HAL
open science

Contrôle plateforme pour la validation du véhicule autonome sur simulateur dynamique à hautes performances

Carolina Rengifo Cadavid

To cite this version:

Carolina Rengifo Cadavid. Contrôle plateforme pour la validation du véhicule autonome sur simulateur dynamique à hautes performances. Automatique / Robotique. HESAM Université, 2020. Français. NNT : 2020HESAE023 . tel-03048468

HAL Id: tel-03048468

https://pastel.hal.science/tel-03048468v1

Submitted on 9 Dec 2020

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.

HESAM UNIVERSITÉ

Arts et Métiers Sciences et Technologies

ÉCOLE DOCTORALE SCIENCES ET MÉTIERS DE L'INGÉNIEUR
[LISPEN - Campus de Cluny]

THÈSE

présentée par : **Carolina RENGIFO CADAVID**

soutenue le : **30 Juin 2020**

pour obtenir le grade de : **Docteur d'HESAM Université**

préparée à : **École Nationale Supérieure d'Arts et Métiers**

Spécialité : **Automatique**

Contrôle plateforme pour la validation du véhicule autonome sur simulateur dynamique à hautes performances

THÈSE dirigée par :
[**M. KEMENY Andras**]

et co-encadrée par :
[**M. CHARDONNET Jean-Rémy, M. PAILLOT Damien, M. MOHELLEBI Hakim**]

Jury

M. Daniel R. MESTRE	Directeur de Recherche, CRVM (CNRS)	Président
M. Jean-Christophe POPIEUL	Professeur, Univ. Polytechnique Hauts-de-France	Rapporteur
M. Stéphane ESPIÉ	Directeur de Recherche, IFSTTAR	Rapporteur
M. Andras KEMENY	Professeur, Arts & Métiers	Examinateur
M. Jean-Rémy CHARDONNET	Maître de conférences, Arts & Métiers	Examinateur
M. Damien PAILLOT	Maître de conférences, Univ. de Bourgogne	Examinateur
M. Hakim MOHELLEBI	Docteur, Renaut S.A.S.	Invité
M. Hans-Peter SCHOENER	Docteur, Driving Simulation Association	Invité

T H È S E

Résumé

Les travaux présentés dans ce manuscrit s'inscrivent dans le contexte de la simulation de conduite et plus concrètement dans celui des simulateurs de conduite dynamique à hautes performances utilisés pour la validation des systèmes avancés et du véhicule autonome. Bien qu'ils continuent à gagner en performance, le niveau de perception induite par les algorithmes de restitution de mouvement (MCA pour *Motion Cueing Algorithms*) de ces simulateurs est encore insuffisant dans de nombreuses situations de conduite. Pour cette raison, il est essentiel d'améliorer les MCA. Afin de répondre aux enjeux de performance et de perception du mouvement, nous présentons différentes approches d'amélioration des MCA. Dans un premier temps, nous avons effectué une étude de différents simulateurs à hautes performances et des stratégies de commande existantes, afin d'identifier l'algorithme qui correspond le mieux à la validation du véhicule autonome. Nous avons ainsi sélectionné la commande prédictive (MPC pour *Model predictive Control*) parmi d'autres stratégies avancées de contrôle (classique, adaptative, optimale, logique floue, neuronale). Après avoir analysé les enjeux de cette stratégie, nous avons abordé deux éléments principaux de sa formulation : l'optimisation et le modèle mathématique.

Concernant l'optimisation, nous avons proposé une alternative à l'optimisation quadratique sous contraintes généralement proposée dans le MPC. Elle consiste à utiliser des réseaux de neurones récurrents (RNR) représentant un ensemble d'équations différentielles sous la forme d'un circuit associé à partir d'une fonction de coût qui définit l'objectif de l'optimisation. Nous avons effectué une analyse des conditions de stabilité et de faisabilité de la commande prédictive, pour assurer une stabilité des réseaux de neurones en tant que mécanisme d'optimisation. Nous avons comparé plusieurs méthodes qui favorisent la faisabilité du MCA soumis aux contraintes.

Concernant le modèle mathématique, nous avons intégré les systèmes sensoriels, en particulier le système vestibulaire, dans le contrôle plateforme, afin de restituer la sensation de mouvement la plus réaliste possible au cours d'une simulation. La détermination du modèle mathématique correspondant le plus approprié s'est faite en réalisant une expérimentation dans le simulateur de conduite à hautes performances de Renault. Les objectifs de cette étude étaient : analyser l'impact du modèle du système vestibulaire dans le contrôle plateforme, évaluer les différences entres les modèles existants de façon subjective et objective, comparer le comporte-

ment auto-déclaré de conduite des participants avec des mesures de performance et déduire le type de modèle à implémenter pour chaque participant. Les tests ont été menés en conduite autonome et en conduite interactive, afin d'évaluer les différents modèles dans les deux conditions. Pour cette étude, nous avons accordé une attention particulière aux cas d'usage à implémenter ainsi qu'au comportement du modèle autonome.

L'ensemble de nos études montre que la stratégie de contrôle prédictif est le meilleur choix pour contrôler les mouvements des nouveaux simulateurs à hautes performances. En effet, elle permet de mieux profiter de l'espace de travail sans mettre en danger le simulateur et/ou le conducteur, comparativement aux autres stratégies. Cependant, dans cette stratégie de restitution de mouvement, l'optimisation en temps réel et le modèle de perception doivent être garantis afin d'améliorer l'immersion du conducteur dans l'environnement virtuel. Nous avons donc comparé différentes techniques pour résoudre les problèmes d'optimisation sous contraintes et avons proposé une technique innovante d'optimisation à partir de circuits intégrés, qui propose une solution intuitive et rapide au problème d'optimisation sous contraintes du MPC. Enfin, nous avons établi des recommandations de paramétrage des MCA en fonction du comportement de conduite auto-déclaré qui permet une meilleure perception du mouvement dans un simulateur de conduite, en conduite interactive et en mode autonome.

Mots clés : algorithmes de restitution du mouvement, commande prédictive, conduite autonome, simulateurs de conduite dynamiques, perception du mouvement humaine.

Abstract

The work presented in this manuscript takes part in the context of driving simulation and more specifically in the one of dynamic driving simulators used for the validation of advanced systems and the autonomous vehicle. Although, these simulators continue to gain in performance, the level of perception induced by the motion cueing algorithms (MCA) is still insufficient. For this reason, it is essential to improve the MCA. In order to address the issues of performance and motion perception, we have presented different approaches to improve the MCA. As a first step, we conducted a study of different high-performance simulators and existing control strategies in order to identify the algorithm that best fits the validation of the autonomous vehicle. From this comparative study, we chose Model Predictive Control (MPC) among other advanced control strategies (classical, adaptive, optimal, fuzzy logic, neural). After analyzing the stakes of this strategy, we approached two main elements of its formulation : optimization and mathematical models.

Regarding optimization, we proposed an alternative to the quadratic optimization under constraints generally proposed in the MPC. It consists of using recurrent neural networks (RNN) that represent a set of differentiable equations in the form of an associated circuit from an objective function that defines the objective of the optimization. We performed an analysis of the stability and feasibility conditions of predictive control to stabilize the RNN as an optimization mechanism. We performed comparisons with several methods used in the platform control that support the feasibility of the constrained control strategy over the entire prediction horizon.

Regarding the mathematical model, we integrated the sensory systems, and in particular the vestibular system, into the platform control, in order to reproduce as-realistic-as-possible driving sensations during a simulation. The determination of the most appropriate mathematical model was made by conducting an experiment in Renault's high-performance driving simulator. The objectives of this study were : to analyse the impact of the vestibular system model in the platform control, to evaluate the differences between existing models in a subjective and objective way, to compare the participants' self-reported driving behaviour with performance measures and to deduce the type of model to be implemented according to each participant. The tests were conducted in autonomous and interactive driving, in order to evaluate the different models under both conditions. For this study, we paid particular attention to the use cases to be implemented and to the behaviour

of the autonomous model.

All our studies show that the predictive control strategy is the best choice to control the motion of new high-performance simulators. Indeed, it allows to better take advantage of the workspace without endangering the simulator and/or the driver, compared to other strategies. However, in this motion restitution strategy, the real-time optimization and the perception model must be guaranteed in order to improve the driver's immersion in the virtual environment. Therefore, we compared different techniques to solve constrained optimization problems. We proposed a optimization-based technique, which provides an intuitive and fast solution to the MPC constrained optimization problem. Finally, we established recommendations for MCA parameterization according to the self-declared driving behaviour allowing a better perception of motion in a driving simulator, in interactive driving and in an autonomous mode.

Keywords : motion cueing algorithms, model predictive control, autonomous driving, dynamic driving simulators, human motion perception.

Remerciements

Je remercie chaleureusement tous ceux qui ont contribué directement ou indirectement à ce projet de thèse, tant dans le domaine professionnel que personnel. Toutefois, je me permettrai d'être concise dans mes remerciements et de ne mentionner que ceux dont le soutien a eu un impact direct sur le résultat de ces travaux.

J'exprime ma gratitude à Andras Kemeny, mon directeur de thèse et à Hakim Mohellebi, mon encadrant entreprise pour m'avoir accueillie dans leur service de Renault en tant que stagiaire puis doctorante. Je tiens ensuite à remercier mes encadrants du laboratoire, Damien et Jean-Rémy pour la confiance qu'ils m'ont accordé. Je remercie particulièrement Jean-Rémy qui a assuré l'encadrement de ma thèse. Je lui suis très reconnaissante pour sa disponibilité, ses retours toujours pertinents et tout son support pendant ces trois années.

J'adresse mes remerciements à Jean-Christophe Popieul et à Stéphane Espié pour avoir accepté d'être rapporteurs de ma thèse. Merci également à Daniel Mestre et à Hans-Peter Schoener pour avoir accepté d'évaluer mon travail.

Je voudrais remercier également tous mes camarades, aujourd'hui mes amis, que j'ai rencontré chez Renault : Lucie, Benoit, Antoine, Valentin, Martin, Morgane, Léa, Younesse. Sans eux, mon expérience n'aurait pas été aussi agréable. Je suis profondément reconnaissante à Bérénice et à Guillaume avec qui j'ai partagé la plupart de mon temps et que je considère comme de grands amis. Merci également à tous ceux qui font partie de la communauté des doctorants et qui ont généré une bonne ambiance et un groupe de support pendant ces trois années de thèse.

« Agradezco igualmente a mi familia, mis padres y mi hermana por el apoyo brindado durante todos estos años. Ustedes siempre han sido y serán una fuente de inspiración y motivación para dar lo mejor de mí. Sé que en la distancia no es fácil pero siempre supieron estar ahí para mí. Gracias. »

Je remercie bien évidemment mon mari Jean pour m'avoir soutenue, conseillée et encouragée tout au long de ces trois années. Merci pour le réconfort que tu as su m'apporter chaque jour.

Bien que cette thèse soit gravée à mon nom, elle est néanmoins le résultat d'un travail qui n'aurait pas pu se faire tout seul. À vous tous : Merci.

Table des matières

Introduction générale **1**

1 Rôle de la simulation dans l'automobile **7**
1.1 Systèmes avancés d'aide à la conduite (ADAS) 7
1.1.1 Catégorisation et technologies utilisées 8
1.2 Conduite autonome 9
1.2.1 Avantages et enjeux 9
1.2.2 Modes de fonctionnement 11
1.3 Apport des simulateurs à la validation automobile 13
1.4 Simulateurs de conduite 14
1.4.1 Composition 15
1.4.2 Avantages et enjeux des simulateurs 17
1.4.3 Types de plateformes 18
Structure en série 18
Structure parallèle 19
Structure hybride 20
1.4.4 Différentes échelles de simulateurs 20
1.4.5 Comparaison de simulateurs à hautes performances 21
Simulateurs dynamiques à hautes performances du groupe Renault : 23

2 Mouvement et perception humaine **27**
2.1 Système visuel 28
2.1.1 L'anatomie et la physiologie oculaire 28
2.1.2 Effet du système visuel sur la conduite automobile 29
2.1.3 Modèle du système visuel pour la perception du mouvement . 30
2.2 La somesthésie 31
2.2.1 La proprioception 32
2.3 Système vestibulaire 32
2.3.1 Anatomie vestibulaire 33
2.3.2 Les organes otolithiques 33
2.3.3 Les canaux semi-circulaires 35
2.3.4 Modèles pour les composants du système vestibulaire : Modèle de perception humaine 36
2.3.5 Modèle des organes otolithiques 36

2.3.6 Modèle des canaux semi-circulaires 40
2.3.7 Modèle du système vestibulaire 42
2.3.8 Technique de *tilt coordination* 43
2.4 L'intégration sensorielle 45
2.4.1 Couplage visuo-vestibulaire 46
2.4.2 Modèle de perception complet 47
2.5 Synthèse du chapitre 49

3 Stratégies de contrôle de plateforme 51
3.1 Commande classique 52
3.1.1 Influence des filtres sur le comportement du système 54
3.1.2 Avantages et inconvénients 57
3.2 Commande adaptative 58
3.2.1 Avantages et inconvénients 60
3.3 Commande optimale 62
3.3.1 Performance en tant que stratégie de contrôle de mouvement 64
3.4 Commande prédictive 65
3.4.1 Modèle mathématique du processus 67
3.4.2 Prédiction 69
3.4.3 Optimisation 71
3.4.4 Avantages et inconvénients 75
3.5 Autres techniques de contrôle avancées pour la restitution de mouvement 75
3.5.1 Commande par logique floue 76
La logique floue dans le contrôle plateforme 77
3.5.2 Contrôle neuronal 78
3.6 Comparaison des algorithmes de contrôle de plateforme 79
3.6.1 Axes d'amélioration de la commande choisie 84

4 Optimisation de la commande prédictive par réseaux de neurones récurrents 87
4.1 Faisabilité dans la commande prédictive 88
4.1.1 Paramétrage 89
Horizons de prédiction et contrôle 89
Réglage des poids 91
4.1.2 Méthodes pour assurer la faisabilité 92
État de l'art 92
Comparaison des stratégies 93
Simulation 95
4.1.3 Discussion 98
4.2 Application de l'optimisation avec des réseaux neuronaux récurrents 100
4.2.1 Introduction aux réseaux de neurones 100
4.2.2 État de l'art spécifique 102

4.3 Formulation du problème d'optimisation en termes de réseaux de neurones récurrents . . . 104
4.3.1 Méthode de pénalisation . . . 105
Mise en œuvre . . . 107
Simulation . . . 110
4.3.2 Méthode duale . . . 113
Implémentation . . . 114
4.3.3 Limitations et perspectives . . . 117
4.4 Conclusion . . . 118

5 Évaluation du modèle perceptif dans le simulateur de conduite dynamique 121
5.1 Introduction et contexte . . . 122
5.1.1 Présentation des différents modèles . . . 125
5.1.2 Réglage préliminaire pour l'expérimentation . . . 126
5.1.3 Restitution de mouvement selon le type de modèle . . . 128
5.2 Conditions expérimentales . . . 130
5.2.1 Matériel . . . 130
5.2.2 Procédure expérimentale . . . 131
5.2.3 Scénario de conduite . . . 131
5.2.4 Modèle du véhicule autonome . . . 132
5.2.5 Questionnaire d'auto-évaluation sur le comportement en situation de conduite . . . 134
5.2.6 Questionnaire sur la restitution de mouvement dans le simulateur . . . 135
5.2.7 Acquisition de données . . . 136
5.3 Analyse expérimentale . . . 137
5.3.1 Comparaison selon le type de conduite . . . 138
Mode de conduite active « Libre » . . . 139
Mode de conduite autonome « AD » . . . 141
5.3.2 Comparaison entre le mode « AD » et « Libre » . . . 142
5.3.3 Division des groupes selon le type de MCA préféré . . . 143
5.3.4 Classification du type de conduite auto-déclaré . . . 145
MDSI : inventaire multidimensionnel des styles de conduite . 146
DBQ : questionnaire sur le comportement des conducteurs . . 148
Comparaison entre MDSI et DBQ . . . 150
5.3.5 Relation entre le comportement de conduite et la préférence de mouvement de la plateforme . . . 151
5.3.6 Relation objective entre les questionnaires auto-déclarés et le comportement de conduite sur simulateur . . . 151
5.4 Discussion . . . 153
5.4.1 Contrôle de restitution du mouvement fondé sur le modèle prédictif . . . 155

5.4.2 Influence du comportement de conduite dans la préférence du type de contrôle de plateforme 156
5.4.3 Limites 158
5.5 Conclusion 159

Discussion Générale **161**

Annexes **171**

A Système d'état utilisé dans l'algorithme optimal **171**

B Exemple de réseau neuronal récurrent **173**

C Influence des à-coups dans les algorithmes de contrôle de mouvement **175**

D Questionnaires **177**
D.1 Questionnaire sur le comportement du conducteur au volant 178
D.2 Questionnaire sur la stratégie de restitution de mouvement 181

Bibliographie **183**

Table des figures **203**

Liste des tableaux **207**

Introduction générale

Les simulateurs de conduite dynamiques à hautes performances sont des outils indispensables dans l'industrie automobile. Ils offrent des environnements de simulation flexibles, reproductibles et hautement répétables. Ils sont utilisés pour mesurer les comportements des conducteurs, simuler des environnements de circulation, évaluer des situations réelles, étudier les interactions entre le conducteur et les autres usagers de la route, et valider des systèmes avancés d'aide à la conduite et conduite autonome.

Concernant la validation du véhicule autonome, l'utilisation des simulateurs de conduite est de plus en plus évidente, car bien que les essais routiers soient la clé pour le déploiement des véhicules autonomes sur la route, le nombre de kilomètres de roulage est difficile à réaliser avec ce seul type de test. En effet, dans l'industrie automobile, le nombre "un milliard de miles" [Castignani, 2018] est généralement considéré comme celui pour lequel le véhicule doit être capable de remplir toutes les fonctions dans certaines conditions de conduite. Par exemple, afin d'atteindre un bon niveau de fiabilité par des statistiques de conduite, il faudrait parcourir $20\text{x}10^9$ kilomètres de roulage [Wautier, 2018]. Comme c'est irréalisable, il faut établir d'autres alternatives afin de prouver la sécurité du conducteur et du système. Pour les systèmes où les risques de défaillance sont faibles, la façon de prouver la sécurité n'est pas par les statistiques de conduite, mais par la conception formelle du système. Cela implique l'identification des causes pertinentes des défaillances et l'utilisation de la simulation pour vérifier les performances du système. Dans les cas où l'homme est dans la boucle, les simulateurs de conduite peuvent être utilisés afin de vérifier efficacement le comportement du véhicule et du conducteur dans des environnements sûrs et contrôlés. Toutefois, la validation de ces technologies en matière de simulation dépend fortement du simulateur de conduite, car la relation entre la performance du simulateur et la validité n'est pas directe [Wynne *et al.*, 2019]. Il est donc nécessaire de sélectionner avec soin les caractéristiques du simulateur de conduite.

En effet, l'un des principaux défis de ces plateformes est de procurer au conducteur et aux passagers la sensation la plus proche possible de celle vécue sur route en tenant compte des limites de l'espace de travail. Pour surmonter ce problème, tous les simulateurs de conduite dynamique utilisés par les constructeurs automobiles et les laboratoires de recherche mettent en œuvre des algorithmes de restitution de mouvement (MCA). Ils visent à reproduire au mieux tous les signaux provenant

du modèle de véhicule simulé tout en maintenant la plateforme dans ses limites physiques. En effet, les commandes de mouvement telles que les accélérations et les rotations ne peuvent pas être envoyées directement du modèle du véhicule simulé à la plateforme, en raison des contraintes et des limites physiques.

Plusieurs efforts ont été réalisés dans le développement de MCA afin d'augmenter le réalisme de la simulation et réduire le mal du simulateur induit par les effets contradictoires entre plusieurs sous-systèmes, par exemple, entre le système visuel et le système vestibulaire. Bien que ces algorithmes continuent de gagner en performance, le niveau de perception induit par les MCA est insuffisant pour remplir les besoins de la simulation de conduite. Pour cette raison, il est essentiel d'étudier de nouvelles techniques pour ces stratégies afin de les rendre plus performantes et adaptées aux nouveaux systèmes de l'industrie automobile. Ce contexte donne lieu à notre première question de recherche : quelle est la stratégie de contrôle la plus appropriée pour les simulateurs à hautes performances utilisés pour la validation du véhicule autonome ?

Il existe principalement quatre stratégies de restitution de mouvement, trois sont des stratégies de contrôle fondées sur des filtres : la classique, l'adaptative, l'optimale ; et une stratégie est fondée sur la prédiction des états, la commande prédictive [Garrett & Best, 2010]. Par ailleurs, deux autres stratégies de contrôle avancé peuvent être adaptées aux techniques de restitution de mouvement : la logique floue et la commande par réseaux de neurones. Parmi ces stratégies, nous avons sélectionné la commande prédictive (MPC pour *Model Predictive Control*) [Dagdelen *et al.*, 2004]. Elle constitue la stratégie la plus appropriée pour les simulateurs à hautes performances dédiés à la validation du véhicule autonome, car contrairement aux autres MCA existants, cette technique possède de nombreux avantages en termes de gestion des contraintes [Fang & Kemeny, 2012a; Garrett & Best, 2013], d'utilisation de l'espace de travail [Cleij *et al.*, 2019], un réglage relativement facile, une meilleure robustesse [Beghi *et al.*, 2012; Fang & Kemeny, 2012b; Nehaoua *et al.*, 2006; Venrooij *et al.*, 2016]. De plus, elle utilise la prédiction du système et de la trajectoire afin de trouver une loi de contrôle optimale. Le MPC est une stratégie bien connue et utilisée dans les simulateurs de conduite dynamique, ce qui donne lieu à notre deuxième question de recherche : quels sont les axes d'amélioration nécessaires pour que la stratégie de contrôle choisie maximise l'espace de travail du simulateur et améliore la perception du mouvement des conducteurs ?

En effet, la commande prédictive possède des enjeux qui doivent être traités afin de donner une perception de mouvement adéquate au conducteur et respecter le temps de calcul des simulateurs à hautes performances. Dans cette thèse, nous nous concentrerons sur deux éléments principaux : l'optimisation et le modèle de prédiction.

Ainsi, l'un des principaux avantages de cette stratégie peut devenir l'un de ses principaux inconvénients si elle n'est pas traitée avec la vigilance nécessaire. Nous parlons ici du processus d'optimisation. Celle-ci permet de trouver la loi de contrôle optimale du système en se basant sur la prédiction des états de la plateforme, tout

en respectant les contraintes physiques et perceptives imposées. Cependant, la dynamique du système varie rapidement et la solution numérique requise en temps réel est souvent considérée comme un obstacle pour les algorithmes d'optimisation existants. Cette solution du problème d'optimisation en temps réel a été étudiée de manière intensive [Goodwin *et al.*, 2006]. Une approche prometteuse consiste à utiliser les réseaux de neurones récurrents artificiels (RNR) comme méthode de calcul pour l'optimisation dynamique fondée sur l'implémentation de circuits analogiques. Les réseaux de neurones récurrents possèdent des caractéristiques intéressantes, telles qu'un haut degré de parallélisme, l'adaptabilité, la vitesse de convergence et la mise en œuvre aisée dans des logiciels embarqués, qui sont reconnues par les chercheurs et sont utilisées dans de nombreux domaines d'application différents tels que : la conception de systèmes de contrôle, la mémoire associative, la programmation mathématique, et le traitement des signaux et des images.

