trop pénalisant pour la fonction de coût. Le premier chemin trouvé passe donc par la première allée et n'effectue qu'une seule marche arrière, pour se garer, ce qui est plus proche du comportement conducteur.

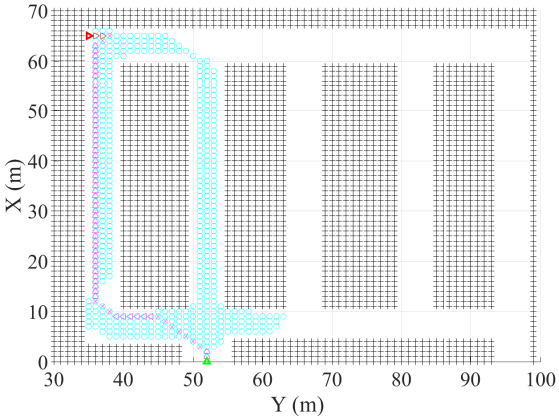


FIGURE 5.19 – Cas 4

	Cas 1 (a)	Cas 1 (b)	Cas 2	Cas 3 (a)	Cas 3 (b)	Cas 4
d_1	$\sqrt{2}$	1	1	1	1	1
d_2	2	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{2}$
Temps de calcul (s)	0,63	0,14	1,41	7,21	4,34	4,38
Nombre de nœuds explorés	470	41	837	2365	1221	1235

TABLE 5.1 – Configuration et résultats des différents scénarios présentés

Bilan : avantages et problématiques

L'algorithme A* présenté ici permet d'explorer un environnement statique représenté par une grille régulière tout en respectant les contraintes géométriques du rayon de courbure minimum admissible, avec des manœuvres en marche arrière si nécessaire. Les avantages de cette méthode sont le calcul d'un chemin admissible par le véhicule et la propriété de complétude. En effet, avec la représentation de l'environnement par une grille régulière, la recherche de chemins par A* est sûre de trouver une solution, si elle existe.

Cependant, comme nous l'avons vu avec les différents scénarios présentés, le temps de calcul peut s'avérer très important pour des chemins complexes dans un environnement contraint. De plus, tel quel, l'algorithme ne permet pas de gérer les sens de circulation de la route et le positionnement dans les voies. Ces informations pourraient cependant être portées par la grille de

l'environnement, ou par l'algorithme de recherche de chemin.

Lorsqu'un conducteur se déplace en voiture manuellement, son positionnement dans l'infrastructure est conditionné à la fois par le Code de la route et par l'infrastructure elle-même. Il est donc possible de définir des points de passage admissibles. L'ensemble de ces points de passage va donc former une grille plus « macroscopique » de l'environnement. En déterminant dans un premier temps les points de passage par lequel doit passer le véhicule pour effectuer sa trajectoire globale, on va ainsi découper le problème de recherche de chemin en plusieurs sous-problèmes. Le A* sera alors utilisé entre chaque point de passage pour calculer la trajectoire finale.

5.2.3 Dijkstra

Principe et algorithme

L'algorithme de Dijkstra permet de résoudre le problème du plus court chemin dans un graphe à partir d'un nœud de départ et pour tous les nœuds possibles. Cet algorithme peut se rapporter à l'algorithme A* avec une heuristique nulle. Ainsi, seule la distance entre chaque nœud, c'est-à-dire le poids de chaque lien, est considérée. Comme pour le A*, le poids entre deux nœuds s_1 et s_2 s'écrit $d(s_1, s_2)$. À l'origine, l'algorithme de Dijkstra n'a pas de nœud d'arrivée en entrée, l'algorithme se termine une fois tous les nœuds du graphe visités. Cela permet d'avoir tous les chemins possibles pour un point de départ.

Sachant un noeud de départ s et un graphe orienté ou non à liens positifs G, le pseudo-code de l'algorithme de Dijkstra s'écrit de la manière suivante.

Algorithme 2 Fonction Dijkstra

Entrée : s_S, G
Sortie : *parent*

Création d'un tableau *distance*
Création d'un tableau *parent*
Création d'une liste Q contenant tous les nœuds de G
Pour Tous les nœuds s dans G **faire**
  Initialiser $distance[s] = \infty$
  Initialiser $parent[s] = -1$
Fin Pour
$distance[s_S] = 0$
Tant que Q n'est pas vide **faire**
  Trouver le nœud u dans Q avec la plus petite *distance*
  Enlever u de Q
  Pour Chaque voisin v de u **faire**
    Calculer $alt = distance[u] + d(u, v)$
    Si $alt < distance[v]$ **alors**
      Définir $distance[v] = alt$
      Définir $parent[v] = u$
    Fin Si
  Fin Pour
Fin Tant que
Retourner *parent*

Objectif dans le cadre de la conduite autonome

L'objectif du Dijkstra dans le cadre de la conduite autonome est de déterminer les points de passage par lesquels devra passer le véhicule pour se déplacer dans l'environnement. Pour franchir une intersection ou un rond-point, un conducteur s'appuie sur le Code de la route, les marquages au sol et la disposition des voies. En utilisant ces informations, l'environnement peut être découpé en plusieurs zones. Ce sont ces zones qui constitueront les différents points de passage. Les points de passage forment alors un graphe de l'environnement à plus grande échelle que la grille d'occupation utilisée précédemment. Ici, les objets dynamiques ne sont pas considérés. L'infrastructure de la route est uniquement utilisée. C'est le A*, en utilisant la grille d'occupation, qui s'occupera dans un deuxième temps de trouver un chemin admissible tout en évitant les obstacles.

En plus de trouver un chemin conforme au code de la route et l'infrastructure, le fait d'utiliser un sous-graphe pour découper le problème dans un premier temps devrait permettre de réduire le temps de calcul. En effet, la distance entre deux points de passage sera plus faible, il y aura donc moins de points à explorer.

Pour pouvoir utiliser cette méthode, il faut définir les règles permettant le découpage de l'environnement. Pour créer le graphe nécessaire au Dijkstra, il faut donc positionner correctement les points de passage dans l'environnement, définir les liens entre ces nœuds et déterminer les poids de chaque lien.

5.2.4 Cas applicatif : le carrefour giratoire

Dijkstra en carrefour à sens giratoire

Le passage d'un carrefour à sens giratoire, plus communément appelé rond-point par abus de langage, est conditionné par le nombre de voies intérieures et la sortie vers laquelle va se diriger le véhicule. Ainsi, suivant si le véhicule va tout droit ou à gauche, il va se positionner plus ou moins proche du centre du rond-point et y rester plus ou moins longtemps. L'algorithme de Dijkstra va permettre, à partir de points de passage positionnés au préalable, de respecter le Code de la route pour franchir un rond-point.

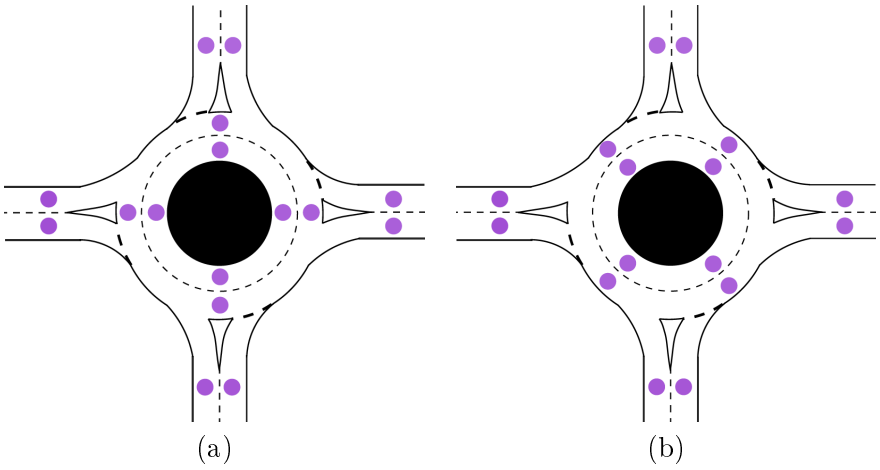


FIGURE 5.20 – Positionnement des points de passage pour un rond-point à deux voies. Configuration en ligne (a) et en diagonale (b)

La première étape est donc de déterminer comment positionner ces points de passage. Prenons le cas d'un carrefour giratoire avec deux voies de circulation internes, voir Figure 5.20. Logiquement, des points sont positionnés à chaque voie d'entrée et de sortie du carrefour. Ensuite, il faut placer les points de passage internes. Une solution est de les placer « en ligne », c'est-à-dire dans l'axe des routes qui arrivent au carrefour, comme sur la Figure 5.20 (a). L'autre solution consiste à positionner les points de passage « en diagonale », au milieu des intersections des routes, voir Figure 5.20 (b). Le problème de la première solution est qu'elle ne permet pas un franchissement du carrefour conforme au Code de la route si le véhicule prend la troisième sortie ou fait demi-tour. En effet, pour passer la troisième sortie d'un carrefour giratoire à deux voies, il faut que le véhicule se positionne sur la voie interne, puis se déporte sur la voie externe après avoir passé la deuxième sortie. Or, les points de passages de la Figure 5.20 (a) favorisent un rabattement avant le passage de la deuxième sortie. Le positionnement de la Figure 5.20 (b) est donc privilégié, d'autant plus qu'il conduit à une solution avec un point de passage en plus que le positionnement en ligne. Pour un nombre de noeuds équivalent, l'environnement est donc mieux découpé en faveur du code de la route et de la recherche de chemin avec plus de points intermédiaires et donc moins de temps de calcul.

Ensuite, il s'agit de définir les liens entre les nœuds du graphe. En effet, pour pouvoir définir correctement des points de passage, tous les nœuds ne peuvent pas être reliés entre eux. Ainsi, les nœuds sur les voies d'entrée peuvent accéder aux deux nœuds les plus proches dans le carrefour giratoire et les voies de sortie ne sont accessibles que par les deux nœuds du carrefour les plus proches de la sortie, voir Figure 5.21 (a). Les liens de la Figure 5.21 (b) constituent les moyens de circulation interne dans le carrefour giratoire sans changement de voie. Enfin, les changements de voie à l'intérieur du carrefour giratoire sont représentés par les liens sur la Figure 5.21 (c).

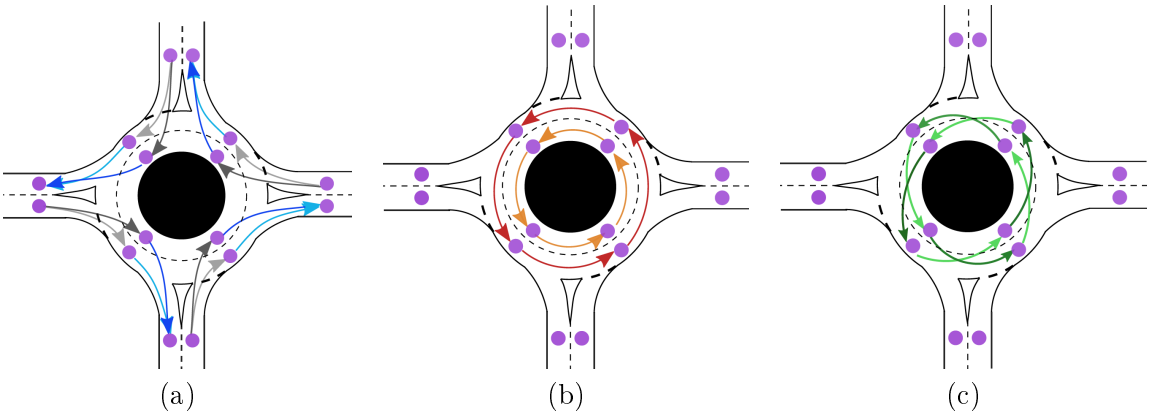


FIGURE 5.21 – Liens dans le graphe du rond-point à deux voies, pour les entrées et sorties (a), pour les maintiens dans les voies (b) et les changements de voie (c)

Enfin, la dernière étape de constitution du graphe est de pondérer les liens définis précédemment. C'est l'étape la plus importante, car c'est celle qui va conditionner le choix des points de passage finaux. Pour les liens d'entrée et de sortie du carrefour giratoire, comme il est plus facile d'entrer et de sortir du carrefour par la voie externe, ces transitions sont celles avec la pondération la plus faible, voir Figure 5.21 (a). Sur la circulation sur la même voie, à l'intérieur du carrefour, comme le fait de passer par la voie interne est plus rapide, la pondération des liens internes est deux fois plus faible que ceux externes, voir Figure 5.21 (b). Enfin, une fois dans le carrefour giratoire, il est plus facile de sortir du rond-point que d'y rentrer. Ainsi, la transition de la voie externe à la voie interne est fortement pénalisée alors que celle de la voie interne à la

voie externe est favorisée, voir Figure 5.21 (c).

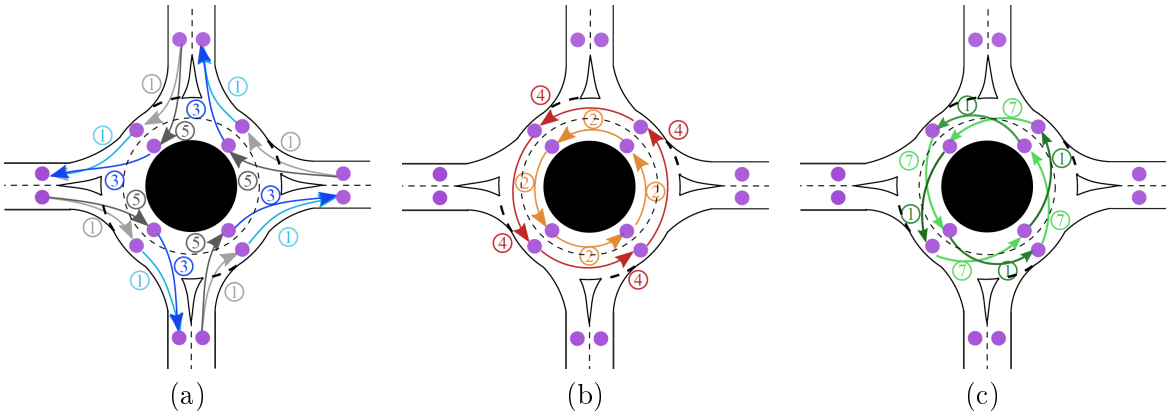


FIGURE 5.22 – Pondérations des liens dans le graphe du rond-point à deux voies, pour les entrées et sorties (a), pour les maintiens dans les voies (b) et les changements de voie (c)

La même démarche peut être effectuée pour les carrefours à sens giratoire à trois voies. Pour ce type de carrefour, il y a cependant deux voies d'entrée ce qui oblige à rajouter un point de passage dans la voie de droite de la route en entrée du carrefour, voir Figure 5.23. Il faut donc rajouter des liens d'entrée dans le carrefour giratoire.

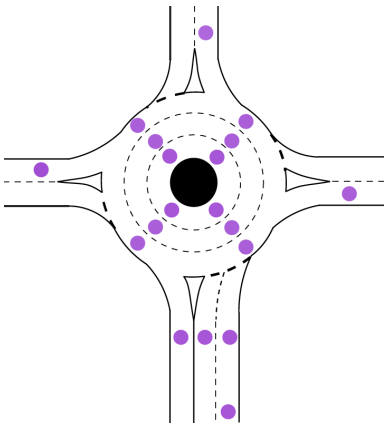


FIGURE 5.23 – Positionnement des points de passage pour un rond-point à trois voies

Ainsi, pour entrer sur la voie du milieu ou la voie interne dans le carrefour, il faut forcément être positionné à gauche sur la route d'entrée, voir Figure 5.23 (a). De même, on considère qu'il n'est pas possible de sortir du rond-point directement depuis la voie la plus au centre s'il y a plus de deux voies. Les pondérations sont identiques aux précédentes. On limite cependant la pondération de la voie interne, car elle est déjà très fortement pondérée en amont. Pour la circulation sur les voies internes, Figure 5.23 (b). Comme précédemment, plus la voie est proche du centre, moins les pondérations sont élevées, avec un rapport de deux entre les voies. Enfin, les transitions entre les voies à l'intérieur même du carrefour ont aussi des poids identiques. Il n'y a pas de liens directs entre les voies extrêmes.

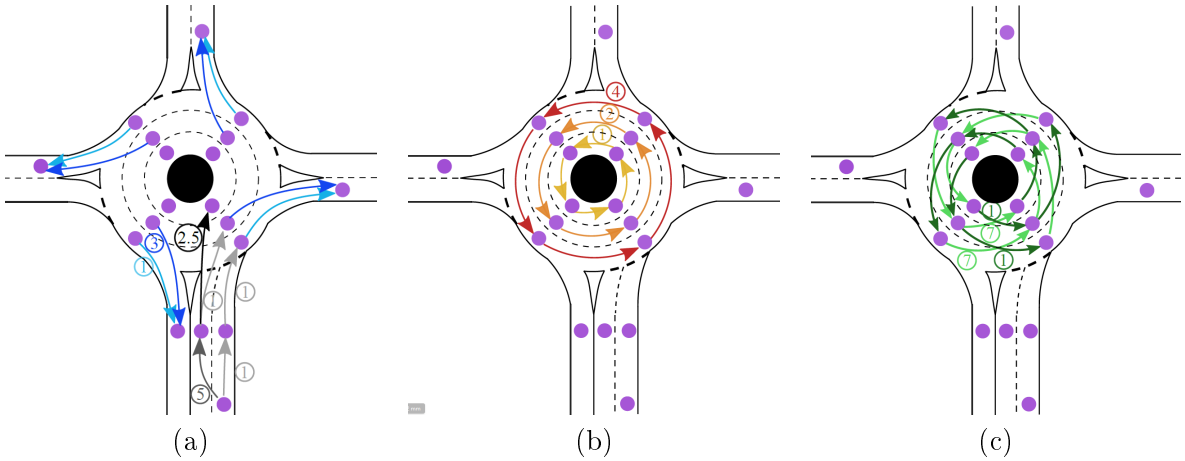


FIGURE 5.24 – Les liens et pondérations dans le graphe du rond-point à trois voies, pour les entrées et sorties (a), pour les maintiens dans les voies (b) et les changements de voie (c)

Pour les deux graphes pondérés définis précédemment pour des carrefours à sens giratoire à deux et trois voies, les résultats de l'algorithme de Dijkstra sont tracés sur la Figure 5.25. Ces figures montrent, à partir du point de départ en bas, quels sont les liens à prendre pour atteindre chaque nœud du graphe avec la pondération la plus faible. Ainsi, dans les deux cas, pour aller tout droit ou à droite, le véhicule doit bien passer par la voie externe et entre à l'intérieur du rond-point avant de se rabattre pour aller à gauche. Dans le cas du carrefour giratoire à trois voies, le véhicule s'insère dans la voie la plus au centre.

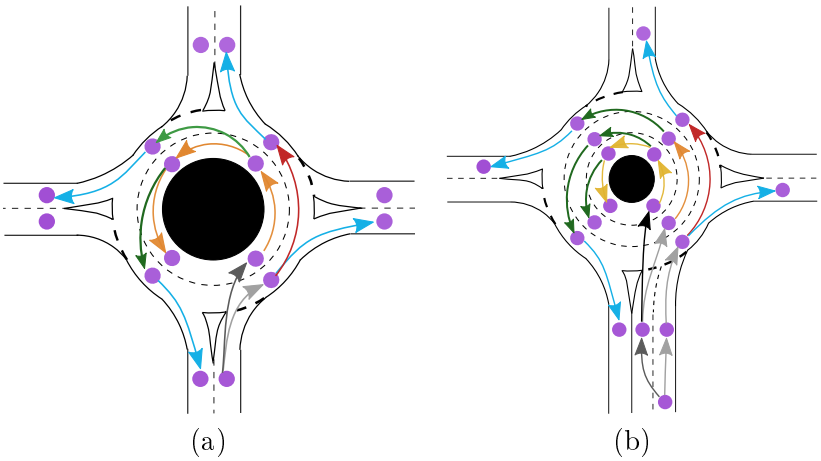
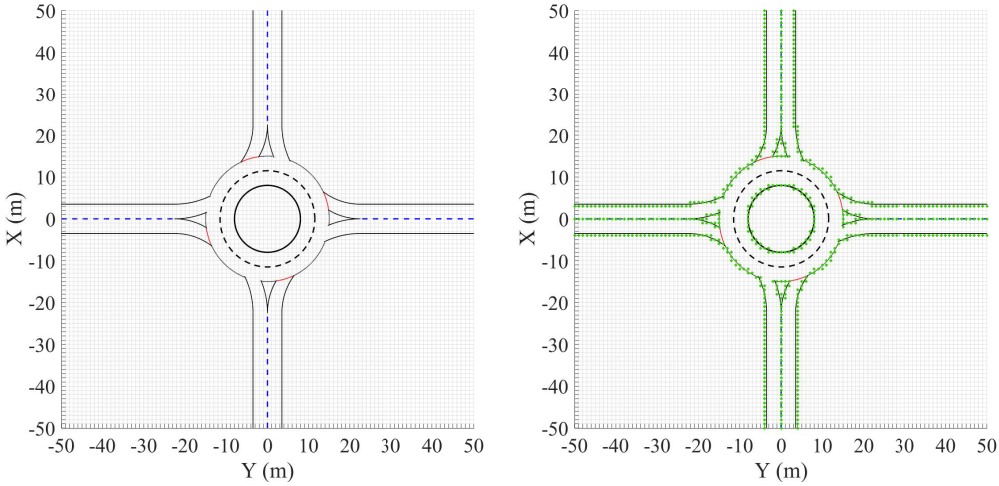


FIGURE 5.25 – Résultats du Dijkstra pour les carrefours à sens giratoire à deux voies (a) et trois voies (b)

L'intérêt de cette méthode est qu'elle est valide pour tout type de rond-point. Ici, des ronds points à quatre sorties avec deux ou trois voies internes ont été présentés. En réalité, ces paramètres varient et le positionnement des sorties n'est pas forcément régulier. Comme cette méthode fonctionne pour tout type de configuration, il n'est pas nécessaire d'enregistrer au préalable les points de passage pour chaque type de rond-point, au risque d'en oublier ou de rencontrer un cas particulier.

Génération de trajectoire par combinaison des deux algorithmes

Maintenant que les points de passage dans l'infrastructure sont définis, il est possible de les combiner avec l'algorithme A* décrit dans la section précédente. Pour nous rapprocher le plus possible d'un carrefour avec des dimensions réelles, les normes de construction de [50] sont utilisées pour représenter le carrefour sous Matlab. Ainsi, pour un rayon extérieur de 15 m, avec deux voies intérieures, la Figure 5.26 (a) trace le carrefour giratoire considéré pour la recherche de chemin. La grille d'occupation de 1 m par 1 m utilisée pour la recherche de chemin est représentée en verte sur la Figure 5.26 (b).



(a) (b)

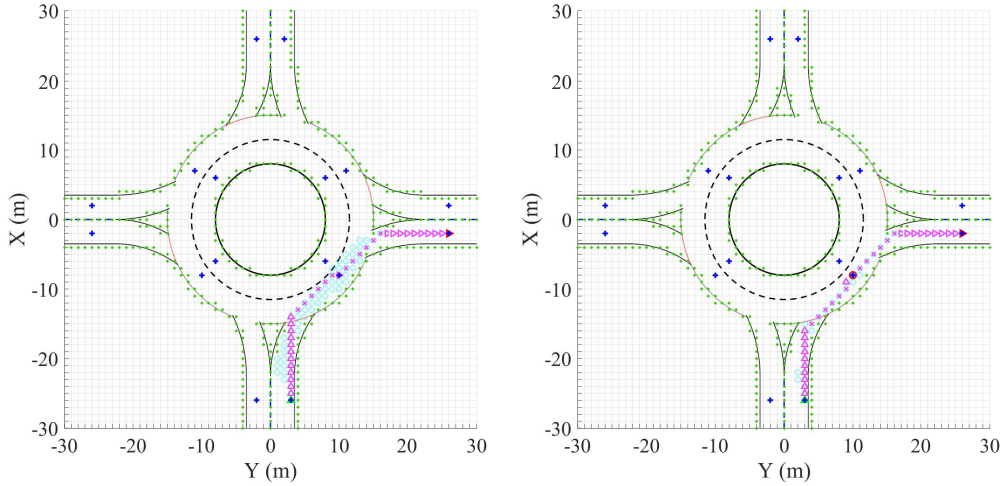
FIGURE 5.26 – Carrefour à sens giratoire (a) et la grille d'occupation qui en résulte (b)

Les quatre figures suivantes, 5.27, 5.28, 5.29 et 5.30 tracent, quant à elles, les résultats de la recherche de chemin avec et sans points de passage pour aller : à la première sortie à droite, la deuxième sortie en haut, la troisième sortie à gauche et la quatrième sortie en bas. La Table 5.2 récapitule ensuite les résultats pour chacune des simulations avec le temps de calcul et le nombre de nœuds explorés dans la grille pour trouver le chemin.

Pour le virage à droite, les résultats avec et sans points de passage sont assez similaires, car le chemin à trouver est relativement court. Sans points de passage, l'algorithme A* explore 2,5 fois plus de points, mais réussit à trouver un chemin assez proche, Figure 5.27.

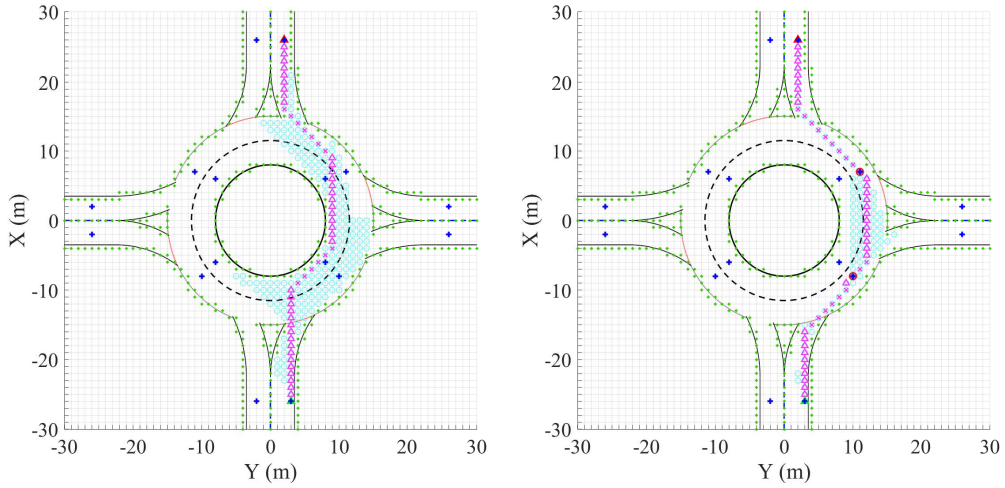
L'intérêt des points de passage commence à être visible pour la recherche de chemin vers la deuxième sortie, Figure 5.28. En effet, sans ces points intermédiaires, le chemin trouvé passe par la voie interne du rond-point ce qui n'est pas conforme aux règles de conduite, voir Figure 5.28 (a). En mettant des positions intermédiaires, Figure 5.28 (b), le chemin passe bien dans la voie extérieure du rond-point.