La perception de mouvement est le second axe d'amélioration de cette stratégie. En effet, lors de l'utilisation du MPC, la qualité de la restitution des mouvements dépend fortement du modèle mathématique intégré dans la boucle de contrôle. Cependant, trouver le modèle le plus approprié est une tâche majeure qui nécessite des efforts considérables. L'une des approches les plus courantes pour améliorer la perception de mouvement grâce aux MCA est l'intégration du modèle mathématique du système vestibulaire dans la conception de la loi de contrôle [Asadi *et al.*, 2015a, 2017; Augusto, 2009; Baseggio *et al.*, 2011; Beghi *et al.*, 2012; Bruschetta *et al.*, 2019; Chen & Fu, 2010; Fang & Kemeny, 2014; Garrett & Best, 2013; Lamprecht *et al.*, 2019; Mohammadi *et al.*, 2016; Reid & Nahon, 1985; Telban & Cardullo, 2005; Zacharias, 1978; Zaychik *et al.*, 2009]. Bien que la plupart des chercheurs suggèrent que l'utilisation d'un modèle vestibulaire précis améliore la qualité de la restitution des mouvements, l'implémentation d'un tel modèle dans l'algorithme de contrôle sans aucun test expérimental n'est pas évidente. En effet, certains auteurs continuent à concevoir des MCA sans tenir compte d'aucun modèle de perception humaine [Dagdelen *et al.*, 2009; Fang *et al.*, 2017]. En outre, l'intégration du modèle de perception du mouvement rend la conception des commandes plus complexe et augmente le temps d'optimisation à mesure que le nombre d'états dans le système augmente. Par conséquent, dans ce travail, nous étudierons le rôle du modèle de perception du mouvement sur les MCA du point de vue de l'utilisateur. Comme tous les conducteurs ne se comportent pas de la même manière, nous formulons l'hypothèse selon laquelle le type du comportement de conduite a un impact sur la perception de mouvement dans un simulateur de conduite. C'est pourquoi nous analyserons subjectivement et objectivement l'influence que le comportement de conduite des sujets peut avoir sur la préférence du MCA.

Dans ce contexte nous avons abordé deux questions de recherche, car elles correspondent aux piliers sur lesquels repose ce travail de thèse. Néanmoins, ce ne sont certainement pas les seules questions scientifiques abordées tout au long de cette thèse. Les questions de recherches plus précises seront abordées dans chaque chapitre.

Cadre de la thèse

Le travail présenté dans cette thèse a été réalisé au sein de l'équipe DEA-TDV au centre de réalité virtuelle et de simulation immersive du Groupe Renault. Il s'inscrit dans le cadre d'une Convention Industrielle de Formation par la REcherche (CIFRE) entre Arts et Métiers et l'entreprise Renault.

Depuis quelques années, le groupe automobile Renault s'engage pour une mobilité durable, avec des solutions innovantes comme les véhicules électriques, hybrides, les nouvelles technologies d'aide à la conduite (ADAS) et le développement de véhicules autonomes. Dans cette optique, cette thèse s'intéresse aux activités de validation effectuées sur les simulateurs de conduite à hautes performances, afin de les rendre plus adaptés en utilisant les stratégies de contrôle de mouvement.

Objectifs

L'objectif de ce travail de recherche est de concevoir ou améliorer des stratégies de restitution de mouvement des simulateurs dynamiques à hautes performances, en proposant des algorithmes qui permettent de :

- Répondre aux besoins de validations du véhicule autonome et des nouvelles technologies d'aide à la conduite.
- Améliorer la perception de mouvement des conducteurs dans le simulateur de conduite dynamique.
- S'adapter au comportement de conduite des conducteurs afin de proposer un environnement virtuel et une perception de mouvement appropriés.
- Permettre un temps de calcul en temps réel qui s'adapte aux systèmes d'exploitation afin de respecter les contraintes du système d'intégration.
- Être facilement modifiable pour les autres types de plateformes de simulation et pour les nouvelles générations des simulateurs de conduite à hautes performances.

Afin de valider les différentes approches, les algorithmes de restitution de mouvement seront testés d'abord en simulation numérique, puis dans le simulateur à hautes performances du groupe Renault.

Plan de lecture

Les thématiques abordées dans cette thèse sont représentées dans le schéma 1. Chaque thématique correspond à un chapitre de ce manuscrit.

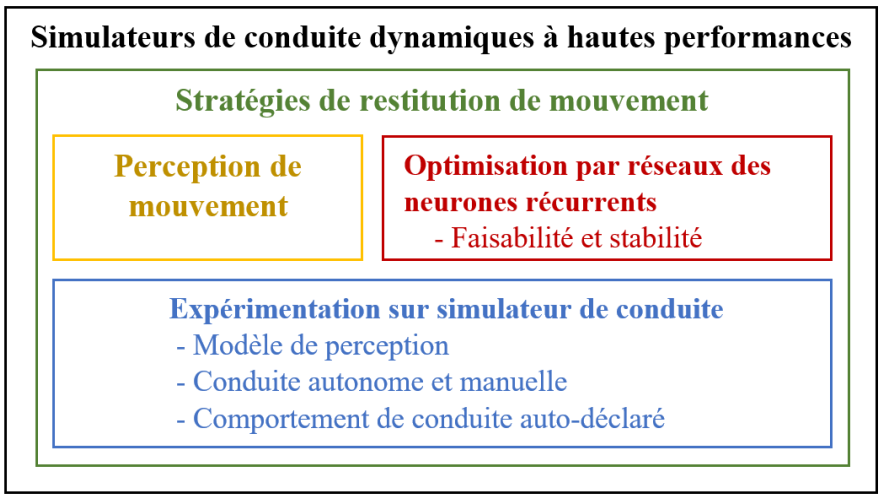


FIGURE 1 – Organisation de la thèse.

Le premier chapitre commence par une présentation générale de la conduite autonome et les différents systèmes d'aide à la conduite avancés. Ensuite, l'importance des simulateurs pour la validation de ces systèmes et le véhicule autonome est exposée. Dans la dernière partie, nous présentons les simulateurs de conduite dynamique. Nous décrivons leur importance pour l'industrie automobile, leurs architectures, leurs avantages, leurs enjeux, ainsi qu'une comparaison des simulateurs existants et futurs.

Le deuxième chapitre présente en détail les sens de la perception du mouvement de l'être humain, tels que la vision, la proprioception et le système vestibulaire. Tout au long de ce chapitre, nous montrons les différents modèles mathématiques associés à ces systèmes, afin d'évaluer leur pertinence pour les algorithmes de restitution de mouvement.

Dans le troisième chapitre, nous présentons une étude bibliographique des différents types d'algorithmes utilisés dans le contrôle de simulateurs de conduite dynamique. Nous incluons les différentes techniques de contrôle avancé qui peuvent potentiellement être utilisées pour la génération de nouveaux algorithmes. Ensuite, nous comparons la technique qui convient le plus à la validation du véhicule autonome, la commande prédictive, aux algorithmes existants afin de justifier notre choix. Pour introduire les approches traitées dans les chapitres suivants, nous terminons ce chapitre par un état de l'art qui expose les axes d'amélioration de cette stratégie de contrôle.

Le quatrième chapitre commence par la présentation des conditions nécessaires pour assurer la stabilité de la commande prédictive grâce à la faisabilité du pro-

blème d'optimisation. Ces conditions permettent d'introduire l'approche d'optimisation proposée dans la deuxième partie du chapitre : les réseaux de neurones récurrents. Nous concluons le chapitre en évoquant les avantages, limitations ainsi que les perspectives pour la mise en œuvre dans le MPC basé sur les algorithmes de restitution de mouvement.

Le cinquième chapitre est consacré à l'application de l'algorithme de restitution du mouvement sur le simulateur de conduite dynamique. Dans ce chapitre, nous présentons une expérimentation qui évalue la perception du mouvement des conducteurs dans le simulateur en tenant compte des mesures objectives du mouvement réalisé. À cet égard, nous faisons un état de l'art exhaustif des différents modèles de perception ainsi que des questionnaires de conduite auto-déclarés, afin de déterminer une relation entre le type de comportement sur la route et la préférence de l'algorithme de contrôle. Cette étude est menée à la fois en utilisant un modèle de conduite manuelle et un modèle virtuel du véhicule autonome.

Enfin, nous concluons ce document en faisant le bilan des contributions apportées et nous présentons également les limites de nos approches ainsi que des perspectives d'évolution.

Chapitre 1

Rôle de la simulation dans l'automobile

Ce premier chapitre décrit de manière générale la simulation de conduite, plus spécifiquement, les simulateurs de conduite dynamique et les avantages qu'ils procurent à l'industrie automobile. Dans un premier temps, nous présentons les technologies et les enjeux liés au développement des nouveaux véhicules, et plus particulièrement, dans le cadre de cette thèse, les systèmes avancés d'aide à la conduite et la conduite autonome. Ensuite, nous présentons les différents simulateurs de conduite, afin de comprendre leur importance dans la validation de ces systèmes. Dans la dernière partie, nous faisons une comparaison des types de simulateurs qui seront utilisés dans cette thèse.

1.1 Systèmes avancés d'aide à la conduite (ADAS)

Leur objectif principal est de baisser la probabilité d'occurrence d'un accident et d'aider activement le conducteur à améliorer sa conduite au volant [Mas, 2012]. Les ADAS donnent des informations supplémentaires au conducteur afin de lui donner une aide dans le contrôle de son véhicule. De plus, ces dispositifs électroniques potentiellement « intelligents », tels que le régulateur de vitesse adaptatif, les systèmes d'évitement des collisions, le maintien de la voie et les systèmes de commande latérale, offrent des avantages importants pour la sécurité routière et le confort du conducteur et d'autres acteurs du trafic. Ils informent ou aident activement le conducteur à améliorer la conduite d'un véhicule automobile [van Wees, 2004]. Leur fonctionnement dépend du niveau d'automatisation du véhicule. Ces systèmes participent à la compétitivité industrielle, le développement durable, la sécurité, l'efficacité, la mobilité étendue, l'aide pour les personnes âgées, le confort et la convenance [Schreurs & Steuwer, 2016].

Il n'existe pas de passage direct entre une conduite interactive où le conducteur a complètement le contrôle de son véhicule et la conduite autonome. Au contraire, entre ces deux phases, il existe une phase intermédiaire de conduite assistée dans laquelle les ADAS sont directement intégrées dans le véhicule afin de faciliter la conduite humaine et éviter les risques sur la route. Par conséquent, la conduite autonome sera possible uniquement en intégrant les ADAS. Afin de comprendre cette phase d'assistance intermédiaire, nous présentons les systèmes qui seront présents dans le véhicule autonome.

1.1.1 Catégorisation et technologies utilisées

Nous avons focalisé cet état de l'art sur le cas spécifique du groupe Renault, du fait que chaque entreprise peut développer et catégoriser ses technologies selon son besoin spécifique. En outre, les stratégies de commande de mouvement développées dans ce document qui seront utilisées pour la validation de ces systèmes et du véhicule autonome seront intégrées dans le simulateur dynamique ULTIMATE et dans le nouveau simulateur ROADS du groupe Renault. Par conséquent, il est important de connaître leur fonctionnement et la façon dont l'entreprise les catégorise.

Les ADAS se présentent en plusieurs catégories : les systèmes de sécurité, les systèmes d'alerte, les aides au parking et les systèmes d'affichage d'informations. Chacune de ces catégories vise à réduire à zéro le nombre d'accidents dans les rues, et à aider le conducteur dans les manœuvres de conduite ; soit en informant le conducteur par signaux visuels ou auditifs, soit en fournissent des actions sur les systèmes existants, tels que les ceintures de sécurité et les airbags. Ces catégories sont constituées d'une ou plusieurs technologies ADAS qui transforment le véhicule en un moyen de transport plus sûr et interactif. Parmi eux, se trouvent :

- **Multiples caméras (mono ou stéréo) :** ces systèmes sont capables de fournir au conducteur toutes les informations dont il a besoin dans son environnement. Ils donnent des avertissements de départ, assistance à la signalisation routière, des avertissements de collision directe, assistance au freinage d'urgence et estimation du profil routier.
- **Multiples radars (ondes électromagnétiques – longue (77GHz) et courte (24GHz) distance) :** contrôle de vitesse adaptatif, assistance au freinage d'urgence, assistance de changement de voie et avertissement de collision directe.
- **LIDAR (impulsions dans le spectre visible courte portée, caméra multifonction) :** assistance au freinage d'urgence dans la ville ou freinage imminent.

1.2 Conduite autonome

Le véhicule autonome est un concept de véhicule sans conducteur qui doit prendre lui-même des décisions adaptées pour gérer le véhicule dans certaines conditions de circulation et climatiques. Son comportement en conduite doit se faire de manière compatible avec les règles et les contraintes liées à la sécurité des usagers et à son environnement, tout ceci sans intervention humaine. En effet, la sécurité des personnes dans la circulation routière est l'une des motivations principales dans le développement des véhicules autonomes, car environ 90% des accidents [Brown, 2017] sont de cause humaine [Wachenfeld & Winner, 2016]. Ainsi, la capacité autonome d'un véhicule ouvre de nouvelles opportunités pour la mobilité individuelle et permet de contribuer à l'optimisation de la circulation en toute sécurité.

La conduite autonome est un sujet de recherche depuis plus de quatre décennies. Il a fait l'objet de nombreux projets : le California PATH, le NAVLAB (depuis 1986), le projet PROMETHEUS (1987-1995), le projet US du National Automated Highway System Research Program (NAHS) (1994-1997), pour en nommer quelques-uns [Bengler *et al.*, 2014]. Dans ces projets, peuvent interagir à la fois l'État, le monde académique et l'industrie.

1.2.1 Avantages et enjeux

Les avancées technologiques, l'ingénierie assistée par ordinateur, et les divers modèles d'intelligence artificielle et d'aide à la décision permettront de développer la conduite autonome du véhicule. Ces véhicules apporteront plusieurs avantages tels que : profiter du temps de trajet pour faire d'autres activités, réduire les accidents, augmenter la mobilité des jeunes, des personnes âgées et handicapées, optimiser le flux de la circulation, diminuer les embouteillages, réduire la pollution et augmenter l'efficacité de carburant et des sources d'énergie alternatives [Anderson *et al.*, 2016]. De plus, ils contribueront à développer de nouveaux services de transports publics souples et adaptables, et à renforcer la sécurité routière. Selon quelques études, les véhicules autonomes auraient le potentiel d'éliminer 90% des accidents de la route [Sender, 2016].

Ces avantages déprendront du type d'usage de la voiture autonome. Ces types sont classés en trois groupes : la voiture autonome particulière, la voiture autonome industrielle et le système de transport public autonome [Nouvelle France Industrielle, 2015]. Pour les particuliers, le véhicule doit gérer les situations dès son point de départ jusqu'au point d'arrivée, soit sur autoroute, en milieu urbain ou en péri-urbain. Le véhicule industriel concerne les voitures synchronisées, le transport de marchandises, les bennes à ordures ménagères, les véhicules destinés aux livraisons, entre autres. Enfin, pour le transport public autonome, nous trouvons les flottes en libre-service, le convoi de navettes autonomes ou les transports à la

demande des usagers.

Voici quelques exemples d'applications possibles dans ces trois groupes de véhicules autonomes :

- « ***Valet Parking*** » **autonome :** avec cette fonction, le conducteur économise du temps pour trouver une place de stationnement, ainsi que pour marcher vers / depuis cette place. En outre, l'accès au véhicule est facilité spatialement et temporellement.
- **Pilotage autonome :** le véhicule autonome prend en charge la tâche de conduite exclusivement sur autoroutes ou voies rapides. Le conducteur devient juste un passager pendant le voyage autonome et il peut poursuivre d'autres activités.
- **Véhicule sur demande :** le véhicule autonome reçoit la destination demandée par les occupants ou des entités externes (utilisateurs, fournisseurs de services, etc.), et s'y rend, de façon autonome.
- **Commercial :** « points de collecte », location de voiture, taxi, auto-partage. Les véhicules sont fournis à l'utilisateur, mais il doit aller les chercher et les ramener au point de prise en main.

Bien que le développement du véhicule autonome présente de nombreux avantages, il reste encore beaucoup de changements au niveau sociétal et réglementaire avant qu'il ne circule sur nos routes. Or les essais sur route des véhicules autonomes et leur développement nécessitent une adaptation inévitable des réglementations [Gerdes & Thornton, 2015]. Les progrès rapides de la technologie peuvent parfois dépasser l'environnement réglementaire, mais pour que le système de véhicule autonome fonctionne, il est nécessaire de mettre en place une prise de conscience de cette nouvelle technologie.

Avec les véhicules autonomes, la prévention des accidents dans la rue n'est pas satisfaite complètement. En effet, certaines conditions non intentionnelles sont difficiles à prévoir et peuvent causer des situations dangereuses pour le système et le conducteur. Pour les véhicules semi-autonomes, c'est le conducteur qui est responsable, mais il faut analyser de manière critique qui est responsable quand le véhicule est complètement autonome. Peut-être, sont-ce aux constructeurs et équipementiers de s'assurer auprès de grandes compagnies avec des contrats couvrant toute leur flotte autonome [Jacqué, 2016]. Cet impact à moyen et long terme sur le secteur de l'assurance s'annonce considérable, sans oublier ce qui est dans la balance : moins d'accidents, moins de blessés, moins de pertes humaines. Ce n'est qu'un seul point de discussion concernant ce sujet si controversé.

La technologie de conduite autonome sera le plus grand progrès de ces dernières années en matière de sécurité automobile même si pour le moment il reçoit considérablement moins d'attention par rapport à d'autres problèmes, tels que la compétitivité, la durabilité, l'efficacité des ressources, ou l'innovation [Reschka, 2016]. Ces

arguments de sécurité et de confort que promet la voiture autonome semblent séduire la majorité des automobilistes. En effet, un sondage réalisé par l'observatoire Cetelem en 2016, avec auprès de 8500 personne venant de 15 pays, indique que trois automobilistes sur quatre sont convaincus que la voiture autonome deviendra réalité et ils sont 55 % à se déclarer intéressés par l'achat d'un tel véhicule [Bataille & Chapus, 2016].

1.2.2 Modes de fonctionnement

Le passage entre un véhicule sans ADAS à un véhicule complètement autonome est constitué de différents niveaux d'automatisation. L'organisation mondiale SAE (pour *Society of Automotive Engineers*) propose 5 niveaux d'automatisation dans la figure 2 [1], dans un cadre règlementaire initial avec des pratiques exemplaires pour guider les constructeurs et d'autres entités dans la conception, la mise au point, les essais et le déploiement en toute sécurité de véhicules hautement automatisés [Shuttleworth, 2019].

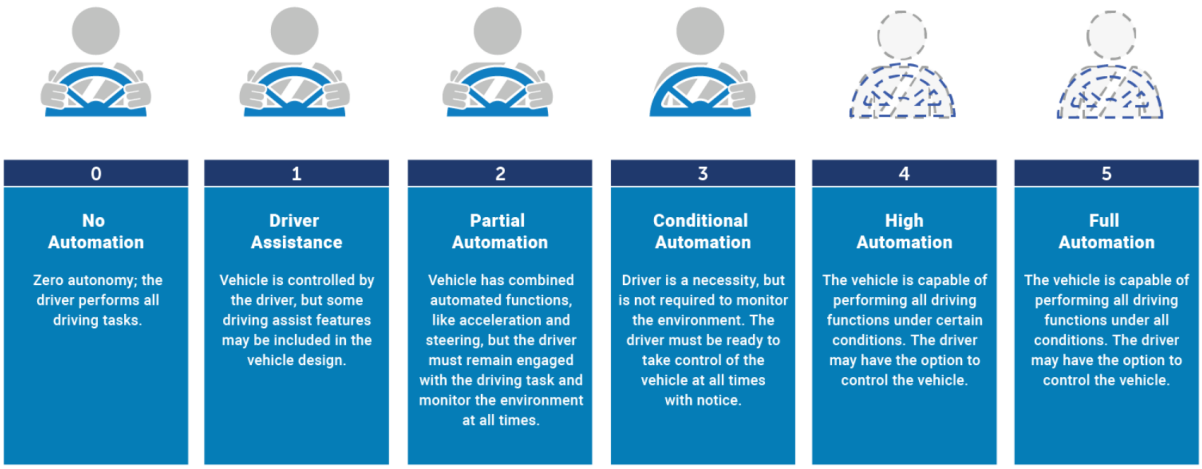


FIGURE 2 – Niveaux d'automatisme de la SAE.

De son côté, le programme du groupe Renault « véhicule autonome » a modifié les 5 niveaux de la SAE en différents modes de fonctionnement. Ceux-ci sont montrés dans le tableau 1.

Voici quelques définitions importantes :

- ***Eyes on*** **:** le conducteur doit garder le regard sur la route.
- ***Eyes off*** **:** le conducteur ne doit pas nécessairement garder le regard sur la scène de conduite et peut effectuer d'autres tâches (lecture, divertissement, etc.).

1. https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety

Eyes on	**Eyes off**	
Contrôle longitudinal	AD1/TJP : Pilotage en situation de trafic. Tenue de la voie.	AD1.1 : Autoconduite sur l'autoroute en situation de trafic dense.
Voies multiples	AD2 : Pilotage sur Autoroute. Insertion de voie automatique.	AD2.1 : Autoconduite sur l'autoroute en situation de trafic dense. Changement de voie automatique.
Intersections	AD3 : Pilotage en ville (Négociation en intersection)	AD3.1 : Négociation en intersection, reconnaissance des feux, piétons, priorité et décision (tourner à droite –gauche). Valet parking.

TABLEAU 1 – Modes et niveaux de fonctionnement du véhicule autonome.

- **Contrôle longitudinal :** le véhicule ne peut pas changer de voie automatiquement avec ou sans supervision du conducteur. Ce mode intègre AD1 pour *Autonomous driving 1* (conduite autonome) ou TJP pour *Traffic Jam Pilot* (pilotage en congestion de trafic).

- **Voies multiples** : le véhicule peut changer de voie.

- **Intersection :** le mode intègre la gestion de tous les cas présents dans une conduite en ville (feux de signalisation, intersections, prise de décisions, etc.).

Le degré d'automatisation défini plus haut dans lequel le véhicule conduit par lui-même sans surveillance humaine, donc entièrement automatisée, a été nommé « *Full Automation* ». D'un point de vue technique, le plus grand défi réside dans l'absence totale d'un superviseur humain qui connait les limites et les défauts du système. Cependant, l'objectif de Renault est d'être le premier constructeur généraliste à offrir une technologie « *eyes-off/hands off* » sur des voitures grand public et à des prix abordables.