Pour les troisième et quatrième sorties, l'avantage d'utiliser un graphe macroscopique concerne plus le gain en temps de calcul que le résultat final. Les chemins trouvés sont similaires, voir Figures 5.29 et 5.30. Cependant, le recours aux points intermédiaires aide grandement la recherche de chemin dans ces deux cas. Pour la sortie à gauche, le nombre de nœuds explorés est divisé d'environ 4,5. Sans points intermédiaires, le temps de calcul est alors supérieur à 2 s alors qu'il reste du même ordre que les sorties précédentes en cas de découpage du problème. Pour le chemin le plus complexe de la quatrième sortie, quasiment toute la grille est explorée avant de trouver le point d'arrivée pour l'algorithme A* classique. Cela conduit à un temps de calcul



(a) (b)

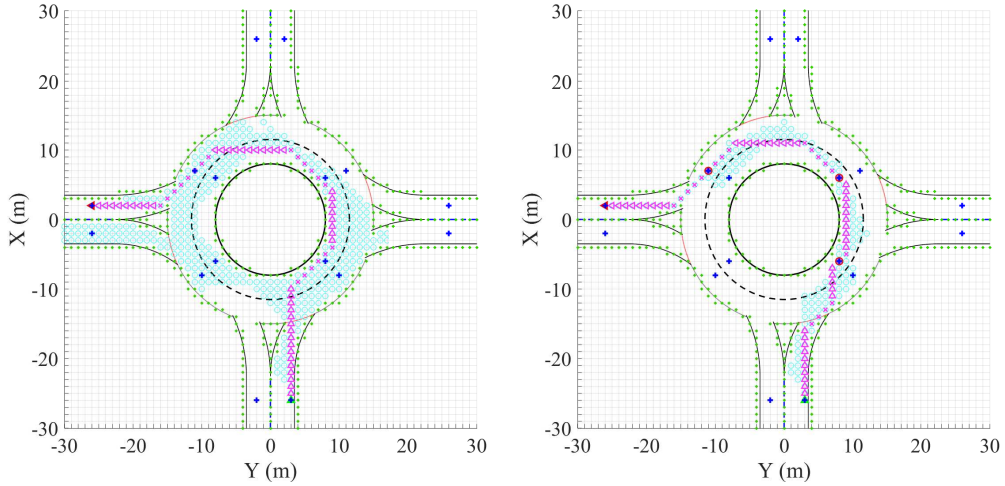
FIGURE 5.27 – Calcul du chemin vers la première sortie sans (a) et avec (b) points de passage



(a) (b)

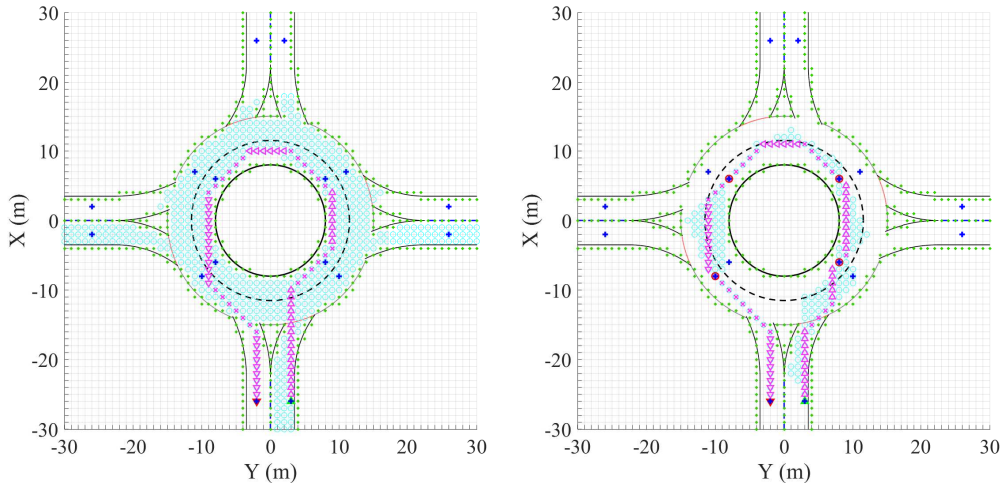
FIGURE 5.28 – Calcul du chemin vers la deuxième sortie sans (a) et avec (b) points de passage

total de plus de 8 s ce qui n'est pas admissible dans le cadre de la conduite autonome. Avec l'ajout du Dijkstra, 8 fois moins de nœuds sont ajoutés dans la liste fermée.



(a) (b)

FIGURE 5.29 – Calcul du chemin vers la troisième sortie sans (a) et avec (b) points de passage



(a) (b)

FIGURE 5.30 – Calcul du chemin vers la quatrième sortie sans (a) et avec (b) points de passage

	Droite		Haut		Gauche		Bas	
Points de passage	non	oui	non	oui	non	oui	non	oui
Temps de calcul (s)	0,33	0,25	0,71	0,32	2,41	0,43	8,14	0,49
Nombre de nœuds explorés	108	44	330	128	907	212	1879	233

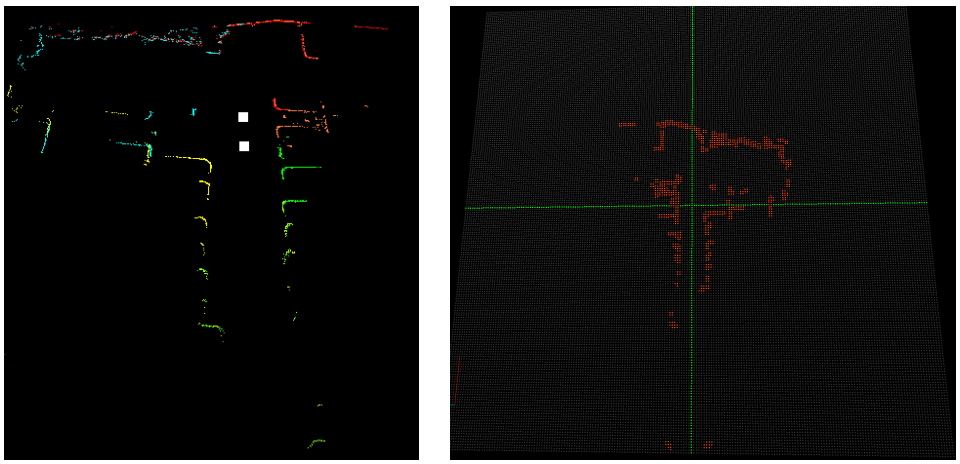
TABLE 5.2 – Résultats du passage de carrefour giratoire pour les différentes sorties

L'utilisation de points intermédiaires pour le calcul du chemin à suivre par le véhicule est donc une manière efficace pour réduire le temps de calcul. De plus, positionnés correctement, ces points de passage peuvent aussi être utilisés pour respecter le Code de la route pour le passage de ronds-points ou d'intersections. Sachant que les prototypes disposent de cartes HD ou d'un équivalent, il est tout à fait possible d'imaginer une utilisation de cette méthode embarquée dans le véhicule avec le positionnement des points de passage en temps réel.

5.2.5 Résultats sur prototype en parking

L'algorithme A* tenant compte des contraintes géométriques a été implanté dans les calculateurs du prototype HZTR. Les capteurs lidars (un frontal et quatre dans les coins) sont utilisés pour construire la grille d'occupation. L'algorithme de recherche de chemin a été testé dans un parking du site PSA de Vélizy, voir Figure 5.13. Pour générer la grille d'occupation nécessaire pour la recherche de chemin, les informations des capteurs lidars sont utilisées. Le prototype effectue ensuite un tour du parking pour reconstruire entièrement la grille.

À l'instant t_0 de l'enregistrement de la grille, la Figure 5.31 montre le nuage de points envoyé par les capteurs. Sur celle de gauche, chaque couleur correspond à un capteur et tous les points reçus sont tracés. La figure de gauche représente la grille d'occupation de résolution 0,5 m par 0,5 m qui découle des points des lidars.



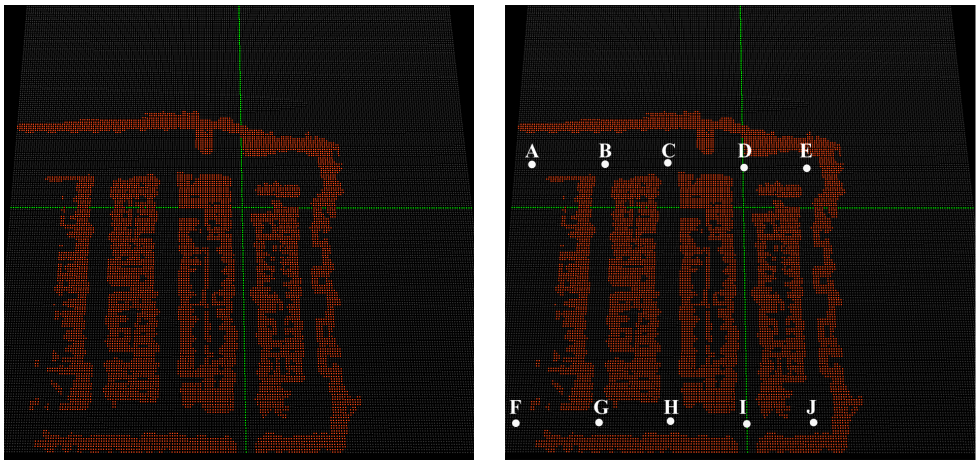
(a) (b)

FIGURE 5.31 – Nuage de points de capteurs lidar (a) et la grille d'occupation résultante

La grille complète du parking enregistrée par les capteurs est ensuite visible sur la Figure 5.32 (a). Comme nous l'avons vu dans le cas des ronds-points, le fait d'ajouter des points de passage augmente l'efficacité de la recherche de chemin en termes de temps de calcul. Des points de passage sont donc ajoutés dans la grille du parking à chaque intersection d'allée. Ces points intermédiaires sont représentés sur la Figure 5.32 (b). Pour tester l'algorithme développé précédemment, on choisit un scénario où le point de départ est au niveau du point « I » et l'arrivée est dans l'allée entre les points « H » et « C » avec un cap orienté vers le bas ↓. On impose les points de passage « D », « E », « J », « I », « H », « G », « F », « A » et « B ». Le résultat est visible sur la Figure 5.33.

La trajectoire finale est conforme aux contraintes géométriques imposées pour avoir un chemin admissible concernant le rayon de courbure minimum du véhicule. L'algorithme proposé est capable de générer un tel chemin très rapidement, en 190 ms, et peut donc être utilisé en temps réel dans le véhicule. La contrainte majeure de la recherche de chemin qu'est le temps de calcul, dans le cadre de la conduite autonome, est donc respectée.

L'algorithme, comme présenté jusqu'ici, prend uniquement en compte les objets statiques. Dans la section suivante, la méthode est améliorée pour prendre en compte des objets dynamiques dans des scénarios urbains.



(a) (b)

FIGURE 5.32 – Grille complète du parking par les informations capteurs (a) et points de passages associés (b)

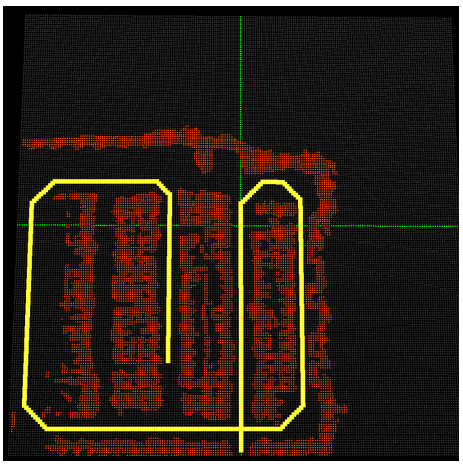


FIGURE 5.33 – Résultat de la recherche de chemin dans le parking de l'ADN

5.3 Génération de trajectoire en environnement dynamique

La méthode de recherche de chemin en environnement statique a été validée tant en simulation qu'en pratique sur les véhicules. En dehors du cas du parking, l'environnement n'est en réalité jamais statique. En environnement urbain, que ce soit sur les ronds-points, les intersections ou les tronçons à double sens de circulation, le véhicule autonome doit être capable d'interagir avec les autres utilisateurs de la route. Le nombre de situations, ou scénarios, en fonction du type de route et des utilisateurs étant extrêmement grand, l'étude de cette section se focalise sur un type de scénario particulier : la circulation sur route à double sens de circulation. Dans ces scénarios, le véhicule devra être capable de doubler un vélo ou un autre véhicule plus lent. Une évolution de ce scénario consiste à ajouter un véhicule arrivant en sens inverse.

5.3.1 Intégration de la dimension temporelle pour la recherche de trajectoire

Le fait de considérer non plus un environnement statique, mais un environnement dynamique introduit la notion de temps dans la recherche de chemin. Ce n'est donc plus une recherche de

chemin, mais une recherche de trajectoire. Pour calculer cette trajectoire, il est nécessaire d'ajouter une dimension temporelle à l'algorithme A*. Ainsi, deux changements majeurs interviennent pour la recherche de trajectoire en environnement dynamique :

- Modification de la grille d'occupation $\Longrightarrow$ Grille d'occupation temporelle
- Modification de la fonction de coût $\Longrightarrow$ Intégration d'un coût lié à la proximité des objets dynamiques

Grille d'occupation temporelle

Dans le paragraphe sur la recherche de chemin en environnement statique, la grille d'occupation est uniquement constituée de 0 et de 1 pour indiquer si la case est occupée ou non. En fonction, le A* est autorisé à s'étendre sur la case. En environnement dynamique, les objets vont occuper des cases, mais uniquement à un certain instant. En soit, l'ego véhicule peut se déplacer sur cette case à un instant t tant qu'elle n'est pas déjà occupée par un autre objet au même instant. L'idée est donc d'utiliser une grille temporelle en trois dimensions. Les deux premières dimensions sont toujours les positions longitudinale et latérale des objets, la troisième indique si la case est occupée par un objet statique ou dynamique. Si cet objet est dynamique, elle indique à quel instant elle sera occupée. Ainsi, pour une grille temporelle $G(X, Y)$ créée à l'instant $t_0 = 0$ s :

- $G(X, Y) = t > 0$ indique que la case sera occupée par un objet dynamique à l'instant t
- $G(X, Y) < 0$ indique que la case est occupée par un objet statique

Nouvelle fonction de coût

En tenant compte de ces modifications dans la grille, il faut aussi modifier la façon dont le A* se déplace dans cette grille. La première étape est d'ajouter des attributs aux nœuds explorés. En plus de ceux listés précédemment, on ajoute pour un nœud s :

- temps pour arriver à s, $s.t$

Le temps de déplacement dépend de la vitesse longitudinale désirée et de la taille de la grille.

Pour se déplacer sur une case s de la grille, le A* doit donc faire en sorte que la case soit libre d'objet statique et que $s.t \neq G(s.X, s.Y)$. Cependant, cette contrainte sera quasiment toujours respectée, car la grille est discrète. Il faut donc un moyen d'inclure une contrainte sur le déplacement du A* sur les nœuds temporels. Pour cela, un autre terme est ajouté à la fonction de coût f :

$$f = h + g + g_t, \tag{5.6}$$

avec g_t la fonction de coût temporelle telle que :

$$g_t(s) = s.t + \frac{K_t}{|s.t - G(s.X, s.Y)|}, \tag{5.7}$$

avec K_t le gain de la fonction de coût.

Ainsi, plus un point exploré se rapproche d'un point ayant la même position et la même valeur temporelle, plus la fonction de g_t, et donc la fonction de coût f, augmente. Le déplacement sur ce point est donc contraint. Pour éviter que le A* ne soit trop contraint par les objets temporels, la contrainte suivante est ajoutée sur la fonction g_t :

- Si $|s.t - G(s.X, s.Y)| > t_{max}$: $K_t = 0$

- Sinon : K_t reste inchangé,

avec t_{max} la valeur du delta de temps entre objet et ego véhicule en dessous de laquelle la fonction de coût intègre une pondération temporelle à l'approche d'objets dynamiques. De cette manière, le A* pourra quand même favoriser les déplacements sur les cases temporelles non nulles.

Remarque. Il est à noter qu'avec la méthode décrite ci-dessus, l'exploration des nœuds de la grille se fait avec une vitesse longitudinale constante de l'ego véhicule. Ainsi, uniquement le déplacement latéral est traité. On considère alors que l'ACC, couplé avec les modules de génération de trajectoire et de décision, s'occupe de la dimension longitudinale.

5.3.2 Application : dépassement d'un vélo

Dans ce premier exemple applicatif, l'ego véhicule doit dépasser un vélo. Aucun véhicule ne roule en sens inverse. L'ego véhicule roule à 50 km/h et le vélo à 20 km/h. À l'instant, $t = 0$ s de création de la grille, le vélo a une position initiale $(X = 100, Y = 3)$ m, sachant que les bords de la route sont situés à $Y = 5$ m et $Y = -5$ m et sont considérés comme des points occupés sur lesquels le véhicule ne peut pas rouler. Ce scénario donne alors la grille d'occupation temporelle de la Figure 5.34. Sur cette figure, le déplacement du vélo est visible par l'augmentation du temps sur l'axe vertical.

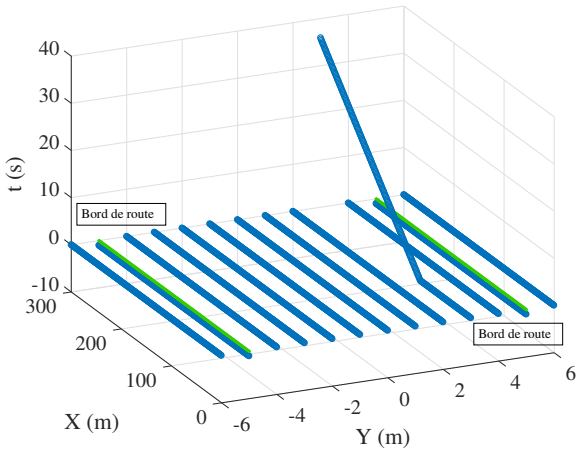


FIGURE 5.34 – Grille temporelle pour le dépassement d'un vélo

Un premier test est généré pour $K_t = 1$ et $t_{max} = 5$ s. Les résultats, à différents instants de déplacement, sont tracés sur la Figure 5.35. Sur ce test, 310 noeuds ont été explorés. Sachant que le chemin final compte 301 points, le A* n'a pas beaucoup dévié. Le temps de calcul pour trouver la trajectoire est de 0,56 s. Avec cette trajectoire, le véhicule se déplace de 1 m latéralement à $t = 8,5$ s pour éviter le vélo, puis se rabat dans sa voie à $t = 15,5$ s.

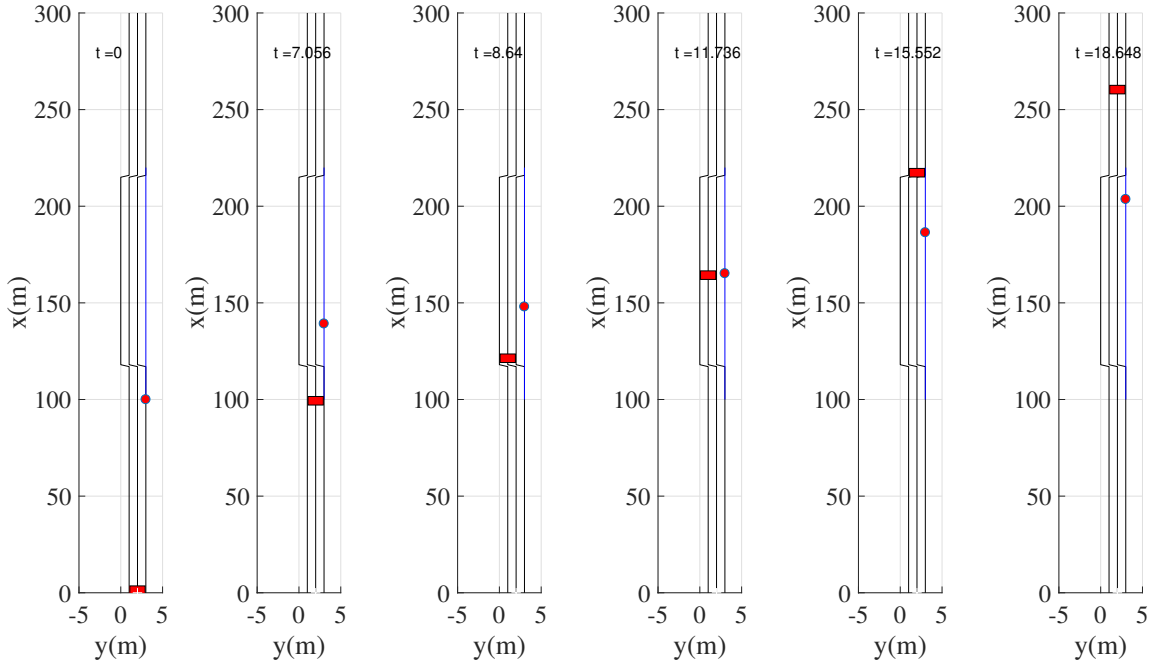


FIGURE 5.35 – Positionnement de l'ego véhicule (rectangle rouge) et du vélo (cercle rouge) à différents instants, $K_t = 1$ et $t_{max} = 5$ s

Il est ensuite intéressant, pour un même scénario, d'évaluer l'influence des paramètres t_{max} et K_t sur le moment de l'évitement et du rabattement. La Figure 5.36 montre l'évolution de la position latérale de l'ego véhicule en fonction de ces paramètres. Le temps entre l'ego véhicule et le vélo est aussi tracé pour évaluer le temps restant avant collision au moment de l'évitement.

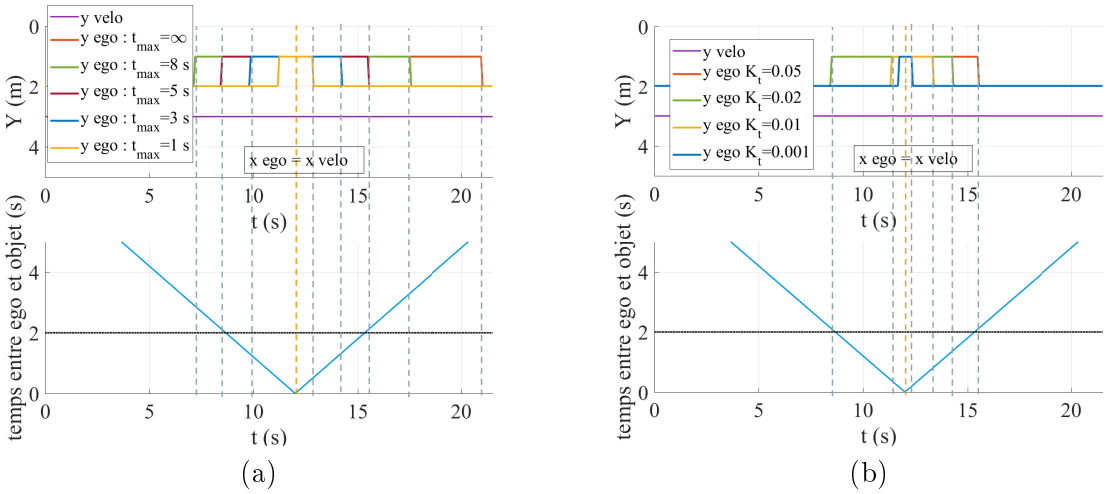


FIGURE 5.36 – Influence des paramètres du A* temporel sur le moment du changement de voie et du rabattement, en fonction de t_{max} (a) et K_t (b)

Le paramètre t_{max} influence à la fois le temps d'évitement et le temps de rabattement, Figure 5.36 (a). Lorsque la fonction de coût inclut systématiquement le coût temporel, *i.e* $t_{max} = \infty$, le véhicule ne se rabat que pour atteindre le point d'arrivée. Ce comportement n'est pas souhaitable, l'ego véhicule doit se rabattre dès qu'il en a la possibilité, d'où l'intérêt du paramètre t_{max}. Plus t_{max} est faible, plus le temps du dépassement est faible. Le dépassement est symétrique par

rapport à l'instant où l'ego véhicule et le vélo se croisent longitudinalement. Sur la Figure 5.36 (b), l'influence de K_t n'est visible que lorsque K_t diminue énormément. Il n'y a aucune différence pour des valeurs entre 1 et 0,05. En dessous de 0,05, K_t fait varier le moment de l'évitement beaucoup trop brusquement, la situation devient dangereuse pour l'ego véhicule et pour le vélo. À très faible valeur, le paramètre est trop sensible et est donc fixé à $K_t = 1$ pour la suite. À noter que dans tous les cas présentés ici, le nombre de nœuds dans la liste fermée est entre 301 et 310. Le temps de calcul est donc très proche : environ 0,5 s.

L'intégration d'un coût temporel lié au temps de déplacement des objets de l'environnement semble être une bonne solution pour la génération de trajectoire en environnement dynamique. Cependant, la trajectoire générée avec la méthode telle quelle, introduit des déplacements brusques. Ces déplacements géométriques conviennent à faibles vitesses, mais ne sont pas réalisables dynamiquement. Pour parvenir à une solution admissible en fonction de la vitesse, l'extension de la recherche des nœuds par A* Hybride [127] est utilisée.

5.3.3 Intégration des contraintes dynamiques dans le A*

Le A* Hybride [127], utilisé pour le DARPA Urban Challenge, étend les nœuds du graphe de manière itérative en respectant un certain rayon de courbure. De cette manière, la recherche de trajectoire se rapproche de la méthode des tentacules. La différence réside dans le nombre de branches, ou de tentacules, et que le A* Hybride calcule une trajectoire globale et la méthode des tentacules une trajectoire locale sur un horizon de temps très court, et donc avec moins de visibilité future.

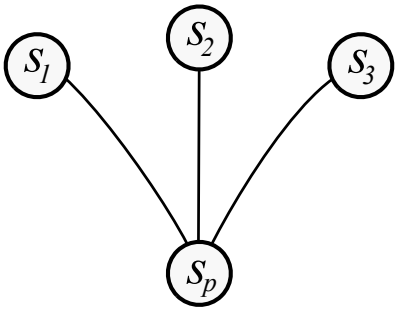


FIGURE 5.37 – Extension du A* Hybride

Comme précédemment, à partir d'un nœud parent s_S, trois nœuds peuvent être créés en marche avant, voir Figure 5.37. La position des nœuds résultants s_1, s_2 et s_3 dépend de la position et de l'orientation de s_S, de la vitesse longitudinale V_x, de l'accélération latérale engendrée par les virages A_y et de la discrétisation longitudinale, dans le repère véhicule, dX. Le modèle cinématique du véhicule est utilisé pour les calculs. Nous avons vu dans le deuxième chapitre que ce modèle n'est pas adapté pour les grandes vitesses longitudinales. Il reste cependant assez représentatif pour les cas en situation urbaine. Ainsi les dérivées des positions longitudinale X_G et latérale Y_G du centre de gravité du véhicule dans le repère absolu s'écrivent :

$$\begin{aligned} \dot{Y}_G(t) &= V_x \sin(\Psi(t)) \\ \dot{X}_G(t) &= V_x \cos(\Psi(t)), \end{aligned} \tag{5.8}$$

avec $\Psi(t)$ le cap du véhicule.

Avec une période d'échantillonnage dT, il devient possible d'écrire, pour l'instant k :

$$\begin{aligned}\dot{Y}_G(kdT) &= \frac{Y_G(kdT) - Y_G((k-1)dT}{dT}) \\ \dot{X}_G(kdT) &= \frac{X_G(kdT) - X_G((k-1)dT}{dT}),\end{aligned} \tag{5.9}$$

on a donc :

$$\begin{aligned}Y_G(kdT) &= \dot{Y}_G(kdT)dT + Y_G((k-1)dT) \\ X_G(kdT) &= \dot{X}_G(kdT)dT + X_G((k-1)dT).\end{aligned} \tag{5.10}$$

Ce qui donne en utilisant (5.8) :

$$\begin{aligned}Y_G(kdT) &= V_x dT \sin(\Psi(kdT)) + Y_G((k-1)dT) \\ X_G(kdT) &= V_x dT \cos(\Psi(kdT)) + X_G((k-1)dT).\end{aligned} \tag{5.11}$$

L'évolution de l'angle de lacet se fait en fonction de l'angle précédent et d'un échantillonnage $d\Psi$ dépendant de l'accélération latérale et de la vitesse longitudinale. Ainsi, sur la Figure 5.37, à partir de s_S, les nœuds s_1, s_2 et s_3 auront une orientation :

$$\begin{aligned}s_1.cap &= s_S.cap - d\Psi \\ s_2.cap &= s_S.cap \\ s_3.cap &= s_S.cap + d\Psi.\end{aligned} \tag{5.12}$$

Leurs positions sont calculées, par exemple pour s_1, par :

$$\begin{aligned}s_1.Y &= V_x dT \sin(s_S.cap - d\Psi) + s_S.Y \\ s_1.X &= V_x dT \cos(s_S.cap - d\Psi) + s_S.X.\end{aligned} \tag{5.13}$$

À partir des paramètres V_x, A_y et dX, les constantes dT et $d\Psi$ sont obtenues par :

$$\begin{aligned}dT &= \frac{dX}{V_x} \\ \dot{\Psi} &= \frac{A_y}{V_x} \\ d\Psi &= \dot{\Psi} dT.\end{aligned} \tag{5.14}$$

Comme le nœud n'est plus forcément sur la grille d'occupation, la vérification se fait dans un rayon $d_v = 1,5$ m autour du véhicule pour inclure l'encombrement du véhicule. Ainsi, si un point de la grille à une distance inférieure à d_v du point considéré est occupé, le nœud n'est pas visitable et n'est donc pas ajouté à la liste ouverte. De même, on considère que l'algorithme à atteint l'objectif si un point dans la liste fermée est à une distance d_G du point d'arrivée avec une erreur de cap de Ψ_e.

Remarque. Ici, le nombre de nœuds pouvant être étendus depuis le parent s_S est limité à trois pour se rapprocher de la méthode précédente. Cependant, il est tout à fait possible d'ajouter d'autres variations de cap, permettant alors d'avoir des rayons de courbure variables. La méthode se rapprocherait encore plus de celle des tentacules, mais le nombre de nœuds visitables, et donc la taille des listes, augmenterait alors considérablement, impactant le temps de calcul. Il conviendrait aussi, en ajoutant d'autres angles de lacet possibles, de vérifier la continuité de la trajectoire finale.