Bien avant d'arriver au degré *full automation*, chaque mode d'automatisation vu auparavant doit être testé et évalué en utilisant la simulation numérique. Pour ce faire, il faut analyser les comportements des conducteurs et d'agents externes tels que les piétons ou les cyclistes. Ces comportements sont transformés en cas d'usages en tant que situations réalistes, critiques ou potentiellement critiques qui mettent à l'épreuve le système autonome. L'idée est de mettre le véhicule autonome face à des évènements programmés pour qu'il respecte des tâches spécifiques. Ces situations n'impliquent pas le comportement du véhicule, mais peuvent influer sur la manière dont il peut réagir.

La qualité de ces modèles comportementaux du véhicule autonome ainsi que ceux de son environnement influencent la qualité, le réalisme et les résultats obtenus à partir de la simulation ou les tests routiers. Ces modèles sont un support intégré

pour la génération de scénarios qui regroupent divers cas d'usages. Cet environnement de simulation ne doit pas seulement prendre en compte le modèle du véhicule autonome, mais aussi, intégrer les conditions de circulation, météo, piétons, cyclistes ou animaux, simuler correctement les capteurs intégrés dans le véhicule autonome, entre autres. Il est important qu'ils soient intégrés soigneusement afin de reproduire à l'identique leur vrai comportement, pour que les interactions et réactions du véhicule autonome en simulation soient les plus précises et réalistes possible.

Afin d'évaluer les différents algorithmes de mouvement dans le simulateur, nous utiliserons plusieurs scénarios de conduite comprenant différents cas d'usages qui sont communément utilisés dans la validation des systèmes d'aide à la conduite et de conduite autonome. Ils seront présentés tout au long de cette thèse.

1.3 Apport des simulateurs à la validation automobile

La simulation est un moyen d'explorer de nouvelles technologies qui permet de représenter fidèlement le monde réel qui nous entoure. Ainsi, lorsque l'on souhaite évaluer la performance ou l'impact d'une nouvelle technologie, sa simulation permet de produire une grande quantité de données utiles pour son amélioration. De plus, elle peut être utilisée comme outil d'aide à la conception et d'analyse afin de tester différentes hypothèses sans avoir à les réaliser, évitant ainsi des dépenses supplémentaires de fabrication et des situations de risque.

Dans le domaine automobile, la simulation est un composant essentiel pour la conception et la production de tests des nouvelles technologies ou systèmes de conduite autonome. Les tests automobiles pour la validation des modèles, les études prenant en compte le conducteur ou les interactions entre le véhicule et les systèmes intégrés, nécessitent un environnement de simulation fidèle. Par fidélité, nous entendons l'intégration d'environnements de conduite, du véhicule lui-même, de la circulation, d'infrastructures, mais aussi d'éléments de scénarisation permettant une mise en situation du conducteur la plus proche de la réalité.

L'un des besoins actuels de l'industrie automobile, qui peut être comblé par la simulation, est de tester et de valider les nouvelles technologies, en particulier, les ADAS et le véhicule autonome.

Comme nous l'avons vu précédemment, les ADAS sont développés dans le but d'accroitre la sécurité, l'efficacité, le confort et de réduire la charge de travail du conducteur. Afin que ces technologies soient intégrées dans les prochains modèles de véhicules, il est nécessaire de faire un grand nombre des tests et d'analyses avec chacun d'entre eux. [Mas, 2012] a montré qu'une première implémentation des ADAS dans les simulateurs de conduite permet aux conducteurs de s'adapter

au système d'aide à la conduite en permettant l'adoption d'une conduite plus sûre, pour éviter les obstacles, par exemple après un freinage.

Le groupe Renault ne cherche pas seulement à ce que ces systèmes soient utiles pour le consommateur, mais aussi qu'ils reçoivent une notation 5 étoiles à l'EuroNCAP, un système de notation de sécurité européenne. Pour ce faire, les systèmes sont mis à l'épreuve avec des situations récurrentes ou à haut risque qui ne sont pas faciles à développer dans la vie réelle et qui nécessitent de nombreuses heures de tests. Dans ce sens, l'industrie automobile se sert des simulateurs de conduite, car ils fournissent un environnement de test polyvalent, contrôlable et reproductible qui peut facilement être modifié pour simuler tout type de modification sur la conception du véhicule, du scénario de conduite et de son environnement.

Dans la validation des véhicules autonomes, les simulateurs de conduite sont indispensables. Par exemple, afin de valider un système autonome de niveau AD2 du tableau 1 ou niveau 4 dans la SAE (figure 2), le groupe Renault exige d'atteindre un niveau de fiabilité de 10^{-8} échecs par heure. Ce chiffre représente $20\text{x}10^{9}$ kilomètres de roulage [Wautier, 2018] équivalant à 100 ans d'essais sur les routes. Néanmoins, avec les simulateurs numériques, une grand part de ces kilométrages seront déployés en simulation.

En effet, c'est avec l'utilisation de l'ingénierie numérique que les simulateurs de conduite deviennent intéressants, car ils créent un environnement de simulation complet, sûr et parfaitement contrôlé. Ils sont équipés d'un système en temps réel qui reproduit le comportement dynamique du véhicule autonome et des agents externes, d'un environnement virtuel, d'un système audio et de mouvement. De plus, en simulation, le conducteur peut se sentir à l'aise et percevoir l'environnement comme s'il était dans une vraie voiture, tout en évitant les dangers, le gaspillage d'argent et de temps.

La figure 3 [Wautier, 2018] montre le rôle important des simulateurs de conduite dans toute la chaîne de validation que déploie l'alliance Renault-Nissan, afin de valider le véhicule autonome et les ADAS.

1.4 Simulateurs de conduite

Les simulateurs de conduite fournissent des environnements virtuels et interactifs comparables à ceux de la vie réelle afin d'être utilisés pour de nombreux aspects, tels que le comportement du conducteur en situation de conduite, l'évaluation des situations réelles de jour ou de nuit, la représentation de scénarios à risque pour le conducteur ou son entourage, le développement des systèmes véhicule ou de circulation, la validation des ADAS et la conduite autonome, entre autres.

Les simulateurs de conduite sont apparus vers 1960 [Kemeny, 2014]. Néanmoins, la première génération de simulateurs intégrant une plateforme dynamique dans

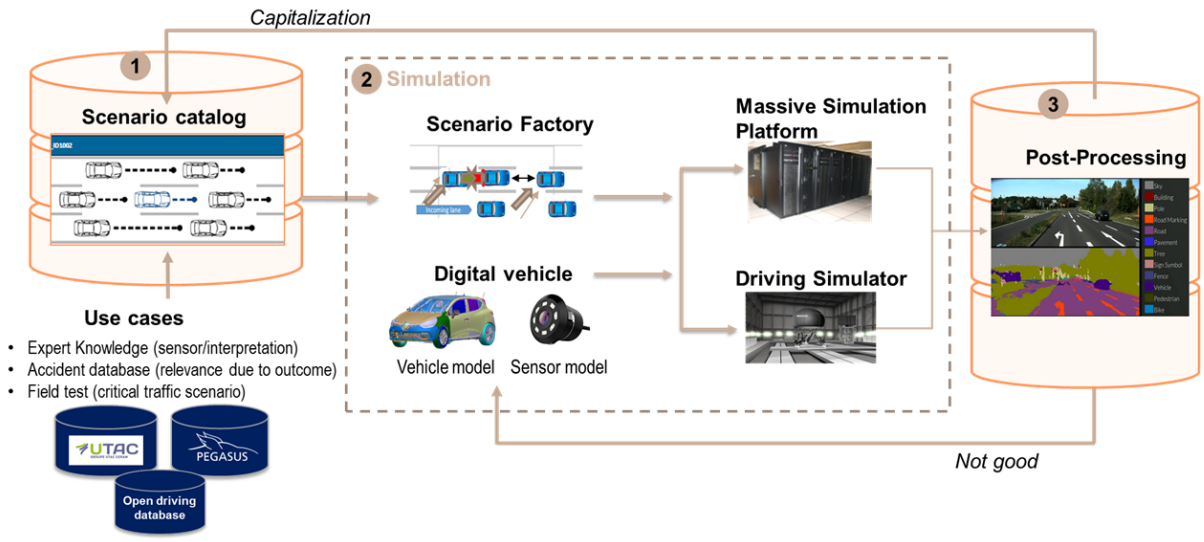


FIGURE 3 – Chaîne de simulation AD / ADAS de l'alliance Renault-Nissan.

l'industrie automobile a été construite à partir de 1980 [Drosdol & Panik, 1985; Nordmark *et al.*, 2004]. Depuis, de nombreux constructeurs industriels dans l'automobile ou l'aéronautique, ainsi que différents organismes de recherche ont conçu leurs propres systèmes, selon leurs besoins [Blana, 1996].

Avec le progrès des nouvelles interfaces interactives ou des outils informatiques de plus en plus performants, l'industrie automobile se dote de simulateurs à hautes performances afin de fournir aux conducteurs un environnement de plus en plus réaliste.

1.4.1 Composition

La conduite interactive en simulation nécessite de nombreux stimuli qui interagissent simultanément pour donner au conducteur une impression cohérente du monde réel. L'humain reçoit les informations de son environnement à partir de stimuli visuels principalement. Il est raisonnable de supposer alors qu'un système de projection d'images à haute résolution avec un grand champ de vision [1] suffira pour réaliser un simulateur de conduite. Cependant, ce n'est pas le cas, car les informations de mouvement, captées par le système vestibulaire, contribuent aussi de manière importante à l'immersion du conducteur et doivent donc être prises en compte dans la conception des simulateurs [Kemeny & Panerai, 2003].

Bien qu'un système de mouvement et un système visuel soient suffisants pour l'élaboration d'un simulateur de conduite dynamique, d'autres facteurs sont à considérer. Par exemple, les coûts de construction, les domaines d'application et les

1. Un champ de vision est considéré comme grand s'il est supérieur à 180 degrés. Cette valeur permet de reproduire la plage naturelle de la vision centrale et périphérique du conducteur en améliorant son immersion dans le simulateur [Hirsch & Quimper, 2015].

performances souhaitées sont autant de paramètres à prendre en compte. Pour la construction de simulateurs à grande échelle, des contraintes supplémentaires s'ajoutent, telles que la taille du bâtiment, la performance du système de mouvement, la course des pistons, la vitesse maximale, le bruit acoustique généré, la bande passante, la fluidité, la flexibilité de la configuration, pour n'en citer que quelques-uns [Mas, 2012]. L'ensemble de ces éléments permettent de créer une simulation avec un degré de réalisme nécessaire pour l'intégration et la validation des ADAS et des modèles de conduite autonome.

Ici, nous présentons quelques sous-systèmes qui, ensemble, reproduisent au mieux le comportement d'un véhicule réel :

- *Cockpit* ou cabine : pour les simulateurs à hautes performances, il s'agit généralement du corps d'un véhicule réel.
- Dispositifs de commande : pédales, boîte de vitesses et volant avec un niveau de retour d'efforts.
- Plateforme de pilotage de mouvement : vérins hydrauliques ou électriques, et rails. Ce système de mouvement reproduit les signaux dynamiques dans l'espace de travail du simulateur. Les signaux les plus importants dans la perception de mouvement sont ceux des accélérations linéaires et angulaires à hautes et basses fréquences. La perception de mouvement sera détaillée dans le chapitre 2.
- Modèle dynamique du véhicule : modèle approximatif du véhicule réel qui reproduit en simulation l'information fournie par les dispositifs de commande. Les signaux produits par ce modèle sont envoyés au système de contrôle de restitution de mouvements puis reproduits sur la plateforme dynamique. Ces signaux sont la position, la vitesse et l'accélération pour les 6 degrés de liberté (DDL) dans l'espace [Woon-Sung Lee *et al.*, 1998].
- Systèmes visuels qui affichent l'environnement de simulation en temps réel et garantissent une projection avec un grand champ de vision, afin d'augmenter l'immersion du conducteur.
- Système de son pour accroître l'immersion du conducteur à travers le bruit simulé du moteur ou du contact roue-route, et des différents ADAS.

Un système de simulation en temps réel et un logiciel de simulation de conduite sont également nécessaires afin d'intégrer les différents sous-systèmes. Il est primordial de valider en premier lieu le modèle dynamique du véhicule et la capacité du simulateur à reproduire les déplacements, vitesses et accélérations en temps réel. Il faut préciser que le réalisme de la simulation déprendra de la fidélité et de l'interaction de ces sous-systèmes. En effet, les sens de l'humain, tels que la vision, le système vestibulaire et la proprioception feront une pondération de tous les stimuli donnés par ces systèmes, afin de reproduire la sensation de vitesse et du mouvement dans le simulateur

1.4.2 Avantages et enjeux des simulateurs

Les simulateurs de conduite dynamique reproduisent des environnements virtuels et interactifs de circulation qui sont comparables à ceux de la vie réelle. Ils offrent de multiples avantages, parmi lesquels :

- **Sécurité :** le conducteur n'est pas exposé directement à des situations de conduite risquées et tout son environnement est entièrement simulé et contrôlé. Ainsi, les simulateurs optimisent la performance humaine en situation de conduite sans prendre de risque.
- **Polyvalence :** les simulateurs peuvent simuler divers scénarios tels que les conditions météorologiques, l'état de surface de la route, l'interaction avec d'autres agents (voitures, piétons ou objets) et même des situations qui ne se présentent que rarement dans le monde réel.
- **Simulation massive :** pour la validation de la conduite autonome, des centaines de millions de kilomètres sont nécessaires [Kalra & Paddock, 2016]. Cependant, si on ne considère que des tests de roulage sur route réelle, ce chiffre est difficile à atteindre. Par conséquent, les simulateurs de conduite offrent une reproductibilité des scénarios et permettent de représenter virtuellement le comportement du véhicule grâce à leur modèle dynamique, les systèmes de sécurité, les modèles de comportement de conduite, entre autres. Tout en fournissant un environnement virtuel sûr et réaliste pour les différents outils de *Hardware-In-the-Loop (HIL)*.
- **Simulation *Driver-in-the-loop* :** l'étude du comportement humain en situation de conduite est l'une des applications les plus répandues dans les simulateurs. Ils permettent d'évaluer la réaction des conducteurs face aux nouvelles technologies ou cas d'usages, et d'analyser l'interaction entre le conducteur et les éléments présents sur la route, tels que piétons, objets ou autres voitures. Enfin, ces simulateurs sont utilisés également pour l'entraînement de conducteurs professionnels.
- **Coût :** la représentation et la reproductivité de certaines situations sont très coûteuses lorsque seuls les tests réels sont considérés. Les simulateurs permettent la reproduction des mêmes situations à un coût bien inférieur, car ceux-ci sont réutilisables indéfiniment. De plus, ils constituent un outil d'aide aux constructeurs automobiles dans la phase de conception de modèles dynamiques et de nouvelles technologies. Ainsi, en amont de la fabrication, la simulation représente un moyen d'évaluation permettant de faire les modifications nécessaires sur le produit final.

Grâce aux avantages cités précédemment et à la qualité immersive et sensorielle de l'environnement virtuel, les simulateurs sont utilisés de plus en plus pour valider les systèmes d'aide à la conduite et les différents niveaux d'autonomie des véhicules.

Dans un premier temps, ils intègrent un modèle de véhicule afin d'être testés en simulation. Ensuite, les simulateurs fournissent aux concepteurs une grande quantité de données nécessaires lors du processus de validation.

Bien que les simulateurs de conduite possèdent de nombreux avantages indéniables comme indiqué auparavant, ils souffrent d'inconvénients non négligeables, tels que : le mal du simulateur [1], l'absence de réalisme de l'environnement virtuel, le coût de construction, la cohérence du mouvement, le niveau d'immersion et le fait que le comportement de conduite en simulation est forcément différent de celui dans le monde réel. Ce dernier s'explique par la connaissance au préalable de l'état d'immersion du conducteur en simulation [Blana, 1996].

Concernant la validation de la conduite autonome, les résultats obtenus avec les simulateurs de conduite peuvent être biaisés en raison d'une mauvaise qualité du simulateur, d'une prédisposition de l'utilisateur dans l'environnement virtuel, du comportement irréaliste du trafic ou de l'absence de vraisemblance entre le modèle autonome et la conduite réelle, entre autres. Certains de ces inconvénients restent présents avec les simulateurs de dernière génération, ce qui rend difficile la validation de la conduite autonome. Dans cette thèse, nous explorons diverses options d'algorithmes de contrôle de mouvement dédiés à l'amélioration de l'immersion du conducteur et la cohérence de mouvement du simulateur.

1.4.3 Types de plateformes

Au cours des années, différentes plateformes pour les simulateurs de conduite à base mobile ont été développées, afin de donner au conducteur une perception du mouvement comparable à celle de la vie réelle. Ces systèmes sont classés en deux catégories selon leur architecture mécanique et la capacité à reproduire les mouvements : les simulateurs de conduite avec une plateforme fixe (simulateurs statiques) ou une plateforme mobile (simulateurs dynamiques) [Mohellebi, 2005]. Nous nous intéressons exclusivement à la seconde catégorie, les simulateurs dynamiques, car la plateforme mobile permet une meilleure restitution de mouvements. Selon leur structure cinématique, il est possible de les classer en trois types : série, parallèle ou hybride.

Structure en série

Les systèmes en série ont été inspirés de la topologie du bras humain comme nous l'observons sur les figures 4a et 4b, qui montrent les simulateurs de DLR et de l'Institut Max Planck (MPI) pour la cybernétique biologique [Nieuwenhuizen &

1. Ensemble de symptômes qui résultent, entre autres, d'une différence dans la latence du système visuel et du système de mouvement. Parmi eux, les plus communes sont la transpiration, le vertige, le mal de tête et les nausées.

Bulthoff, 2013], respectivement. Les mécanismes de mouvement de ces systèmes sont intégrés en cascade, c'est-à-dire, la chaîne cinématique est en boucle ouverte. Parmi les avantages de ce type de structure, nous pouvons citer la disponibilité d'un large espace de travail et la manœuvrabilité [Ish–Shalom, 1982].

En revanche, si la structure bouge à des vitesses élevées ou porte un poids important, des problèmes dynamiques tels que des vibrations peuvent se produire. Les déplacements sont par ailleurs limités à ceux des articulations et dans la plupart des cas, ces limites sont plus restrictives que celles du système lui-même [Arioui & Nehaoua, 2013]. Enfin, comme pour toute structure poly-articulée de ce type, la gestion des redondances cinématiques et des positions singulières contraignent un peu plus les mouvements.

(a) Simulateur en structure série de DLR

(b) Simulateur « CyberMotion » : *MPI for Biological Cybernetics*

FIGURE 4 – Simulateurs à structure série.

Structure parallèle

Les structures parallèles sont conçues comme une alternative à celles en série. Avec ce type de structure, la capacité de charge utile, la performance dynamique et la précision de mouvement sont améliorées. Structurellement, elles contiennent une ou plusieurs chaînes cinématiques en boucle fermée.

Ce type de plateforme est la structure la plus courante pour les simulateurs de conduite dynamiques. Généralement la plateforme est constituée d'un hexapode autorisant 6 degrés de liberté (DDL) : trois en translations, x, y, z, et trois en rotation, roulis, tangage et lacet. L'hexapode est aussi nommé plateforme Gough-Stewart en honneur à ses inventeurs qui l'ont imaginé en 1965 [Stewart, 1965]. Ce mécanisme est monté sur un plateau fixe et contrôle un plateau mobile sur lequel se situe la cabine de conduite. Les deux sont reliés en au moins trois points par 6 vérins linéaires hydrauliques ou électriques.

(a) Simulateur DLR

(b) Simulateur VIRTTEX

FIGURE 5 – Simulateurs à structure parallèle.

Les limitations de mouvement de ces systèmes dépendent de la longueur minimale et maximale des vérins, de leur vitesse maximale et de leur force maximale. Comparées aux structures en série, les structures parallèles offrent des avantages en termes de rigidité, de précision, de vitesse et de charge admissible. Les inconvénients sont un espace de travail restreint et une dynamique difficile à modéliser.

Structure hybride

Les structures cinématiques hybrides possèdent des chaînes en boucle fermée et en boucle ouverte. Elles sont de plus en plus utilisées pour la réalisation de simulateurs à hautes performances, car elles permettent de combiner les avantages des deux structures citées auparavant. La structure se présente sous la forme d'une plateforme parallèle mise au-dessus d'une structure en série (habituellement une table X-Y), comme indiqué dans la partie 1.4.5.

La fabrication de ce type de structures est née du besoin d'améliorer la perception d'accélération de la part des conducteurs, particulièrement les accélérations linéaires continues et transitoires. Les travaux de cette thèse utiliseront ce type de structure pour tester les algorithmes de restitution de mouvements développés.

1.4.4 Différentes échelles de simulateurs

Connaître les capacités en matière d'espace de travail, vitesses ou accélérations attendues est essentiel pour choisir la stratégie de contrôle de mouvement la plus appropriée pour chaque simulateur. Aussi, nous nous intéressons à comparer les caractéristiques entre les différents types de simulateurs.

D'après la littérature, les simulateurs sont répartis en trois échelles selon leur degré de complexité et capacité : grande, moyenne et faible [Blana, 1996].

- **Échelle moyenne ou faible :** les simulateurs à bas coûts consistent, généralement, en un seul ordinateur avec des systèmes qui fournissent une fidélité raisonnable en termes de son, de rendu du mouvement et visuel avec un affichage graphique relativement peu coûteux. Les simulateurs de moyenne échelle utilisent des systèmes avancés d'imagerie et une cabine plus équipée. Ils sont soit à base fixe soit montés sur des systèmes vibratoires.
- **Simulateurs à grande échelle :** appelés également simulateurs à hautes performances. Il existe actuellement plus de 100 simulateurs de conduite à grande échelle exploités et construits par des universités, des institutions gouvernementales ou bien des constructeurs automobiles. Ces simulateurs ont un grand champ de vision et un système de restitution de mouvement qui permet de restituer non seulement des vibrations mais aussi des accélérations à hautes et basses fréquences.

Dans cette thèse, les stratégies de contrôle de mouvements seront développées pour des simulateurs de conduite à structure hybride à hautes performances.

1.4.5 Comparaison de simulateurs à hautes performances

Afin d'évaluer les stratégies de commande plateforme, il est nécessaire de connaître les caractéristiques dynamiques, ainsi que les différentes applications des simulateurs pour lesquelles les algorithmes de contrôle seront employés. Nous présentons ici une évaluation comparative de certains simulateurs existants et de leurs applications. Nous nous focalisons seulement sur les simulateurs ayant au moins 6 DDL de mouvement et avec un grand espace de travail pour effectuer des mouvements de translation.

Dans le domaine académique, le simulateur NADS pour *national advanced driving simulator* et le simulateur de l'Université de Leeds « UoLDS-Leeds » sont considérés comme parmi les plus avancés. NADS est le premier simulateur à implémenter une table de lacet qui tourne de 330° ; combiné à un hexapode et une table X-Y, le simulateur permet de fournir des signaux d'accélération avec une grande précision en mélangeant les accélérations fournies par les différents composants [Wilkinson, 2012]. Il est utilisé pour l'étude des facteurs humains et d'aspects liés à la sécurité routière, dans le but de réduire la mortalité routière aux États-Unis [Chen *et al.*, 2001]. Quant à UoLDS, il permet d'évaluer précisément le comportement des conducteurs, la distraction, la sécurité du trafic, la conception de véhicules et des routes, la conduite pour personnes handicapées et les facteurs humains. Afin de reproduire les mouvements de la cabine, UoLDS [Romano *et al.*, 2017] et NADS [Schwarz *et al.*, 2003] utilisent des stratégies de contrôle constituées de filtres passe-bas et passe-haut. Contrairement à UoLDS, dans NADS, les commandes d'accélération de translation à basse fréquence sont envoyées à la table X-Y, tandis que les commandes en haute fréquence sont envoyées à l'hexapode.

L'institut des systèmes de transport DLR a développé une conception différente par rapport aux autres simulateurs. Le simulateur de DLR utilise un hexapode avec les vérins hydrauliques articulés au niveau de la partie supérieure de la cabine, comme illustré dans la figure 5a. Cette disposition offre un large déplacement à six mètres de hauteur du sol, ainsi qu'une meilleure restitution des accélérations continues grâce aux angles d'inclinaison du simulateur. Même si les systèmes mécaniques sont différents, les thématiques de recherche ne varient pas beaucoup d'un simulateur à l'autre. DLR étudie le comportement, l'interaction et les problèmes de fonctionnalités que rencontrent les conducteurs avec les systèmes ADAS et les interfaces homme-machine dans le trafic réel [Suikat, 2005]. Sa commande repose également sur des filtres qui reproduisent les accélérations du véhicule dans l'espace de travail restreint de la plateforme [Seehof *et al.*, 2014].

Bien que de nombreux simulateurs soient utilisés pour ne faire que de la recherche, de nombreuses entreprises dans les secteurs de l'automobile et de l'aéronautique ont construit également leurs propres simulateurs. Voici quelques exemples de simulateurs dans l'industrie automobile :

- **Nissan :** leur principal intérêt est l'étude des relations entre l'humain et le véhicule. Nissan s'intéresse aux situations de freinage avec accélérations à haute fréquence, afin d'améliorer le sentiment d'immersion [Nobuyuki *et al.*, 2002]. Le dernier simulateur hybride à hautes performances a été nommé NIDS2.

- **Daimler Benz** : son hexapode peut se déplacer selon les deux directions, latérale (MBS1) et longitudinale (MBS2) [Klüver *et al.*, 2015]. Son but est d'analyser et de développer les fonctions du véhicule et d'évaluer la sécurité routière, le comportement humain sur les routes à travers différents scénarios et en conditions normales de circulation [Blana, 1996], entre autres.

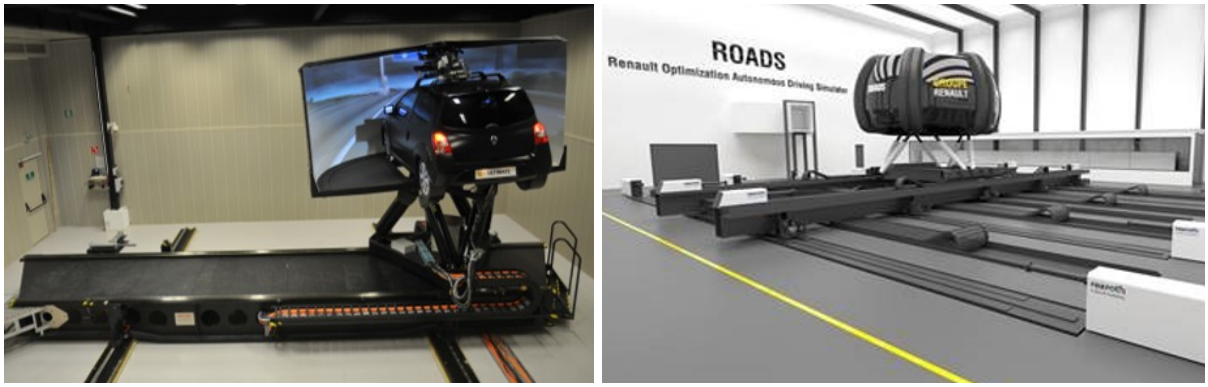
- **SHERPA-PSA- Peugeot-Citroën :** construit en 2007, le simulateur a une large base de mouvement permettant d'améliorer significativement le réalisme lors d'une simulation de conduite. Le dôme dans lequel se trouve la cabine peut tourner de 90°, ce qui permet d'utiliser une course de déplacement plus longue pour les accélérations latérales ou longitudinales [Chapron & Colinot, 2007]. Son système de mouvement est basé sur une stratégie classique de contrôle [Stratulat *et al.*, 2011].

- **Toyota :** le simulateur est utilisé pour analyser les caractéristiques de conduite et la vérification des nouvelles technologies pour la sécurité routière. Il permet une rotation complète de 360 degrés dans l'angle de lacet, une base X-Y de 35mx20m et un hexapode, formant un système à 12 DDL [Slob, 2008]. Malheureusement, il est difficile de trouver des informations supplémentaires sur le type de contrôle employé pour ce simulateur.

Simulateurs dynamiques à hautes performances du groupe Renault :

Le projet de cette thèse est à l'initiative du groupe Renault, pour améliorer le contrôle de la plateforme des simulateurs existants et de les intégrer dans les simulateurs de conduite dynamique actuels et futurs :

- **Ultimate :** développé pour la première fois en 2001 dans le cadre d'un projet de recherche européen (Eureka 1493) et renouvelé en 2011. Il est utilisé pour évaluer le comportement humain en situation de conduite, la validation des systèmes d'aide à la conduite et des scénarios pour les véhicules autonomes, entre autres. La cabine est montée sur un hexapode construit par Rexroth, Pays-Bas, lui-même monté sur une table X-Y. Le système a donc 8 DDL. Le contrôle de la plateforme est basé sur la commande par modèle prédictif [Fang & Kemeny, 2014].
- **ROADS -Renault Optimization Autonomous Driving Simulator :** ce nouveau simulateur à très haute performance avec des caractéristiques uniques au monde fera partie du groupe Renault à partir du 2021. Il sera conçu principalement pour la validation des véhicules autonomes.



(a) Simulateur ULTIMATE (b) Simulateur ROADS

FIGURE 6 – Simulateurs dynamiques à hautes performances.

> *« Pour proposer un véhicule autonome sûr pour tous, nous devons le valider sur des milliards de kilomètres. Pour compléter les tests physiques, la seule option est la simulation massive, avec un grand nombre de scénarios de conduite. L'un des pires scénarios est le transfert entre les modes manuel et autonome dans différentes situations de circulation. Le nouveau simulateur dynamique de conduite haute performance (ROADS) aidera Renault à tester tous ces cas de la manière la plus efficace et la plus réaliste avec de vrais pilotes (DIL) et à concevoir la meilleure expérience pour le client final (simple et sécurisée) »* - Olivier COLMARD, Directeur CAE et PLM, VP Renault Engineering.

Des informations précédentes, nous pouvons déduire que les applications des simulateurs sont assez similaires d'une organisation à l'autre. Les applications les plus courantes, que ce soit pour la recherche ou l'industrie sont les facteurs humains concernant la perception de mouvements ou sa sécurité et l'impact des nouvelles technologies ADAS.

En termes de capacités physiques et de type de contrôle, le tableau 2 présente les limites de déplacement, vitesse et accélération des sept simulateurs, choisis selon deux critères : leur structure physique innovante comme celle de DLR, Daimler, et le nombre élevé de degrés de liberté comme pour NADS, NIDS2, ROADS qui intègrent une table de lacet en plus. Pour cette thèse, les algorithmes de contrôle seront développés et intégrés exclusivement dans le simulateur ULTIMATE.

Parmi ces sept simulateurs, DLR, NADS, Daimler et SHERPA utilisent une stratégie de contrôle appelée classique (voir partie 3.1), les autres utilisent une stratégie de contrôle par commande prédictive (voir partie 3.4). En effet, si l'on ne considère que les applications ou les caractéristiques physiques de chaque simulateur, le choix de l'algorithme de contrôle n'est pas évident. Aussi, dans le troisième chapitre, nous présenterons une comparaison détaillée entre les différents algorithmes existants, avec une attention particulière pour les algorithmes classique et prédictif.

Simulateur	**DLR**	**NADS**	**Daimler**	**SHERPA**	**NIDS2**	**ULTIMATE**	**ROADS**
Axes				**± Position**			
Longi. (m)	1.5	9.75	1.3	—	0.3	0.28	0.73
Latéral (m)	1.4	9.75	1.3	—	0.3	0.26	0.70
Vertical (m)	1.4	0.61	1	0.2	0.25	0.20	0.68
Roulis (°)	21	25	20	18	15	15	-22
Tangage (°)	21	25	+24-19	18	15	15	21
Lacet (°)	21	330	38	23	15	15	22
X rail (m)	ras	9.7	5.6	5	3	5,5	15
Y rail (m)	ras	9.7	–	2.75	11	6	15
				± Vitesse			
Longi. (m/s)	2	6.1	1.25	—	0.6	0.7	0.9
Latéral (m/s)	2	6.1	1	—	0.6	0.7	0.9
Vertical (m/s)	2	1.52	1	2	0.5	0.4	0.8
Roulis (°/s)	50	45	25	20	30	40	28
Tangage (°/s)	50	45	25	20	30	40	26
Lacet (°/s)	50	60	25	30	30	60	26
X rail (m/s)	—	6	4	3	2.5	2	10
Y rail (m/s)	—	6	—	3	10	3	10
				± Accélération			
Longi. (m/s^2)	10	0.6 g	1g	—	5	7.5	13
Latéral (m/s^2)	10	0.6 g	1	—	5	7.5	12
Vertical (m/s^2)	10	1g	1	5	9	5.0	15
Roulis ($°/s^2$)	250	120	200	300	230	300	400
Tangage ($°/s^2$)	250	120	200	300	230	300	400
Lacet ($°/s^2$)	250	120	200	600	230	600	400
X rail (m/s^2)	—	6	6.8	5	4	6,5	10
Y rail (m/s^2)	—	6	—	5	12	6,5	10

TABLEAU 2 – Limites physiques des différents simulateurs hybrides à hautes performances.

Chapitre 2

Mouvement et perception humaine

La perception de notre environnement résulte d'une série d'étapes permettant de déterminer notre comportement et nos réactions aux stimuli externes, tels que les images, les sons, les arômes, etc. Ces stimuli sont détectés par nos systèmes sensoriels puis envoyés au cerveau sous la forme d'informations [Coren, 2003], afin d'interpréter l'environnement et ces éléments. La perception d'un individu étant subjective, sélective et temporelle, elle peut varier d'un sujet à un autre en fonction des besoins, des motivations ou des expériences.

Concernant la perception du mouvement, le corps humain est sensible à trois types d'équilibres : l'équilibre statique qui représente le maintien de la position du corps sous l'effet de la gravité, l'équilibre cinétique qui est généré par des forces passives, par exemple, le déplacement d'une voiture, et l'équilibre dynamique généré par le déplacement de notre corps en réponse à des mouvements de rotation et d'accélération. Les trois équilibres sont perçus par notre cerveau grâce aux informations provenant du système vestibulaire, des récepteurs tactiles, du système visuel, du système proprioceptif, du cortex moteur, du système de coordination musculaire et des centres de diencéphales [Arruñada, 2015]. Dans le cadre de cette thèse, nous nous intéressons plus particulièrement aux systèmes sensoriels qui coopèrent pour la perception du mouvement du corps et le contrôle de la posture et qui sont le système visuel, le système vestibulaire et la proprioception musculoarticulaire [Murgovski, 2007]. Ces systèmes sensoriels sont indépendants, mais coopèrent pour transmettre des informations au cerveau permettant de localiser le corps dans l'espace.

En simulation de conduite, la compréhension de ces systèmes est primordiale pour permettre une bonne immersion dans l'environnement virtuel. En effet, lors d'un déplacement passif en environnement naturel, les informations visuelles, vestibulaires et tactiles coïncident physiquement afin que notre cerveau perçoive le déplacement. Toutefois, malgré l'évolution des technologies, certaines incohérences apparaissent toujours dans la perception du mouvement sur les simulateurs de conduite. Pour réduire ces incohérences perceptives, les stratégies de restitution de mouvement (MCA pour *Motion Cueing Algorithms*) doivent intégrer les modèles de

perception humaine dans un environnement virtuel.

C'est dans cette volonté que ce chapitre introduit le fonctionnement des différents organes sensoriels, ainsi que leur impact sur la perception du mouvement. Ainsi, nous présentons le système visuel, la proprioception et le système vestibulaire. Pour chacun de ces systèmes sensoriels, nous nous concentrons sur les aspects importants dans la restitution de mouvement dans les simulateurs de conduite : la vitesse et les accélérations linéaires et angulaires. Ces aspects sont accompagnés de leur modèle mathématique, ainsi que d'une analyse théorique, afin de choisir ceux qui vont intégrer les MCA développées dans cette thèse. Enfin, nous présentons l'importance d'une intégration sensorielle et les méthodes mises en œuvre dans le contrôle de plateformes permettant d'améliorer la perception du mouvement grâce à l'intégration des organes perceptifs.

2.1 Système visuel

La vision donne une perception du monde qui nous entoure. Cette perception commence lorsqu'un stimulus, la lumière, est réfléchi vers les yeux. Ceux-ci forment une image dans le cerveau, qui à son tour, analyse les informations afin de définir les objets, les couleurs et les formes.

En simulation de conduite, le système visuel stimule le conducteur pour générer l'information relative aux images visualisées sur l'écran. Ces images sont générées en temps réel avec une résolution et une fréquence comprise entre 30 Hz et 60 Hz, donnant ainsi une visualisation confortable. Afin de favoriser l'immersion pendant la simulation, les images virtuelles doivent être compatibles avec la vision humaine avec une bonne qualité d'affichage à tout moment de la simulation [Kemeny, 1999].

2.1.1 L'anatomie et la physiologie oculaire

L'œil est un système optique hautement spécialisé dans le traitement de la lumière. Celle-ci pénètre dans l'œil à travers la cornée, une zone dure et transparente qui fournit la plus grande puissance optique. À l'intérieur de cette couche, la membrane choroïdienne empêche la réflexion de la lumière et nourrit la couche externe de la rétine [LaValle, 2016]. Les rayons lumineux contrôlés en quantité par l'iris continuent leur trajet à travers la pupille et le cristallin, ce dernier donne une image inversée du monde extérieur sur la couche sensible de la rétine à l'arrière du globe oculaire [Puell Marin, 2006]. Pour que l'image soit nette, les muscles ciliaires corrigent la courbure du cristallin, un mécanisme d'accommodation. Enfin, un message électrique est transmis au nerf optique du cerveau où l'information est traitée et consciemment appréciée comme vision [Kolb, 2003]. La physiologie de

l'oeil humain est montrée dans la figure 7 [1].

La rétine possède des neurones sensoriels qui réagissent à la lumière, et des circuits complexes neuronaux qui effectuent les premières étapes du traitement de l'image. Elle contient au moins deux types de cellules sensoriels ou photorécepteurs : les bâtonnets et les cônes. Respectivement, ils sont responsables de la vision à faible luminosité et à la lumière du jour et de la distinction entre les couleurs. Le reste de la surface externe de l'œil est protégé et entouré par la sclérotique, une couche dure et blanche.

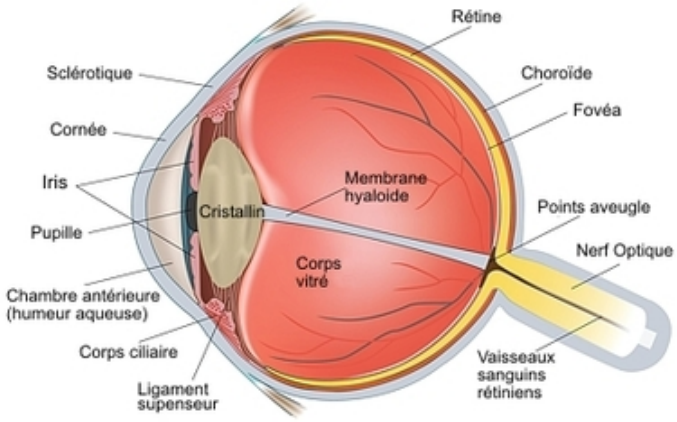


FIGURE 7 – Physiologie de l'œil humain et ses différentes structures.

2.1.2 Effet du système visuel sur la conduite automobile

Le comportement humain au volant est déterminé par divers facteurs liés à la vision, qui fournit au conducteur des informations relatives à l'exploration visuelle à tout moment. Ainsi, la vision est la modalité sensorielle principale dans ce domaine [Direction des routes et de la circulation routière, 1981].

L'information obtenue par la vision, permet à l'homme de connaître sa vitesse de déplacement, les caractéristiques de l'environnement, la position relative occupée, son orientation et celle des éléments de l'environnement. Parmi ces informations, nous soulignons les suivantes [Egea-Caparrós, 2001] :

- **Vitesse de déplacement :** pendant la conduite, la vision doit s'adapter rapidement à la position des objets perçue par la succession d'images dans les différentes régions de la rétine. La perception de vitesse peut varier selon les éléments présents dans la périphérie visuelle (arbres, panneaux, entre autres). Pendant une longue période de temps sur la route, la capacité de réponse des

1. https://www.provisu.ch/fr/maladies-les-plus-frequentes/plaies-et-traumatismes.html

détecteurs de mouvement dans la rétine est réduite à cause de l'exposition à une stimulation continue. Ce phénomène peut provoquer des situations à haut risque, car l'homme ne perçoit pas correctement la vitesse à laquelle il circule.

- **Direction de déplacement :** le déplacement de l'humain vers une direction provoque des changements dans le flux optique qui stimulent la rétine. Par conséquent, les éléments proches de notre vision qui sont dans un sens différent du déplacement semblent se retirer vers la périphérie dans une sorte de flux radial à l'extérieur. C'est ainsi que nous pouvons connaître la direction de déplacement dans l'environnement.
- **Temps de collision :** le conducteur est informé du temps qu'il faut pour atteindre un objet sur la route par le taux d'expansion de l'image rétinienne de cet objet au fil du temps. Ce temps est une information très utile lorsqu'il s'agit d'éviter une collision.

Dans la vie réelle ou en simulation, il existe deux types d'indices visuels : la vision périphérique et la vision fovéale. La première est responsable des objets qui se déplacent rapidement et la deuxième analyse plus précisément ces objets dans le champ central de la vision [Direction des routes et de la circulation routière, 1981]. Parmi ces deux indices visuels, c'est la vision périphérique qui a le plus d'importance dans la perception du mouvement propre [Telban & Cardullo, 2005].

2.1.3 Modèle du système visuel pour la perception du mouvement

La sensation de mouvement dans un environnement stationnaire due à la présence de mouvement visuel est communément appelée vection [Kim *et al.*, 2015]. Celle-ci peut être linéaire pour le mouvement de translation ou circulaire pour le mouvement de rotation en lacet, tangage ou roulis.

La réponse du système visuel à la vection linéaire est plus rapide que celle à la vection induite par rotation. D'après la littérature, le délai pour percevoir une vection linéaire, suite à la présence d'un stimulus visuel, est de 1 seconde, et d'environ 10 secondes pour la sensation de vection circulaire [Mohellebi, 2005]. Au bout de 5-15 secondes, il y aura une saturation de l'effet de vection, suivie par une diminution après 30-50 secondes [Colombet, 2010]. La vection est influencée également par les stimuli vestibulaires dûs à la présence de mouvement en plus de la stimulation visuelle [Reymond, 2000].

Le système visuel n'est ni rapide ni stable lors de mouvements à haute fréquence [Laurens & Droulez, 2008]. Toutefois, l'information visuelle est essentielle à basse fréquence pour la perception du mouvement. Dans ce sens, la dynamique de la vection ressemble à une fonction de transfert passe-bas qui adapte la perception

visuelle à des stimuli constants. Ainsi, les indices visuels dominent la perception de la vitesse et l'orientation en régime permanent lorsque les fréquences sont au-dessus de 0,1 Hz [Young, 1978].

L'influence de l'estimation visuelle sur la perception du mouvement, également nommée influence optocinétique des stimuli visuels, a été étudiée principalement pour la vitesse de rotation en lacet [Van der Steen, 1998]. Afin de la représenter pour tous les degrés de liberté, nous prenons le modèle proposé par [Telban & Cardullo, 2005] dans la figure 8, inspiré par les modèles de [Young, 1978] et [Van der Steen, 1998].

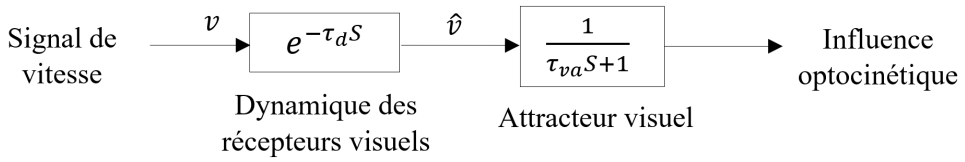


FIGURE 8 – Schéma fonctionnel du modèle de perception visuelle.

La dynamique des récepteurs visuels est représentée par un retard de temps τ_d de 90 millisecondes, car seuls les indices visuels périphériques sont considérés. Ces indices ont un délai des perceptions entre 10 et 150 millisecondes [Van der Steen, 1998]. En fonction des conditions expérimentales, la constante de temps de l'attracteur visuel τ_{va} peut varier sur une plage de 1 à 8 secondes. Afin de produire les latences notées dans la littérature, la valeur de 2 secondes est choisie [Telban & Cardullo, 2005]. Ensuite, la composante appelée attracteur visuel représente l'estimation visuelle du mouvement [Van der Steen, 1998].

D'autre part, les travaux développés par [Van der Steen & Brockhoff, 2000] suggèrent que la perception d'accélération par les stimuli visuels n'est pas perçue par les sujets entre deux amplitudes différentes de vitesse. Cela renforce l'idée que les stimuli visuels sont importants dans la simulation, principalement pour la perception de vitesse et l'orientation de soi-même.

2.2 La somesthésie

Le système sensoriel somatique comprend la sensibilité cutanée, la sensibilité viscérale et la proprioception (ou sensibilité musculo-articulaire). Dans cette thèse, nous nous intéressons exclusivement à la proprioception qui désigne la perception, de la position, de la vitesse et les mouvements effectués des différentes parties du corps dans l'espace [Priot, 2012].

2.2.1 La proprioception

Ce concept fait référence à la capacité de l'humain à détecter le mouvement du corps, la position des articulations, la volonté de ses membres, le poids, la régulation de la direction et l'amplitude de mouvement, les réactions automatiques et la relation entre les différentes parties du corps et sa position dans l'espace.