Une fois le calcul des nœuds s_1, s_2 et s_3 effectué, la méthode de recherche de trajectoire est identique à celle de A* temporelle précédente. La fonction de coût globale f est toujours constituée de l'heuristique, de la longueur du chemin parcouru et du temps parcouru. Une modification est apportée sur la fonction calculant la longueur du chemin parcouru g. Dans le cas du respect des contraintes géométriques, l'incrémentation de cette fonction dépendait du changement d'orientation. Ici, étant donné que le changement d'orientation est conforme à la dynamique latérale, pour chaque nœud admissible s_i depuis s_S, $s_i.g$ est calculé par :

$$s_i.g = s_S.g + \sqrt{(s_i.X - s_S.X)^2 + (s_i.X - s_S.Y)^2}. \tag{5.15}$$

5.3.4 Application : dépassement sur voie à double sens

Dépassement d'un vélo

Reprenons le cas du dépassement d'un vélo sur la route à double sens précédent. Le premier test utilise le même paramétrage que précédemment : $K_t = 1$, $t_{max} = 5$ s et avec $dX = 1$ m et $A_y = 1$ m.s^{-2}. Le résultat du A* hybride temporel est visible sur la Figure 5.38 pour différents instants.

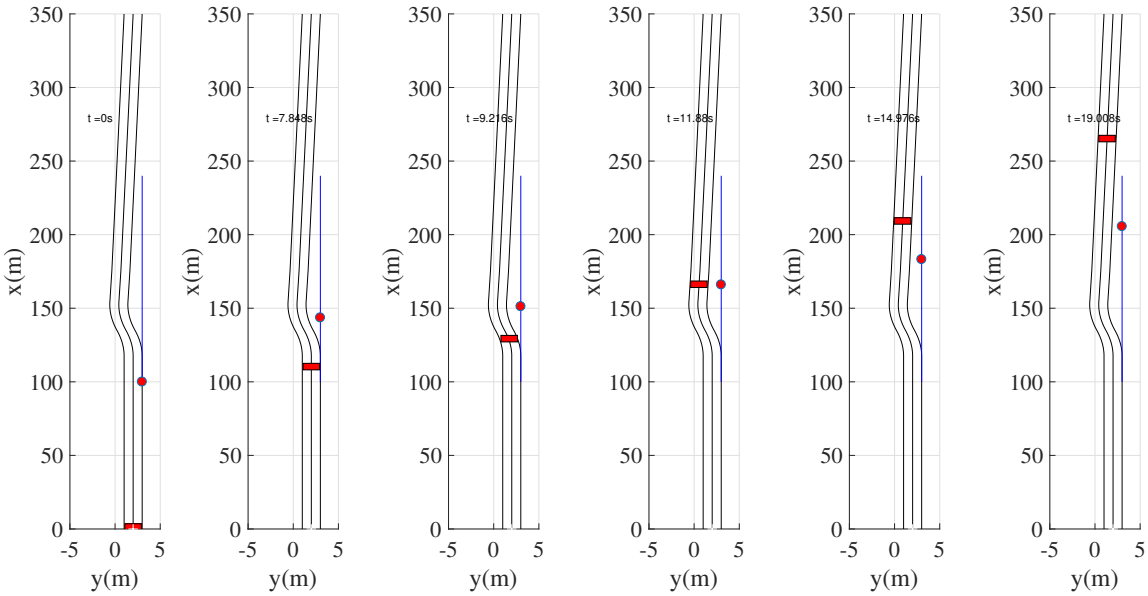


FIGURE 5.38 – Positionnement de l'ego véhicule (rectangle rouge) et du vélo (cercle rouge) à différents instants dans le cas du A* Hybride temporel, $K_t = 1$ et $t_{max} = 5$ s

Grâce aux contraintes dynamiques, et non plus géométriques, le dépassement de voie du vélo est plus fluide. Le dépassement commence au même instant que précédemment. Lorsque l'ego véhicule et le vélo se croisent longitudinalement, l'ego s'est déporté de 1,5 m. Il y a alors 1,5 m entre le vélo et les roues droites de l'ego. De la même manière que les analyses sur l'influence des variations de K_t et t_{max}, il est possible ici de modifier la valeur de dX. Les résultats sont tracés sur la Figure 5.39. Ainsi, plus dX augmente et plus le moment du dépassement est retardé. De même, avec un dX plus grand, le rabattement dans la voie commence plus tard.

Plus dX augmente, plus les noeuds explorés se rapprochent du point d'arrivée rapidement, cela a donc un impact sur le nombre de points explorés. La Table 5.3 présente ces résultats.

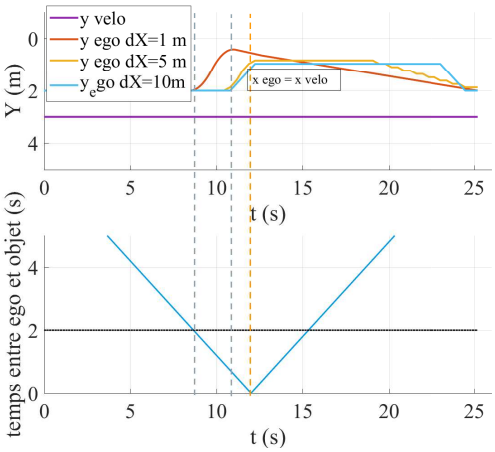


FIGURE 5.39 – Influence de dX sur le calcul de la trajectoire

dX (m)	1	5	10
Nombre de nœuds explorés	695	81	41
Temps de calcul (s)	3,25	0,28,	0,16

TABLE 5.3 – Résultats du dépassement du vélo pour différents dX

Le problème qui apparait ici, pour $dX = 1$ m notamment, est que le véhicule ne se rabat pas dans sa voie, il va se déporter lentement jusqu'à atteindre le point d'arrivée. Ceci est compréhensible, car c'est effectivement le chemin le plus court. Cependant, d'un point de vue du comportement routier, il faudrait que le véhicule se rabatte complètement dès le vélo dépassé. Sachant que ce comportement est dû en grande partie à l'heuristique h de la fonction de coût, il faut trouver une nouvelle heuristique favorisant le retour dans la voie.

Il existe principalement deux heuristiques pour la recherche de chemin par A* : la distance euclidienne utilisée précédemment, et la distance de Manhattan. Pour un nœud s et le nœud d'arrivée s_G, la distance de Manhattan se calcule par :

$$d_M = |s.X - s_G.X| + |s.Y - s_G.Y|. \tag{5.16}$$

Ainsi, avec une distance de Manhattan comme heuristique, les points explorés auront plutôt tendance à se rapprocher de la voie de départ. Cependant, en utilisant la distance de Manhattan avec $dX = 1$ m de l'exemple précédent, la position latérale oscille autour de la position désirée, voir courbe verte de la Figure 5.40. La distance de Manhattan permet bien un retour latéral plus rapide dans la voie, mais ne se stabilise pas une fois la position latérale désirée atteinte. Ce phénomène d'oscillation n'est pas souhaitable et doit être évité. Pour combiner les effets des deux heuristiques, une pondération entre distance euclidienne d_E et distance de Manhattan d_M est utilisée telle que pour un point s :

$$s.h = \frac{Ed_E + Md_M}{E + M}, \tag{5.17}$$

avec $E, M \in \mathbb{R}_+$. La Figure 5.40 montre les résultats du dépassement du vélo pour différentes combinaisons de M et E. Le fait d'augmenter la part de distance euclidienne dans l'heuristique globale permet de réduire les oscillations, et même de complètement les arrêter pour $E = 25$ et $M = 1$. Avec cette combinaison, le véhicule revient dans la voie ce qui est le comportement

attendu par le Code de la route. Le résultat de cette configuration, avec le mouvement des objets, est visible sur la Figure 5.41. Les temps de calcul et le nombre de points explorés sont indiqués dans la Table 5.4.

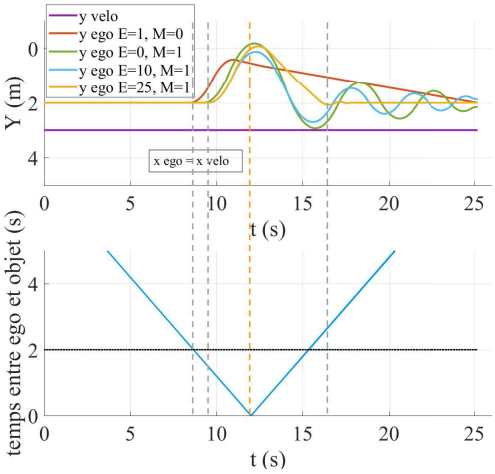


FIGURE 5.40 – Influence de l'heuristique sur le calcul de la trajectoire

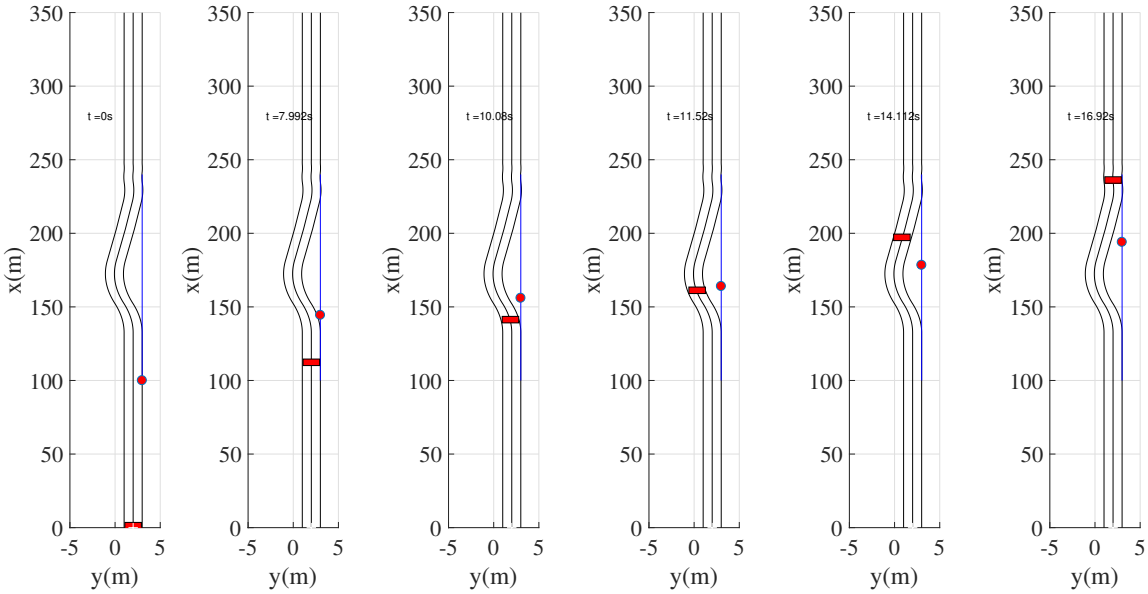


FIGURE 5.41 – Positionnement de l'ego véhicule (rectangle rouge) et du vélo (cercle rouge) à différents instants dans le cas du A* Hybride temporel, $K_t = 1$, $t_{max} = 5$ s, $dX = 1$ m, $E = 25$ et $M = 1$

M	0	1	1	1
E	1	0	10	25
Nombre de nœuds explorés	695	1121	732	677
Temps de calcul (s)	3,25	5,48	3,41	3,08

TABLE 5.4 – Résultats du dépassement du vélo pour différentes combinaisons de E et M

Dépassement d'une voiture avec une voiture en sens inverse

La méthode étant validée pour un cas simple, le scénario est complexifié. Maintenant, l'ego véhicule doit dépasser un autre véhicule roulant à 30 km/h sachant qu'un deuxième véhicule, avec une vitesse de 60 km/h, arrive en sens inverse. Trois cas peuvent alors se présenter :

- Cas 1 : l'ego véhicule dépasse le véhicule de devant avant de croiser le véhicule en face
- Cas 2 : l'ego véhicule dépasse le véhicule de devant après avoir croisé le véhicule en face
- Cas 3 : les trois véhicules arrivent à la même position longitudinale en même temps.

Sachant que la méthode ne permet pas au véhicule de freiner, le dernier cas donnera forcément lieu à une collision. En cas de collision, c'est l'ACC qui prend le relais. Il ne faut cependant pas que le véhicule se soit déporté. En effet, le choc frontal reste le scénario le plus critique pour le véhicule, il est donc à proscrire. Une grille temporelle de l'environnement pour ce type de scénario est tracée Figure 5.42.

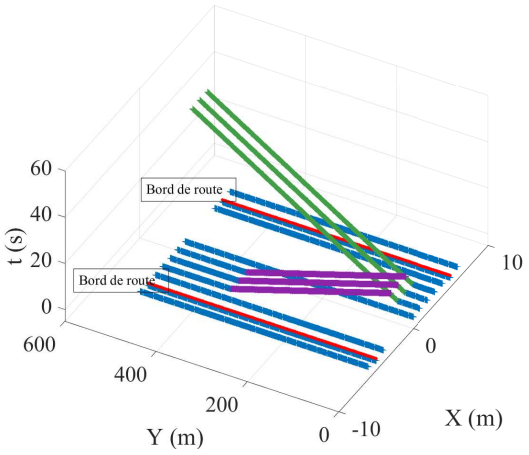


FIGURE 5.42 – Grille temporelle pour le dépassement d'une voiture avec une voiture arrivant en sens inverse

Dans chaque cas, les paramètres choisis sont : $dX = 2$ m, $A_y = 1$ m.s^{-2}, $K_t = 1$, $t_{max} = 5$, $E = 25$ et $M = 1$. Les Figures 5.43, 5.44 et 5.45 montrent les trajectoires calculées et l'évolution de l'environnement à différents instants pour les différents cas.

Dans les deux premiers cas, le A* Hybride temporel parvient à calculer une trajectoire permettant un dépassement sans collision tout en respectant les contraintes dynamiques imposées. Pour le premier cas, le temps de calcul est de 10,37 s alors qu'il est de 3,12 s pour le deuxième. La trajectoire met plus de temps à être calculée dans le premier cas, car le véhicule est influencé plus longtemps par des objets dynamiques. Il y a donc beaucoup de points contraints par la présence d'objets dynamique ce qui force l'algorithme à tester un grand nombre d'autres points.

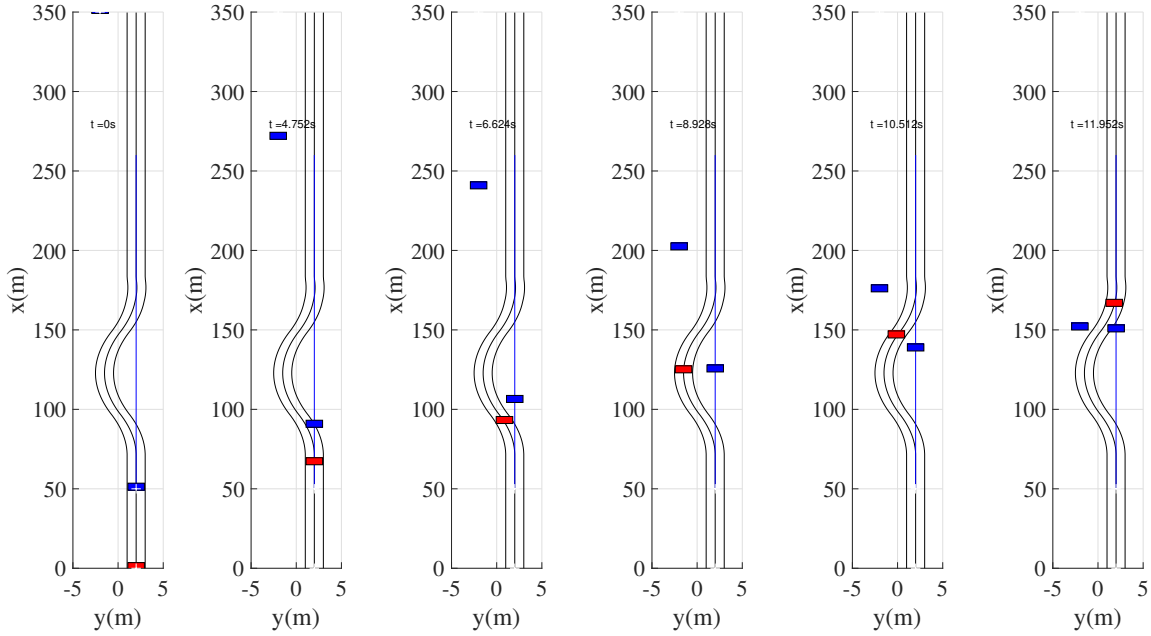


FIGURE 5.43 – Résultats du cas 1 : Positionnement de l'ego véhicule (rectangle rouge) et des autres véhicules (rectangles bleus) à différents instants dans le cas du A* hybride temporel

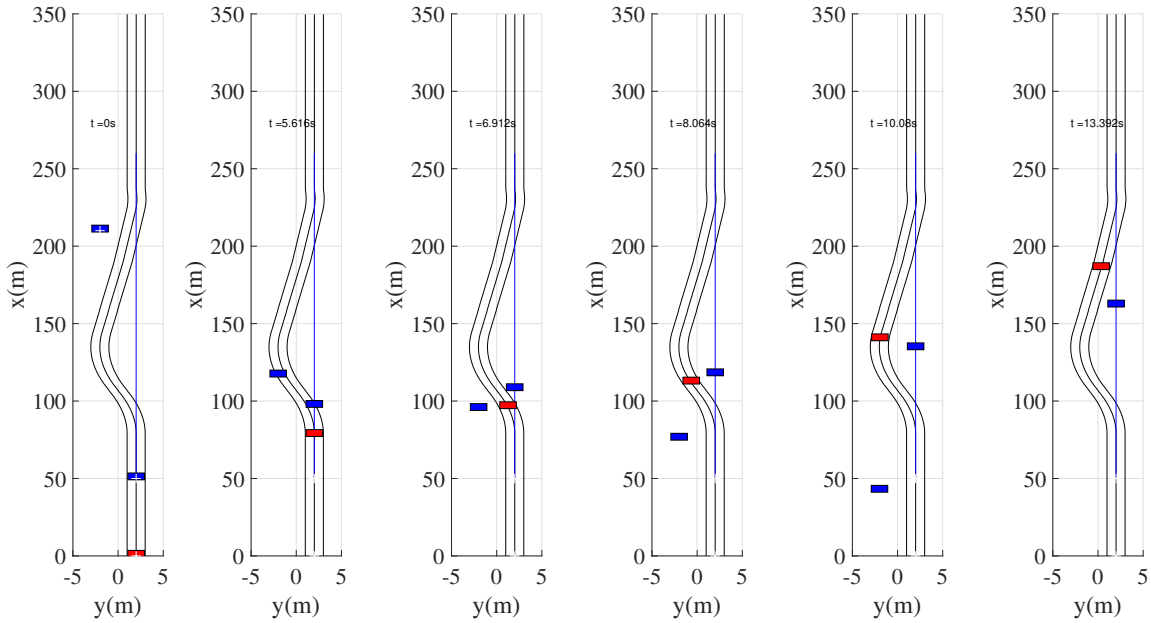


FIGURE 5.44 – Résultats du cas 2 : Positionnement de l'ego véhicule (rectangle rouge) et des autres véhicules (rectangles bleus) à différents instants dans le cas du A* hybride temporel

Le troisième cas est celui d'une collision. Comme expliqué précédemment, le module de génération de trajectoire est couplé à l'ACC qui calcule la vitesse longitudinale. Ainsi, si la trajectoire calculée comporte une collision, l'ACC gère sa vitesse sur le véhicule de devant et le véhicule suit sa trajectoire latérale. La trajectoire calculée dans ce cas effectue quand même le dépassement. Il est préférable d'éviter ce type de résultat pour que l'ego véhicule reste dans sa voie. La trajectoire calculée s'explique par le fait que le A* voit d'abord le véhicule en face.

Il va donc explorer les nœuds sur la gauche pour l'éviter. Une fois à gauche, il est alors aussi pénalisant de continuer tout droit que de revenir explorer les nœuds de la ligne de droite.

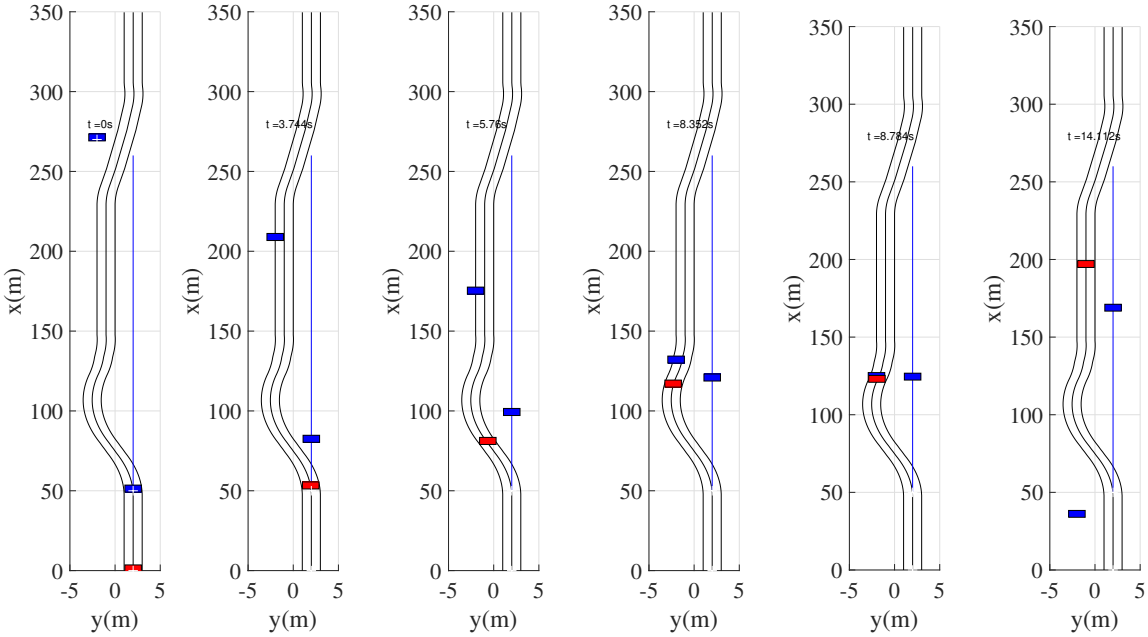


FIGURE 5.45 – Résultats du cas 3 : Positionnement de l'ego véhicule (rectangle rouge) et des autres véhicules (rectangles bleus) à différents instants dans le cas du A* hybride temporel

Pour éviter ce type de problème, il faut anticiper davantage l'arrivée de véhicules en sens inverse et augmenter la pondération K_t dans ce cas. Ce raisonnement est aussi réalisé lors d'une conduite manuelle. En effet, un conducteur regarde plus « loin » les véhicules arrivant en sens inverse et fait plus attention à eux que les véhicules dans le même sens de circulation qui sont moins dangereux. Deux nouveaux paramètres sont ajoutés pour les véhicules en sens opposés : $K_{topp} = 5$ et $t_{maxopp} = 10$ s. La nouvelle trajectoire calculée avec ces paramètres est tracée Figure 5.46.

En donnant plus d'importance aux véhicules arrivant en sens inverse, la trajectoire calculée est une ligne droite. Dans les deux cas cependant, le temps de calcul est extrêmement long : supérieur à 110 s. À cause de la fonction de coût g_t qui augmente fortement autour des objets dynamiques, le A* va donc ajouter à la liste fermée tous les points en amont qui étaient certes plus loin de l'arrivée, mais avait un coût total plus faible.

Les trajectoires des cas 1 et 2 sont recalculées avec les nouveaux paramètres K_{topp} et t_{maxopp}. Les trajectoires, ainsi que tous les nœuds explorés, sont tracées sur la Figure 5.47. On peut y observer l'augmentation de la zone de points explorés lorsque le sens inverse est pondéré plus fortement. Pour une analyse plus complète des résultats, la Table 5.5 indique les temps de calcul et les nombres de points explorés.

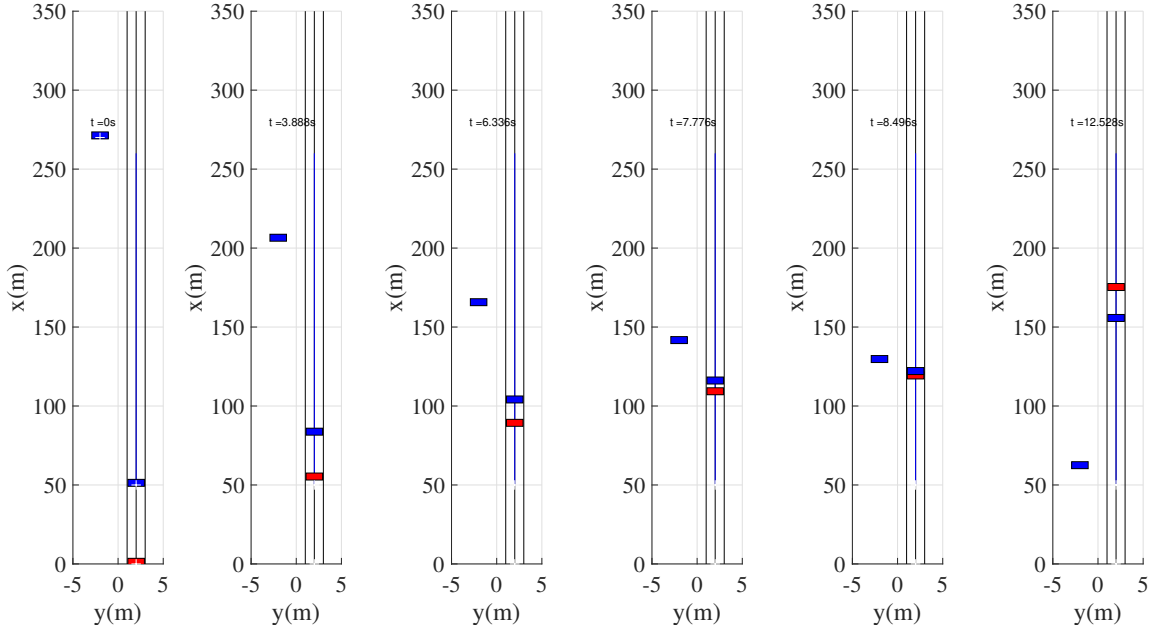


FIGURE 5.46 – Résultats du cas 3 avec pondération sur le sens opposé : Positionnement de l'ego véhicule (rectangle rouge) et des autres véhicules (rectangles bleus) à différents instants dans le cas du A* hybride temporel

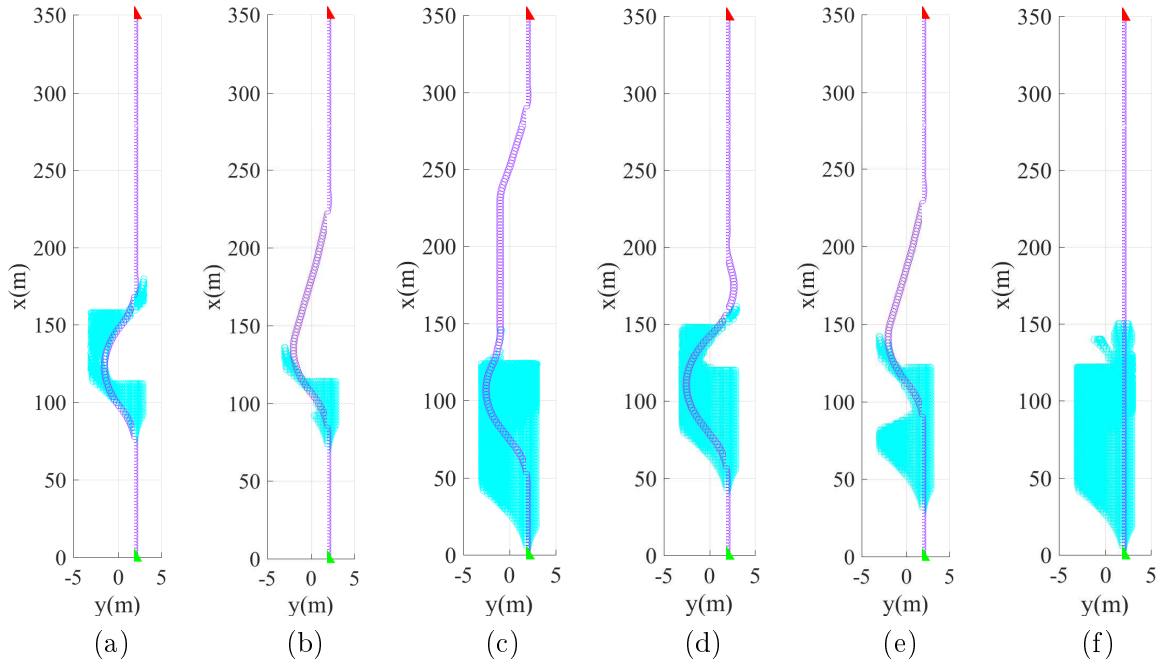


FIGURE 5.47 – Trajectoire (en violet) et nœuds explorés (en cyan) pour : (a) le cas 1, (b), le cas 2, (c) le cas 3, (d) le cas 1 avec pondération du sens inverse, (e) le cas 2 avec pondération du sens inverse, (f) le cas 3 avec pondération du sens inverse

La pondération du sens opposé introduit une augmentation non négligeable du temps de calcul. Il est multiplié par quatre pour les deux premiers cas. De plus pour les deux premiers cas, la trajectoire finale est similaire.