Le système proprioceptif est composé d'une série de nerfs récepteurs, nommés propriocepteurs. Ils aident à atteindre un mouvement désiré et à détecter le degré d'étirement musculaire et de stress grâce aux informations transmises à la moelle et au cerveau. L'information concernant tous les changements dans la géométrie du corps, comme la position relative de la tête par rapport au corps, est fournie par les récepteurs musculaires exposés ci-dessous :

- **Fuseaux neuromusculaires :** ils se trouvent dans le muscle et assurent le contrôle du tonus musculaire dû à un étirement.
- **Les organes tendineux de Golgi :** ils se situent dans les tendons des muscles et sont responsables de la mesure de la tension développée par le muscle.
- **Les mécanorécepteurs articulaires :** les ligaments et les capsules articulaires.
- **Récepteurs cutanés :** ils fournissent des informations sur l'état tonique musculaire et sur le mouvement, ce qui contribue au sens de la position et du mouvement, en particulier des membres du corps.

L'information concernant la perception du mouvement par la proprioception est surtout importante lorsque des mouvements actifs de la tête ou du corps se produisent. Par exemple, en simulation de conduite, ces stimuli sont induits par le siège grâce aux vibrations, entre autres. Dans cette thèse, le conducteur du simulateur est considéré comme n'étant pas en mouvement par rapport à la plateforme, ainsi, nous nous concentrerons principalement sur les stimuli qui peuvent être fournies par la plateforme du simulateur. Par conséquent, les informations obtenues par le système sensoriel somatique ne seront pas prises en compte dans le développement des stratégies de contrôle de mouvement.

2.3 Système vestibulaire

Dans la perception du mouvement, le système vestibulaire joue le rôle le plus important. Il possède tous les capteurs de mouvement de l'homme qui lui permettent

de sentir les forces spécifiques [1] [Strachan, 2014]. Par exemple, un accéléromètre linéaire répond à une force spécifique comme le fait le système sensoriel humain [Reid & Nahon, 1985].

En tant que récepteur sensoriel, le système vestibulaire, situé dans l'oreille interne, est l'un des centres d'information sur l'état d'équilibre du corps. Parmi ses fonctions nous pouvons citer : l'équilibre, la stabilisation de l'image sur la rétine, afin de permettre une vision claire lors de mouvements de la tête, la perception de l'orientation statique et du mouvement, la position de la tête dans l'espace, la perception des accélérations linéaires, angulaires et les forces gravitationnelles dans le corps, entre autres. Le système vestibulaire fait également la communication entre l'appareil vestibulaire périphérique, le système oculaire, les muscles de la posture, le tronc cérébral, le cervelet et le cortex [Khan & Chang, 2013].

2.3.1 Anatomie vestibulaire

L'oreille interne se compose d'un labyrinthe osseux et d'un labyrinthe membraneux. Dans ce dernier se trouvent les capteurs principaux pour la perception du mouvement chez l'homme. Cette structure sert à informer sur la position et la vitesse de la tête, instaurer les réflexes pour stabiliser la tête, les yeux et le corps, décoder l'information visuelle pour interpréter l'information de mouvement et décoder l'information proprioceptive.

Le labyrinthe membraneux est divisé en un labyrinthe antérieur composé de la cochlée et un labyrinthe postérieur nommé aussi vestibulaire qui contient le saccule, l'utricule et les canaux semi-circulaires [Arruñada, 2015], tel que montré dans la figure 9. La cochlée est l'organe qui s'occupe de la perception auditive et le labyrinthe vestibulaire s'occupe de la fonction d'équilibre. Ce dernier contient les cinq capteurs qui mesurent l'accélération linéaire et angulaire de la tête. Parmi ces capteurs, trois sont les canaux semi-circulaires latéraux, supérieurs et postérieurs, et les deux autres sont les organes otolithiques : l'utricule et le saccule [Khan & Chang, 2013]. Les canaux semi-circulaires détectent les rotations et sont disposés perpendiculairement les uns aux autres. Les organes otolithiques sentent le mouvement linéaire. Chacun des cinq organes récepteurs est rempli de liquide endolymphe et possède un groupe de mécanorécepteurs sensoriels appelés cellules ciliées.

2.3.2 Les organes otolithiques

L'utricule et le saccule sont deux sacs du labyrinthe membraneux qui répondent à l'accélération ou décélération linéaire, aux forces gravitationnelles et à l'inclinaison

1. Force de réaction totale agissant sur un corps par unité de masse en m/s^2. La force spécifique est définie comme étant la différence vectorielle entre l'accélération inertielle et l'accélération due à la gravité.

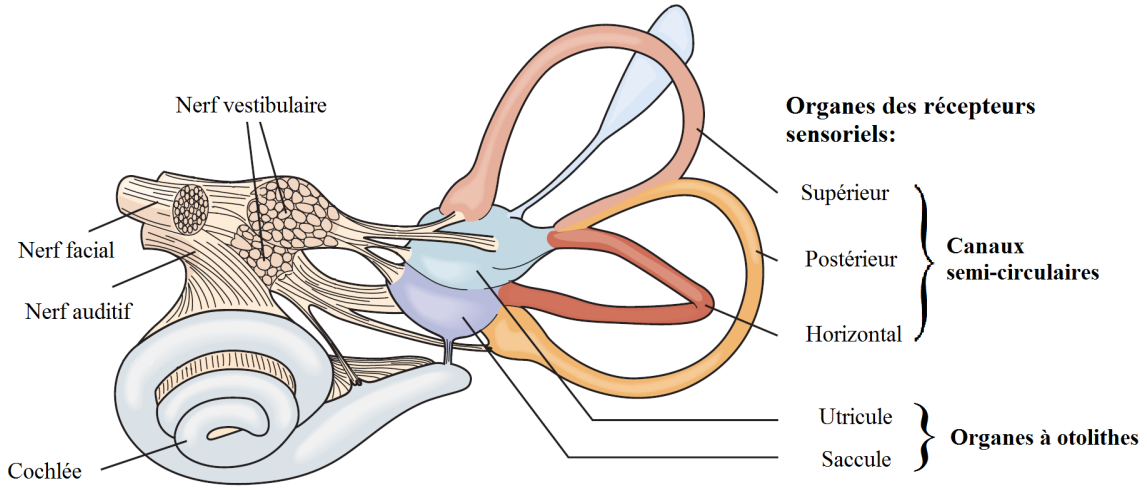


FIGURE 9 – Anatomie du système vestibulaire. Image tirée de [Kandel *et al.*, 2000].

de la tête. Ils vont donc sentir l'orientation de la tête dans l'espace [Khan & Chang, 2013]. Chaque organe a des macules, des organes sensoriels qui contiennent les cellules ciliées. Dans la figure 10 nous pouvons observer que les macules sont couvertes d'une lame gélatineuse appelée la membrane otolithique sous une fine couche de cristaux nommés otolithes [Kandel *et al.*, 2000].

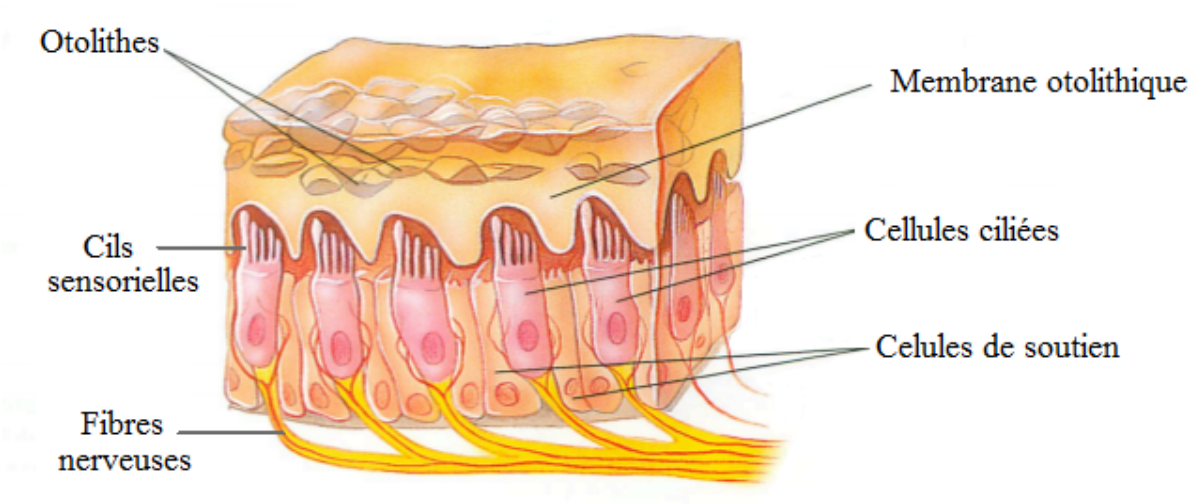


FIGURE 10 – Agrandissement d'une coupe du saccule et de l'utricule. Image tirée de [Raven, 2011].

La présence d'un mouvement linéaire, l'action de la gravité ou l'inclinaison de la tête provoque une résistance inertielle et une force de cisaillement entre la membrane otolithiques et la surface maculaire. Cela induit une inclinaison des cils, suivie d'une modification des fibres nerveuses et l'envoi des informations au cerveau afin de permettre la perception du mouvement. En fonction de la direction de la force de cisaillement, les cellules sensorielles peuvent être excitées ou inhibées. Dans

l'utricule, les macules otolithiques détectent le mouvement dans le plan horizontal et dans le saccule, les macules détectent le mouvement dans le plan vertical.

Bien que ces organes soient les principaux capteurs de mouvement linéaire, ils nécessitent des informations sensorielles supplémentaires pour différencier entre l'accélération linéaire et l'inclinaison de la tête. Ces informations sont fournies entre autres par les canaux semi-circulaires.

2.3.3 Les canaux semi-circulaires

Ces organes du système vestibulaire détectent la rotation de la tête et l'accélération angulaire. Chaque canal représente un tube cylindrique semi-circulaire, ouvert par deux orifices dans la cavité vestibulaire, orientés vers l'utricule et vers l'ampoule, qui contient les fibres du nerf vestibulaire. L'ampoule est recouverte de cupule, une substance gélatineuse dans laquelle se trouvent les cellules ciliées, qui servent à détecter les mouvements [Khan & Chang, 2013], tel que représenté dans le schéma 11.

Le système vestibulaire détecte les accélérations de la tête grâce à la réponse en inertie des cellules ciliées. La rotation angulaire de la tête cause une rotation dans la direction contraire dans l'un des plans des canaux semi-circulaires. Ce mouvement provoque le déplacement du liquide endolymphe dans les canaux semi-circulaires vers la cupule à cause de la pression exercée, ce qui plie les cils dans la direction opposée à la rotation. L'inclinaison des cils entraine la stimulation des cellules ciliées et des fibres du nerf vestibulaire.

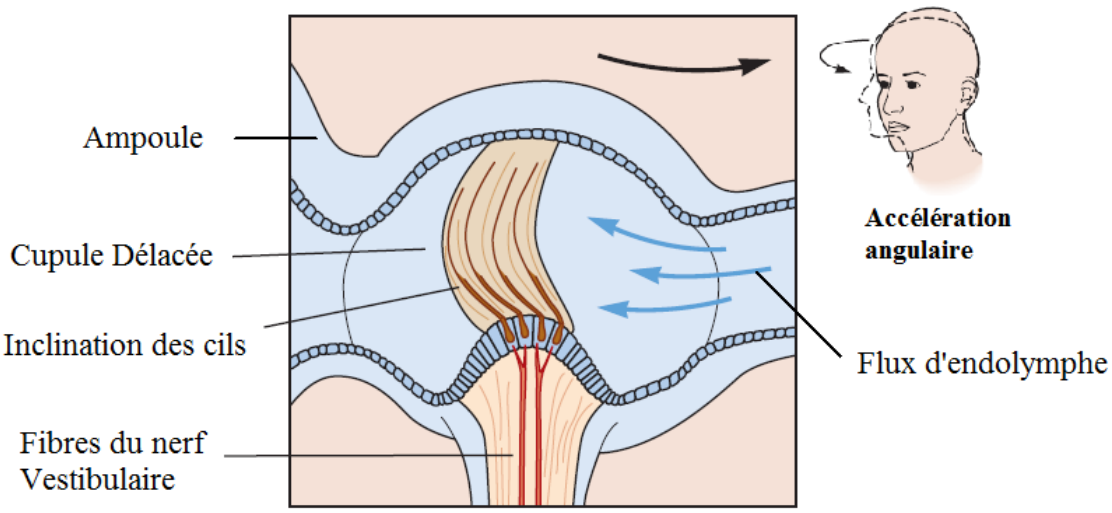


FIGURE 11 – L'ampoule d'un canal semi-circulaire en présence d'une accélération rotationnelle. Image tirée de [Kandel *et al.*, 2000].

Ainsi, l'accélération est transformée en déviation de la cupule par le système cupule-endolymphe. Ensuite, cette déviation se transforme en impulsions électriques par le système de transduction mécanoneurale constitué des cellules ciliées sensorielles et des fibres du nerf vestibulaire.

2.3.4 Modèles pour les composants du système vestibulaire : Modèle de perception humaine

Sans la connaissance du fonctionnement du système vestibulaire et des seuils de perception du mouvement, la restitution de mouvement dans les simulateurs de conduite devient limitée. En effet, les différentes techniques d'illusion de mouvement telle que le *tilt coordination* dans la partie 2.3.8 ou le *washout* dans la partie 3.1 ne pourront pas être appliquées correctement. En outre, les modèles du système vestibulaire sont de plus en plus utilisés dans les algorithmes de contrôle de plateformes, afin d'améliorer le ressenti du conducteur pendant la simulation. Ainsi, le conducteur aura les repères inertiels nécessaires pour lui faire croire qu'il conduit comme dans la vraie vie.

Depuis les années 60, la littérature présente plusieurs modèles de système vestibulaire. Nous trouvons principalement deux modèles distincts, un premier dédié aux organes otolithiques [Young & Meiry, 1968] qui sentent les accélérations linéaires et un second dédié aux canaux semi-circulaires [Young & Oman, 1969] qui sentent les accélérations rotationnelles. Leur unification forme le modèle complet du système vestibulaire.

Par la suite, nous décrirons les différents modèles du système vestibulaire les plus significatifs et implémentés dans le déploiement des MCA. Ces modèles mathématiques seront décrits sous la forme de fonctions de transfert, tel que montré dans la figure 15.

2.3.5 Modèle des organes otolithiques

Précédemment, nous avons vu que lorsque la tête perçoit une force spécifique, il se produit un déplacement de la membrane otolithique. Les forces spécifiques sont le résultat d'un déplacement vers l'avant et vers l'arrière (poussée), vers le haut et vers le bas (soulèvement) ou vers la droite et vers la gauche (balancement). Par exemple, avec l'inclinaison de la tête vers l'avant, un mouvement relatif est généré, ce qui produit une déformation de la surface maculaire et de la membrane otolithique qui est en cisaillement. C'est ainsi que l'homme perçoit un mouvement linéaire.

La modélisation des organes à otolithes est basée sur sa géométrie et les caractéristiques de la force de cisaillement interne. C'est pourquoi l'utricule et le saccule sont

modélisés par un système de masse-ressort-amortisseur qui décrit le mouvement des otolithes [Zacharias, 1978].

Les modèles mathématiques des organes otolithiques sont représentés en fonction de la force spécifique, l'accélération ou la réponse du tir afférent (AFR) [Momani & Cardullo, 2018]. Dans ce document, nous utiliserons le modèle qui utilise la force spécifique, puisque ce modèle est le plus utilisé dans les algorithmes de restitution de mouvement. Ainsi, notre modèle a pour entrée la force spécifique f (l'accélération de la tête) et pour sortie l'accélération linéaire perçue $\hat{f}$. La fonction de transfert qui correspond au modèle de perception de la force spécifique subjective, selon tous les axes de mouvement, est [Reid & Nahon, 1985] :

$$\frac{\hat{f}}{f} = \frac{G_{oto}(\tau_{aoto}s + 1)}{(\tau_{Loto}s + 1)(\tau_{soto}s + 1)} \tag{1}$$

où, τ_{Loto} représente la constante de temps long, τ_{aoto} la constante de temps de délai ou *lead constant*, τ_{soto} la constante de temps court et la constante G_{oto} représente la sensibilité statique. Ces constantes ont été détaillées dans les travaux de [Zacharias, 1978]. Elles sont obtenues grâce aux réponses subjectives collectées durant diverses expériences, néanmoins, leurs valeurs peuvent varier d'un auteur à l'autre. Ainsi, nous pouvons conclure que les zones de sensibilité du système vestibulaire représentées par différentes constantes de temps varient d'un sujet à un autre.

Différents modèles des organes otolithiques ont été proposés pour représenter la force spécifique perçue. Plusieurs sont présentés dans l'état de l'art réalisé par [Asadi *et al.*, 2016]. Toutefois, pour la conception des MCA, seuls deux modèles sont principalement utilisés dans un contexte de théorie du contrôle et de modélisation. Le premier modèle proposé par [Young & Meiry, 1968], nommé **Mo1**, a été utilisé dans de nombreuses études et expérimentations du MCA [Asadi *et al.*, 2015a; Augusto, 2009; Chen & Fu, 2010; Fang & Kemeny, 2014; Garrett & Best, 2013; Lamprecht *et al.*, 2019; Reid & Nahon, 1985; Zacharias, 1978] ; le second modèle **Mo2**, plus récent, a été proposé par [Telban & Cardullo, 2005]. Il est également utilisé par de nombreux auteurs dans la conception de MCA [Asadi *et al.*, 2017; Baseggio *et al.*, 2011; Beghi *et al.*, 2012; Bruschetta *et al.*, 2019; Mohammadi *et al.*, 2016; Zaychik *et al.*, 2009]. Le tableau 3 montre les valeurs des paramètres pour ces deux modèles.

Modèle	**Paramètres** (s)			
	G_{oto}	τ_{aoto}	τ_{Loto}	τ_{soto}
Mo1	0,4	13,2	5,33	0,66
Mo2	0,4	10	5	0,016

TABLEAU 3 – Paramètres des modèles du système vestibulaire.

La figure 12 présente la réponse des organes otolithiques à un stimulus de force spécifique. Ici, le profil d'entrée correspond à un créneau d'accélération linéaire de

2 m/s^2 pendant 8 secondes.

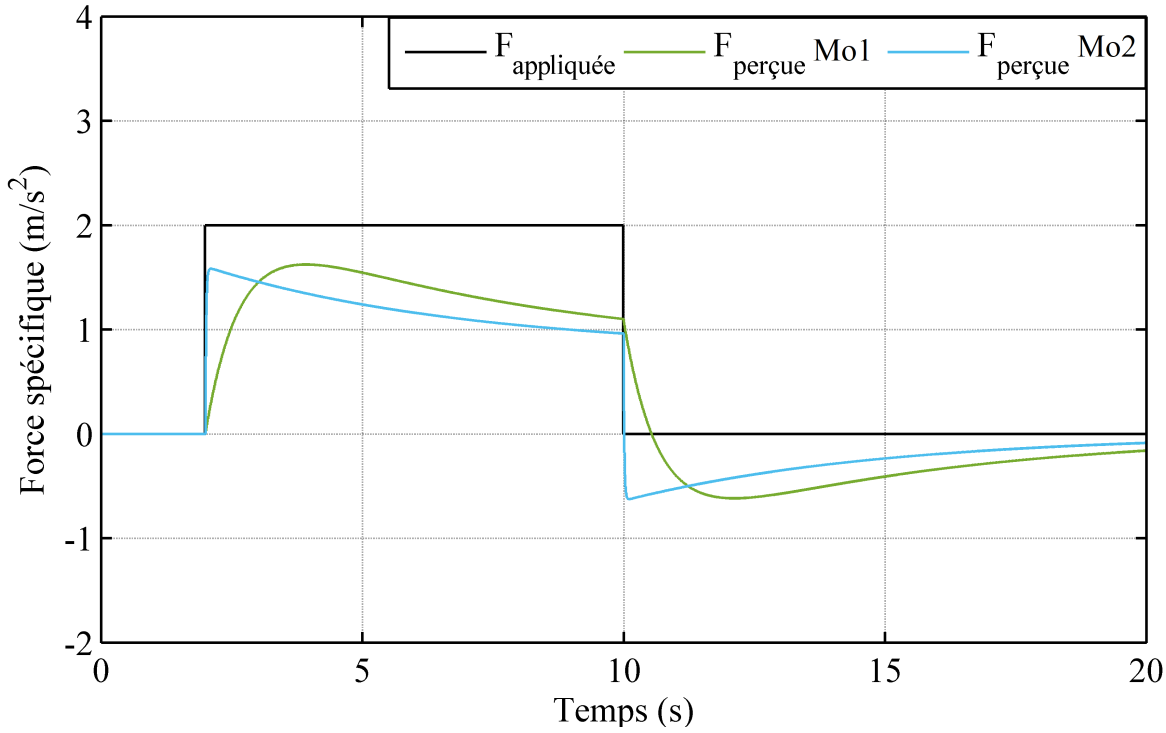


FIGURE 12 – Réponse temporelle des organes otolithiques.

Les algorithmes fondés sur le contrôle des plateformes sont configurés comme un problème de suivi, dans lequel le mouvement perçu dans le simulateur suit le mouvement du véhicule réel tout en respectant les contraintes physiques et dynamiques de la plateforme. Certes, le mouvement du simulateur dépend de la force perçue par le conducteur, d'où l'importance de choisir le bon modèle de perception. Cependant, la figure 12 nous montre qu'avec les paramètres de la fonction de transfert des organes otolithiques, les deux modèles se comportent comme des filtres d'accélération différents, ce qui rend difficile le choix du modèle le plus approprié.

Afin de représenter clairement le comportement fréquentiel de chaque modèle mathématique, nous avons tracé le diagramme d'amplitude et de phase pour chaque modèle dans la figure 13.

Le diagramme d'amplitude montre que les deux modèles ont tendance à atténuer le signal d'entrée parce que leur réponse en fréquence n'est jamais supérieure à 0 dB. Ils agissent comme des filtres passe-bas. On peut également noter que les deux systèmes ont la même pente aux fréquences supérieures à la coupure, ce qui correspond à l'ordre du filtre de la fonction de transfert (1). Toutefois, ces pentes ne sont pas superposées, car la constante de temps court τ_{soto} est différente pour chaque modèle.

En outre, nous pouvons observer que les organes otolithiques sont plus sensibles

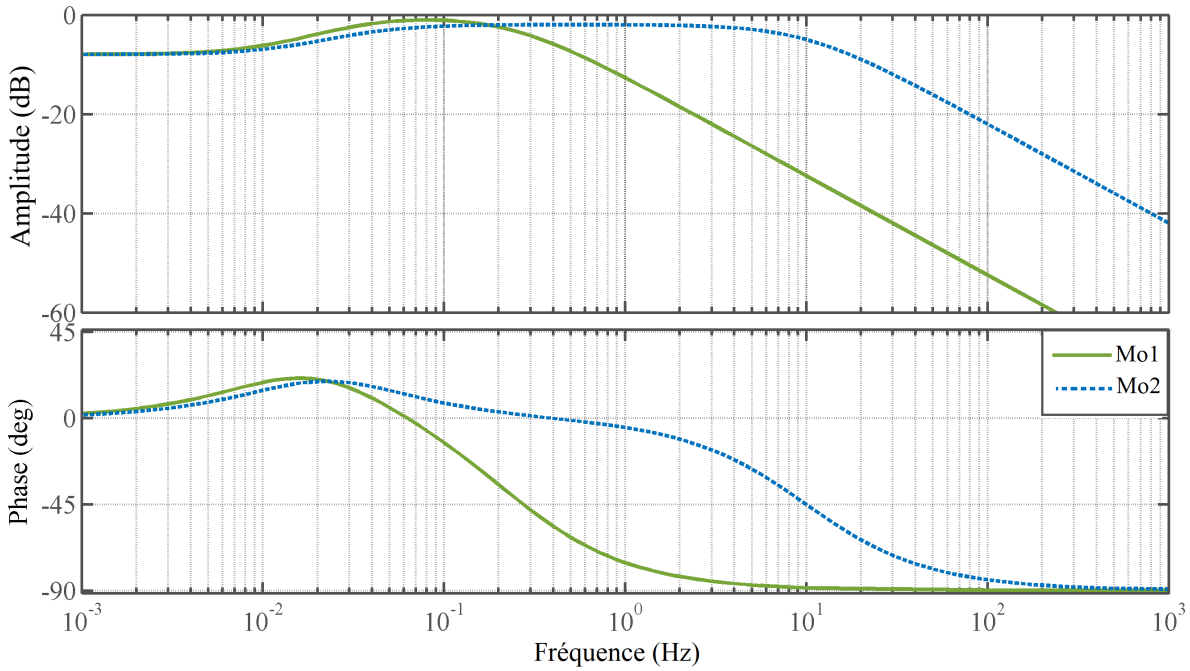


FIGURE 13 – Réponses en fréquence du modèle Mo1, proposé par [Young & Meiry, 1968] et du modèle Mo2, proposé par [Telban & Cardullo, 2005].

aux fréquences entre 0,03 et 0,25 Hz pour le modèle Mo1 et entre 0,06 et 5 Hz pour le modèle Mo2. Ainsi, le modèle Mo2 permet de ressentir des fréquences plus élevées que le modèle Mo1, celles-ci correspondent aux fréquences dans la gamme des mouvements normaux de la tête allant de 0,1 à 1 Hz [Telban & Cardullo, 2005]. Dans le chapitre 5 nous comparerons ces deux modèles du point de vue de l'utilisateur, afin de choisir le modèle du système vestibulaire le plus approprié dans la conception du contrôle de la plateforme.

D'autres paramètres à considérer dans la modélisation des MCA sont les seuils de perception des accélérations linéaires, principalement pour éviter le mal du simulateur provoqué par les *false cues* [1] ou faux indices du mouvement. Cet effet se produit lorsque le simulateur restitue une accélération de signe contraire à celle de l'accélération simulée, par exemple lorsque la plateforme freine pour respecter son espace de travail. La valeur du seuil de perception peut varier d'un chercheur à l'autre, elle est pourtant située aux alentours de $5\ cm/s^2$ [Colombet, 2010]. Dans cette thèse, nous utiliserons les valeurs du tableau 4, ces seuils étant déjà implémentés dans la conception de MCA [Reid & Nahon, 1985; Wang & Fu, 2004]. Certains auteurs proposent que pendant une stimulation visuelle combinée, c'est-à-dire, pendant la simulation, la valeur du seuil de perception d'accélération linéaire peut être modifiée par une valeur supérieure aux valeurs du tableau 4 [Reymond, 2000].

1. Perception du mouvement dans la direction opposée au déplacement en simulation.

	Axes de déplacement (m/s^2)		
	X	Y	Z
Seuils de perception linéaire	0,17	0,17	0,28

TABLEAU 4 – Valeurs des seuils de perception pour les otolithes.

2.3.6 Modèle des canaux semi-circulaires

Comme nous l'avons vu précédemment, l'accélération de la tête provoque un déplacement de la cupule, ce qui provoque une réponse sensorielle à travers le système de transduction mécano-neural. Ce déplacement produit la sensation de rotation par les canaux semi-circulaires.

Le premier modèle mathématique des canaux semi-circulaires a été défini par la dynamique d'un pendule en torsion, car la déflexion de la cupule se comporte comme un système sur-amorti de ce modèle. Malheureusement, ce modèle ne représentait pas complètement la sensation de rotation résultant des différentes expérimentations [Zacharias, 1978]. Afin de résoudre l'incohérence entre la réponse d'un pendule de torsion et la réponse mesurée dans les expérimentations, le modèle a été amélioré en intégrant un opérateur d'adaptation. De plus, un dernier terme a été ajouté pour représenter la sensibilité des cellules ciliées dans le traitement de l'information de la vitesse angulaire de la cupule. Ce modèle est analysé et caractérisé en détail dans [Telban *et al.*, 2000].

Ainsi, le modèle final représente à la fois la dynamique du canal semi-circulaire et la dynamique de transduction neurale en utilisant un pendule de torsion de deuxième ordre, un terme d'adaptation et un retard. Le système pendule-torsion provient du système cupule-endolymphe, tandis que la partie d'adaptation et le retard proviennent du système de transduction mécano-neuronal.

La fonction de transfert qui sert à lier la vitesse angulaire perçue par la cupule chez le conducteur $\hat{\omega}$ et la vitesse angulaire réelle ω est représentée par l'équation suivante :

$$\frac{\hat{\omega}}{\omega} = \frac{\tau_L \tau_a s^2 (1 + \tau_l s)}{(1 + \tau_a s)(1 + \tau_L s)(1 + \tau_s s)} \tag{2}$$

avec τ_L la constante de temps long pour le modèle de pendule de torsion, qui représente le rapport entre la déviation de la cupule et la vitesse angulaire, τ_a l'opérateur d'adaptation, τ_l la constante de délai et τ_s la constante de temps court [Zacharias, 1978].

Cette fonction de transfert est utilisée pour les trois rotations autour des axes x, y et z appelées respectivement roulis, tangage et lacet. Les paramètres sont difficiles à mesurer, car ils sont basés sur des réponses subjectives. Comme pour le modèle des organes otolithiques, nous avons choisi deux ensembles des paramètres parmi ceux existants dans la littérature. Ils correspondent au complément des modèles

Mo1 et Mo2 proposés dans la partie précédente. Le premier modèle, nommé **Mc1**, a été proposé par [Young & Oman, 1969] et le second modèle, nommé **Mc2**, a été proposé par [Telban & Cardullo, 2005].

Les valeurs de chaque composante de temps sont obtenues d'après une étude de la sensation en réponse à la rotation verticale de la Terre autour des trois axes du mouvement. Le propos était d'analyser la capacité des sujets à détecter et estimer l'amplitude de leur accélération [Dagdelen, 2005]. Les valeurs des constantes de temps sont indiquées dans le tableau 5 :

Modèle	**Paramètres** (s)			
	τ_L	τ_s	τ_l	τ_a
Mc1	X : 6,1 Y : 5,3 Z : 10,2	0,1	0	30
Mc2	5,73	0,005	0,06	80

TABLEAU 5 – Paramètres du modèle de canaux semi-circulaires.

Pour implémenter ces modèles dans les algorithmes de restitution de mouvement, certains auteurs négligent la constante de temps court, car elle peut causer une instabilité dans le système. De plus, la constante de délai est également négligée [Telban *et al.*, 2000]. En effet, la constante de temps court et la constante de délai τ_l influencent le système à des fréquences supérieures à 2,2 Hz [Telban & Cardullo, 2005]. Or, comme la gamme de mouvement de la tête est comprise entre 0,1 et 1 Hz, l'élimination de ces constantes ne semble pas être problématique pour la perception des vitesses angulaires.

Le diagramme de Bode de la figure 14 montre le comportement de ces deux modèles.

Le diagramme montre que les deux filtres ont un comportement similaire à celui d'un filtre passe-haut. La gamme de fréquences pour lesquelles le système des canaux semi-circulaires fonctionne comme un transducteur de vitesse angulaire est entre 0,2 et 0,5 Hz pour le modèle Mc1 et à partir de 0,2Hz pour le modèle Mc2. Cela indique que Mc2 couvre toutes les fréquences dans la plage des mouvements de la tête. Comme pour les organes otolithiques, les modèles seront évalués dans le chapitre 5.

Pour certaines valeurs, les canaux semi-circulaires ne peuvent pas détecter les accélérations angulaires. Ils ont été beaucoup étudiés pour leur utilisation dans les algorithmes de contrôle de mouvement en tant que seuils de perception [Colombet *et al.*, 2017]. Cependant, dans les MCA, le seuil qui possède le rôle principal est celui de la vitesse angulaire ω, car il est particulièrement utile lorsque l'on applique la technique de *tilt coordination* qui sera détaillée dans la partie 2.3.8. Selon les axes de rotation, la valeur du seuil de ω est différent : 2,5 deg/s en roulis, 3,6 deg/s en tangage et 4,2 deg/s en lacet. Cependant, l'étude réalisée par [Telban & Cardullo,

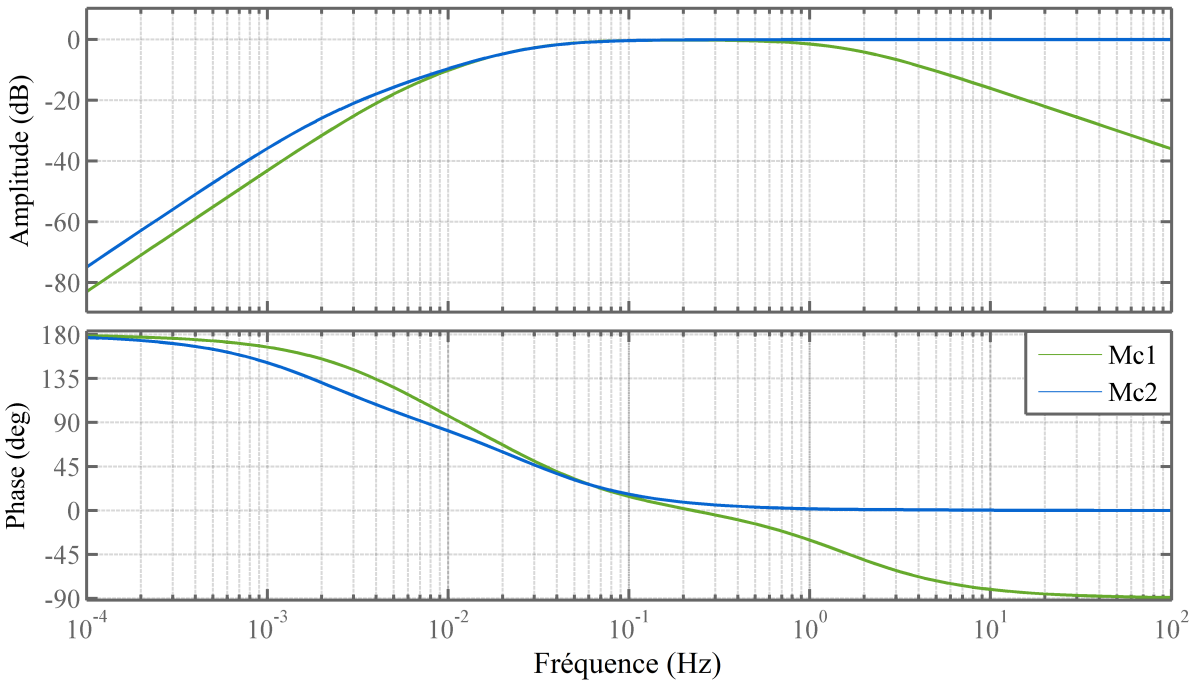


FIGURE 14 – Réponses en fréquence du modèle des canaux semi-circulaires Mc1 et Mc2.

2005] propose un seuil de perception différent pour le roulis (3 deg/s) et le lacet (2,6 deg/s). Dans la plupart des MCA, la valeur du seuil de perception peut varier entre 2-4 deg/s pour les trois axes de mouvement [Asadi *et al.*, 2015a; Stratulat *et al.*, 2011].

Pour simuler la réponse de chaque modèle mathématique, nous définissons la fonction de transfert dans sa forme canonique :

$$\frac{\hat{\omega}}{\omega} = \frac{G_{\text{csc}}(T_1 s^3 + T_2 s^2)}{s^3 + T_3 s^2 + T_4 s + T_5} \tag{3}$$

avec : $T_1 = \frac{\tau_l}{\tau_s}$; $T_2 = \frac{1}{\tau_s}$; $T_3 = \frac{\tau_L\tau_s + \tau_a\tau_s + \tau_a\tau_L}{\tau_a\tau_L\tau_s}$; $T_4 = \frac{\tau_s + \tau_a + \tau_L}{\tau_a\tau_L\tau_s}$; $T_5 = \frac{1}{\tau_a\tau_L\tau_s}$

2.3.7 Modèle du système vestibulaire

Le modèle du système vestibulaire est le couplage des modèles mathématiques des organes otolithiques et des canaux semi-circulaires. Ce modèle a été implémenté pour la première fois dans l'algorithme de contrôle optimal [Sullivan, 1985]. Dans celui-ci, les signaux d'entrée sont les accélérations linéaires et angulaires, traitées respectivement par les organes otolithiques et les canaux semi-circulaires. Plusieurs auteurs ont proposé d'unifier ces deux modèles en un seul système d'état [Augusto, 2009]. Cependant, pour analyser la contribution de chacun à la perception des accélérations linéaires et angulaires, nous prendrons indépendamment chaque système.

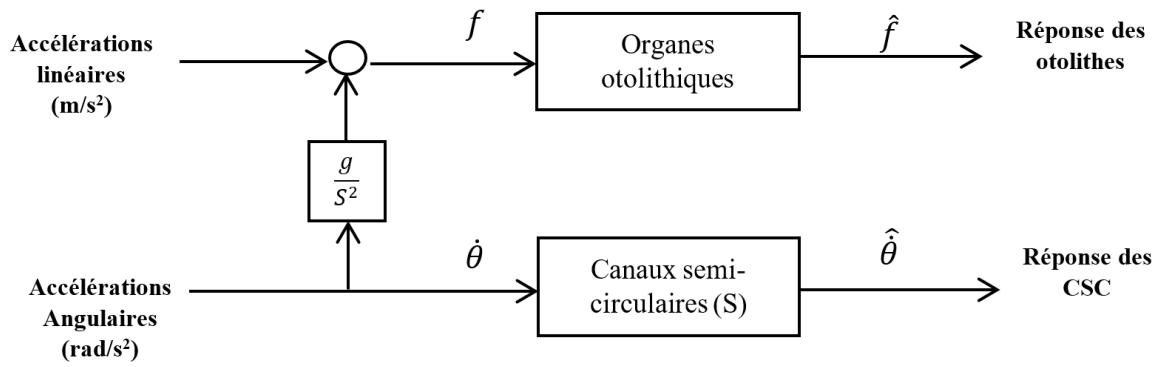


FIGURE 15 – Modèle du système vestibulaire soumis à un mouvement linéaire et rotationnel.

Comme nous pouvons l'observer dans le schéma de la figure 15, il est nécessaire de considérer la composante du vecteur gravité pour intégrer les deux sous-systèmes dans la perception de l'accélération linéaire. En effet, les organes otolithiques ne sont pas capables de discriminer par eux-mêmes les forces gravitationnelles de celles longitudinales, autrement dit, l'inclinaison de la tête de l'accélération linéaire. Ainsi, la composante de gravité basée sur l'angle de roulis ou de tangage fournit au conducteur une sensation d'accélération soutenue selon l'axe x ou selon l'axe y. Cette ambigüité sensorielle est limitée par un seuil de perception que nous verrons par la suite.

2.3.8 Technique de *tilt coordination*

« *Tilt coordination* » est le nom utilisé par les concepteurs de MCA pour désigner la restitution inertielle supplémentaire à partir des axes d'inclinaison de la plate-forme, pour fournir une accélération linéaire continue (soutenue). Ainsi, l'ambigüité des organes otolithiques est essentielle dans la modélisation de MCA, car grâce à cela, l'on peut reproduire le comportement basse fréquence des accélérations longitudinales. De plus, les limites de l'espace de travail des plateformes empêchent la restitution des accélérations continues si cette technique d'inclinaison n'est pas prise en compte. La figure 16 montre ce comportement en vue latérale, dans laquelle le conducteur est incliné vers l'arrière, grâce à la composante du vecteur gravité (en tangage). a_x et a_z représentent l'accélération selon les axes x et z, θ l'angle d'inclinaison en tangage et g la gravité, 9,81 m/s^2.

La somme des efforts appliquée sur l'axe x est donnée par l'équation 4.

$$f_x = a_x cos(\theta) + gsin(\theta) \tag{4}$$

Cette équation représente l'amplitude de la force spécifique f_x selon l'axe de déplacement longitudinal, formée par la composante du vecteur accélération $a_x cos\theta$ et

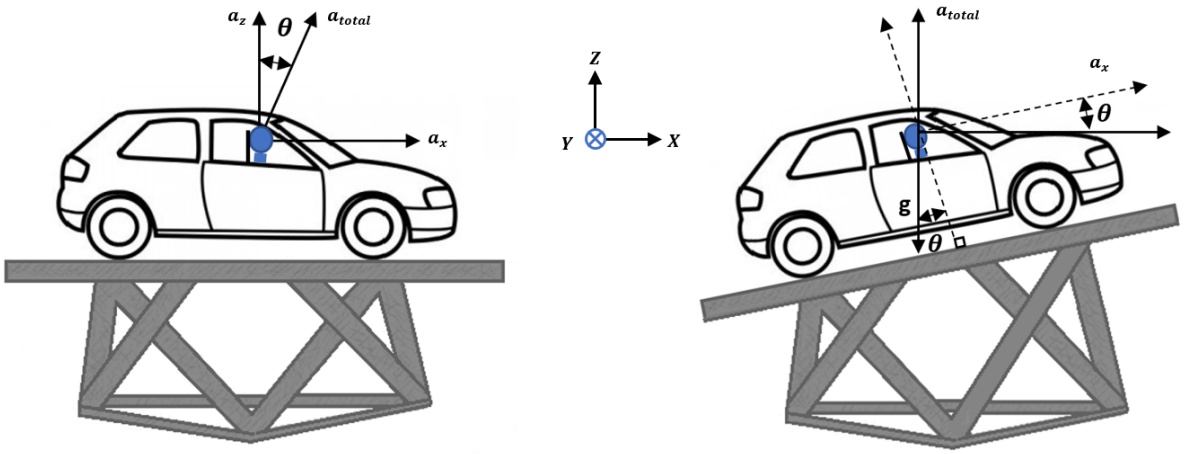


FIGURE 16 – Technique de tilt-coordination : accélération selon l'axe x.

la composante vectorielle de gravité $gsin\theta$ selon le même axe de mouvement. Dans cette condition, le conducteur du simulateur percevra une accélération d'amplitude $g = 9,81m/s^2$ selon l'axe vertical et une accélération $g\theta m/s^2$ selon l'axe horizontal qui intervient sur la somme des forces spécifiques perçues.

Toutefois, les accélérations continues simulées par le *tilt coordination* ne doivent pas dépasser $0,5m/s^2$. Le cas échéant, il peut se produire l'effet Aubert [1], une distorsion de la verticale du conducteur qui se produit lors de l'utilisation d'angles d'inclinaison supérieurs à $20° - 30°$ [Weib, 2006]. En outre, une étude réalisée par [Scotto Di Cesare *et al.*, 2014] montre qu'une inclinaison de la plateforme supérieure à $8°$ peut être perçue par les participants. En conséquence, les angles utilisés pour cette technique sont de faibles orientations, $\theta \ll 1\ rad/s$, $cos\theta = 1$ et $sin\theta = \theta$.

En remplaçant ces valeurs dans l'équation 4, nous obtenons l'équation suivante dans le domaine de Laplace, en termes d'accélération linéaire et rotationelle

$$f_x(s) = a_x(s) + \frac{g}{s^2}\ddot{\theta}(s) \tag{5}$$

Certains seuils de perception doivent être respectés afin de pouvoir produire la technique correctement. Dans la littérature on trouve que la vitesse de rotation doit être inférieure à $3°/s$ [Telban & Cardullo, 2005]. D'autres auteurs proposent des seuils plus élevés si la rotation est accompagnée d'un environnement visuel cohérent. Par exemple, une étude élaborée par [Fang & Kemeny, 2014], suggèrée que si pendant pour une stimulation en roulis, l'angle est de $5°$, la vitesse de $5°/s$ et une accélération de $8°/s^2$, les conducteurs acceptent la restitution du mouvement.

Le système postural ne permet pas de distinguer entre l'inclinaison et l'accélération linéaire du corps si l'on ne considère que les organes otolithiques. C'est pourquoi la synchronisation entre le système visuel et le système vestibulaire doit être faite de

1. Effet Aubert : déviation subjective, dans l'obscurité et si l'on est incliné, d'une ligne lumineuse verticale, dans le sens inverse de l'inclinaison. Repéré sur `http://definitions-de-psychologie.psyblogs.net/2017/01/effet-aubert.html`

manière à ne pas interférer dans la mise en œuvre de la technique *tilt coordination*. Il est bien établi qu'une action coordonnée du système visuel donnant une référence par rapport aux repères horizontaux et verticaux est fondamentale pour obtenir une performance satisfaisante de cette technique. Sans casque de visualisation stéréoscopique, l'écran doit être fixe et s'incliner par rapport à la plateforme en assurant la stabilité de l'horizon perçu lors des mouvements à accélérations non nulles [Kemeny, 1999]. Si l'écran n'est pas fixe, l'environnement virtuel doit être modifié afin de ne pas introduire de malaises lors des mouvements de la tête dûs au réflexe visio-vestibulaire [1]. Il y existe deux types de réflexes différents : réflexe de rotation qui compense le mouvement de rotation et l'inclinaison de la tête dans le plan vertical, et le réflexe de translation qui compense le mouvement linéaire. Le premier réflexe est le mouvement des yeux dans la direction opposée à la rotation de la tête perçue par les canaux semi-circulaires afin de minimiser la direction du mouvement de la rétine. Pour compenser le mouvement linéaire, le système vestibulaire doit tenir compte de la distance des objets, sachant que plus l'objet est éloigné, plus le mouvement de l'œil est petit [Kandel *et al.*, 2000].

Ainsi, si les indices visuels et vestibulaires ne détectent pas une inclinaison de la plateforme, la perception visio-vestibulaire donne l'illusion d'une accélération linéaire dans l'environnement virtuel.

2.4 L'intégration sensorielle

En simulation en conduite et plus particulièrement dans la mise en œuvre de stratégies de contrôle du mouvement, il est important de considérer les stimuli artificiels, les illusions sensorielles et les conflits sensoriels [Mohellebi, 2005]. Le premier stimulus renforce les indices de perception perçus par les autres sens, c'est le cas de l'intégration de vérins dans les simulateurs afin de produire des vibrations. Celles-ci sont traitées par le système somatique qui en retour, perçoit un faux déplacement sur la route. Les illusions sensorielles, comme la technique du *tilt coordination*, profitent des ambiguïtés sensorielles afin de compenser les limites de la plateforme, certes, en trompant les sens dans l'environnement virtuel. Le dernier est traité pour éviter les situations qui provoquent le mal du simulateur au conducteur. Pour aborder ces thèmes, il faut d'abord comprendre les interactions de tous les sens impliqués dans la perception du mouvement.