	Cas 1		Cas 2		Cas 3	
t_{max} (s)	5	5	5	5	5	5
t_{maxopp} (s)	5	10	5	10	5	10
K_t (s)	1	1	1	1	1	1
K_{topp} (s)	1	5	1	5	1	5
Nombre de nœuds explorés	1698	4108	675	1957	7329	7531
Temps de calcul (s)	10,37	42,78	3,12	13,14	114,46	122,81

TABLE 5.5 – Résultats du dépassement du véhicule pour les différents cas et les différents paramètres

5.3.5 Bilan : avantages, problématiques et perspectives

Nous avons présenté dans les paragraphes précédents un nouveau A* Hybride temporel pour la recherche de trajectoire en environnement dynamique. Cette méthode a été testée en simulation dans le cas de scénarios urbains sur des routes à double sens de circulation. La méthode proposée permet à la fois d'inclure la dynamique latérale du véhicule avec un modèle cinématique et d'éviter les objets dynamiques grâce à l'ajout d'une nouvelle fonction de coût dans le coût total des nœuds explorés par le A*. En supposant que la trajectoire des véhicules de l'environnement est estimée, on sait donc où ils se trouveront à chaque instant. Il est alors possible d'éviter ces positions lors de l'exploration de l'environnement par le A*.

La méthode est testée pour le dépassement d'un vélo et le dépassement d'une voiture avec une deuxième voiture arrivant en sens inverse. Dans tous les scénarios, la trajectoire finale évite les objets au maximum en essayant de dépasser et se rabattre lorsque le temps avant collision est d'environ 2 s. Si un véhicule arrive en face, la trajectoire finale commence le dépassement une fois le véhicule en face passé.

L'inconvénient majeur de cette méthode, telle que présentée ici, est qu'elle ne permet pas une variation de la dynamique longitudinale de l'ego véhicule. La vitesse longitudinale est supposée fixe et le véhicule peut donc calculer des trajectoires s'approchant fortement des autres véhicules voir entrant en collision avec eux. C'est pourquoi cette méthode de planification de trajectoire est forcément liée au module de décision. Ainsi, si la trajectoire calculée se rapproche trop d'autres véhicules malgré la contrainte imposée, alors le véhicule doit rester dans sa voie et la vitesse est gérée par l'ACC.

Un autre inconvénient est le temps de calcul dans les cas complexes. En effet, si la collision est inévitable, les temps de calcul sont trop importants alors qu'il est possible de calculer très simplement le fait qu'il est impossible de changer de voie. Le problème du temps de calcul peut être limité en augmentant la valeur de dX. Cela se traduira alors par des itérations avec des mouvements latéraux plus grands.

Enfin, le A* Hybride perd la propriété de complétude du A*. Étant donné que l'exploration de l'environnement ne se fait plus sur une grille fixe, le fait de ne pas trouver une solution ne signifie pas qu'elle n'existe pas.

Au vu de ces résultats, la méthode paraît prometteuse pour la planification de trajectoire dans des infrastructures plus complexes : intersection ou rond-point. En la couplant avec un module de planification de mission, le temps de calcul pourrait être réduit considérablement. On peut aussi imaginer que la dynamique longitudinale soit directement incorporée dans l'exploration de l'environnement du A* avec des nœuds pouvant avoir une accélération longitudinale différente.

Conclusion générale et perspectives

Cette thèse traite de l'amélioration de fonctions existantes et de la création de nouveaux modules pour le guidage latéral et la planification de mouvement d'un véhicule autonome en environnement urbain. Les principaux résultats, ainsi que les perspectives, sont présentés dans cette conclusion.

Le **Chapitre 1** présente l'évolution des systèmes mécatroniques à bord des véhicules. La multiplication des systèmes d'aide à la conduite introduit une augmentation de l'électronique à bord des véhicules : capteurs, calculateurs, câblage. La multiplication des capteurs permet donc aux constructeurs automobiles de proposer des assistances de conduite toujours plus performantes et automatisées. Le véhicule autonome est la prochaine étape de cette automatisation. Le chapitre présente ensuite les apports nécessaires en termes d'équipement et de fonctionnalité pour franchir cette étape. Ensuite, les prototypes robotisés et équipés de capteurs extéroceptifs du Groupe PSA sont introduits. Ces prototypes ont été utilisés pour l'intégration et la validation des modules développés dans cette thèse.

Le **Chapitre 2** compare différents modèles d'évolution de la dynamique latérale en vue d'une observation de composantes de cette dynamique qui sont nécessaires aux fonctions d'autonomisation. Il est aussi montré que la vitesse longitudinale est une composante importante, non négligeable, de la dynamique latérale. Deux comportements distincts peuvent être identifiés : un à basse vitesse et un à haute vitesse. Un modèle non linéaire sert de référence. Trois modèles simplifiés lui sont comparés dans les domaines fréquentiel et temporel : un modèle linéaire, un modèle linéaire en régime circulaire uniforme et un modèle cinématique. Le modèle cinématique n'est représentatif qu'aux basses vitesses alors que le modèle linéaire représente correctement la dynamique latérale tant que les pneumatiques sont dans leur zone linéaire de fonctionnement. Le chapitre présente aussi une méthode d'observation de la dynamique latérale utilisant une approche multimodèle. L'objectif est d'avoir une estimation plus précise de la vitesse de lacet et de l'accélération latérale, car ces variables sont primordiales et utilisées pour les modules de guidage et de localisation du véhicule autonome. Les variables estimées par la méthode développée sont comparées à celles d'une centrale inertielle, servant de référence, et validées sur les prototypes. La méthode offre de meilleurs résultats que la mesure utilisée habituellement sur le véhicule.

Le **Chapitre 3** utilise les modélisations de la dynamique latérale et les conclusions apportées sur l'influence de la vitesse longitudinale pour développer une méthode de guidage latéral du véhicule. L'objectif du chapitre étant la mise au point d'une méthode de régulation capable de réguler l'angle volant suivant un cahier des charges défini sur toute la plage de vitesse autorisée. Le chapitre compare alors plusieurs régulateurs pour des simulations de dépassement. La vitesse longitudinale étant un paramètre variant et mesurable, l'approche développée utilise cette connaissance pour le calcul du régulateur. Ainsi, au lieu d'utiliser un seul régulateur devant

fonctionner à toutes les vitesses, l'approche multirégulateur combine plusieurs régulateurs positionnés à différents points de fonctionnement qui dépendent de la vitesse. L'approche permet à la fois de déterminer les valeurs de ces points de fonctionnement et le nombre de régulateurs nécessaires.

La consigne est une composante cruciale de la boucle de régulation. Ainsi, il est montré que l'utilisation d'un point de visée comme consigne latérale améliore le degré de stabilité du système à la variation de vitesse longitudinale. Il est donc plus facile de concevoir un régulateur avec ce type de consigne. Finalement, un multi-PID avec point de visée est implanté dans un prototype et testé sur route ouverte. La régulation proposée est validée pour une conduite autonome dans les cas de maintien dans la voie, à haute vitesse et en embouteillage, et de changement de voie. La méthode doit cependant être validée en environnement urbain, dans les intersections et les ronds-points.

Pour aller plus loin, la méthode de multirégulation pourrait utiliser différents types de régulateurs en fonction du point de fonctionnement considéré. Ainsi, il serait intéressant de comparer les multi-PID calculés ici à des multirégulateurs à base de P, PI, PID et CRONE. Une autre perspective serait d'utiliser la méthode afin de coupler la régulation des dynamiques longitudinale et latérale. Enfin, l'étude de la stabilité globale doit être finalisée.

Le **Chapitre 4** traite de la prédiction d'intention et de trajectoire des autres véhicules de l'environnement. Cette étape est nécessaire pour la génération de trajectoire de l'ego véhicule. Dû à une base de données plus conséquente sur route à chaussées séparées qu'en environnement urbain, le chapitre développe des méthodes de prédiction de changement de voie : une méthode probabiliste à base de règles et une méthode par apprentissage automatique utilisant des réseaux de neurones. L'objectif de ce chapitre est de montrer la pertinence d'une méthode pour l'appliquer ensuite aux scénarios urbains lorsque les données seront disponibles. Il est alors montré que les réseaux de neurones sont plus efficaces pour détecter les changements de voie, mais ces résultats dépendent fortement des variables utilisées en entrée du réseau. Ainsi, une meilleure estimation de la vitesse latérale donnera une meilleure prédiction. Ces deux méthodes doivent cependant être testées dans les prototypes afin d'analyser les prédictions en temps réel et de déterminer la durée entre la détection de changement de voie et le franchissement de ligne effectif.

La deuxième partie du chapitre présente une méthode, toujours à base de réseaux de neurones, et avec les mêmes entrées dynamiques que précédemment pour la prédiction de trajectoire. La prédiction se fait sur un horizon d'une seconde. La méthode à base de neurones semble être une bonne solution pour la prédiction de trajectoire avec une erreur de prédiction de 20 cm à 1 s. Cependant, comme pour la prédiction d'intention, elle doit être étudiée avec attention pour utiliser les bonnes variables en entrée.

Une ouverture sur la prédiction de trajectoire en environnement urbain, dans un rond-point, est présentée. La visibilité des capteurs est ainsi illustrée dans ce type d'environnement à partir d'enregistrements. Les capteurs arrivent à percevoir correctement les autres véhicules dans le rond-point. Étant donné que la dynamique latérale est davantage sollicitée, la suite de l'étude consisterait à tester la méthode présentée sur autoroute pour évaluer si d'autres variables, de la dynamique ou de l'infrastructure, sont nécessaires.

Après un état de l'art sur les méthodes de génération de mouvement pour le véhicule autonome, le **Chapitre 5** présente une méthode de planification de trajectoire, d'abord en environnement statique puis en environnement dynamique dans le cas urbain. Pour beaucoup des méthodes présentées dans l'état de l'art, le temps de calcul est le principal inconvénient. La solution développée dans ce chapitre respecte un temps de calcul admissible dans le cadre de

la conduite autonome. Cette méthode est basée sur l'exploration de l'environnement, représenté par une grille régulière, par l'algorithme A*. Cette exploration est contrainte de manière à respecter les contraintes géométriques du véhicule et ainsi générer une trajectoire admissible. Pour améliorer le temps de calcul et respecter le Code de la route, des points de passages sont ajoutés. Une méthode générique pour le placement de ces points et leurs pondérations, formant alors un graphe, est aussi présentée dans le cas du rond-point. La méthode de génération de trajectoire est validée sur les prototypes dans un scénario de type parking.

L'algorithme est ensuite amélioré pour les environnements dynamiques. La recherche de chemin ne se fait plus dans une grille fixe, car trop contraignante au niveau de la dynamique du véhicule. L'exploration de l'environnement se fait alors en utilisant le modèle cinématique du véhicule. L'environnement dynamique est intégré dans une autre dimension de la grille d'occupation avec les trajectoires prédites des véhicules de l'environnement. Une contrainte dans l'exploration de l'environnement est ajoutée à l'approche de tels objets de manière à ce que l'ego véhicule évite les autres objets dynamiques au maximum. La solution proposée est validée en simulation dans les cas de dépassement sur route à double sens de circulation avec un véhicule arrivant en sens opposé.

La méthode proposée semble être prometteuse pour la génération de trajectoire complexe en environnement dynamique. Cependant, elle doit être améliorée pour être couplée à la consigne de vitesse et au module de décision. Enfin, la méthode doit aussi être testée en intersections et intégrée dans les calculateurs des prototypes. Une extension pourrait porter sur l'intégration de la variation de la vitesse longitudinale dans la recherche de trajectoire.

Bibliographie

[1] J. Ackermann, J. Guldner, W. Sienel, R. Steinhauser, and V. I. Utkin. Linear and nonlinear controller design for robust automatic steering. *IEEE Transactions on Control Systems Technology*, 3(1) :132–143, 1995.

[2] AFNOR. Mécatronique - Vocabulaire, NF E01-010. Novembre 2008.

[3] C. Alia, T. Gilles, T. Reine, and C. 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.

[4] F. Altché and A. de La Fortelle. An lstm network for highway trajectory prediction. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 353–359. IEEE, 2017.

[5] G. S. Aoude and J. P. How. Using support vector machines and bayesian filtering for classifying agent intentions at road intersections. Technical report, 2009.

[6] K. J. Åström and T. Hägglund. *PID controllers : theory, design, and tuning*, volume 2. Instrument society of America Research Triangle Park, NC, 1995.

[7] R. Attia, R. Orjuela, and M. Basset. Combined longitudinal and lateral control for automated vehicle guidance. *Vehicle System Dynamics*, 52(2) :261–279, 2014.

[8] G. Baffet, A. Charara, and D. Lechner. Estimation of vehicle sideslip, tire force and wheel cornering stiffness. *Control Engineering Practice*, 17(11) :1255–1264, 2009.

[9] A. Benine-Neto, S. Mammar, B. Lusetti, and S. Scalzi. Piecewise affine control for lane departure avoidance. *Vehicle System Dynamics*, 51(8) :1121–1150, 2013.

[10] T. Besselmann and M. Morari. Autonomous vehicle steering using explicit lpv-mpc. In *2009 European Control Conference (ECC)*, pages 2628–2633. IEEE, 2009.

[11] S. Bezzaoucha. *Fault tolerant control for Takagi-Sugeno nonlinear systems*. Theses, Université de Lorraine, Oct. 2013. URL https://tel.archives-ouvertes.fr/tel-01750104.

[12] B. Boada, M. Boada, and V. Diaz. Vehicle sideslip angle measurement based on sensor data fusion using an integrated anfis and an unscented kalman filter algorithm. *Mechanical Systems and Signal Processing*, 72 :832–845, 2016.

[13] F. Bocklisch, S. F. Bocklisch, M. Beggiato, and J. F. Krems. Adaptive fuzzy pattern classification for the online detection of driver lane change intention. *Neurocomputing*, 262 :148–158, 2017.

[14] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv :1604.07316*, 2016.

[15] F. Boyer and F. Lamiraux. Trajectory deformation applied to kinodynamic motion planning for a realistic car model. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 487–492. IEEE, 2006.

[16] P. Boyraz, M. Acar, and D. Kerr. Signal modelling and hidden markov models for driving manoeuvre recognition and driver fault diagnosis in an urban road scenario. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 987–992. IEEE, 2007.

[17] S. Brechtel, T. Gindele, and R. Dillmann. Probabilistic mdp-behavior planning for cars. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1537–1542. IEEE, 2011.

[18] S. Brechtel, T. Gindele, and R. Dillmann. Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 392–399. IEEE, 2014.

[19] L. Breiman. Random forests. *Machine learning*, 45(1) :5–32, 2001.

[20] L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and regression trees. wadsworth & brooks. *Cole Statistics/Probability Series*, 1984.

[21] O. Brock and O. Khatib. High-speed navigation using the global dynamic window approach. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 1, pages 341–346. IEEE, 1999.

[22] A. Broggi, M. Bertozzi, A. Fascioli, C. G. L. Bianco, and A. Piazzi. The ARGO autonomous vehicle's vision and control systems. *International Journal of Intelligent Control and Systems*, 3(4) :409–441, 1999.

[23] G. Brusaglino. Safe and effective mobility in Europe-the contribution of the PROMETHEUS programme. In *Prometheus and Drive, IEE Colloquium on*, pages 1–1. IET, 1992.

[24] L. Cai, A. B. Rad, W.-L. Chan, and K.-Y. Cai. A robust fuzzy PD controller for automatic steering control of autonomous vehicles. In *The 12th IEEE International Conference on Fuzzy Systems, 2003. FUZZ'03.*, volume 1, pages 549–554. IEEE, 2003.

[25] S. Chaib, M. S. Netto, and S. Mammar. H_∞, adaptive, PID and fuzzy control : a comparison of controllers for vehicle lane keeping. In *IEEE Intelligent Vehicles Symposium, 2004*, pages 139–144. IEEE, 2004.

[26] X. Chalandon. *Conscience de la situation : invariants internes et invariants externes.* PhD thesis, Cnam, 2007.

[27] A. C. Charalampidis and D. Gillet. Speed profile optimization for vehicles crossing an intersection under a safety constraint. In *Control Conference (ECC), 2014 European*, pages 2894–2901. Ieee, 2014.

[28] F. Cheli, E. Sabbioni, M. Pesce, and S. Melzi. A methodology for vehicle sideslip angle identification : comparison with experimental data. *Vehicle System Dynamics*, 45(6) : 549–563, 2007.

[29] C. Chen and H.-S. Tan. Experimental study of dynamic look-ahead scheme for vehicle steering control. In *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, volume 5, pages 3163–3167. IEEE, 1999.

[30] C. Chen, Y. He, C. Bu, J. Han, and X. Zhang. Quartic bézier curve based trajectory generation for autonomous vehicles with curvature and velocity constraints. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6108–6113. IEEE, 2014.

[31] L.-K. Chen, A. G. Ulsoy, et al. Identification of a driver steering model, and model uncertainty, from driving simulator data. *Transactions-American Society of Mechanical Engineers Journal of Dynamic Systems Measurement and Control*, 123(4) :623–629, 2001.

[32] Y. Chen, Y. Ji, and K. Guo. A reduced-order nonlinear sliding mode observer for vehicle slip angle and tyre forces. *Vehicle System Dynamics*, 52(12) :1716–1728, 2014.

[33] H. Cherouat, M. Braci, and S. Diop. Vehicle velocity, side slip angles and yaw rate estimation. In *Proceedings of the IEEE International Symposium on Industrial Electronics, 2005. ISIE 2005.*, volume 1, pages 349–354. IEEE, 2005.

[34] M. Chevrieé. *Modélisation électrothermique de composants électriques et électroniques automobiles et estimation des résistances de contact dans les connecteurs.* PhD thesis, Université de Bordeaux, 2016.

[35] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3) :273–297, 1995.

[36] L. B. Cremean, T. B. Foote, J. H. Gillula, G. H. Hines, D. Kogan, K. L. Kriechbaum, J. C. Lamb, J. Leibs, L. Lindzey, C. E. Rasmussen, et al. Alice : An information-rich autonomous vehicle for high-speed desert navigation. *Journal of Field Robotics*, 23(9) : 777–810, 2006.

[37] A. G. Cunningham, E. Galceran, R. M. Eustice, and E. Olson. Mpdm : Multipolicy decision-making in dynamic, uncertain environments for autonomous driving. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1670–1677. IEEE, 2015.

[38] D. A. De Lima and G. A. S. Pereira. Navigation of an autonomous car using vector fields and the dynamic window approach. *Journal of Control, Automation and Electrical Systems*, 24(1-2) :106–116, 2013.

[39] D. A. de Lima and A. C. Victorino. A hybrid controller for vision-based navigation of autonomous vehicles in urban environments. *IEEE Transactions on Intelligent Transportation Systems*, 17(8) :2310–2323, 2016.

[40] V. De Oliveira and A. Karimi. Robust and gain-scheduled PID controller design for condensing boilers by linear programming. *IFAC Proceedings Volumes*, 45(3) :335–340, 2012.

[41] P. Del Moral and A. Doucet. Particle methods : An introduction with applications. In *ESAIM : proceedings*, volume 44, pages 1–46. EDP Sciences, 2014.

[42] V. Delsart and T. Fraichard. Navigating dynamic environments using trajectory deformation. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 226–233. IEEE, 2008.

[43] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1) :269–271, 1959.

[44] N. Ding, Y. Zhang, F. Gao, and G. Xu. A Gain-Scheduled PID Controller for Automatic Path Following of a Tractor Semi-Trailer. *SAE International Journal of Commercial Vehicles*, 6(2013-01-0687) :110–117, 2013.

[45] M. Doumiati, A. C. Victorino, A. Charara, and D. Lechner. Onboard real-time estimation of vehicle lateral tire–road forces and sideslip angle. *IEEE/ASME Transactions on Mechatronics*, 16(4) :601–614, 2011.

[46] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis. State-space solutions to standard H_2 and H_∞ control problems. *IEEE Transactions on Automatic control*, 34(8) : 831–847, 1989.

[47] H. Du, J. Lam, K.-C. Cheung, W. Li, and N. Zhang. Side-slip angle estimation and stability control for a vehicle with a non-linear tyre model and a varying speed. *Proceedings of the Institution of Mechanical Engineers, Part D : Journal of Automobile Engineering*, 229(4) : 486–505, 2015.

[48] J. Du and T. A. Johansen. A gap metric based weighting method for multimodel predictive control of MIMO nonlinear systems. *Journal of Process Control*, 24(9) :1346–1357, 2014.

[49] X. Du, H. Sun, K. Qian, Y. Li, and L. Lu. A prediction model for vehicle sideslip angle based on neural network. In *2010 2nd IEEE International Conference on Information and Financial Engineering*, pages 451–455. IEEE, 2010.

[50] L. DUPONT, L. PATTE, P. BOIVIN, P. FLACHAT, B. GUICHET, J. GIRARD, G. DUPRE, and J. SANGOUARD. *Aménagement des carrefours interurbains sur les routes principales. Carrefours plans - Guide technique*. 1998. Service d'Etudes Techniques des Routes et Autoroutes (SETRA). Accessed 24 April 2019 at http://dtrf.setra.fr/pdf/pj/Dtrf/0001/Dtrf-0001974/DT1974.pdf?openerPage=resultats&qid=sdx_q0.

[51] C. Edwards and S. Spurgeon. *Sliding mode control : theory and applications*. Crc Press, 1998.

[52] A. El Hajjaji and S. Bentalba. Fuzzy path tracking control for automatic steering of vehicles. *Robotics and Autonomous Systems*, 43(4) :203–213, 2003.

[53] M. T. Emirler, K. Kahraman, M. Şentürk, B. Aksun Güvenç, L. Güvenç, and B. Efendioğlu. Vehicle yaw rate estimation using a virtual sensor. *International Journal of Vehicular Technology*, 2013, 2013.

[54] M. R. Endsley. Toward a theory of situation awareness in dynamic systems. *Human factors*, 37(1) :32–64, 1995.

[55] G. Erinc and S. Carpin. A genetic algorithm for nonholonomic motion planning. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 1843–1849. IEEE, 2007.

[56] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat. Predictive active steering control for autonomous vehicle systems. *IEEE Transactions on control systems technology*, 15(3) :566–580, 2007.

[57] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1) :23–33, 1997.

[58] T. Fraichard and H. Asama. Inevitable collision states—a step towards safer robots ? *Advanced Robotics*, 18(10) :1001–1024, 2004.

[59] T. Fraichard and A. Scheuer. From reeds and shepp's to continuous-curvature paths. *IEEE Transactions on Robotics*, 20(6) :1025–1035, 2004.

[60] A. Furda and L. Vlacic. Enabling safe autonomous driving in real-world city traffic using multiple criteria decision making. *IEEE Intelligent Transportation Systems Magazine*, 3 (1) :4–17, 2011.

[61] R. Garcia, O. Aycard, T.-D. Vu, and M. Ahrholdt. High level sensor data fusion for automotive applications using occupancy grids. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 530–535. IEEE, 2008.

[62] T. Gindele, S. Brechtel, and R. Dillmann. A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 1625–1631. IEEE, 2010.

[63] S. Glaser, B. Vanholme, S. Mammar, D. Gruyer, and L. Nouveliere. Maneuver-based trajectory planning for highly autonomous vehicles on real road with traffic and driver interaction. *IEEE Transactions on Intelligent Transportation Systems*, 11(3) :589–606, 2010.

[64] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning.* MIT press, 2016.

[65] C. Götte, M. Keller, C. Rösmann, T. Nattermann, C. Haß, K.-H. Glander, A. Seewald, and T. Bertram. A real-time capable model predictive approach to lateral vehicle guidance. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1908–1913. IEEE, 2016.

[66] R. Graf, H. Deusch, M. Fritzsche, and K. Dietmayer. A learning concept for behavior prediction in traffic situations. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 672–677. IEEE, 2013.

[67] Y. Granjon. Automatique - systèmes linéaires, non linéaires, à temps continu, à temps discret, représentation d'état, événements discrets. *Dunod, Paris*, 2015.

[68] H. F. Grip, L. Imsland, T. A. Johansen, T. I. Fossen, J. C. Kalkkuhl, and A. Suissa. Nonlinear vehicle side-slip estimation with friction adaptation. *Automatica*, 44(3) :611–622, 2008.

[69] T. Gu and J. M. Dolan. Toward human-like motion planning in urban environments. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 350–355. IEEE, 2014.

[70] J. Guldner, W. Sienel, H.-S. Tan, J. Ackermann, S. Patwardhan, and T. Bunte. Robust automatic steering control for look-down reference systems with front and rear sensors. *IEEE transactions on control systems technology*, 7(1) :2–11, 1999.

[71] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús. *Neural network design*, volume 20. Pws Pub. Boston, 1996.

[72] D. L. Hall and J. Llinas. An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1) :6–23, 1997.

[73] N. Hamzah, Y. Md Sam, H. Selamat, M. Aripin, and M. Ismail. Yaw stability improvement for four-wheel active steering vehicle using sliding mode control. 03 2012. doi : 10.1109/CSPA.2012.6194704.

[74] G. Han, W. Fu, and W. Wang. The study of intelligent vehicle navigation path based on behavior coordination of particle swarm. *Computational intelligence and neuroscience*, 2016 :1, 2016.

[75] L. Han, H. Yashiro, H. T. N. Nejad, Q. H. Do, and S. Mita. Bezier curve based path planning for autonomous vehicle in urban environment. In *2010 IEEE Intelligent Vehicles Symposium*, pages 1036–1042. IEEE, 2010.

[76] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2) : 100–107, 1968.

[77] C. Hatipoglu, U. Ozguner, and K. A. Redmill. Automated lane change controller design. *IEEE transactions on intelligent transportation systems*, 4(1) :13–22, 2003.

[78] X. He, D. Xu, H. Zhao, M. Moze, F. Aioun, and F. Guillemard. A human-like trajectory planning method by learning from naturalistic driving data. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 339–346. IEEE, 2018.

[79] C. Hermes, J. Einhaus, M. Hahn, C. Wöhler, and F. Kummert. Vehicle tracking and motion prediction in complex urban scenarios. In *2010 IEEE Intelligent Vehicles Symposium*, pages 26–33. IEEE, 2010.

[80] M. Himmelsbach, T. Luettel, F. Hecker, F. von Hundelshausen, and H.-J. Wuensche. Autonomous off-road navigation for mucar-3. *KI-Künstliche Intelligenz*, 25(2) :145–149, 2011.

[81] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao. Vehicle trajectory prediction based on motion model and maneuver recognition. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4363–4369. IEEE, 2013.

[82] R. A. Howard. Dynamic programming and markov processes. 1960.

[83] Y.-H. J. Hsu, S. M. Laws, and J. C. Gerdes. Estimation of tire slip angle and friction limits using steering torque. *IEEE Transactions on Control Systems Technology*, 18(4) : 896–907, 2010.

[84] C. Hu, H. Jing, R. Wang, F. Yan, and M. Chadli. Robust H_∞ output-feedback control for path following of autonomous ground vehicles. *Mechanical Systems and Signal Processing*, 70 :414–427, 2016.

[85] J. Huang and H.-S. Tan. Vehicle future trajectory prediction with a dgps/ins-based positioning system. In *2006 American Control Conference*, pages 6–pp. IEEE, 2006.

[86] C. Hubmann, M. Aeberhard, and C. Stiller. A generic driving strategy for urban environments. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1010–1016. IEEE, 2016.

[87] J. hwan Jeon, S. Karaman, and E. 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.

[88] S. Jalel. *Optimisation de la navigation robotique.* PhD thesis, 2016.

[89] S. J. Julier and J. K. Uhlmann. New extension of the kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–194. International Society for Optics and Photonics, 1997.

[90] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2) :99–134, 1998.

[91] E. Käfer, C. Hermes, C. Wöhler, H. Ritter, and F. Kummert. Recognition of situation classes at road intersections. In *2010 IEEE International Conference on Robotics and Automation*, pages 3960–3965. IEEE, 2010.

[92] R. Kala and K. Warwick. Motion planning of autonomous vehicles in a non-autonomous vehicle environment without speed lanes. *Engineering Applications of Artificial Intelligence*, 26(5-6) :1588–1601, 2013.

[93] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1) :35–45, 1960.

[94] S. Karaman and E. Frazzoli. Incremental sampling-based algorithms for optimal motion planning. *arXiv preprint arXiv :1005.0416*, 2010.

[95] D. Kasper, G. Weidl, T. Dang, G. Breuel, A. Tamke, A. Wedel, and W. Rosenstiel. Object-oriented bayesian networks for detection of lane change maneuvers. *IEEE Intelligent Transportation Systems Magazine*, 4(3) :19–31, 2012.

[96] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka. Real-time motion planning methods for autonomous on-road driving : State-of-the-art and future research directions. *Transportation Research Part C : Emerging Technologies*, 60 :416–442, 2015.

[97] M. Keller, F. Hoffmann, C. Hass, T. Bertram, and A. Seewald. Planning of optimal collision avoidance trajectories with timed elastic bands. *IFAC Proceedings Volumes*, 47 (3) :9822–9827, 2014.

[98] M. Khatib, H. Jaouni, R. Chatila, and J.-P. Laumond. Dynamic path modification for car-like nonholonomic mobile robots. In *Proceedings of International Conference on Robotics and Automation*, volume 4, pages 2920–2925. IEEE, 1997.

[99] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Autonomous robot vehicles*, pages 396–404. Springer, 1986.

[100] U. Kiencke and A. Daiß. Observation of lateral vehicle dynamics. *Control Engineering Practice*, 5(8) :1145–1150, 1997.

[101] B. Kim, C. M. Kang, J. Kim, S. H. Lee, C. C. Chung, and J. W. Choi. Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 399–404. IEEE, 2017.

[102] K.-D. Kim and P. R. Kumar. An mpc-based approach to provable system-wide safety and liveness of autonomous ground traffic. *IEEE Transactions on Automatic Control*, 59(12) : 3341–3356, 2014.

[103] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli. Kinematic and dynamic vehicle models for autonomous driving control design. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 1094–1099. IEEE, 2015.

[104] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[105] M. Kuderer, S. Gulati, and W. Burgard. Learning driving styles for autonomous vehicles from demonstration. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2641–2646. IEEE, 2015.

[106] Y. Kuwata, J. Teo, G. Fiore, S. Karaman, E. Frazzoli, and J. P. How. Real-time motion planning with applications to autonomous urban driving. *IEEE Transactions on Control Systems Technology*, 17(5) :1105–1118, 2009.

[107] F. Lamiraux, D. Bonnafous, and O. Lefebvre. Reactive path deformation for nonholonomic mobile robots. *IEEE transactions on robotics*, 20(6) :967–977, 2004.

[108] J.-C. Latombe. *Robot motion planning*, volume 124. Springer Science & Business Media, 2012.

[109] S. M. LaValle. Rapidly-exploring random trees : A new tool for path planning. 1998.

[110] A. Lawitzky, D. Althoff, C. F. Passenberg, G. Tanzmeister, D. Wollherr, and M. Buss. Interactive scene prediction for automotive applications. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 1028–1033. IEEE, 2013.

[111] S. Lefèvre, D. Vasquez, and C. Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH journal*, 1(1) :1, 2014.

[112] R. D. Leighty. DARPA ALV (Autonomous Land Vehicle) Summary. Technical report, ARMY ENGINEER TOPOGRAPHIC LABS FORT BELVOIR VA, 1986.

[113] D. J. Leith and W. E. Leithead. Survey of gain-scheduling analysis and design. *International journal of control*, 73(11) :1001–1025, 2000.

[114] G. Li, S. E. Li, Y. Liao, W. Wang, B. Cheng, and F. Chen. Lane change maneuver recognition via vehicle state and driver operation signals—results from naturalistic driving data. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 865–870. IEEE, 2015.

[115] J. Li and J. Zhang. Vehicle sideslip angle estimation based on hybrid kalman filter. *Mathematical Problems in Engineering*, 2016, 2016.

[116] X. Li, X. Song, and C. Chan. Reliable vehicle sideslip angle fusion estimation using low-cost sensors. *Measurement*, 51 :241–258, 2014.

[117] M. Likhachev, G. J. Gordon, and S. Thrun. Ara* : Anytime a* with provable bounds on sub-optimality. In *Advances in neural information processing systems*, pages 767–774, 2004.

[118] P. Liu, A. Kurt, et al. Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 942–947. IEEE, 2014.

[119] D. Luenberger. An introduction to observers. *IEEE Transactions on automatic control*, 16(6) :596–602, 1971.

[120] P. Lytrivis, G. Thomaidis, M. Tsogas, and A. Amditis. An advanced cooperative path prediction algorithm for safety applications in vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 12(3) :669–679, 2011.

[121] S. Mammar, N. M. Enache, S. Glaser, B. Lusetti, and A. B. Neto. Lane keeping automation at tire saturation. In *Proceedings of the 2010 American Control Conference*, pages 6466–6471. IEEE, 2010.

[122] Y. Mansour. Pessimistic decision tree pruning based on tree size. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 195–201. Citeseer, 1997.

[123] R. Marino, S. Scalzi, and M. Netto. Nested PID steering control for lane keeping in autonomous vehicles. *Control Engineering Practice*, 19(12) :1459–1467, 2011.

[124] M. McNaughton, C. Urmson, J. M. Dolan, and J.-W. Lee. Motion planning for autonomous driving with a conformal spatiotemporal lattice. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 4889–4895. IEEE, 2011.

[125] S. Melzi and E. Sabbioni. On the vehicle sideslip angle estimation through neural networks : Numerical and experimental results. *Mechanical Systems and Signal Processing*, 25(6) : 2005–2019, 2011.

[126] R. Miller and Q. Huang. An adaptive peer-to-peer collision warning system. In *Vehicular Technology Conference. IEEE 55th Vehicular Technology Conference. VTC Spring 2002 (Cat. No. 02CH37367)*, volume 1, pages 317–321. IEEE, 2002.

[127] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke, et al. Junior : The stanford entry in the urban challenge. *Journal of field Robotics*, 25(9) :569–597, 2008.

[128] A. Morand. *Commande assistée au conducteur basée sur la conduite en formation de type "banc de poissons"*. Thèse, Université de Bordeaux, 2014. URL https://tel.archives-ouvertes.fr/tel-01142147.

[129] A. Morand, X. Moreau, P. Melchior, M. Moze, and F. Guillemard. CRONE cruise control system. *IEEE Transactions on Vehicular Technology*, 65(1) :15–28, 2016.

[130] B. Morris, A. Doshi, and M. Trivedi. Lane change intent prediction for driver assistance : On-road design and evaluation. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 895–901. IEEE, 2011.

[131] H. Mouhagir, V. Cherfaoui, R. Talj, F. Aioun, and F. Guillemard. Using evidential occupancy grid for vehicle trajectory planning under uncertainty with tentacles. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–7. IEEE, 2017.

[132] B. Nagy and A. Kelly. Trajectory generation for car-like robots using cubic curvature polynomials. *Field and Service Robots*, 11, 2001.

[133] G. Naus, J. Ploeg, M. Van de Molengraft, W. Heemels, and M. Steinbuch. Design and implementation of parameterized adaptive cruise control : An explicit model predictive control approach. *Control Engineering Practice*, 18(8) :882–892, 2010.

[134] N. J. Nilsson. A mobile automaton : An application of artificial intelligence techniques. Technical report, SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGENCE CENTER, 1969.

[135] M. Nolte, M. Rose, T. Stolte, and M. Maurer. Model predictive control based trajectory generation for autonomous vehicles—an architectural approach. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 798–805. IEEE, 2017.

[136] R. T. O'Brien, P. A. Iglesias, and T. J. Urban. Vehicle lateral control for automated highway systems. *IEEE Transactions on Control Systems Technology*, 4(3) :266–273, 1996.

[137] J. J. Oh and S. B. Choi. Vehicle velocity observer design using 6-d imu and multiple-observer approach. *IEEE Transactions on Intelligent Transportation Systems*, 13(4) : 1865–1879, 2012.

[138] S.-Y. Oh, J.-H. Lee, and D.-H. Choi. A new reinforcement learning vehicle control architecture for vision-based road following. *IEEE Transactions on Vehicular Technology*, 49 (3) :997–1005, 2000.

[139] M. G. Ortiz, J. Fritsch, F. Kummert, and A. Gepperth. Behavior prediction at multiple time-scales in inner-city scenarios. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 1068–1073. IEEE, 2011.

[140] A. Oustaloup. La commande crone : commande robuste d'ordre non entier. 1991. *Hermès, Paris*.

[141] A. Oustaloup and B. Dubuisson. *Diversity and non-integer differentiation for system dynamics*. Wiley Online Library, 2014.

[142] A. Oustaloup, O. Cois, and L. Le Lay. *Représentation et identification par modèle non entier*. Lavoisier, 2005.

[143] H. Pacejka. *Tire and vehicle dynamics*. Elsevier, 2005.

[144] J.-H. Park and Y.-W. Tai. A simulation based method for vehicle motion prediction. *Computer Vision and Image Understanding*, 136 :79–91, 2015.

[145] R. Parthasarathi and T. Fraichard. An inevitable collision state-checker for a car-like vehicle. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3068–3073. IEEE, 2007.

[146] J. Pérez, V. Milanés, T. De Pedro, and L. Vlacic. Autonomous driving manoeuvres in urban road traffic environment : a study on roundabouts. *IFAC Proceedings Volumes*, 44 (1) :13795–13800, 2011.

[147] J. Pérez, V. Milanés, and E. Onieva. Cascade architecture for lateral control in autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 12(1) :73–82, 2011.

[148] B. Perrin. La voiture sans accident. *Auto Journal*, 895 :32–34, 2013.

[149] M. Pivtoraiko and A. Kelly. Generating near minimal spanning control sets for constrained motion planning in discrete state spaces. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3231–3237. IEEE, 2005.

[150] J. R. Quinlan. Induction of decision trees. *Machine learning*, 1(1) :81–106, 1986.

[151] S. Quinlan and O. Khatib. Elastic bands : Connecting path planning and control. In *[1993] Proceedings IEEE International Conference on Robotics and Automation*, pages 802–807. IEEE, 1993.

[152] M. Raharijaona, M. Duc, and M. Mammar. Linear parameter-varying control and H-infinity synthesis dedicated to lateral driving assistance. In *IEEE Intelligent Vehicles Symposium, 2004*, pages 407–412. IEEE, 2004.

[153] R. Rajamani. *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[154] Y. Rasekhipour, A. Khajepour, S.-K. Chen, and B. Litkouhi. A potential field-based model predictive path-planning controller for autonomous road vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(5) :1255–1267, 2017.

[155] L. R. Ray. Nonlinear state and tire force estimation for advanced vehicle control. *IEEE Transactions on Control Systems Technology*, 3(1) :117–124, 1995.

[156] C. Raymond. *Décodage conceptuel : co-articulation des processus de transcription et compréhension dans les systèmes de dialogue*. PhD thesis, These de doctorat, Universite d'Avignon et des Pays de Vaucluse, 2005.

[157] J.-B. Receveur, S. Victor, and P. Melchior. Multi-criteria trajectory optimization for autonomous vehicles. *IFAC-PapersOnLine*, 50(1) :12520–12525, 2017.

[158] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once : Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[159] T.-J. Ren, T.-C. Chen, and C.-J. Chen. Motion control for a two-wheeled vehicle using a self-tuning pid controller. *Control Engineering Practice*, 16(3) :365–375, 2008.

[160] P. Resende and F. Nashashibi. Real-time dynamic trajectory planning for highly automated driving in highways. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 653–658. IEEE, 2010.

[161] A. Rezaeian, R. Zarringhalam, S. Fallah, W. Melek, A. Khajepour, S.-K. Chen, N. Moshchuck, and B. Litkouhi. Novel tire force estimation strategy for real-time implementation on vehicle applications. *IEEE Transactions on Vehicular Technology*, 64(6) :2231–2241, 2015.

[162] D. E. Rumelhart, G. E. Hinton, R. J. Williams, et al. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3) :1, 1988.

[163] K. Saadeddin, M. F. Abdel-Hafez, M. A. Jaradat, and M. A. Jarrah. Performance enhancement of low-cost, high-accuracy, state estimation for vehicle collision prevention system using anfis. *Mechanical Systems and Signal Processing*, 41(1-2) :239–253, 2013.

[164] A. Sathyanarayana, P. Boyraz, and J. H. Hansen. Driver behavior analysis and route recognition by hidden markov models. In *Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference on*, pages 276–281. IEEE, 2008.

[165] A. Scheuer and T. Fraichard. Continuous-curvature path planning for car-like vehicles. In *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems. Innovative Robotics for Real-World Applications. IROS'97*, volume 2, pages 997–1003. IEEE, 1997.

[166] S. M. Shinners. *Modern control system theory and design*. John Wiley & Sons, 1998.

[167] M. Sokolova and G. Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4) :427–437, 2009.

[168] A. Stentz. Optimal and efficient path planning for partially known environments. In *Intelligent Unmanned Ground Vehicles*, pages 203–220. Springer, 1997.

[169] J. Stephant, A. Charara, and D. Meizel. Contact roue-sol : comparaison de modèles d'efforts. *Journées d'Etude : Automatique et Automobile*, 2001.

[170] J. Stéphant, A. Charara, and D. Meizel. Evaluation of a sliding mode observer for vehicle sideslip angle. *Control Engineering Practice*, 15(7) :803–812, 2007.

[171] R. S. Sutton, A. G. Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

[172] G. Tagne, R. Talj, and A. Charara. Higher-order sliding mode control for lateral dynamics of autonomous vehicles, with experimental validation. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 678–683. IEEE, 2013.

[173] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. In *Readings in Fuzzy Sets for Intelligent Systems*, pages 387–403. Elsevier, 1993.

[174] W. Tan, H. J. Marquez, and T. Chen. Operating point selection in multimodel controller design. In *Proceedings of the 2004 American Control Conference*, volume 4, pages 3652–3657. IEEE, 2004.

[175] F. Thau. Observing the state of non-linear dynamic systems. *International journal of control*, 17(3) :471–479, 1973.

[176] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, et al. Stanley : The robot that won the DARPA Grand Challenge. *Journal of field Robotics*, 23(9) :661–692, 2006.

[177] K. Tin Leung, J. F. Whidborne, D. Purdy, and A. Dunoyer. A review of ground vehicle dynamic state estimations utilising gps/ins. *Vehicle System Dynamics*, 49(1-2) :29–58, 2011.

[178] Q. Tran and J. Firl. Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 918–923. IEEE, 2014.

[179] M. Tsogas, X. Dai, G. Thomaidis, P. Lytrivis, and A. Amditis. Detection of maneuvers using evidence theory. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 126–131. IEEE, 2008.

[180] S. Tsugawa. Vision-based vehicles in Japan : Machine vision systems and driving control systems. *IEEE Transactions on industrial electronics*, 41(4) :398–405, 1994.

[181] S. Ulbrich and M. Maurer. Probabilistic online pomdp decision making for lane changes in fully automated driving. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 2063–2067. IEEE, 2013.

[182] United Nations. Convention on road traffic, Novembre 1968. URL https://treaties.un.org/doc/Treaties/1977/05/19770524%2000-13%20AM/Ch_XI_B_19.pdf. Vienne. Accessed 24 April 2019 at https://treaties.un.org/doc/Treaties/1977/05/19770524%2000-13%20AM/Ch_XI_B_19.pdf.

[183] United Nations Economic Comission for Europe. Rapport du groupe de travail de la sécurité et de la circulation routières sur sa soixante-huitième session, Avril 2014. URL http://www.unece.org/fileadmin/DAM/trans/doc/2014/wp1/ECE-TRANS-WP1-145-f.pdf. ECE/TRANS/WP.1/145, Genève. Accessed 24 April 2019 at http://www.unece.org/fileadmin/DAM/trans/doc/2014/wp1/ECE-TRANS-WP1-145-f.pdf.

[184] United Nations Economic Comission for Europe. Rapport du groupe de travail de la sécurité et de la circulation routières sur sa soixante-quatorzième session, Mars 2017. URL https://www.unece.org/fileadmin/DAM/trans/doc/2017/wp1/ECE-TRANS-WP1-157f.pdf. ECE/TRANS/WP.1/157, Genève. Accessed 24 April 2019 at http://www.unece.org/fileadmin/DAM/trans/doc/2014/wp1/ECE-TRANS-WP1-145-f.pdf.

[185] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, et al. Autonomous driving in urban environments : Boss and the urban challenge. *Journal of Field Robotics*, 25(8) :425–466, 2008.

[186] F. Von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wuensche. Driving with tentacles : Integral structures for sensing and motion. *Journal of Field Robotics*, 25(9) :640–673, 2008.

[187] J.-Y. Wang and M. Tomizuka. Robust H_∞ lateral control of heavy-duty vehicles in automated highway system. In *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, volume 5, pages 3671–3675. IEEE, 1999.

[188] G. Welch, G. Bishop, et al. An introduction to the Kalman filter. 1995.

[189] M. Wielitzka, M. Dagen, and T. Ortmaier. State estimation of vehicle's lateral dynamics using unscented kalman filter. In *53rd IEEE Conference on Decision and Control*, pages 5015–5020. IEEE, 2014.

[190] M. T. Wolf and J. W. Burdick. Artificial potential functions for highway driving with collision avoidance. In *2008 IEEE International Conference on Robotics and Automation*, pages 3731–3736. IEEE, 2008.

[191] G. Xie, H. Gao, L. Qian, B. Huang, K. Li, and J. Wang. Vehicle trajectory prediction by integrating physics-and maneuver-based approaches using interactive multiple models. *IEEE Transactions on Industrial Electronics*, 65(7) :5999–6008, 2018.

[192] W. Xu, J. Wei, J. M. Dolan, H. Zhao, and H. Zha. A real-time motion planner with trajectory optimization for autonomous vehicles. In *2012 IEEE International Conference on Robotics and Automation*, pages 2061–2067. IEEE, 2012.

[193] Z. Yacine. *Observateurs pour l'estimation de la dynamique latérale du véhicule : application à la détection de situations critiques.* Thèse, Université Paris-Saclay, 2016. URL http://www.theses.fr/2016SACLE031.

[194] J.-H. Yoon, S. E. Li, and C. Ahn. Estimation of vehicle sideslip angle and tire-road friction coefficient based on magnetometer with gps. *International journal of automotive technology*, 17(3) :427–435, 2016.

[195] S. Yoon, S.-E. Yoon, U. Lee, and D. H. Shim. Recursive path planning using reduced states for car-like vehicles on grid maps. *IEEE Transactions on Intelligent Transportation Systems*, 16(5) :2797–2813, 2015.

[196] Y. Yoon, J. Shin, H. J. Kim, Y. Park, and S. Sastry. Model-predictive active steering and obstacle avoidance for autonomous ground vehicles. *Control Engineering Practice*, 17(7) : 741–750, 2009.

[197] M. Zeitz. The extended Luenberger observer for nonlinear systems. *Systems & Control Letters*, 9(2) :149–156, 1987.

[198] B. Zhang, H. Du, J. Lam, N. Zhang, and W. Li. A novel observer design for simultaneous estimation of vehicle steering angle and sideslip angle. *IEEE Transactions on Industrial Electronics*, 63(7) :4357–4366, 2016.

[199] L.-H. Zhao, Z.-Y. Liu, and H. Chen. Design of a nonlinear observer for vehicle velocity estimation and experiments. *IEEE Transactions on Control Systems Technology*, 19(3) : 664–672, 2011.

[200] J. Zheng, K. Suzuki, and M. Fujita. Predicting driver's lane-changing decisions using a neural network model. *Simulation Modelling Practice and Theory*, 42 :73–83, 2014.

[201] J. Ziegler, M. Werling, and J. Schroder. Navigating car-like robots in unstructured environments using an obstacle sensitive cost function. In *2008 IEEE Intelligent Vehicles Symposium*, pages 787–791. IEEE, 2008.

Annexe A

L'approche fréquentielle pour l'analyse des systèmes dynamiques et la conception de régulateurs

L'analyse de fonctions et de signaux dans le domaine fréquentiel est utilisée dans les domaines de l'électronique, automatique et statistique. Contrairement à l'analyse temporelle, basée sur l'évolution des signaux au cours du temps, l'analyse fréquentielle est basée sur la répartition fréquentielle de ces signaux. Des opérateurs mathématiques tels que la transformée de Fourier ou la transformation de Laplace permettent de transposer les fonctions du domaine temporel au domaine fréquentiel et inversement. L'avantage d'utiliser ces fonctions dans le domaine fréquentiel plutôt que le domaine temporel est lié à la simplification des équations. En effet, une fonction différentielle linéaire revient à une équation polynomiale dans le domaine fréquentiel. Autre avantage, de nombreux outils et méthodes existent pour analyser le comportement entrée/sortie des systèmes physiques dans le domaine fréquentiel.

L'approche fréquentielle est grandement utilisée dans ce manuscrit et cette annexe présente donc les outils utilisés.

A.1 Outils pour l'analyse fréquentielle des systèmes dynamiques

A.1.1 Transformée de Fourier

La transformée de Fourier convertit une fonction intégrable du domaine temporel en une autre fonction décrivant son spectre fréquentiel et dont la variable indépendante correspond à la pulsation.

Définition 1. *Soit f une fonction intégrable sur $\mathbb{R}$. Sa transformée de Fourier, notée $\mathcal{F}(f)$, est donnée par :*

$$\omega \mapsto \mathcal{F}(f)(\nu) = \int_{-\infty}^{+\infty} f(t)e^{-2\pi i\nu t}\mathrm{d}t. \tag{A.1}$$

A.1.2 Transformée de Laplace

La transformée de Laplace est très similaire à la transformée de Fourier, sa description est la suivante

Définition 2. *Soit $f(t)$ une fonction d'une variable t intégrable sur $\mathbb{R}^+$. Sa transformée de Laplace unilatérale, notée $F(s)$, est donnée par :*

$$F(s) = \mathcal{L}\{f\}(s) = \int_{0^-}^{+\infty} f(t)e^{-st}\mathrm{d}t, \tag{A.2}$$

avec s une variable complexe telle que $s = \sigma + i\omega$.

Si f est intégrable sur $\mathbb{R}$, sa transformée de Laplace bilatérale s'écrit :

$$F(s) = \mathcal{L}_b\{f\}(s) = \int_{-\infty}^{+\infty} f(t)e^{-st}\mathrm{d}t. \tag{A.3}$$

Les transformées de Laplace unilatérale et bilatérale sont équivalentes dans le cas de fonctions causales, ce qui est le cas pour la dynamique du véhicule. La première notation sera donc privilégiée.

La transformée de Laplace est donc une fonction complexe d'une variable complexe alors que la transformée de Fourier est une fonction complexe d'une variable réelle. Si ces deux transformées permettent aussi bien d'exprimer la dualité temps-fréquence, la transformée de Laplace est plus simple à utiliser, car plus générique et fournit plus d'information que la transformée de Fourier. En effet, pour la transformée de Laplace, la variable s peut contenir une variable réelle. Les transformées de Fourier et de Laplace sont égales dans le cas de fonction causale pour $\sigma = 0$.

Dans ce mémoire, nous utiliserons la transformée de Laplace pour l'analyse fréquentielle des modèles dynamiques. Pour arriver à exprimer ces modèles dans le domaine fréquentiel, les propriétés mathématiques suivantes de la fonction de Laplace sont utilisées.

Propriété 1. *Linéarité de la transformée de Laplace :*
Soient deux fonctions $f(t)$ et $g(t)$ ainsi que deux constantes a et b appartenant à $\mathbb{R}$, alors :

$$\mathcal{L}\{af + bg\}(s) = a\mathcal{L}\{f\}(s) + b\mathcal{L}\{g\}(s). \tag{A.4}$$

Propriété 2. *Transformée de Laplace d'une fonction dérivée d'ordre n :*
Soit $f(t)$ une fonction définie sur $\mathbb{R}^+$ dérivable n fois avec $n \in \mathbb{N}$ et à conditions initiales nulles, alors :

$$\mathcal{L}\{f^{(n)}\}(s) = s^n\mathcal{L}\{f\}(s). \tag{A.5}$$

A.1.3 Représentation d'état

Une représentation d'état est un modèle mathématique représentant un système physique par l'intermédiaire de variables d'état, d'entrées et de sorties. Les variables d'état sont des grandeurs physiques du système qui dépendent de leurs valeurs précédentes et des valeurs des entrées. La représentation d'état permet donc de modéliser le comportement interne du système. Cette représentation peut être continue ou discrète et linéaire ou non linéaire.

L'écriture générique d'une représentation d'état linéaire à p entrées, q sorties et n variables d'état s'écrit :

$$\begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) + D(t)u(t) \end{cases} \tag{A.6}$$

avec

— $x \in \mathbb{R}^n$, le vecteur d'état ;
— $y \in \mathbb{R}^q$, le vecteur de sortie ;
— $u \in \mathbb{R}^p$, le vecteur de commande ;
— $A(t) \in \mathbb{R}^{nxn}$, la matrice d'état ;
— $B(t) \in \mathbb{R}^{nxp}$, la matrice de commande ;
— $C(t) \in \mathbb{R}^{qxn}$, la matrice d'observation ;
— $D(t) \in \mathbb{R}^{qxp}$, la matrice d'action directe.

La représentation d'état est donc une représentation matricielle de l'évolution physique d'un système. À partir de cette représentation d'état, il est possible de calculer la fonction de transfert qui permettra l'analyse fréquentielle du système.

A.1.4 Fonction de transfert

Contrairement à la représentation d'état qui modélise l'état physique interne d'un système physique, la fonction de transfert représente les liens entre une entrée et une sortie de ce système. La fonction de transfert peut s'exprimer à partir de la représentation d'état, pour un système linéaire invariant (*i.e* les matrices d'état ne dépendent pas du temps) :

$$H(s) = C(sI_n - A)^{-1}B + D. \tag{A.7}$$

$H(s)$ est une matrice de transfert à qxp dimensions.

Il est à noter que la fonction de transfert ne tient pas compte des conditions aux limites en supposant les conditions initiales nulles.

Soit $H(s) = \dfrac{Num(s)}{Den(s)}$ une fonction de transfert rationnelle, avec $Num(s)$ et $Den(s)$ les fonctions de Laplace du numérateur et du dénominateur. Les pôles de $H(s)$ sont les racines de $Den(s) = 0$ et les zéros de $H(s)$ sont les racines de $Num(s) = 0$. À partir des pôles de la fonction de transfert, il est possible de déterminer la stabilité du système :

Propriété 3. *Un système représenté par une fonction de transfert $H(s)$ est stable EBSB (Entrée Bornée / Sortie Bornée) si les pôles de $H(s)$ sont à partie réelle strictement négative.*

A.1.5 Diagramme de Bode

À partir de la fonction de transfert du système, il devient possible de représenter graphiquement son comportement dans le domaine fréquentiel avec un diagramme de Bode. Il existe un diagramme de Bode d'amplitude et un diagramme de Bode de phase.

Le diagramme de Bode d'amplitude trace le gain de la fonction de transfert en décibel en fonction de la fréquence angulaire. Pour une fonction de transfert $H(s)$, il est calculé par

$$H_{dB}(\omega) = 20\log_{10}(|H(j\omega)|). \tag{A.8}$$

Le diagramme de Bode de phase trace la phase de la fonction de transfert en degré en fonction de la fréquence angulaire. Pour une fonction de transfert $H(s)$, il est calculé par

$$H_{\varphi}(\omega) = \arg(H(j\omega)). \tag{A.9}$$

La fréquence de coupure est la fréquence pour laquelle l'amplitude est à $1/\sqrt{2}$ de la valeur maximale de $|H(j\omega)|$. En décibel, la fréquence de coupure ω_c se situe à $H_{dB}(j\omega_c) = H_{dBmax} - 3$ dB. La bande passante est l'ensemble des fréquences pour lesquelles $H_{dB} > -3$ dB. Par exemple pour une fonction de transfert du premier ordre :

$$H_0(s) = \frac{1}{1 + \frac{s}{5}}, \tag{A.10}$$

$\omega_c = 5$rad/s et ses diagrammes de Bode en amplitude et en phase sont tracés sur la Figure A.1.