Comme nous l'avons vu précédemment, la perception du mouvement est le résultat de tous les signaux provenant des différents organes sensoriels, dont le système visuel, le système vestibulaire et la proprioception. Dans le cadre de cette thèse, nous nous concentrons sur les stimulations produites par le mouvement de la plateforme, qui affectent principalement la perception du système vestibulaire et visuel. La proprioception n'est pas analysée ici, car elle concerne les mouvements provenant du siège, du volant ou de la cabine du simulateur, autrement dit, les

1. Mouvement des yeux influencé par une accélération perçue par le système vestibulaire.

mouvements actifs du corps. De plus, en l'absence de signaux visuels un conducteur utilise davantage les signaux vestibulaires que les signaux proprioceptifs pour déterminer son orientation spatiale [Sullivan, 1985].

2.4.1 Couplage visuo-vestibulaire

Les systèmes vestibulaire et visuel ont des caractéristiques très différentes, d'une part ils n'utilisent pas les mêmes systèmes de coordonnées. Le système vestibulaire transfère l'information d'accélération mesurée dans les trois dimensions du mouvement pour compenser les réflexes visuels qui ont été pris en deux dimensions. D'autre part, les systèmes vestibulaire et visuel réagissent différemment aux stimuli de mouvement présents dans l'environnement. Par exemple, les repères visuels de mouvement sont dominants dans la perception de la vitesse et de l'orientation en régime permanent aux fréquences inférieures à 0,1 Hz, voire après 10 secondes de stimulation [Telban *et al.*, 1999], alors que la perception du système vestibulaire est faible dans ces fréquences ainsi que pour une vitesse constante puisqu'il ne détecte que les accélérations [Priot, 2012].

Séparément, aucun n'est capable de fournir des informations complètes sur notre environnement. Pendant la vection, par exemple, les noyaux vestibulaires sont activés en suivant la même évolution temporelle que le mouvement propre. Cela permet de constater que la vection est un complément de l'information vestibulaire. Elle apparaît avec une latence qui complète la fréquence du système vestibulaire [Colombet, 2010]. Cela révèle la nécessité de comprendre le comportement et l'influence de ces systèmes sensoriels dans le contrôle des simulateurs dynamiques. Ainsi, avec cette intégration sensorielle, nous pouvons mieux comprendre les différents messages sensoriels et avoir une interprétation unique du mouvement.

Bien que ces deux systèmes se complètent l'un à l'autre dans la perception de fréquences différentes, l'intégration sensorielle est perturbée par les décalages en amplitude ou en latence entre ces deux systèmes. Plusieurs expérimentations ont été menées au fil des années pour définir l'impact de ces décalages dans la restitution du mouvement (voir [Mohellebi, 2005] pour un état de l'art exhaustif). Elles indiquent que le décalage temporel entre le système visuel et vestibulaire provoque davantage de nausées que les décalages en amplitude car les sujets comprennent la source de l'incohérence. En effet, le retard entre la restitution visuelle et la restitution vestibulaire produisent des conflits sensoriels qui peuvent produire des symptômes de mal du simulateur lors de la conduite en simulateur. Ces décalages temporelles se présentent principalement sous la forme d'une dégradation de la vection dans le choix d'un des indices inertiels prédominants pour un stimulus donné. Cela se produit à la fréquence de croisement des filtres qui représentent le système visuel (voir figure 8) et le système vestibulaire (voir figure 15), c'est à dire, la fréquence à laquelle la réponse globale des systèmes d'auto-mouvement visuel et vestibulaire est maximale [Duh *et al.*, 2004].

La présence d'une latence entre le système visuel et le système vestibulaire est fortement déconseillée en simulation en conduite [Kemeny, 1999; Reymond, 2000]. Malheureusement, plusieurs facteurs tels que les éléments mécaniques, le temps d'affichage de la scène visuelle, la dynamique de la plateforme et le modèle de véhicule peuvent provoquer cette latence. Le réglage de ces facteurs est une solution qui peut réduire les conflits visuovestibulaires, mais qui n'entre pas dans le cadre de cette thèse. Nous allons plutôt améliorer les techniques de restitution de mouvement qui donnent une perception plus réaliste et douce des signaux provenant de son environnement afin de prévenir ce conflit sensoriel et le mal du simulateur. C'est pourquoi nous allons intégrer les indices inertiels perçus par le système visuel et par le système vestibulaire sous la forme d'un modèle mathématique complet.

2.4.2 Modèle de perception complet

Il existe dans la littérature différents modèles visuovestibulaires qui reproduisent la réponse perçue du mouvement propre par le couplage entre ces deux sens. Une étude détaillée de l'état de l'art est présentée dans [Hosman *et al.*, 2011]. Parmi les modèles, nous utilisons celui de [Telban & Cardullo, 2005] (figure 17), qui a développé un modèle fonctionnel dont les repères visuels et vestibulaires fonctionnent conjointement pour produire la perception du mouvement humain.

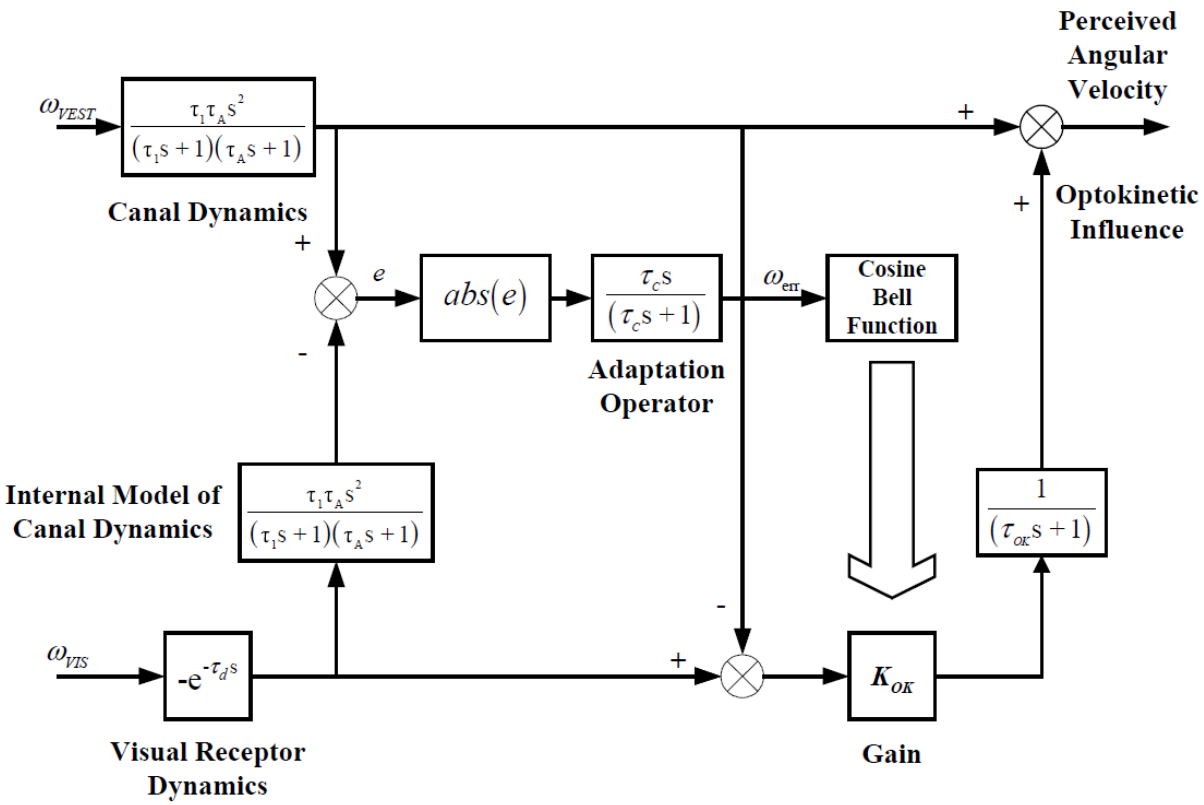


FIGURE 17 – Modèle d'interaction vestibulaire et visuelle proposé par [Telban & Cardullo, 2005].

Ce modèle reçoit en entrée les vitesses angulaires du simulateur, qui sont filtrés par la fonction de transfert des canaux semi-circulaires et la fonction de transfert des récepteurs visuels, afin de produire une estimation perceptive du mouvement à partir de la scène visuelle. Les constantes de temps conservent les mêmes valeurs que dans les parties précédentes. Le seul paramètre différent qui interfère avec la perception du mouvement est le gain K_{OK}. En effet, différentes études ont montré que lorsque les signaux visuels et vestibulaires sont présentés simultanément, une somme linéaire simple ne permet pas de prédire la réponse sensorielle des deux systèmes [Zacharias & Young, 1981]. C'est pourquoi les auteurs ont intégré un opérateur d'adaptation qui agit sur le signal redressé, afin de reproduire le signal d'erreur ω_{err}. Cet opérateur dépend de la constante $\tau_c = 0,2$, choisie afin de respecter les valeurs de latence trouvées dans la littérature [Telban & Cardullo, 2005]. Ensuite, le signal est filtré par un fenêtrage qui définit le gain K_{OK} : un ω_{err} important ramène le gain à zéro, tandis qu'un petit signal le ramène à une valeur de pondération plus importante, jusqu'à l'unité.

La figure 18 montre la réponse du système visuel, du système vestibulaire et du couplage de ces deux systèmes à une vitesse constante de $4°/s$.

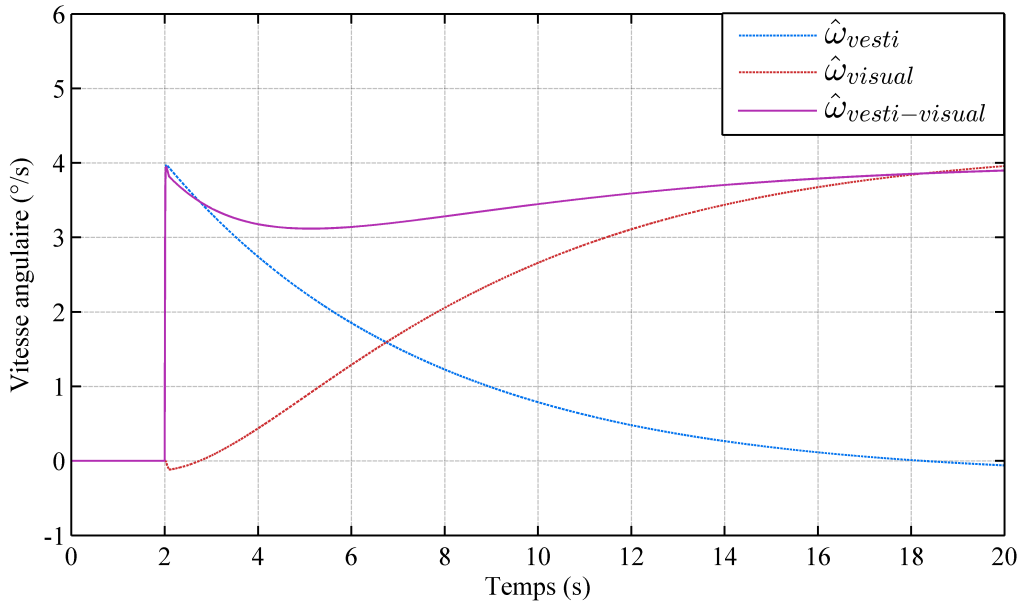


FIGURE 18 – Différence entre la perception du mouvement par le système vestibulaire, visuelle et l'ensemble de ces derniers avec un stimulus constant en vitesse de $4°/s$.

Comme nous pouvons l'observer sur la figure, la contribution des signaux visuels est importante pour les basses fréquences, c'est-à-dire, lors qu'il y a des accélérations continues. Cependant, en raison des limites de perception présentées dans la partie 2.3.8, les indices fournis par l'intégration du modèle de perception du système visuel ne pourraient pas être reproduits sur la plateforme même si la technique de *tilt coordination* est utilisée. Pour les fréquences plus élevées, le système vestibulaire a tendance à dominer la perception de mouvement. Ces fréquences correspondent

aux mouvements normaux de la tête (entre 0,1 et 1 Hz), c'est-à-dire que lorsqu'un mouvement de déplacement est reproduit dans un environnement simulé, une prédominance de la perception vestibulaire apparaît. C'est pourquoi, la perception du mouvement fournie par le système visuel est primordiale lorsque les indices vestibulaires sont éliminés en raison des limites de la plateforme. Ces indices visuels sont fournis uniquement par les écrans à haute résolution des simulateurs de conduite.

Cela nous incite à n'utiliser que le système vestibulaire comme modèle de perception dans le MCA afin de reproduire le mouvement de la plateforme. En effet, lorsque la dérivée de l'accélération (*jerk*) du véhicule varie, c'est le système vestibulaire qui apporte les informations au cerveau pour la perception du mouvement. Les indices visuels, auditifs et haptiques confirment ces informations aux conducteurs, mais ne sont pas intégrés directement en tant que modèles dans les algorithmes de restitution de mouvement.

2.5 Synthèse du chapitre

Dans ce chapitre, nous avons d'abord étudié la perception du mouvement de l'humain et l'importance en simulation de conduite. Les contributions du système visuel, vestibulaire et somatique ont été analysées afin d'identifier les indices nécessaires à leur intégration dans les algorithmes de restitution de mouvement. Cette analyse nous a permis de valider l'intérêt de la mise en place d'un modèle de perception dans la boucle de contrôle du MCA.

L'état de l'art nous a permis de constater la variété des modèles existants et les différents paramètres qui peuvent varier d'un modèle à l'autre. En particulier, nous avons constaté que les paramètres ont un impact important dans la réponse du système vestibulaire puisque ceux-ci sont obtenus par des tests subjectifs. Ainsi, nous constatons la nécessité de faire un test expérimental afin de choisir le modèle qui s'adapte le mieux aux algorithmes de restitution de mouvement pour les simulateurs dynamiques, car la seule analyse fréquentielle ne nous l'en permet pas. Cette étude sera détaillée dans le chapitre 5, car il est d'abord nécessaire d'introduire le MCA auquel le modèle sera intégré. Après avoir présenté les modèles, nous avons montré la technique du *tilt coordination* qui sera utilisée dans les MCA développés dans cette thèse.

La suite du chapitre a été consacrée à l'intégration multisensorielle des entrées visuelles et vestibulaires afin de surmonter les limites de chaque système sensoriel et d'augmenter la sensibilité lorsque les deux systèmes sont actifs. Cependant, le système visuel fournit une perception de mouvement aux basses fréquences difficilement atteignable dans des conditions de conduite normales ou à cause des limites de perception. C'est pourquoi, ce modèle n'est pas retenu pour la mise en œuvre des MCA présents dans la thèse, seul le modèle du système vestibulaire est considéré dans le contrôle.

Chapitre 3

Stratégies de contrôle de plateforme

Le problème principal des plateformes mobiles est de restituer un mouvement adéquat pour permettre une excellente perception du mouvement au conducteur qui, dans une situation idéale, ressentira complètement la dynamique du véhicule au cours d'une simulation. Il existe plusieurs raisons pour lesquelles un système de mouvement est important dans la simulation. Il a une influence sur la performance du conducteur et réduit le mal du simulateur ; de plus, les repères d'accélération ajoutent du réalisme à la simulation [Slob, 2008].

Comme nous l'avons vu dans le chapitre précédent, l'espace de travail limité de la plateforme empêche l'envoi direct des accélérations du cockpit au simulateur. Raison pour laquelle sont implémentées les différentes stratégies de restitution de mouvement (MCA). Leur objectif est de reproduire au mieux les signaux des accélérations provenant du modèle du véhicule simulé, tout en gardant la plateforme dans ses limites dynamiques et physiques. Autrement dit, les MCA sont utilisés pour transformer les accélérations des véhicules en signaux de mouvement admissibles pour le simulateur et ainsi, donner au conducteur une bonne immersion en toute sécurité. Cette immersion est nécessaire pour renforcer la fiabilité des tests effectués pendant la simulation, tels que les tests de validation ADAS ou véhicule autonome ou sur le comportement de conduite sur routes. Le schéma 19 illustre comment les MCA sont intégrés dans la boucle de contrôle d'un simulateur de conduite.

Les MCA ont été conçus il y a une quarantaine d'années pour le contrôle de plateformes dynamiques. Initialement, ils étaient dédiés principalement aux simulateurs aéronautiques, puis, ils ont intégré les simulateurs automobiles. Même si la problématique du contrôle est similaire, les lois de contrôle ont dû être adaptées conformément aux caractéristiques dynamiques d'un simulateur de conduite automobile, puisque dans ces simulateurs, les accélérations hautes fréquences sont plus importantes [Nehaoua *et al.*, 2006]. Ainsi, pour reproduire au plus proche possible le mouvement du véhicule dans le simulateur, les MCA doivent traiter les

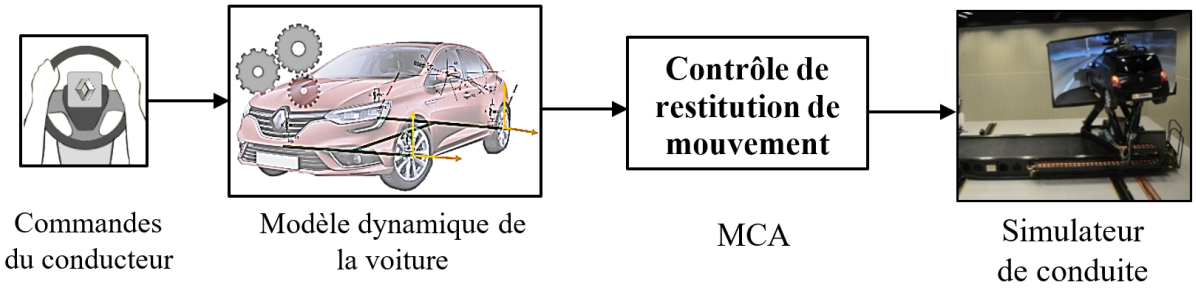


FIGURE 19 – Intégration des stratégies de mouvement dans le contrôle de simulateurs dynamiques.

signaux produits par le conducteur, gérer les contraintes physiques et dynamiques du simulateur et respecter les seuils de perception du conducteur [Cardullo *et al.*, 1999].

D'après la littérature, plusieurs MCA sont utilisés aujourd'hui dans les simulateurs de conduite. Toutefois, chacun d'entre eux ont des limites qui l'empêchent d'être reconnue comme étant la meilleure loi de contrôle pour les simulateurs à hautes performances et plus précisément pour la validation des véhicules autonomes. En effet, nous montrerons dans ce chapitre, les avantages et les enjeux des MCA existants tels que le classique, l'adaptatif, l'optimal et la commande par modèle prédictif. Afin d'analyser les comportements de toutes les stratégies de contrôle, nous avons généré un scénario avec le logiciel de simulation de conduite SCANeR Studio [1]. Le scénario consiste en un simple démarrage suivi d'une accélération continue pendant 15 secondes, puis d'un arrêt complet du véhicule.

Enfin, nous ferons une comparaison afin déterminer la loi de contrôle que nous considérons comme étant la plus appropriée pour la nouvelle génération de simulateurs à hautes performances et pour les prestations de validations des nouvelles technologies d'aide à la conduite.

3.1 Commande classique

Parmi les algorithmes de restitution de mouvement, l'algorithme classique est le premier et le plus utilisé [Conrad & Schmidt, 1970; Nahon, Meyer A. & Reid, Llotd D., 1990]. Cet algorithme filtre les signaux envoyés par le conducteur afin que le simulateur ne dépasse pas ses limites physiques. La commande classique est représentée en boucle ouverte comme le montre la figure 20.

Dans ce schéma, le premier bloc représente la mise à l'échelle des vecteurs d'entrée dont les signaux sont multipliés par une valeur inférieure à l'unité afin de respec-

1. https://www.avsimulation.fr/

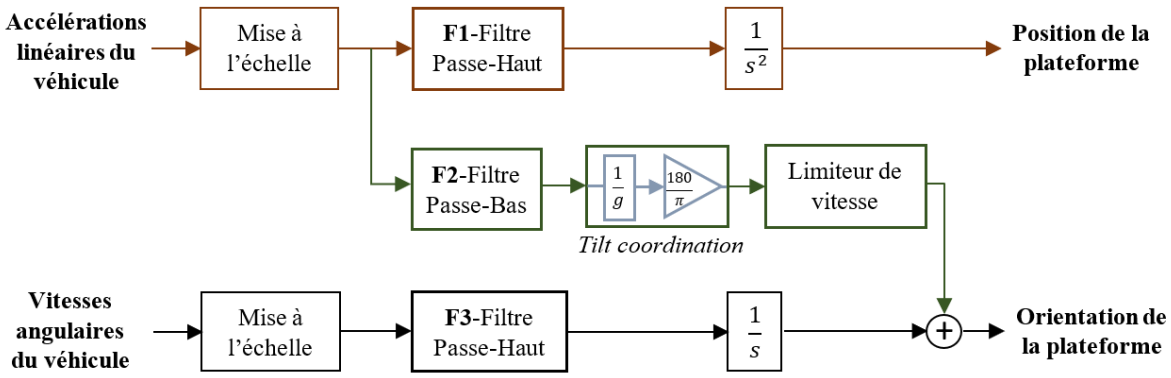


FIGURE 20 – Structure en boucle ouverte du contrôle classique.

ter les contraintes du système [Gutridge, 2004]. Certains auteurs utilisent une matrice de transformation avec des angles d'Euler afin de transformer le système de coordonnées basé sur le corps du conducteur en coordonnées inertielles de la plateforme [Friedland *et al.*, 1973; Nahon *et al.*, 1992]. Contrairement à ces auteurs, nous n'utiliserons pas une matrice de transformation dans le déploiement de la commande classique, car dans notre cas, les accélérations sont restituées au niveau de la tête du conducteur et toutes les stratégies présentées dans ce document seront comparées à partir de ce point.

Pour chaque axe de mouvement, cet algorithme est divisé selon les signaux d'entrée en deux parties principales. La première partie pour les accélérations de translation et la seconde partie pour les vitesses angulaires comme illustré dans la figure 20.

Les accélérations de translation se composent d'accélérations transitoires ou à haute fréquence et d'accélérations continues ou à basse fréquence. Pour les accélérations transitoires, un filtre passe-haut est implémenté afin d'atténuer les basses fréquences en évitant que le simulateur dépasse ses limites physiques [Gutridge, 2004]. Ce filtre est suivi d'un intégrateur double afin d'obtenir la position du simulateur. Pour obtenir les accélérations à basse fréquence, nous appliquons un filtre passe-bas. En l'absence de mouvement de rotation, celles-ci sont perçues par l'humain comme un changement lent de l'amplitude et de la direction de la force spécifique.