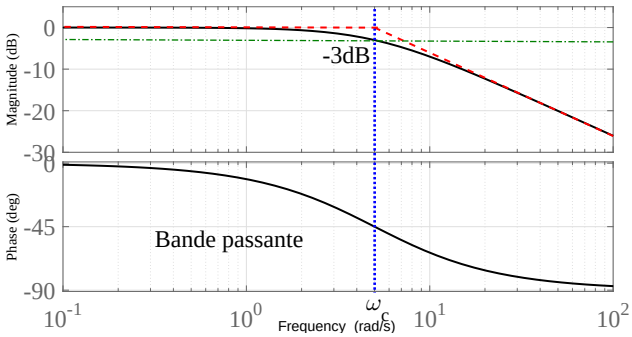


FIGURE A.1 – Diagrammes de Bode de $H_0(s)$

A.2 Les outils pour la conception de régulateurs et l'analyse de systèmes bouclés dans le domaine fréquentiel

La conception de régulateurs dans le domaine fréquentiel utilise comme base les méthodes décrites dans la section précédente et d'autres outils et concepts que nous allons décrire ici.

Un système régulé est un système bouclé pouvant être représenté comme sur la Figure A.2. Sur cette figure, $G(s)$ représente la fonction de transfert du système, $C(s)$ la fonction de transfert du régulateur, Y_{ref} l'entrée désirée, Y_{mes} la sortie mesurée du système, ϵ l'erreur de sortie et U la commande. Des perturbations externes peuvent influencer l'évolution du système, des perturbations sur la commande notées P_u et des perturbations sur la sortie P_y. Les capteurs mesurant la sortie de la boucle de régulation n'étant pas parfaits, il existe un bruit de mesure modélisé sur le schéma bloc de contrôle par le signal B_m.

Le régulateur est le module permettant de calculer la commande à appliquer à l'actionneur afin d'annuler l'erreur de sortie. En règle général, ce régulateur doit respecter un cahier des charges en matière de rapidité, précision et stabilité du système bouclé afin de répondre aux exigences du système et de ses actionneurs. Ses performances peuvent être analysées en utilisant les fonctions de transfert de la boucle ouverte et des fonctions de sensibilité.

A.2.1 Boucle ouverte et boucle fermée

Lorsque l'on parle de boucle ouverte, la rétroaction n'est pas prise en compte. La sortie $y(t)$ du système se calcule donc à partir de la référence $y_{ref}(t)$ en domaine fréquentiel par :

$$\left.\frac{Y(s)}{Y_{ref}(s)}\right|_{BO} = C(s)G(s). \tag{A.11}$$

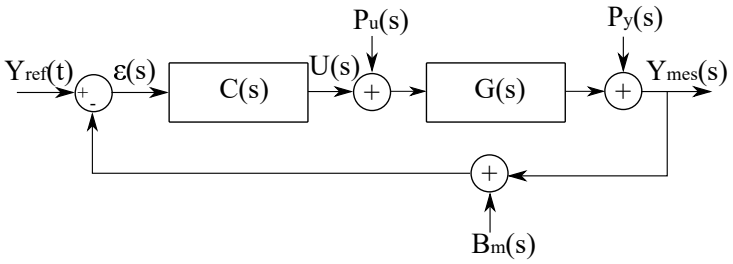


FIGURE A.2 – Schéma bloc d'un système bouclé

(A.11) est la fonction de transfert en boucle ouverte (FTBO) du système bouclé.

Par opposition, la boucle fermée prend bien en compte la rétroaction et le calcul de $y(t)$ donne, toujours en fréquentiel :

$$\left.\frac{Y(s)}{Y_{ref}(s)}\right|_{BF} = \frac{C(s)G(s)}{1+C(s)G(s)}, \tag{A.12}$$

c'est la fonction de transfert en boucle fermée (FTBF).

A.2.2 Point critique

En régime sinusoïdal, il existe une pulsation maximale ω_s pour laquelle le système bouclé est stable. La limite de stabilité est atteinte lorsque le dénominateur de (A.12) est nul, c'est à dire :

$$1 + C(j\omega_s)G(j\omega_s) = 0. \tag{A.13}$$

Cela revient à dire qu'à cette pulsation ω_s, la fonction de transfert en boucle ouverte à un gain de 1 (ou 0 dB) et un argument de $-180°$. C'est le point critique.

A.2.3 Marge de gain et marge de phase

Les marges de phase et de gain sont des indicateurs de stabilité et de précision de la boucle fermée. Elles indiquent à quelle « distance » du point critique se trouve la fonction de transfert en boucle ouverte.

La marge de phase M_ϕ est définie à partir de la fréquence au gain unité de la boucle ouverte ω_u :

$$M_\phi = \arg\left(C(j\omega_u)G(j\omega_u)\right) + \pi, \tag{A.14}$$

sachant que

$$|C(j\omega_u)G(j\omega_u)| = 1. \tag{A.15}$$

La marge de gain M_G est quant à elle définie à partir de la fréquence ω_π pour laquelle la phase de la boucle ouverte vaut $-180°$:

$$\arg\left(C(j\omega_\pi)G(j\omega_\pi)\right) = -180°. \tag{A.16}$$

La marge de gain est alors l'écart entre le gain à cette pulsation et 0 dB :

$$M_G = -20\log_{10}\left|C(j\omega_\pi)G(j\omega_\pi)\right|. \tag{A.17}$$

Un système est stable en boucle fermée si la marge de phase est positive. La marge de gain correspond au gain maximal pouvant être rajouté en boucle ouverte sans risquer de rendre instable la boucle fermée. Des valeurs usuelles garantissent une bonne stabilité et précision. Pour cela, la marge de phase doit être comprise entre 45° et 60° et la marge de gain doit être supérieure à 6 dB.

La lecture de ces deux marges sur un diagramme de Bode est représentée par la Figure A.3. Sur cette figure, la fonction de transfert de la boucle ouverte tracée est :

$$H_1(s) = \frac{1}{s}\frac{1}{1+0.4s+0.1s^2}. \tag{A.18}$$

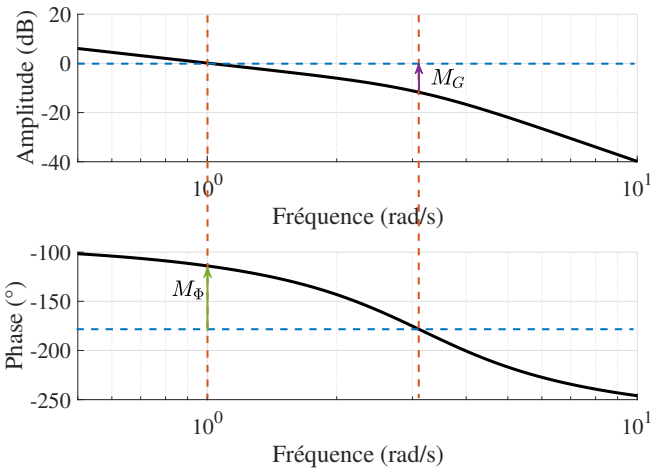


FIGURE A.3 – Lecture des marges de phase et de gain sur un diagramme de Bode

La pulsation au gain unité vaut $\omega_u = 1,02$ rad/s. À cette fréquence, la marge de phase vaut $65,6°$ et la marge de gain, mesurée à $\omega_\pi = 3,16$ rad/s, vaut 12 dB.

A.2.4 Diagramme de Black-Nichols

Le diagramme de Black est une méthode de tracé des fonctions de transfert. Dans ce diagramme, le gain de la fonction de transfert en dB est représenté en fonction de l'argument en degré. Le lieu de la fonction de transfert est paramétré en fonction la pulsation ω. Ainsi, le diagramme de Black regroupe les deux diagrammes de Bode en gain et en phase dans une même figure. La Figure A.4 représente la fonction de transfert $H_0(s)$ (A.10) dans un diagramme de Black.

Il est très fréquent de tracer l'abaque de Nichols dans le diagramme de Black. On parle alors de diagramme de Black-Nichols. Cet abaque permet, à partir de la courbe représentant la boucle ouverte dans le diagramme de Black, de retrouver les valeurs de la boucle fermée. Pour cela, l'abaque trace des contours à isogain et isophase de la boucle fermée en utilisant la transformée :

$$z \to \frac{z}{1+z}, \tag{A.19}$$

modélisant le passage de la boucle ouverte à la boucle fermée. Des contours isogain (en bleu) et isophases (en noir) sont représentés sur la Figure A.5 pour différentes valeurs. Le point critique

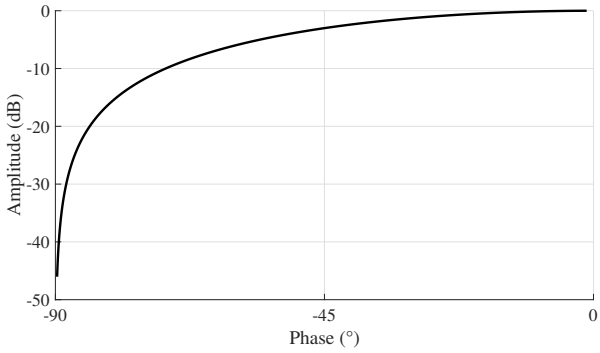


FIGURE A.4 – Fonction $H_0(s)$ tracée dans le plan de Black

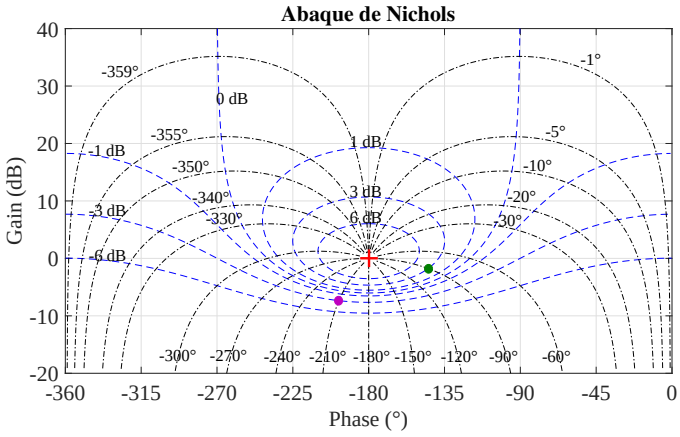


FIGURE A.5 – Abaque de Nichols dans le plan de Black

est marqué d'une croix rouge.

Ainsi, si la boucle ouverte d'un système régulé est tracée sur la Figure A.5 et passe par les points vert et violet, aux pulsations correspondantes de la boucle ouverte, la boucle fermée aura un gain de 6 dB et une phase de $-90°$ pour le point vert et un gain de -3 dB et une phase de $-210°$ pour le point violet.

Le plan de Black-Nichols est donc utile pour le réglage du régulateur et l'évaluation des performances de la boucle fermée. D'un côté, les marges de gain et de phase sont mesurables directement, de l'autre le facteur de résonance de la boucle fermée peut aussi directement être évalué avec l'abaque de Nichols. Ce facteur de résonance est mesuré sur le diagramme de Black-Nichols comme étant la valeur du contour d'amplitude tangenté par le lieu de la boucle ouverte. Une règle usuelle en automatique est d'avoir le lieu de la boucle ouverte tangent, au plus, au contour à 2,3 dB.

A.2.5 Fonctions de sensibilité

À partir du schéma bloc de la Figure A.2, les relations entre la commande et la sortie mesurée par rapport à la consigne et aux perturbations donnent :

$$\begin{aligned} \epsilon(s) &= Y_{ref}(s) - B_m(s) - Y_{mes}(s) \\ U(s) &= C(s)\epsilon(s) \\ Y_{mes}(s) &= P_y(s) + G(s)\left(P_u(s) + U(s)\right), \end{aligned} \tag{A.20}$$

on obtient ensuite :

$$\begin{aligned} U(s) &= C(s)\Big[Y_{ref}(s) - B_m(s) - \Big(P_y(s) + G(s)\big(P_u(s) + U(s)\big)\Big)\Big] \\ Y_{mes}(s) &= P_y(s) + G(s)\Big[P_u(s) + C(s)\big(Y_{ref}(s) - B_m(s) - Y_{mes}(s)\big)\Big], \end{aligned} \tag{A.21}$$

et finalement :

$$\begin{aligned} U(s)\big(1 + C(s)G(s)\big) &= C(s)Y_{ref}(s) - C(s)B_m(s) - C(s)P_y(s) - C(s)G(s)P_u(s) \\ Y_{mes}(s)\big(1 + C(s)G(s)\big) &= C(s)G(s)Y_{ref}(s) - C(s)G(s)B_m(s) + P_y(s) + G(s)P_u(s). \end{aligned} \tag{A.22}$$

À partir de (A.22), quatre fonctions de sensibilité sont définies pour l'analyse des systèmes bouclés. Ainsi, en considérant les autres entrées nulles, ces fonctions s'écrivent :

- la fonction de sensibilité complémentaire : $T(s) = \dfrac{Y_{mes}(s)}{Y_{ref}(s)} = \dfrac{C(s)G(s)}{1 + C(s)G(s)}$,
- la fonction de sensibilité : $S(s) = \dfrac{Y_{mes}(s)}{P_y(s)} = \dfrac{1}{1 + C(s)G(s)}$,
- la fonction de sensibilité aux perturbations en entrée : $GS(s) = \dfrac{Y_{mes}(s)}{P_u(s)} = \dfrac{G(s)}{1 + C(s)G(s)}$,
- la fonction de sensibilité en entrée : $CS(s) = \dfrac{U(s)}{Y_{ref}(s)} = \dfrac{C(s)}{1 + C(s)G(s)}$.

Pour illustrer l'analyse de ces fonctions de sensibilité , le système de régulation suivant est défini. Soit un système physique modélisé par une fonction de transfert du deuxième ordre $H_1(s)$ telle que :

$$H_1(s) = \frac{1}{1 + (2\zeta_1/\omega_1)s + s^2/\omega_1^2} = \frac{1}{1 + 0,2s + 0,1s^2}, \tag{A.23}$$

avec $\zeta_1 = 0,32$ le facteur d'amortissement et $\omega_1 = 3,16$ rad/s la fréquence propre du système. Pour l'analyse des systèmes du premier et second ordre, le lecteur pourra se référer à [67].

La fonction de transfert de H_1 est tracée sur un digramme de Bode Figure A.6. Les principales caractéristiques d'une fonction de transfert du deuxième ordre sont retrouvées : une pente de 40 dB par décade en hautes fréquences, une résonance à ω_R et une phase qui varie de 0 à $-180°$.

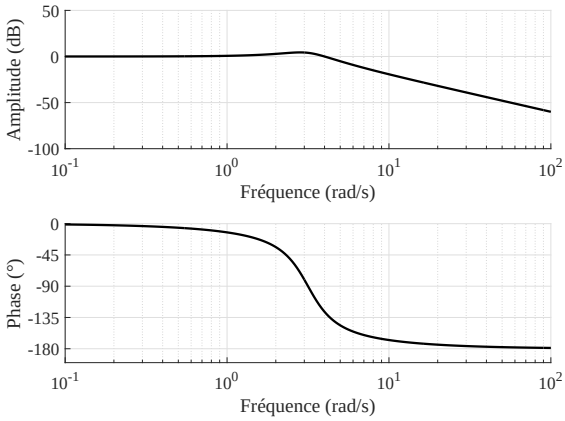
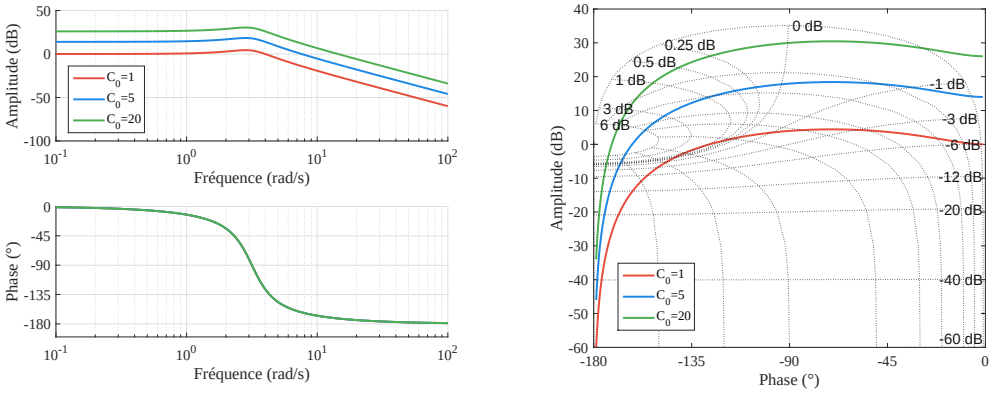


FIGURE A.6 – Diagrammes de Bode de $H_1(s)$



(a) (b)

FIGURE A.7 – Diagrammes de Bode (a) et de Black-Nichols (b) de la boucle ouverte $H_1(s)C_0$

Un régulateur $C(s) = C_0$ proportionnel est utilisé pour réguler la sortie. Plusieurs valeurs de C_0 sont utilisées : 1, 5 et 20. Les diagrammes de Bode et de Black-Nichols de la boucle ouverte sont visibles sur les Figure A.7 (a) et (b) respectivement.

Le régulateur ajouté au système bouclé étant un simple gain, l'argument de la fonction de transfert en boucle ouverte n'est pas modifié. Seule l'amplitude augmente et cela se traduit sur les Figures A.7 (a) et (b) par le glissement des lieux des fonctions de transfert vers le haut. Sur le diagramme de Bode en gain, le fait d'augmenter C_0 augmente la fréquence au gain unité et diminue la marge de gain. En conséquence, le système bouclé sera plus rapide, mais avec un degré de stabilité plus faible. Sur le diagramme de Black-Nichols, l'augmentation de la marge de phase avec un régulateur plus grand est aussi visible. Le fait que le lieu de la fonction de transfert en boucle ouverte se rapproche du point critique avec un régulateur plus grand est beaucoup plus facile à analyser avec le diagramme de Black-Nichols. Sur ce même diagramme, les isogains tangentés par les lieux des boucles ouvertes sont d'autant plus grands que C_0 augmente traduisant un facteur de résonance plus important en boucle fermée.

Les fonctions de sensibilité du système bouclé sont tracées Figure A.8. L'augmentation du gain du régulateur influence la fonction de sensibilité complémentaire $T(s)$ (Figure A.8 (a))

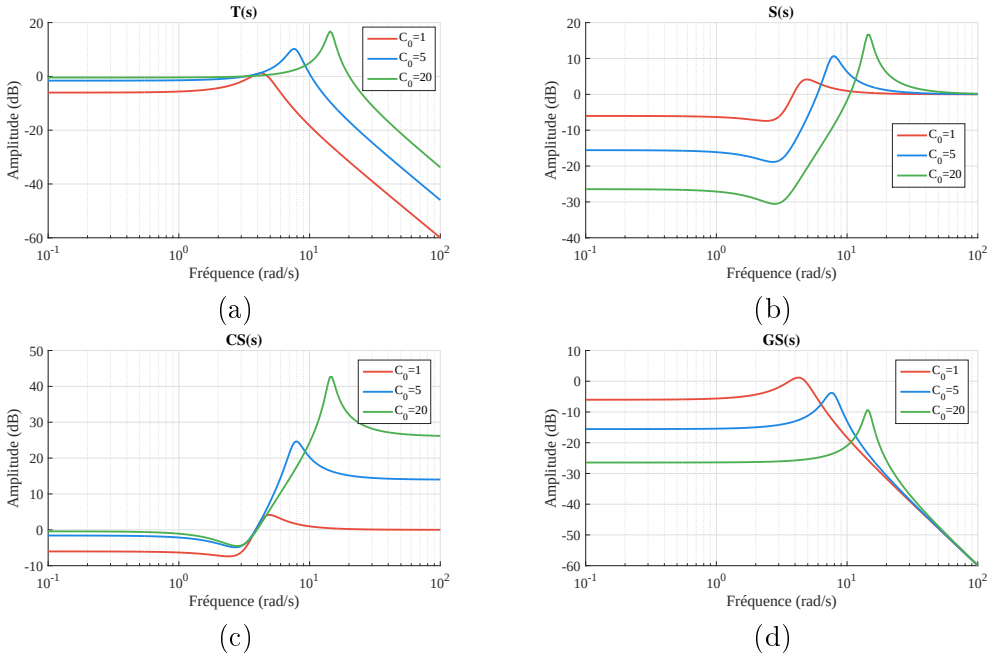


FIGURE A.8 – Diagrammes de Bode des gains des fonctions de sensibilité $T(s)$ (a), $S(s)$ (b), $CS(s)$ (c) et $GS(s)$ (d)

par une augmentation du facteur de résonance et de la bande passante. L'augmentation du facteur de résonance est aussi visible sur le diagramme de Bode de la fonction de sensibilité $S(s)$ (Figure A.8 (b)), où l'on observe une désensibilisation et une diminution de la marge de module quand C_0 augmente. Les fonctions de sensibilité aux perturbations en entrée $GS(s)$ et en entrée $CS(s)$ (Figure A.8 (c) et (d)) indiquent quant à elles une sensibilité de l'entrée accrue, mais une réjection des perturbations plus importante et plus rapide.

Annexe B

Influence des variations paramétriques sur la dynamique latérale du véhicule

De manière générale, un régulateur est calculé en utilisant les paramètres nominaux du procédé. Les performances de la régulation sont alors conditionnées par la connaissance des paramètres du procédé et leurs variations. Dans le cadre de la régulation de la dynamique latérale du véhicule, le Chapitre 3 fait état de l'influence de la vitesse longitudinale sur la dynamique latérale et présente un régulateur adapté à la variation de vitesse.

Outre la vitesse longitudinale, la fonction de transfert entre l'angle volant et la position latérale a ces paramètres dépendant de la masse du véhicule et de l'adhérence. Cette annexe présente comment ces deux paramètres influencent chaque paramètre de la fonction de transfert et la fonction de transfert au global.

B.1 Influence de la masse

Variation des paramètres

Pour rappel, les paramètres nominaux utilisé dans les Chapitres 2 et 3 sont donnés dans la Table B.1.

M_t	1759 kg
M_f	1319 kg
M_r	440 kg
I_z	2638,5 kg.m^2
L_f	0,71 m
L_r	2,13 m
c_{yf} $(\mu = 1)$	94446 N.rad^{-1}
c_{yr} $(\mu = 1)$	48699 N.rad^{-1}

Table B.1 – Paramètres nominaux du véhicule

Avec cette configuration, cela revient à avoir une répartition entre les masses avant et arrière M_f et M_r telle que

$$M_f = 0,75M_t$$
$$M_r = 0,25M_t.$$

Pour un Citroën C4 Picasso, la masse totale autorisée en charge est de 2100 kg et la masse à vide de 1500 kg. Ainsi, pour évaluer l'influence de la masse sur la dynamique latérale du véhicule, les configurations suivantes de répartition de masse sont utilisées :

- $M_t = 1500$ kg et répartition avant/arrière 75/25
- $M_t = 1500$ kg et répartition avant/arrière 60/40
- $M_t = 1759$ kg et répartition avant/arrière 75/25
- $M_t = 1759$ kg et répartition avant/arrière 60/40
- $M_t = 2100$ kg et répartition avant/arrière 75/25
- $M_t = 2100$ kg et répartition avant/arrière 60/40.

Les empattements avant et arrière sont calculés en fonctions des masses avant et arrière par :

$$L_f = M_r \frac{L}{M_t}$$
$$L_r = M_f \frac{L}{M_t}.$$

De même le moment d'inertie I_z dépend aussi de la masse de par la formule :

$$I_z = \rho^2 M_t,$$

avec ρ le rayon de giration, tel que

$$\frac{\rho}{\sqrt{L_f L_r}} = \text{cste} \simeq 1,$$

et avec les paramètres nominaux :

$$\frac{\rho}{\sqrt{L_f L_r}} = 0,9959.$$

Influence sur le procédé

Pour analyser l'influence des variations paramétriques sur la dynamique latérale, le modèle linéaire est utilisé. La fonction de transfert reliant l'angle volant à la position latérale du centre de gravité s'écrit :

$$\frac{Y_G(s)}{\theta_v(s)} = \frac{K_0'}{s^2} \frac{1 + 2\zeta_1'(s/\omega_1') + s^2/\omega_1'^2}{1 + 2\zeta_0'(s/\omega_0') + s^2/\omega_0'^2},$$

avec

$$K_0' = \frac{2}{\lambda} \frac{c_{yf} c_{yr} V_{x0}^2 L}{2 c_{yf} c_{yr} L^2 - M_t {V_{x0}}^2 (L_f c_{yv} - L_r c_{yr})},$$

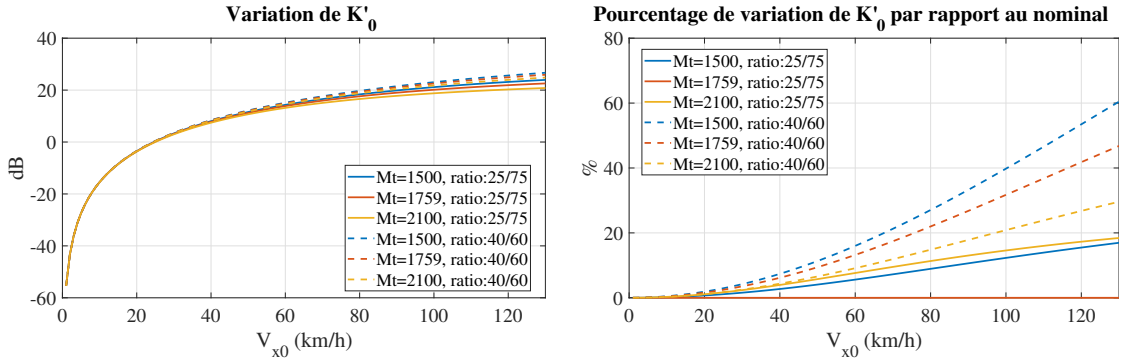
$$\zeta_0' = \frac{M_t (L_f^2 c_{yf} + L_r^2 c_{yr}) + I_z (c_{yf} + c_{yr})}{\sqrt{2 I_z M_t (2 c_{yf} c_{yr} L^2 - M_t {V_{x0}}^2 (L_f c_{yv} - L_r c_{yr}))}},$$

$$\omega_0' = \sqrt{\frac{2}{I_z M_t {V_{x0}}^2}} \sqrt{2 c_{yf} c_{yr} L^2 - M_t V_{x0}^2 (L_f c_{yv} - L_r c_{yr})},$$

$$\zeta_1' = \frac{L_r}{V_{x0}} \sqrt{\frac{c_{yr} L}{2 I_z}},$$

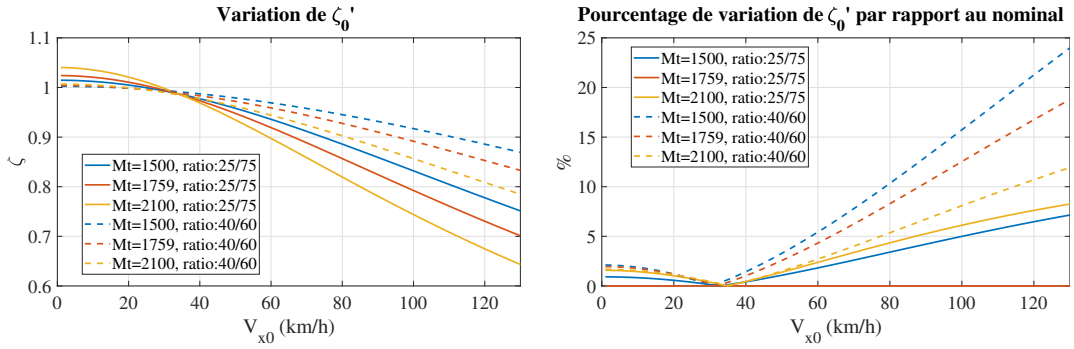
$$\omega_1' = \sqrt{2 \frac{c_{yr} L}{I_z}}.$$

Les Figures B.1, B.2, B.3, B.4 et B.5 tracent en (a) les variations de ces paramètres pour les différentes configurations en fonction de la vitesse longitudinale V_x. En (b), les courbes représentent les variations relatives de ces paramètres à leurs valeurs nominales en fonction de V_x.



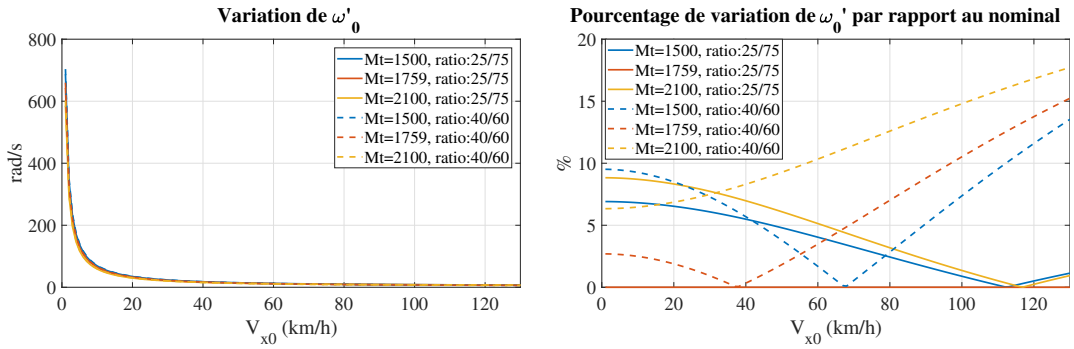
(a) (b)

FIGURE B.1 – Variation de K'_0 en fonction de la vitesse pour plusieurs configurations de répartition de masse (a) et variation relative aux paramètres nominaux (b)



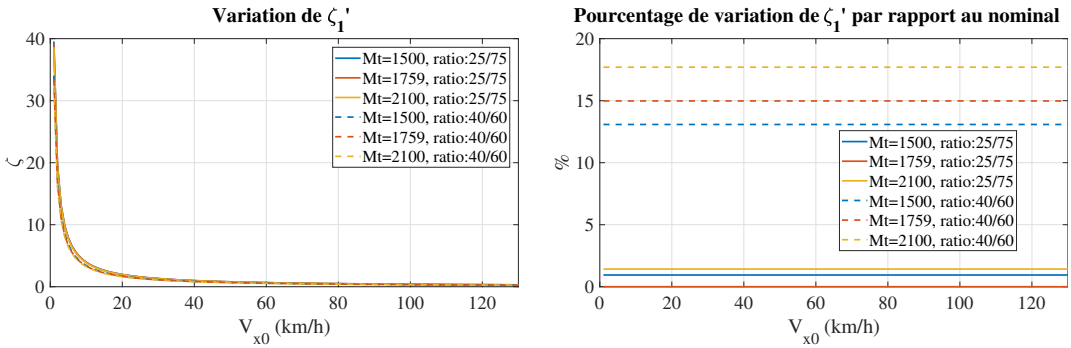
(a) (b)

FIGURE B.2 – Variation de ζ'_0 en fonction de la vitesse pour plusieurs configurations de répartition de masse (a) et variation relative aux paramètres nominaux (b)



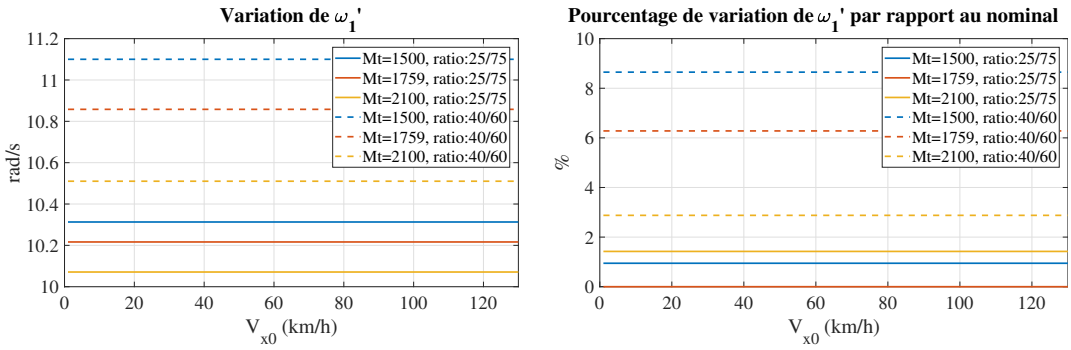
(a) (b)

FIGURE B.3 – Variation de ω'_0 en fonction de la vitesse pour plusieurs configurations de répartition de masse (a) et variation relative aux paramètres nominaux (b)



(a) (b)

FIGURE B.4 – Variation de ζ_1' en fonction de la vitesse pour plusieurs configurations de répartition de masse (a) et variation relative aux paramètres nominaux (b)



(a) (b)

FIGURE B.5 – Variation de ω_1' en fonction de la vitesse pour plusieurs configurations de répartition de masse (a) et variation relative aux paramètres nominaux (b)

Tous les paramètres de la fonction de transfert sont impactés par la configuration de masse du véhicule. Si la masse totale est une composante importante des paramètres de la fonction de transfert, sa répartition sur les essieux avant et arrière l'est tout autant. En effet, à masse totale identique, une répartition différente influence tous les paramètres de la fonction de transfert. Globalement, les variations sont de maximum 20% pour les paramètres ω_0', ζ_0', ω_1' et ζ_1', et de 60% pour le paramètre K_0'.

La Figure B.6 trace les représentations fréquentielles de la fonction de transfert dans les diagrammes de Bode pour toutes les configurations de la masse et pour les vitesses 1 km/h, 10 km/h, 30 km/h, 50 km/h, 70 km/h, 90 km/h, 110 km/h et 130 km/h. Chaque couleur représentant une vitesse. Étant donné que l'influence de la masse est plus importante à mesure que la vitesse augmente, les domaines de variation paramétrique de la fonction de transfert sont plus importants quand la vitesse augmente.

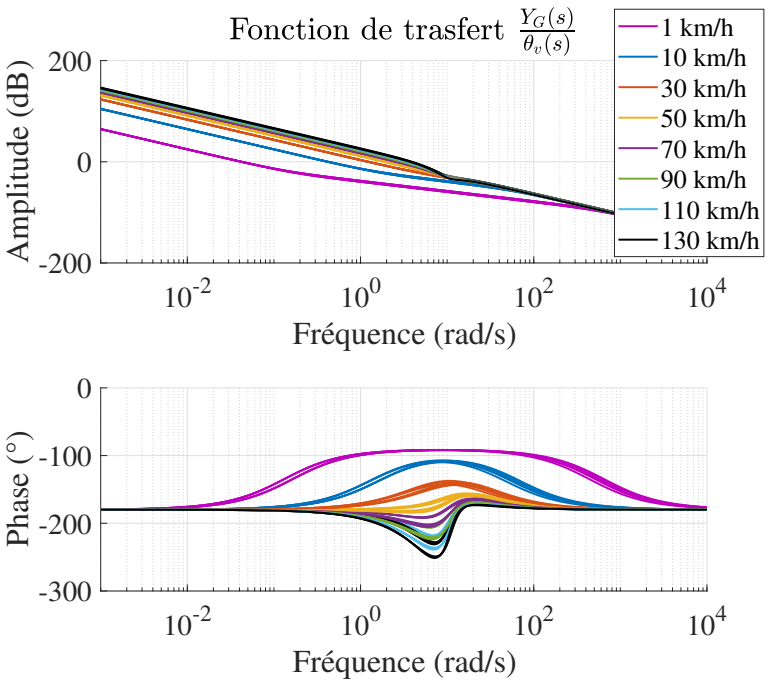


FIGURE B.6 – $G(s)$

B.2 Influence de l'adhérence

Variation des paramètres

L'adhérence de la route μ intervient de manière proportionnelle sur les coefficients de rigidité de dérive des pneumatiques c_{yf} et c_{yr}. Sur une route sèche, $\mu = 1$ et sur une route verglacée $\mu = 0,1$.

Par rapport aux paramètres nominaux c_{yfn} et c_{yrn} sur route sèche avec $\mu = 1$, pour une adhérence μ différente, les coefficients sont calculés par :

$$c_{yf} = \mu c_{yfn}$$
$$c_{yr} = \mu c_{yrn}.$$

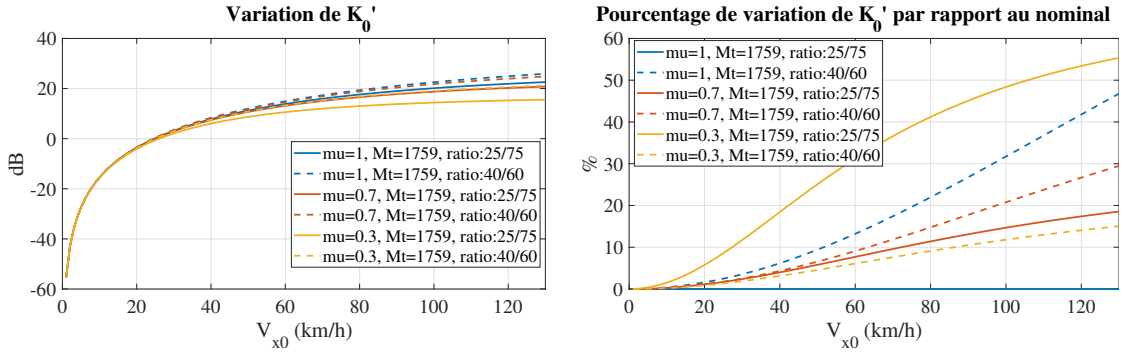
Pour évaluer l'influence de l'adhérence sur les paramètres de la fonction de transfert, les configurations suivantes sont utilisées :

- $\mu = 1$, $M_t = 1759$ kg et répartition avant/arrière 75/25
- $\mu = 1$, $M_t = 1759$ kg et répartition avant/arrière 60/40
- $\mu = 0,7$, $M_t = 1759$ kg et répartition avant/arrière 75/25
- $\mu = 0,7$, $M_t = 1759$ kg et répartition avant/arrière 60/40
- $\mu = 0,3$, $M_t = 1759$ kg et répartition avant/arrière 75/25
- $\mu = 0,3$, $M_t = 1759$ kg et répartition avant/arrière 60/40.

Influence sur le procédé

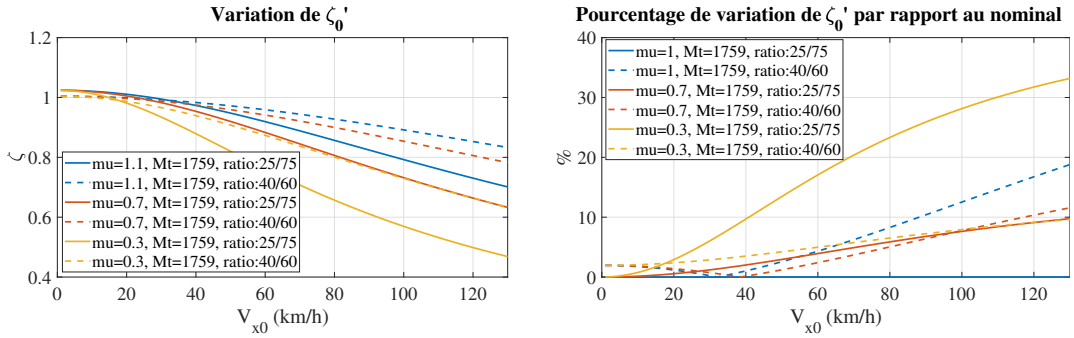
Les Figures B.7, B.8, B.9, B.10 et B.11 tracent en (a) les variations de ces paramètres pour les différentes configurations en fonction de la vitesse longitudinale V_x. En (b), les courbes

représentent les variations relatives de ces paramètres à leurs valeurs nominales en fonction de V_x.



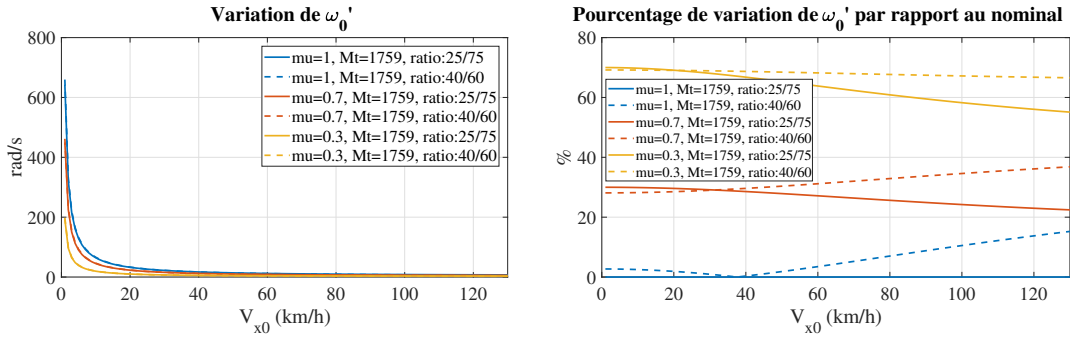
(a) (b)

FIGURE B.7 – Variation de K_0' en fonction de la vitesse pour plusieurs configurations d'adhérence (a) et variation relative aux paramètres nominaux (b)



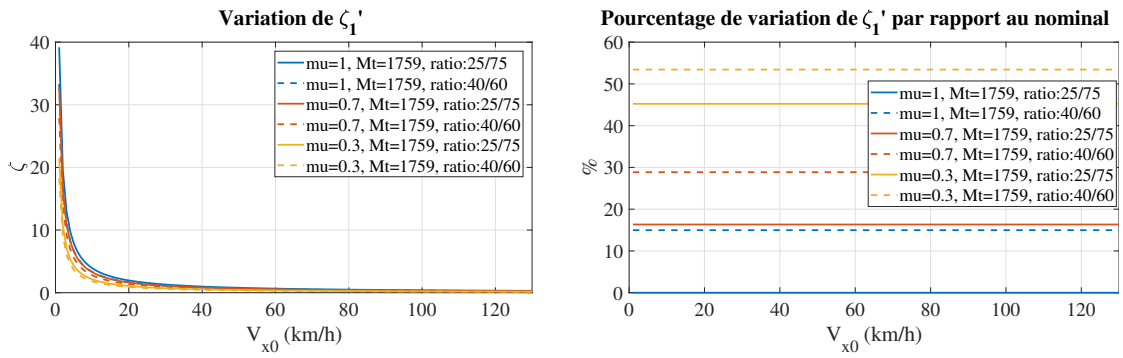
(a) (b)

FIGURE B.8 – Variation de ζ_0' en fonction de la vitesse pour plusieurs configurations d'adhérence (a) et variation relative aux paramètres nominaux (b)



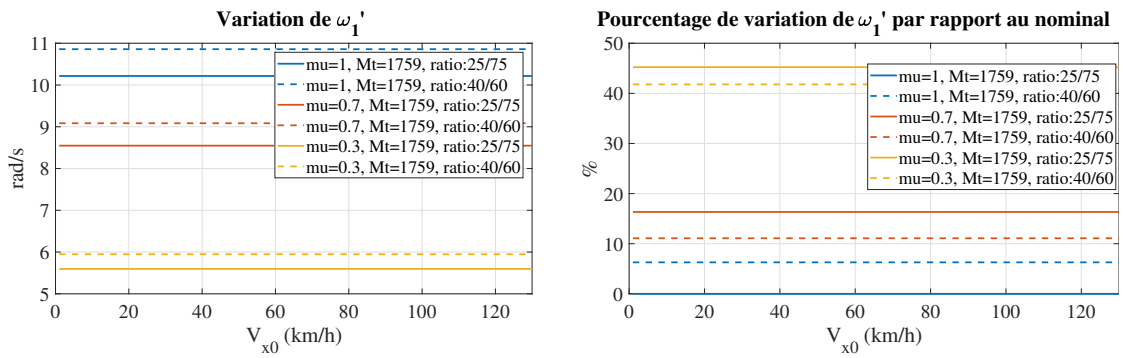
(a) (b)

FIGURE B.9 – Variation de ω_0' en fonction de la vitesse pour plusieurs configurations d'adhérence (a) et variation relative aux paramètres nominaux (b)



(a) (b)

FIGURE B.10 – Variation de ζ_1' en fonction de la vitesse pour plusieurs configurations d'adhérence (a) et variation relative aux paramètres nominaux (b)



(a) (b)

FIGURE B.11 – Variation de ω_1' en fonction de la vitesse pour plusieurs configurations d'adhérence (a) et variation relative aux paramètres nominaux (b)

A mesure que l'adhérence diminue, la variation de tous les paramètres composant la fonction de transfert augmente. Alors que l'influence de la masse était de l'ordre de 20% maximum à haute vitesse, elle est de plus de 50% à faible adhérence, pour $\mu = 0,3$. L'influence de l'adhérence est prépondérante par rapport à celle de la masse. Cela est aussi visible sur les diagrammes de Bode de la fonction de transfert pour les différentes combinaisons de paramètres et d'adhérence, Figure B.12. Les domaines d'incertitudes sont plus « étalés ».

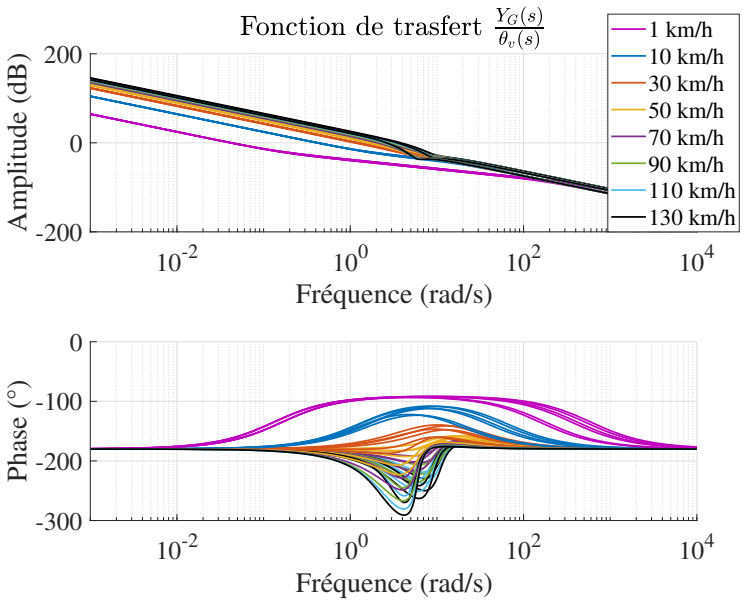


FIGURE B.12 – $G(s)$

Annexe C

Quelques méthodes pour l'observation des systèmes dynamiques

Cette annexe porte sur les observateurs d'état des systèmes dynamiques. On considérera les représentations d'état présentées ici avec la même dimension (*i.e* le même nombre d'entrées, de sorties et de variables d'état) que la représentation d'état (2.48) introduite au paragraphe 2.1.4.

C.1 Observation des systèmes à modèles linéaires

Parmi les modèles linéaires, il convient, pour la conception d'observateurs, de faire la distinction entre les systèmes linéaires à paramètres invariants (LPI) et variants (LPV) dans le temps. Les modèles linéaires LPI sont décrits par une représentation d'état Σ_{LPI} :

$$\Sigma_{LPI} = \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t), \end{cases} \tag{C.1}$$

et les modèles linéaires LPV par Σ_{LPV} :

$$\Sigma_{LPV} = \begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) + D(t)u(t). \end{cases} \tag{C.2}$$

Il existe deux grands types d'observateurs pour l'estimation des variables de modèles linéaires : les observateurs de Luenberger et de Kalman. Un observateur de Luenberger est adapté pour les modèles LPI (C.1) et celui de Kalman peut s'utiliser autant avec un modèle LPI (C.1) qu'un modèle LPV (C.2).

C.1.1 Observateur de Luenberger

Un observateur de Luenberger [119] est un observateur linéaire. Il utilise une matrice de gain L afin de corriger l'estimation du modèle (C.1) à l'aide d'une mesure telle que :

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(y(t) - \hat{y}(t)) \\ \hat{y}(t) = C\hat{x}(t) + Du(t). \end{cases} \tag{C.3}$$

Les notations avec un accent circonflexe représentent les variables estimées et celles sans représentent les variables mesurées. L est un gain multipliant l'erreur de sortie ce qui permet d'ajouter un terme correctif dans la représentation d'état (C.3).

L'objectif de l'observateur est de trouver l'estimation d'état de telle sorte que l'erreur d'estimation tende vers 0, c'est-à-dire :

$$\lim_{t\to\infty} (\hat{x}(t) - x(t)) = 0. \tag{C.4}$$

Or, en utilisant les équations (C.1) et (C.3), il vient :

$$\dot{\hat{x}}(t) - \dot{x}(t) = (A - LC)\,(\hat{x}(t) - x(t))\,. \tag{C.5}$$

Pour garantir la condition de convergence de l'erreur d'estimation vers 0, il faut donc que les valeurs propres de $(A - LC)$ soient à partie réelle négative.

C.1.2 Filtre de Kalman

Le filtre de Kalman [93] est un filtre à réponse impulsionnelle infinie permettant d'estimer l'état d'un système à partir d'une série de mesures. Ces mesures peuvent être incomplètes ou bruitées. Ce filtre est de loin le plus utilisé dans l'industrie automobile. Dans l'utilisation d'observateur à modèle linéaire, l'observateur de Kalman est adapté au modèle LPV et donc approprié au modèle linéaire de la dynamique latérale du véhicule décrit dans la section précédente. La forme discrète du filtre de Kalman est présentée en vue d'une utilisation embarquée. La notation discrète des équations suivantes utilise l'indice k pour l'instant k correspondant au temps $t = kT_e$.

La représentation d'état (C.2) discrétisée s'écrit à l'instant k :

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k \\ \quad y_k = C_k x_k + D u_k. \end{cases} \tag{C.6}$$

Les représentations d'état continue et discrète sont liées par les équations :

$$\begin{aligned} A_k &= e^{A(kT_e)T_e} \\ B_k &= A(kTe)^{-1}(A_k - I)B(kT_e) \\ C_k &= C(kT_e), \end{aligned} \tag{C.7}$$

où T_e est la période d'échantillonnage des calculateurs. La démonstration 1 exprime le raisonnement pour arriver aux équations (C.7). La discrétisation, effectuée de cette manière, suppose qu'il y a un bloqueur d'ordre zéro pour l'entrée u_k et que la matrice A_k est inversible pour toute vitesse V_x.

En traitement du signal, un bloqueur est un dispositif mathématique d'un CNA (Convertisseur Numérique Analogique) permettant la reconstruction de signaux échantillonnés. L'ordre du bloqueur correspond à la complexité de la reconstruction. Ainsi, entre deux échantillons de temps, un bloqueur d'ordre 0 supposera que le signal est constant (voir Figure C.1), et un bloqueur d'ordre 1 calculera l'interpolation linéaire entre ces deux points.

Démonstration 1. *Sachant une représentation d'état linéaire à paramètres invariant dans le temps (C.1), la solution à l'équation différentielle à l'instant t avec un état initial x_0 à l'instant t_0 donne :*

$$x(t) = e^{A(t-t_0)}x_0 + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)\mathrm{d}\tau.$$

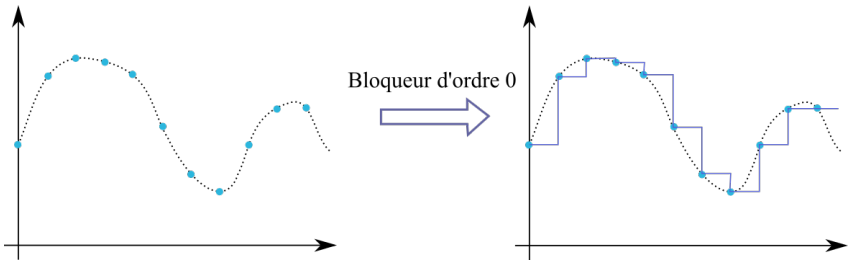


FIGURE C.1 – Fonctionnement d'un bloqueur d'ordre 0

Entre deux échantillons successifs k et $k+1$, sur l'intervalle $[kT_e, (k+1)T_e]$, l'équation précédente donne :

$$x_{k+1} = e^{A((k+1)T_e - kT_e)} x_k + \int_{kT_e}^{(k+1)T_e} e^{A((k+1)T_e - \tau)} Bu(\tau) \mathrm{d}\tau.$$

Avec l'hypothèse du bloqueur d'ordre 0, u(t) est constant sur l'intervalle $[kT_e, (k+1)T_e]$ ce qui permet de simplifier la relation précédente :

$$x_{k+1} = e^{AT_e} x_k + \int_{kT_e}^{(k+1)T_e} e^{A((k+1)T_e - \tau)} \mathrm{d}\tau B u_k.$$

Le changement de variable $\alpha = (k+1)T_e - \tau$ donne ensuite :

$$x_{k+1} = e^{AT_e} x_k + \int_{T_e}^{0} e^{A\alpha} \mathrm{d}\alpha B u_k,$$

et la solution de l'équation est :

$$x_{k+1} = e^{AT_e} x_k + A^{-1} \left(e^{AT_e} - I \right) B u_k,$$

avec I la matrice identité. On retrouve bien les relations (C.7).

Le filtre de Kalman fonctionne en deux phases : une phase de prédiction et une phase de mise à jour. À l'instant k, sachant que $\hat{x}_{k|k}$ est l'estimation de l'état et $P_{k|k}$ la matrice de l'erreur de covariance, l'étape de prédiction utilise l'estimation de l'état à l'instant précédent pour calculer l'estimation de l'état courant tel que :

$$\begin{aligned} \hat{x}_{k|k-1} &= A_k \hat{x}_{k-1|k-1} + B_k u_k \\ P_{k|k-1} &= A_k P_{k-1|k-1} A_k^T + Q_k, \end{aligned} \tag{C.8}$$

avec Q_k la matrice de covariance du bruit du procédé, $\hat{x}_{k|k-1}$ l'état prédit et $P_{k|k-1}$ la matrice prédite, à priori, de l'erreur de covariance.

Lors de l'étape de mise à jour, les observations de l'état du système sont utilisées pour corriger l'état prédit et ainsi avoir une meilleure estimation de celui-ci :

$$\begin{aligned} \tilde{y}_k &= z_k - C\hat{x}_{k|k-1} \\ S_k &= CP_{k|k-1}C^T + R_k \\ K_k &= P_{k|k-1}C^T S_k^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k \tilde{y}_k \\ P_{k|k} &= (I - K_k C) P_{k|k-1}, \end{aligned} \tag{C.9}$$

avec z_k la mesure de l'état, S_k la matrice de covariance de la mise à jour, R_k la matrice de covariance du bruit de mesure, Ka_k le gain de Kalman et C la même matrice que dans (2.53).

C.2 Observations des systèmes à modèles non linéaires

Un modèle non linéaire, comme son nom l'indique, n'établit pas de relation linéaire entre tous les états d'un système. Un tel modèle peut s'écrire :

$$\Sigma_{NL} = \begin{cases} \dot{x}(t) = f\left(x(t), u(t)\right) \\ y(t) = h\left(x(t), u(t)\right), \end{cases} \tag{C.10}$$

avec f et h, des fonctions non linéaires.

Un cas particulier de modèles non linéaires est le modèle non linéaire Lipschitzien :

$$\Sigma_{NLLi} = \begin{cases} \dot{x}(t) = Ax(t) + \phi\left(x(t), u(t)\right) \\ y(t) = Cx(t), \end{cases} \tag{C.11}$$

avec ϕ une fonction non linéaire de Lipschitz par rapport à x. Cette fonction est décrite par la Définition 3.

Définition 3. *Soit $E \subset \mathbb{R}$. Une fonction $f : E \to \mathbb{R}$ est dite de Lipschitz s'il existe $k \in \mathbb{R}^+$ tel que :*

$$\forall (x, y) \in E^2, |f(x) - f(y)| < k\,|x - y|\,. \tag{C.12}$$

Pour la fonction ϕ, cela signifie qu'il existe $\gamma \in \mathbb{R}^+$ tel que :

$$\forall (x_1, x_2, u) \in (\mathbb{R}^n, \mathbb{R}^n, \mathbb{R}^p), \|\phi(x_1, u) - \phi(x_2, u)\| < \gamma\,\|x_1 - x_2\|\,. \tag{C.13}$$

C.2.1 Observateur de Luenberger étendu

Un observateur de Luenberger étendu [197] revient à utiliser un observateur de Luenberger pour un point de fonctionnement du modèle non linéaire. Pour ce faire, le modèle est linéarisé autour du point de fonctionnement et le gain de l'observateur est calculé par placement de pôles. L'observation de l'état est donc pertinente uniquement autour du point de fonctionnement pour lequel a été calculé le gain de l'observateur. En effet, si l'état du système s'éloigne du point de fonctionnement alors des problèmes de stabilité peuvent apparaître. Il est donc rarement retenu dans le domaine automobile.

C.2.2 Filtre de Kalman étendu

La forme discrète du modèle non linéaire Σ_{NL} s'écrit :

$$\Sigma_{NLD} = \begin{cases} x_{k+1} = f(x_k, u_k) \\ y_k = h(x_k, u_k). \end{cases} \tag{C.14}$$

Si la fonction f peut être utilisée pour calculer l'état prédit $\hat{x}_{k|k-1}$, les matrices de covariance P_k, S_k et par conséquent le gain de Kalman K_k ne peuvent plus être calculés de la même manière que précédemment avec le modèle linéaire.

Pour se raccrocher aux équations du filtre de Kalman classique, le filtre de Kalman étendu [188] utilise une linéarisation locale des fonctions f et h. Ainsi, les équivalents des matrices A_k

et C_k, nommés respectivement Ae_k et Ce_k sont estimées comme étant les matrices Jacobiennes de f et h :

$$\begin{aligned} Ae_k &= \left.\frac{\partial f}{\partial x}\right|_{\hat{x}_{k-1|k-1},u_k} \\ Ce_k &= \left.\frac{\partial h}{\partial x}\right|_{\hat{x}_{k|k-1},u_k}. \end{aligned} \tag{C.15}$$

Les équations de prédiction s'écrivent alors :

$$\begin{aligned} \hat{x}_{k|k-1} &= f(\hat{x}_{k-1|k-1}, u_k) \\ P_{k|k-1} &= Ae_k P_{k-1|k-1} Ae_k^T + Q_k, \end{aligned} \tag{C.16}$$

et celles de mise à jour :

$$\begin{aligned} \tilde{y}_k &= z_k - h(\hat{x}_{k|k-1}, u_k), \\ S_k &= Ce_k P_{k|k-1} Ce_k^T + R_k \\ K_{mk} &= P_{k|k-1} Ce_k^T S_k^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_{mk}\tilde{y}_k \\ P_{k|k} &= (I - K_k C) P_{k|k-1}. \end{aligned} \tag{C.17}$$

Le filtre de Kalman étendu utilisant une linéarisation locale des équations dynamiques, sa convergence est donc locale et la convergence globale n'est pas assurée. La stabilité de ce filtrage n'est donc pas garantie et dépend fortement des conditions initiales.

C.2.3 Observateur à grand gain

Un observateur à grand gain [175] s'applique pour les systèmes s'écrivant sous la forme (C.11). Cet observateur est l'équivalent d'un observateur de Luenberger, mais avec un gain important permettant de compenser la non-linéarité du système.

L'estimation d'état de (C.11) par un gain L s'écrit :

$$\dot{\hat{x}}(t) = A\hat{x}(t) + \phi(\hat{x}(t), u(t)) + L(y(t) - C\hat{x}(t)). \tag{C.18}$$

Sachant P et Q des matrices symétriques définies positives et respectant l'équation de Lyapunov

$$(A - LC)^T P + P(A - LC) + Q = 0, \tag{C.19}$$

alors l'erreur d'estimation de l'état est asymptotiquement stable si

$$\gamma < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)}, \tag{C.20}$$

avec $\lambda_{min}(Q)$ et $\lambda_{max}(P)$ les valeurs propres minimales et maximales des matrices Q et P.

La liste des observateurs d'état présentée ici n'est pas exhaustive. Elle couvre cependant les méthodes les plus utilisées dans l'industrie ainsi que ceux répondant à notre problème et notre modélisation de la dynamique latérale.

Annexe D

Influence du choix des points de fonctionnement pour le multi-PID

Le Chapitre 3 présente une méthode de multirégulation, basé sur des régulateurs de type PID, pour le guidage latérale d'un véhicule autonome. La méthode permet de choisir les points de fonctionnement du régulateur : leur nombre et leurs valeurs. Pour cela, la méthode fait en sorte que la variation de phase du procédé soit constante entre chaque point de fonctionnement. Cette annexe présente l'influence de ce découpage pour le calcul du multirégulateur sur la boucle ouverte. Le procédé utilisé est celui de la dynamique latérale et plusieurs multi-PID sont calculés avec des découpages différents.

D.1 Les types de découpages possibles

Le système à réguler peut s'écrire par la représentation d'état :

$$\begin{cases} \dot{\underline{x}}(t) = A(\xi(t))\underline{x}(t) + B(\xi(t))u(t) \\ y(t) = C\underline{x}(t), \end{cases}$$

avec $\underline{x}(t)$ le vecteur d'état, $u(t)$ la commande, $y(t)$ la sortie et $A(\xi(t))$, $B(\xi(t))$, C les matrices d'état. Les deux premières matrices dépendent d'un paramètre variant dans le temps $\xi(t)$. Le paramètre ξ peut varier sur l'intervalle $[\xi_{min}, \xi_{max}]$. L'objectif du multirégulateur est de réguler le système sur tout cet intervalle. Ici, et comme dans le Chapitre 3, $\xi = V_x$.

Les paramètres à déterminer pour le multirégulateur sont :

- N_{op} : le nombre de points de fonctionnement,
- ξ_i : les valeurs du paramètres variant aux points de fonctionnement, avec $i \in \mathbb{N}$ tel que $i \in [1, N_{op}]$.

Il y a deux manières de découper l'intervalle $[\xi_{min}, \xi_{max}]$ pour déterminer les points de fonctionnement :

- Variation constante du paramètre entre deux points de fonctionnement :
  $\xi_{i+1} - \xi_i = \mathrm{Var}_\xi$
- Variation constance de la phase du procédé, à la fréquence au gain unité ω_u, entre deux points de fonctionnement :
  $\left|\arg\left(G(j\omega_u)|_{\xi_{i+1}}\right) - \arg\left(G(j\omega_u)|_{\xi_i}\right)\right| = \mathrm{Var}_\phi,$

avec $G(j\omega)$ la fonction de transfert du procédé telle que

$$G(s) = \frac{Y(s)}{\theta_v(s)}.$$

Une fois les ξ_i identifiés, les régulateurs associés ainsi que leurs pondérations sont calculés comme expliqué dans le Chapitre 3.

D.2 L'influence des découpages sur la boucle ouverte dans le cas du guidage latéral

D.2.1 Spécifications

Pour cette application la fréquence au gain unité et la marge de phase pour le calcul des PID sont choisis de telle sorte que :

- $\omega_u = 1$ rad/s
- $M_\Phi = 45°$.

A $\omega_u = 1$rad/s, la phase du procédé $G(s)$ ne varie pas linéairement, voir Figure D.1.

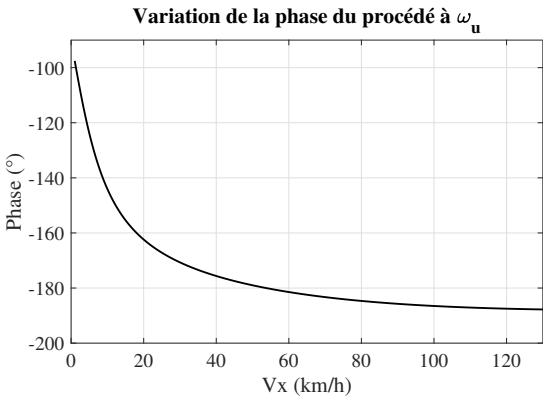


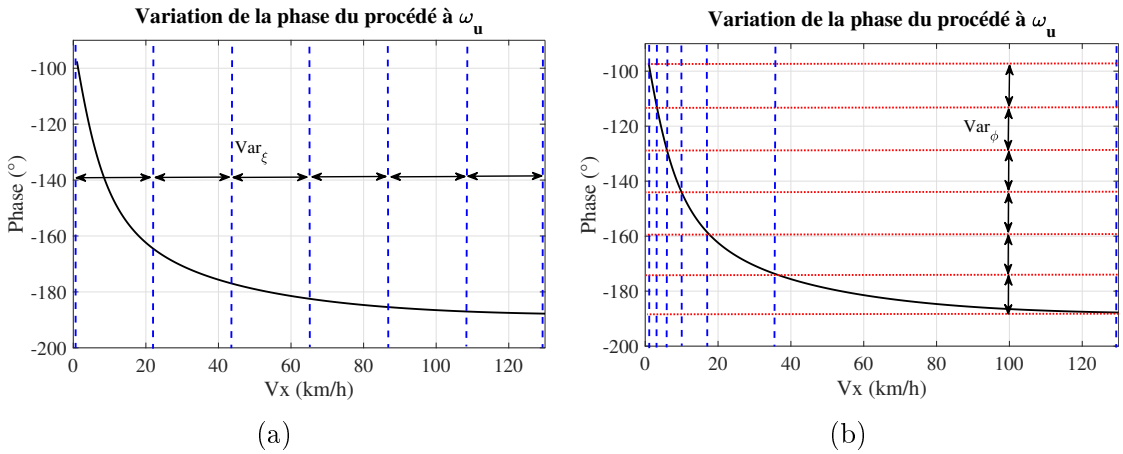
FIGURE D.1 – Variation de la phase de $G(s)$ à ω_u en fonction de la vitesse

En fonction des points de fonctionnement choisis, c'est cette variation qui a un impact important sur la boucle ouverte. Pour évaluer cet impact, plusieurs multi-PID avec différentes valeurs Var_ξ, Var_ϕ et N_{op} sont comparés. La Table D.1 récapitule les six différentes configurations choisies.

TABLE D.1 – Paramètres des multi-PID

	$\text{Var}_{\phi/\xi}$	N_{op}	ξ_i (km/h)
Var_ξ	64.5 km/h	3	1/65/130
	43 km/h	4	1/44/87/130
	21.5 km/h	7	1/22.5/44/65.5/87 /108.5/130
Var_ϕ	45°	3	1/9.6/130
	30°	4	1/5.8/16.7/130
	15°	7	1/3.2/5.9/9.8/17 /35.4/130

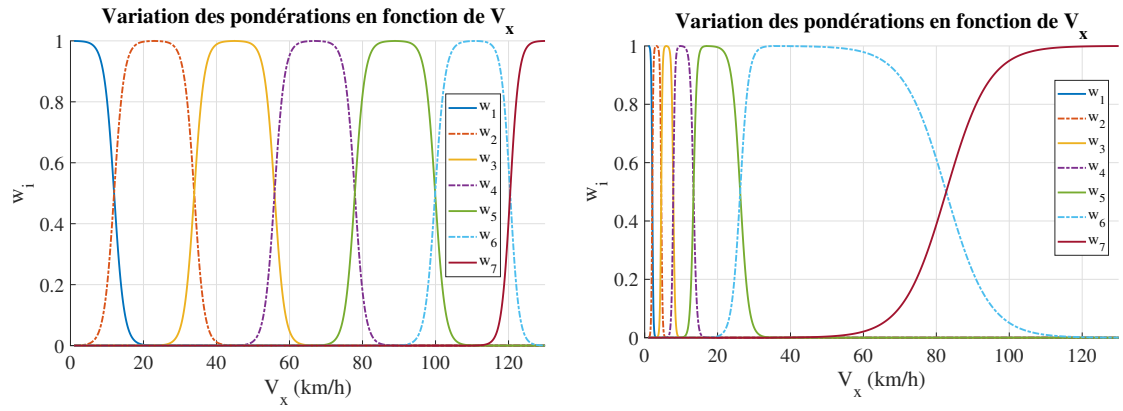
Pour trois différents N_{op} (3, 4 et 7), les intervalles avec Var_ξ ou Var_ϕ constant sont comparés. Il peut être noté que pour un même nombre de points de fonctionnement, les deux méthodes de découpage ne mènent pas du tout aux mêmes points de fonctionnement.



(a) (b)

FIGURE D.2 – Découpages utilisés pour le choix des points de fonctionnement avec Var_ξ constant (a) et Var_ϕ constant (b)

Étant donné que $G(j\omega_u)$ présente une variation de phase plus importante à basse vitesse, le découpage avec des intervalles Var_ϕ constants donne plus de points de fonctionnement aux basses vitesses. Les intervalles, pour les exemples $\mathrm{Var}_\phi = 15°$, menant à $N_{op} = 7$ et $\mathrm{Var}_\xi = 21.5$ km/h peuvent être observés sur la Figure D.2. Les pondérations résultantes, pour chaque régulateur, sont tracés Figure D.3.



(a) (b)

FIGURE D.3 – Pondérations des multi-régulateurs avec 7 PID et des découpages à Var_ξ constant (a) et Var_ϕ constant (b)

D.2.2 Comparaison des boucles ouvertes dans le domaine fréquentiel

La stabilité d'un système bouclé peut-être analysée en utilisant le critère de stabilité de Bode : un système bouclé est stable si la marge de phase de la boucle ouverte est supérieure à 0, $M_\Phi > 0$. Ce critère ne peut être appliqué que sur des systèmes à minimum de phase, c'est à dire que les pôles et les zéros de la fonction de transfert du procédé sont dans la partie gauche

du plan complexe. Ici, $G(s)$ est un système à minimum de phase donc le critère de stabilité de Bode peut être utilisé, ([166]).

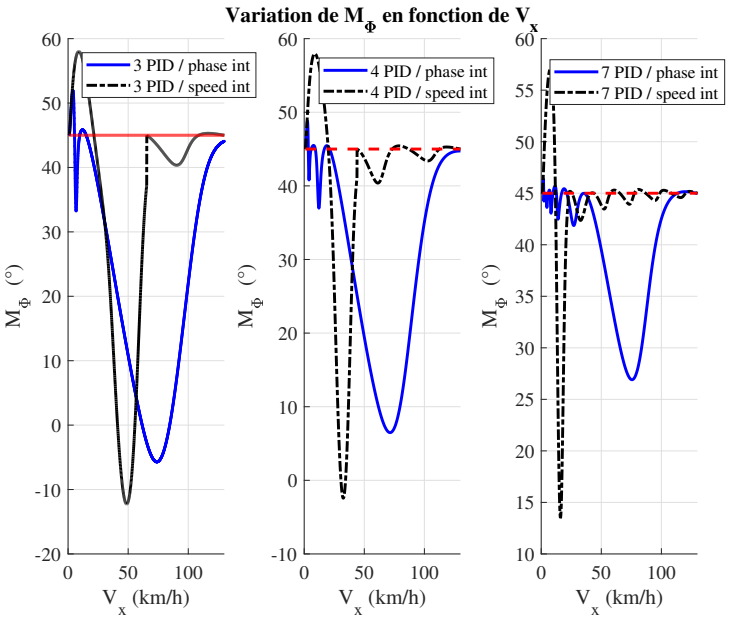


FIGURE D.4 – Marge de phase M_Φ de la boucle ouvertes pour les différentes combinaisons de point de fonctionnement

Afin de vérifier si la stabilité de la boucle ouverte est garantie à chaque vitesse, les marges de phase M_Φ de la boucle ouverte pour les différentes configurations de multi-PID de la Table D.1 sont tracées Figure D.4 en fonction de la vitesse. Il peut être observé que l'utilisation de trois points de fonctionnement n'est pas suffisant pour garantir la stabilité à toutes les vitesses pour les deux découpages avec Var_ξ constant et Var_ϕ constant. Avec quatre points de fonctionnement et Var_ϕ constant, la marge de phase reste positive pour toutes les vitesses tandis que la configuration avec Var_ξ constant n'est toujours pas capable d'assurer la stabilité du système à basse vitesse. Ceci est dû à la grande sensibilité du système aux variations de vitesse lorsque celle-ci est basse et au positionnement sous-optimal des points de fonctionnement. L'utilisation de sept points de fonctionnement est suffisant pour garantir la stabilité avec chacun des découpages. Le multi-PID calculé avec Var_ϕ constant montre un meilleur degré de stabilité à basse vitesse que le multi-PID avec Var_ξ constant. C'est l'inverse à haute vitesse. Cependant, le multi-PID pour Var_ξ constant a des pics de variation de M_Φ plus importants à basse vitesse, et donc un degré de stabilité plus faible, à cause du manque de régulateur aux faibles vitesses.

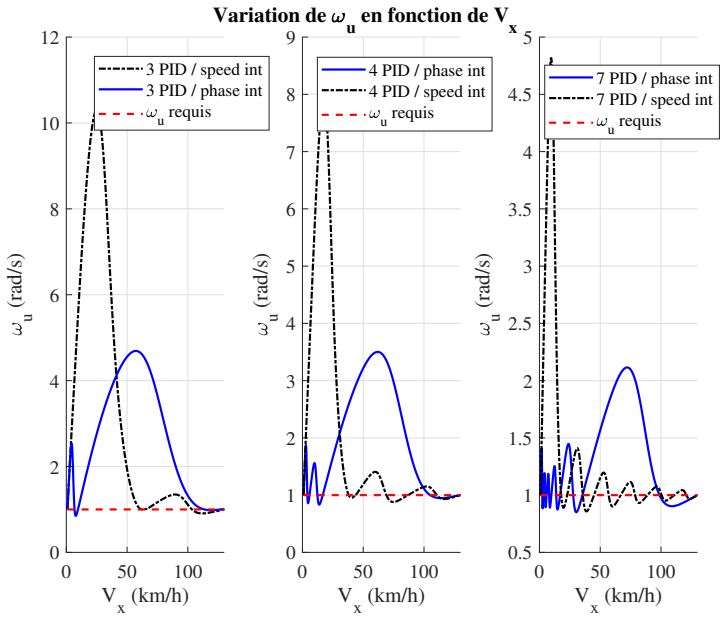


FIGURE D.5 – Fréquence au gain unité ω_u de la boucle ouverte pour les différentes combinaisons de point de fonctionnement

En ce qui concerne la fréquence au gain unité, comme pour la marge de phase, le multi-PID avec Var_ξ constant présente des pics plus importants à basse vitesse où il y a moins de points de fonctionnement, mais reste proche de la valeur désirée à haute vitesse car il y a plus de points de fonctionnement que nécessaire.

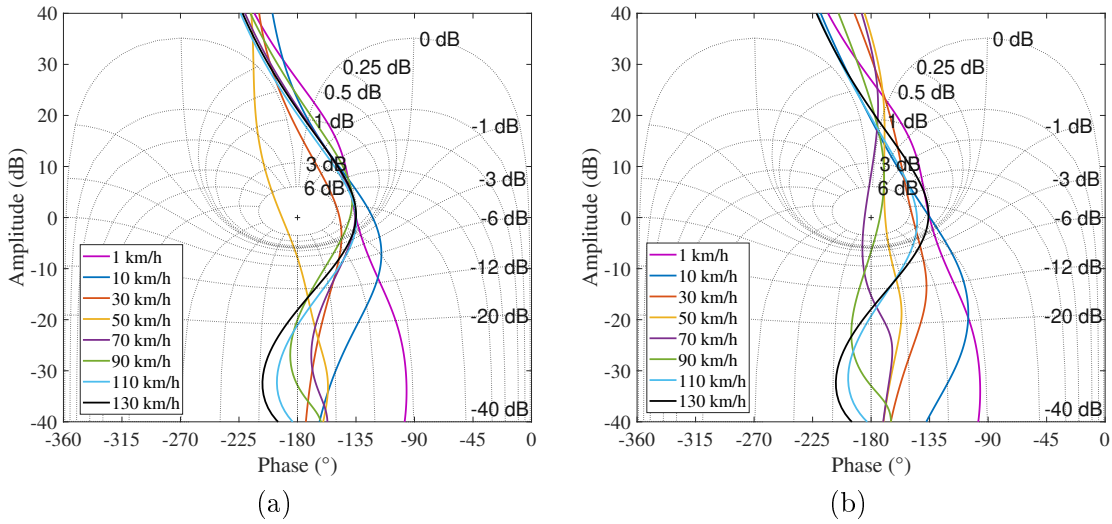
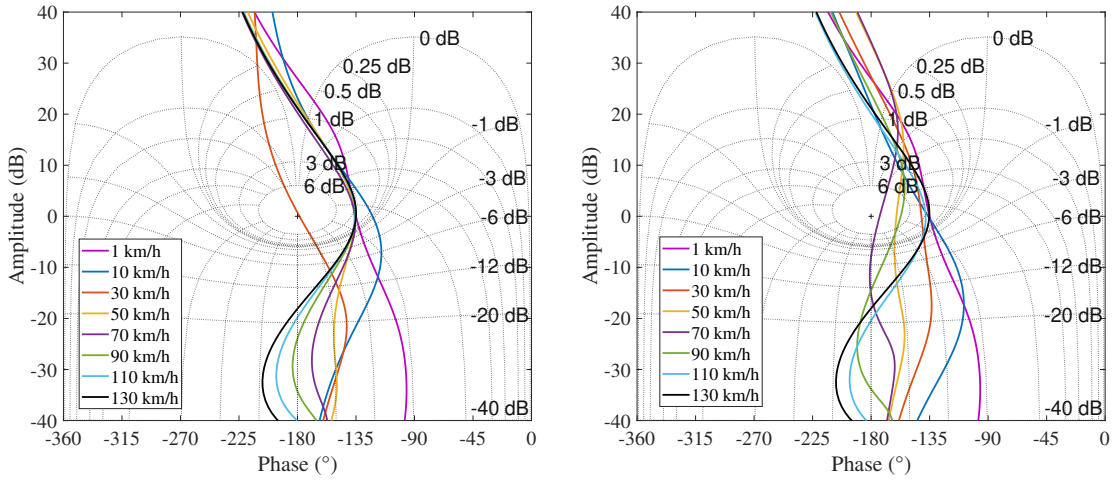


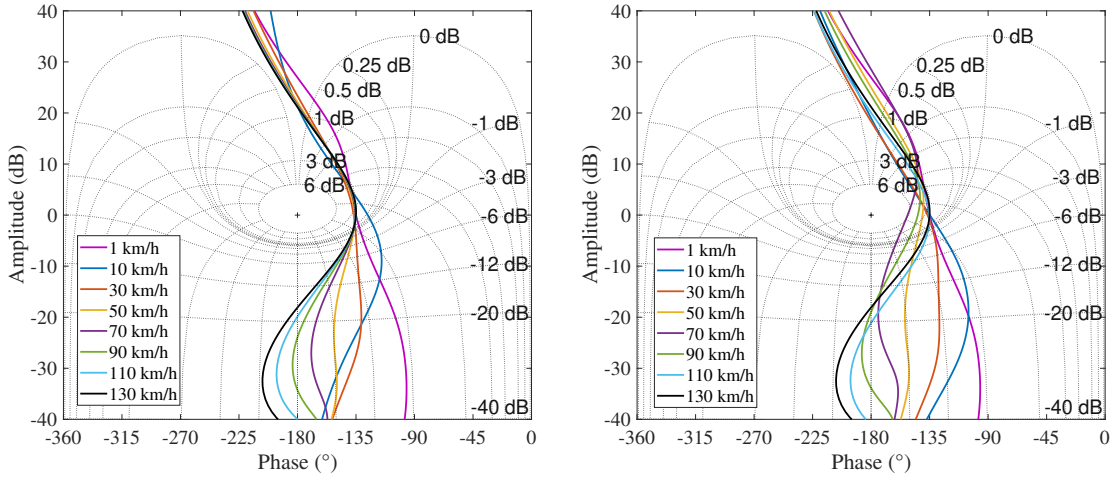
FIGURE D.6 – Réponses fréquentielles de la boucle ouverte dans le diagramme de Black-Nichols pour Var_ξ constant (a) et Var_ξ constant (b) avec $N_{op} = 3$

Les réponses fréquentielles de la boucle ouverte à différentes vitesses sont tracées sur des diagrammes de Black-Nichols. Elles sont visibles sur les Figures D.6-D.8.



(a) (b)

FIGURE D.7 – Réponses fréquentielles de la boucle ouverte dans le diagramme de Black-Nichols pour Var_ξ constant (a) et Var_ξ constant (b) avec $N_{op} = 4$



(a) (b)

FIGURE D.8 – Réponses fréquentielles de la boucle ouverte dans le diagramme de Black-Nichols pour Var_ξ constant (a) et Var_ξ constant (b) avec $N_{op} = 7$

Même si toutes les possibilités de configuration de la boucle ouverte en fonction de la vitesse ne sont pas tracées pour une question de lisibilité, il peut être observé que la boucle ouverte n'est pas robuste aux variations de vitesse lorsque $N_{op} = 3$ et $N_{op} = 4$. En effet, avec $N_{op} = 3$, sur les Figures D.6 (a) et (b), un problème d'instabilité apparaît pour certaines vitesses. Avec $N_{op} = 4$, une grande sensibilité de la boucle ouverte autour de 30 km/h est visible pour le multi-PID avec Var_ξ constant sur la Figure D.7 (a). Pour Var_ϕ constant sur la Figure D.7 (b), le contour d'amplitude tangent aux réponses fréquentielles varie grandement à haute vitesse. Pour $N_{op} = 7$, une robustesse du degré de stabilité est observée pour les deux découpages sur les Figures D.8 (a) et (b).

Titre : Des systèmes d'aide à la conduite au véhicule autonome connecté

Résumé

Cette thèse s'inscrit dans le développement et la conception de fonctions d'aide à la conduite pour les véhicules autonomes de niveau 3 et plus en milieu urbain ou péri urbain. Du fait d'un environnement plus complexe et de trajectoires possibles plus nombreuses et sinueuses, les algorithmes des véhicules autonomes développés pour l'autoroute ne sont pas adaptés pour le milieu urbain. L'objectif de la thèse est de mettre à disposition des méthodes et des réalisations pour permettre au véhicule autonome d'évoluer en milieu urbain. Cette thèse se focalise sur la proposition de solutions pour améliorer le guidage latéral des véhicules autonomes en milieu urbain à travers l'étude de la planification de trajectoire en situation complexe, l'analyse du comportement des usagers et l'amélioration du suivi de ces trajectoires complexes à faibles vitesses. Les solutions proposées doivent fonctionner en temps réel dans les calculateurs des prototypes pour pouvoir ensuite être appliquées sur route ouverte. L'apport de cette thèse est donc autant théorique que pratique.

Après une synthèse des fonctions d'aide à la conduite présentes à bord des véhicules et une présentation des moyens d'essais mis à disposition pour la validation des algorithmes proposés, une analyse complète de la dynamique latérale est effectuée dans les domaines temporel et fréquentiel. Cette analyse permet alors la mise en place d'observateurs de la dynamique latérale pour estimer des signaux nécessaires aux fonctions de guidage latéral et dont les grandeurs ne sont pas toujours mesurables, fortement dégradées ou bruitées. La régulation latérale du véhicule autonome se base sur les conclusions apportées par l'analyse de cette dynamique pour proposer une solution de type multirégulateur capable de générer une consigne en angle volant pour suivre une trajectoire latérale quelle que soit la vitesse. La solution est validée tant en simulation que sur prototype pour plusieurs vitesses sur des trajectoires de changement de voie. La suite de la thèse s'intéresse à la génération d'une trajectoire en milieu urbain tenant compte non seulement de l'infrastructure complexe (intersection/rond-point) mais également des comportements des véhicules autour. C'est pourquoi, une analyse des véhicules de l'environnement est menée afin de déterminer leur comportement et leur trajectoire. Cette analyse est essentielle pour la méthode de génération de trajectoire développée dans cette thèse. Cette méthode, basée sur l'algorithme A* et enrichie pour respecter les contraintes géométriques et dynamiques du véhicule, se focalise d'abord dans un environnement statique complexe de type parking ou rond-point. Des points de passage sont intégrés à la méthode afin de générer des trajectoires conformes au code de la route et d'améliorer le temps de calcul. La méthode est ensuite adaptée pour un environnement dynamique où le véhicule est alors capable, sur une route à double sens de circulation, de dépasser un véhicule avec un véhicule arrivant en sens inverse.

Mots clés : automatique, dynamique du véhicule, assistance à la conduite, régulation, génération de trajectoire, prédiction de trajectoire.

Title: From driving assistance systems to automated and connected driving

Abstract

This thesis is about the design of driving assistance systems for level 3 urban automated driving. Because of a more complex of the environment and a larger set of possible trajectories, the algorithms of highway automated driving are not adapted to urban environment. This thesis objective is to provide methods and algorithms to enable the vehicle to perform automated driving in urban scenarios, focusing on the vehicle lateral guidance and on the path planning. The proposed solutions operate in real-time on board of the automated vehicle prototypes. The contribution of this thesis is as theoretical as practical.

After a synthesis of the driving assistance systems available on current cars and a presentation of the prototypes used for the validation of the algorithms developed in the thesis, a complete analysis of the vehicle lateral dynamics is carried out in time and frequency domains. This analysis enables the design of observers of the lateral dynamics in order not only to estimate signals required for the lateral guidance functions but also to increase reliability of available measurements. Based on the conclusions from the analysis of lateral dynamic, a multi-controller solution has been proposed. It enables the computation of a steering wheel input to follow a trajectory at any longitudinal speed. The solution is validated in simulation and on real road traffic for lane change scenarios. Another contribution consist in an analysis on the other vehicles of the environment is conducted in order to identify their behaviors and which maneuver there are performing. This analysis is essential for the path planning function developed in the thesis. This method, based on the A* algorithm and extended to respect geometric and dynamic constraints, firstly focuses on static environment such as a parking lot. Waypoints are added to the method in order to compute trajectories compatible with traffic regulation and improve the computation time. The method is then adapted for dynamic environment where, in the end, the vehicle is able to perform overtaking manoeuvers in a complex environment.

Keywords: automatic, vehicle dynamics, driving assistance system, control, path planning, trajectory prediction.

Unité de recherche

Laboratoire de l'Intégration du Matériau au Système (IMS) - UMR 5218 CNRS, Pôle P3S, 351 Cours de la Libération, 33405 TALENCE Cedex, France