Malgré les limitations de la plateforme, la force spécifique ne peut pas être simulée par un mouvement de translation, mais par la technique de *tilt coordination* (voir partie 2.3.8). Par exemple, pour atteindre une accélération de $3\ m/s^2$ pendant 5 secondes, un déplacement d'environ 37 mètres est nécessaire dans la direction longitudinale [Dagdelen *et al.*, 2006]. Cette technique permet donc de restituer davantage d'accélération continue lorsque l'espace de travail est insuffisant. L'inclinaison de la plateforme jusqu'à un angle de $30°$ peut générer, par exemple, une perception de force jusqu'à 0,5g [Stahl, Konrad *et al.*, 2015]. Ce mouvement angulaire doit être fait lentement pour qu'il soit imperceptible pour le conducteur.

Dans la seconde partie de cet algorithme, l'entrée du système correspond aux vitesses angulaires. Celles-ci sont filtrées et intégrées une fois pour obtenir leurs angles respectifs. Puis, on additionne à ceux-ci les angles obtenus par la technique de *tilt coordination*, et ensuite, le résultat est envoyé au simulateur.

3.1.1 Influence des filtres sur le comportement du système

Dans cette partie, nous nous focaliserons sur la mise en œuvre et la paramétrisation des filtres passe-haut et passe-bas de l'algorithme classique. L'ordre et les paramètres de ces filtres ne varient pas en temps réel et sont généralement sélectionnés en fonction des caractéristiques dynamiques de la plateforme et du scénario à évaluer. Si le scénario et les signaux d'accélération en entrée sont inconnus, les paramètres des filtres sont choisis pour répondre au pire des cas de conduite, dans le but de respecter les limites de l'espace de travail. Pour le simulateur ULTIMATE, le pire des cas correspond à un échelon de $5\ m/s^2$ [Dagdelen *et al.*, 2009].

Le filtre passe-haut pour les accélérations de translation **F1** est représenté par la fonction de transfert suivante :

$$\underbrace{\frac{k_1 s^2}{(s^2 + 2\zeta_{n1}\omega_{n1}s + {\omega_{n1}}^2)}}_{F1_a}\underbrace{\frac{s}{(s + \omega_{b1})}}_{F1_b} \tag{6}$$

Ce filtre du troisième ordre est composé d'un filtre du deuxième ordre $F1_a$ et d'un filtre du premier ordre $F1_b$. Dans $F1_a$ nous trouvons le gain du filtre k_1, la fréquence propre ou naturelle ω_{n1} et le coefficient d'amortissement ζ_{n1}. Si nous considérons uniquement ce filtre pour les accélérations de translation, la position du simulateur quand $t \to \infty$ sera $\frac{kAcc_{veh}}{{\omega_{n1}}^2}$. Cette valeur doit dans tout le cas être inférieure à la limite de déplacement du simulateur.

$$\frac{k_1 Acc_{veh}}{{\omega_{n1}}^2} \leq Position_{max} \tag{7}$$

Ainsi, les paramètres k_1 et ω_{n1} permettent un compromis entre la fidélité du mouvement et le respect des limites physiques de la plateforme selon les axes x, y et z. La valeur ζ_{n1} doit être supérieure à l'unité pour que le système soit sur-amorti et ne dépasse pas les limites de la plateforme. Afin de ne pas fournir des faux indices de mouvement au conducteur, la valeur ζ_{n1} définit également le seuil d'accélération lorsque celle-ci est dans la direction opposée à celle de la simulation (voir partie 2.3.5). Cette accélération ne peut dépasser 0,2 m/s^2 selon l'axe x.

Dans certaines situations, il est nécessaire de faire revenir la plateforme à sa position de départ, c'est là que le filtre $F1_b$ est intégré. La valeur de la fréquence ω_{b1} vient définir la vitesse de ce déplacement communément appelée *washout* [Reid

& Nahon, 1985]. L'effet de ω_{b1} est détaillé dans [Nehaoua, 2008]. Le retour de la plateforme à la position neutre a été prouvé par le théorème de la valeur finale dans [Stroosma,Olaf *et al.*, 2013]. Cependant, le filtre de *washout* peut ne pas être utilisé lorsque les situations de conduite à simuler sont connues [Fischer *et al.*, 2011].

La figure 21 présente l'accélération du véhicule acc_{veh} simulée dans le scénario présenté en introduction de ce chapitre. Nous montrons également les profils d'accélération et de position après filtrage de acc_{veh} par le filtre du deuxième et troisième ordre de $F1$.

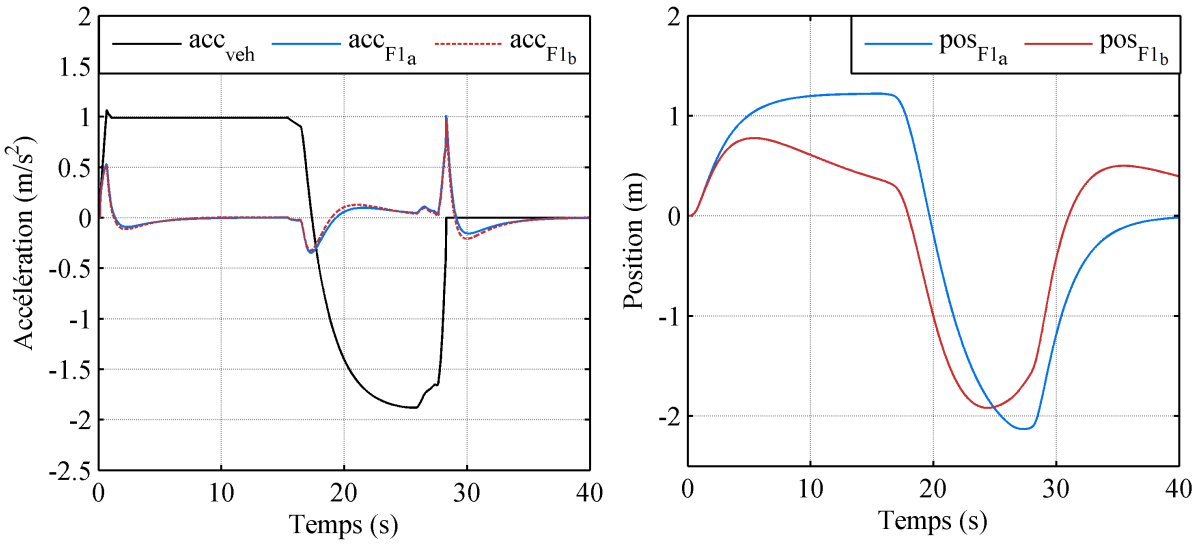


FIGURE 21 – Accélération (à gauche) et déplacement (à droite) du simulateur en utilisant le filtre $F1_a$ en bleu et le filtre $F1_b$ en rouge.

Nous remarquons que le rendu de l'accélération ne varie pas de manière significative avec le degré du filtre $F1$. En revanche, le déplacement du simulateur diminue considérablement avec le filtre $F1_b$, afin de repositionner la plateforme dans la position de départ (x=0, y=0, z=0).

Le filtre $F2$ pour le *tilt coordination* est représenté par la fonction de transfert suivante :

$$\frac{k_2\omega_{n2}^2}{(s^2 + 2\zeta_{n2}\omega_{n2}s + \omega_{n2}{}^2)} \tag{8}$$

Avec, ω_{n2} la pulsation propre, k_2 le gain statique et ζ_{n2} le coefficient d'amortissement. Ce filtre permet de reproduire une inclinaison correspondant à une accélération d'amplitude $g * \theta$ sur le plan horizontal. Les paramètres limitent les angles d'inclinaison par rapport aux conditions dynamiques de la plateforme (voir partie 2.3.8 pour une explication plus détaillée). Une saturation est utilisée pour limiter la vitesse de rotation. La figure 22 montre les signaux en accélération acc_{tilt}, vitesse ω et position angulaire θ de la plateforme en utilisant le *tilt coordination*.

Enfin, le filtre $F3$ est égal au filtre $F1_a$. Certes, les paramètres sont différents afin de respecter les limites en accélération, vitesse et position angulaire.

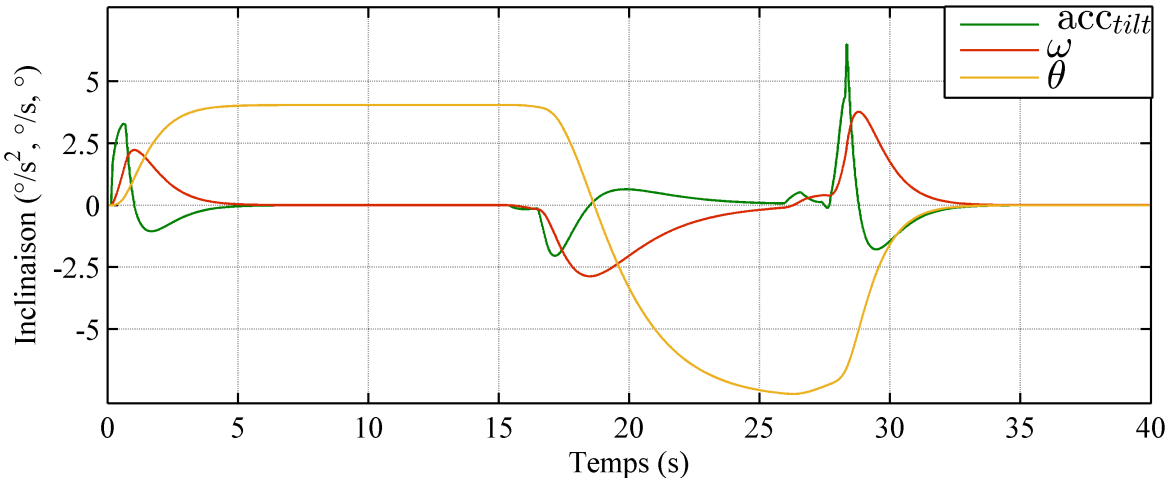


FIGURE 22 – Restitution de mouvement produit par l'hexapode grâce à la technique du *tilt coordination*.

Comme nous l'avons vu précédemment, la perception de la force spécifique dépend d'une composante haute fréquence (HF) et basse fréquence (BF) apportées par l'inclinaison de la plateforme. Afin de montrer les deux types d'accélération, nous traçons ces signaux dans la figure 23.

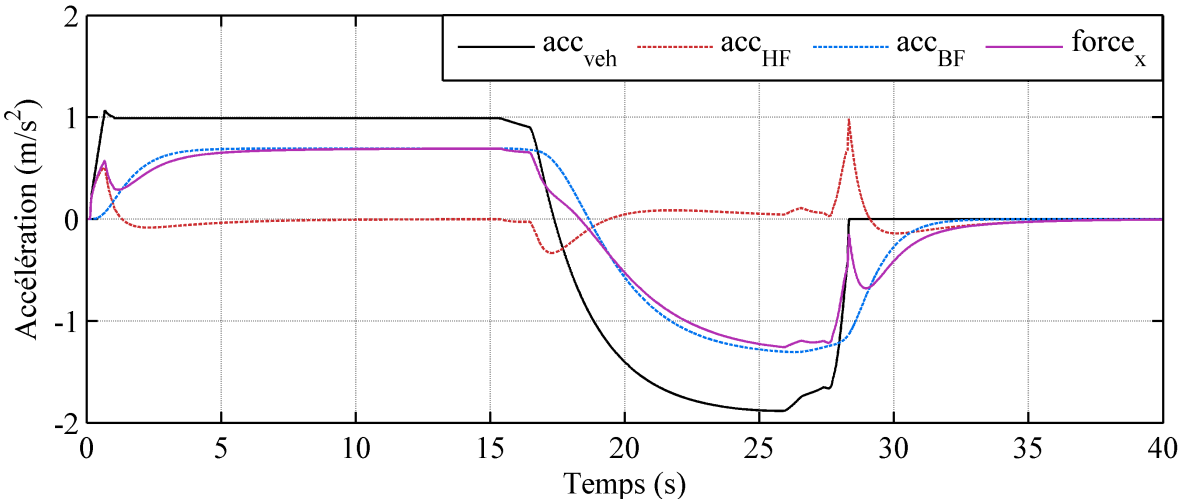


FIGURE 23 – Accélérations hautes fréquences, basses fréquences et force spécifique en utilisant l'algorithme classique selon l'axe de mouvement x.

Les paramètres utilisés pour reproduire le tracé des courbes de cette partie sont montrés dans le tableau 6.

D'après la littérature, il existe des variantes de l'algorithme classique. Par exemple, l'utilisation d'un gain optimal LQR (pour *Linear Quadratic Regulator*) dans la branche du *tilt coordination*. Le filtre $F2$ est alors remplacé par un régulateur linéaire quadratique qui contrôle l'inclinaison de l'hexapode, tout en minimisant la vitesse d'inclinaison et le mouvement total de la plateforme [Romano, 2003]. Cet

F1				***F2 - Tilt coordination***				
k_1	ζ_{n1}	ω_{n1}	ω_{b1}	k_2	ζ_{n2}	ω_{n2}	g	saturation
1	1,4	0,9	0,2	0,6	1	1,5	9,81 m/s^2	$\pm 4\ ^\circ/_s$

TABLEAU 6 – Paramètres utilisés dans l'algorithme classique

algorithme a été implémenté également par [Zywiol Jr & Romano, 2003]. Dans le même ordre d'idée, [Friedland *et al.*, 1973] suggèrent une technique de mouvement non linéaire pour le simulateur de la NASA.

3.1.2 Avantages et inconvénients

Les avantages et inconvénients de la stratégie classique tels que décrits par différents auteurs de la littérature sont les suivants :

Avantages :

- La mise en oeuvre du MCA classique est simple et facile à paramétrer [Nahon *et al.*, 1992].
- Leur structure en boucle ouverte avec des filtres simples font de cette stratégie la plus rapide en termes de temps de calcul [Telban *et al.*, 2000].

Inconvénients :

- Par rapport à d'autres stratégies, les signaux générés par l'algorithme classique ont une amplitude plus faible [Telban *et al.*, 2000]. En effet, les paramètres sont fixes et définis pour le pire des scénarios. Cette considération ne permet pas de profiter de tout l'espace de travail de la plateforme [Nehaoua *et al.*, 2006].
- Les signaux filtrés entraînent une sensation d'accélération irréaliste pour le conducteur [Nehaoua *et al.*, 2006].
- La caractéristique linéaire du filtre passe-haut produit un faux indice de mouvement au conducteur [Regalón *et al.*, 2012].
- Le modèle de perception humaine n'est pas intégré dans la structure de cette stratégie [Asadi *et al.*, 2014].
- Les indices de discordance dans les accélérations hautes fréquences, par exemple le freinage et le démarrage, peuvent provoquerr le mal du simulateur chez le conducteur [Fang & Kemeny, 2016].

Le nombre d'inconvénients par rapport au nombre d'avantages nous fait penser que cet algorithme ne devrait pas être utilisé pour le contrôle de plateforme. Cependant, ses avantages semblent être beaucoup plus importants pour les concepteurs des MCAs. Cet algorithme fait encore l'objet de recherches et est utilisé par un grand nombre de simulateurs à haute performance tels que LEEDS [Romano *et al.*, 2017], VTI Sim IV [Fischer *et al.*, 2011], SHERPA-Groupe PSA [Stratulat *et al.*, 2011], DLR [Seehof *et al.*, 2014], UoLDS [Sadraei *et al.*, 2016] et NADS [Schwarz *et al.*, 2003].

3.2 Commande adaptative

La commande adaptative est un type de contrôle non linéaire qui modifie la loi de contrôle en présence de perturbations dans la dynamique du système. Dans celle-ci, les paramètres du système de contrôle sont adaptées en temps réel. Cette commande a été développée par [Parrish *et al.*, 1975] afin d'améliorer certaines caractéristiques du contrôle classique, telles que sa souplesse et sa performance pour le suivi de trajectoire lors de la restitution du mouvement.

Plusieurs variantes de cet algorithme ont été développées depuis. Par exemple, [Ariel, Dan & Silvan, 1984] simulent en temps réel la réponse du système vestibulaire aux signaux d'entrée et selon le résultat, les filtres de *washout* sont adaptées. [Nahon *et al.*, 1992] proposent un algorithme hybride avec un filtre adaptatif non linéaire incorporé dans l'algorithme classique. Leur objectif est de combiner les caractéristiques les plus avantageuses des algorithmes classiques et adaptatifs sur une plateforme à 6 DDL. Ils utilisent un algorithme classique avec un ajustement en temps réel des filtres passe-haut des accélérations de translation et rotation.

L'objectif du contrôle adaptatif est de gérer les paramètres du filtre lorsque des entrées inattendues apparaissent dans l'accélération linéaire ou angulaire. Par exemple, lors d'un freinage important ou d'un démarrage agressif, les paramètres du filtre passe-haut de l'algorithme classique réduiront le signal d'entrée pour respecter les limites physiques du simulateur. En revanche, si les entrées d'accélération ne varient pas beaucoup, les filtres auront un impact moins significatif. En ce sens, il est nécessaire d'atténuer les valeurs les plus susceptibles d'entraîner des déplacements importants afin de réduire l'apparition de faux indices de mouvement. Le schéma 24 montre comment les filtres de l'algorithme classique sont modifiés en temps réel.

Le régulateur calcule et modifie en temps réel les facteurs d'échelle et les paramètres des filtres passe-haut et passe-bas de l'algorithme classique [Nehaoua *et al.*, 2006]. Cette modification est réalisée avec l'objectif de minimiser la fonction de coût ou fonction de performance.

La fonction de performance est une estimation mathématique de la qualité du comportement du système. Celle-ci contient des paramètres de pondération qui

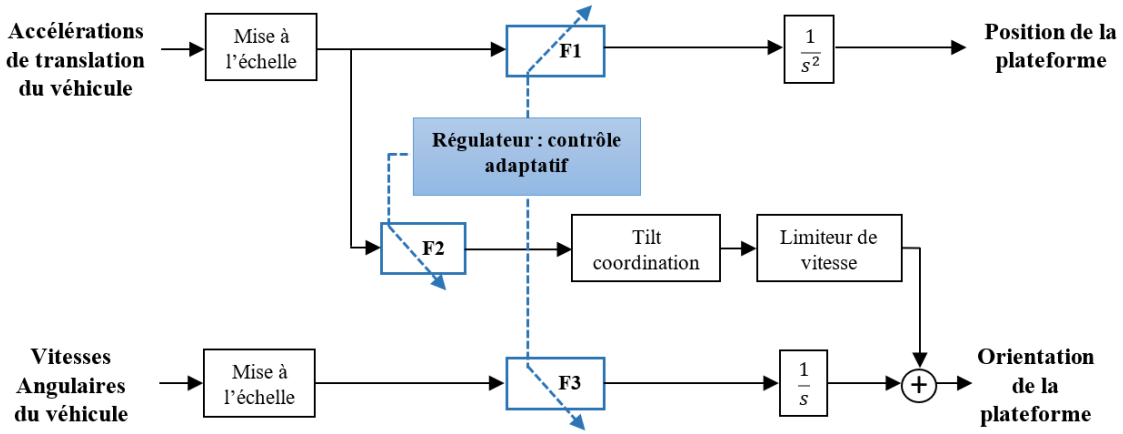


FIGURE 24 – Contrôle adaptatif.

rajoutent de l'importance aux restrictions de la plateforme et à l'erreur résultant de la différence entre le comportement du simulateur et celui du véhicule. Cette fonction varie selon les besoins de chaque concepteur de MCA [Ariel, Dan & Silvan, 1984; Ioannou & Sun, 2012; Rubio & Sánchez, 1996]. Par exemple, le premier algorithme adaptatif pour la restitution de mouvement utilise une fonction de coût quadratique pour faire varier les gains de filtres [Parrish *et al.*, 1975], mais celui de [Nahon *et al.*, 1992] évalue d'autres degrés de liberté dans la fonction afin d'atteindre une meilleur performance de la loi de contrôle résultante.

Normalement, la fonction de coût est exprimée dans un système cartésien qui couple les 6 DDL entre eux. Cette condition fait que le déplacement maximal selon un seul degré de liberté limite la capacité selon les 5 DDL restants. C'est pourquoi, [Grant & Naseri, 2005] ont proposé une solution alternative à ce problème dans laquelle chaque DDL fonctionne de façon indépendante, autrement dit, il existe une fonction de coût (9) définie pour chaque accélération inertielle selon les axes x, y, z et les angles de roulis, tangage et lacet. Cependant, pour des raisons pratiques, seul l'axe x sera développé dans ce document. La fonction de coût selon l'axe x est :

$$J = \frac{1}{2^{N-1}}[w_1(a_{veh} - \ddot{x_{pf}})^N + w_2\dot{x_{pf}}^N + w_3 x_{pf}^N + w_4(P_i - P_{i0})] \tag{9}$$

Le premier terme de la fonction fait référence à l'erreur entre l'accélération souhaitée a_{veh} et l'accélération de la plateforme $\ddot{x}_{pf}$ qui est la sortie du système dynamique. Le deuxième et le troisième termes contiennent les pénalités du mouvement : la vitesse $\dot{x_{pf}}$ et la position de la plateforme x_{pf}. Ils sont introduits dans la fonction afin de limiter le déplacement du simulateur à l'intérieur de l'espace de travail. Les valeurs w_1, w_2, w_3 et w_4 sont les poids de pondération respectifs de l'erreur, la vitesse, le déplacement et le gain. La valeur N représente le degré de liberté ou ordre polynomial de la fonction qui généralement est de 2 [Telban & Cardullo, 2005]. Le dernier terme représente l'écart entre les paramètres adaptatifs et leurs valeurs nominales.

Généralement, le vecteur P_i contient l'ensemble des paramètres adaptatifs des filtres passe-haut du deuxième ordre, tels que le coefficient d'amortissement, le gain et la fréquence naturelle. Cependant, comme indiqué dans la partie 3.1, les paramètres qui affectent le plus le comportement du filtre sont le gain et la fréquence propre du système. Par conséquent, le coefficient d'amortissement reste statique afin de respecter le seuil de perception de l'accélération linéaire. Dans [Grant & Naseri, 2005], il est démontré que parmi ces paramètres restants, le gain impacte davantage le suivi de trajectoire lorsqu'il est adapté dans le contrôle. C'est pourquoi, afin de déployer un MCA adaptatif, seul le gain est calculé en temps réel à partir de l'algorithme classique, à chaque pas de temps, pour minimiser la fonction J dans l'équation 9.

Différentes méthodes d'optimisation sont utilisées pour ajuster en temps réel les paramètres estimés P_i [Ioannou & Sun, 2012]. Parmi elles, nous pouvons citer la méthode de sensibilité, la conception de Liapounov, la méthode des moindres carré et l'algorithme de la descente du gradient (*steepest descent method*). Les deux dernières cherchent à déterminer un contrôle adaptatif P_i qui minimise la fonction de coût J [Nehaoua, 2008; Telban & Cardullo, 2005]. Dans cette thèse nous utiliserons l'algorithme de la descente du gradient, ce qui donne la loi adaptative suivante :

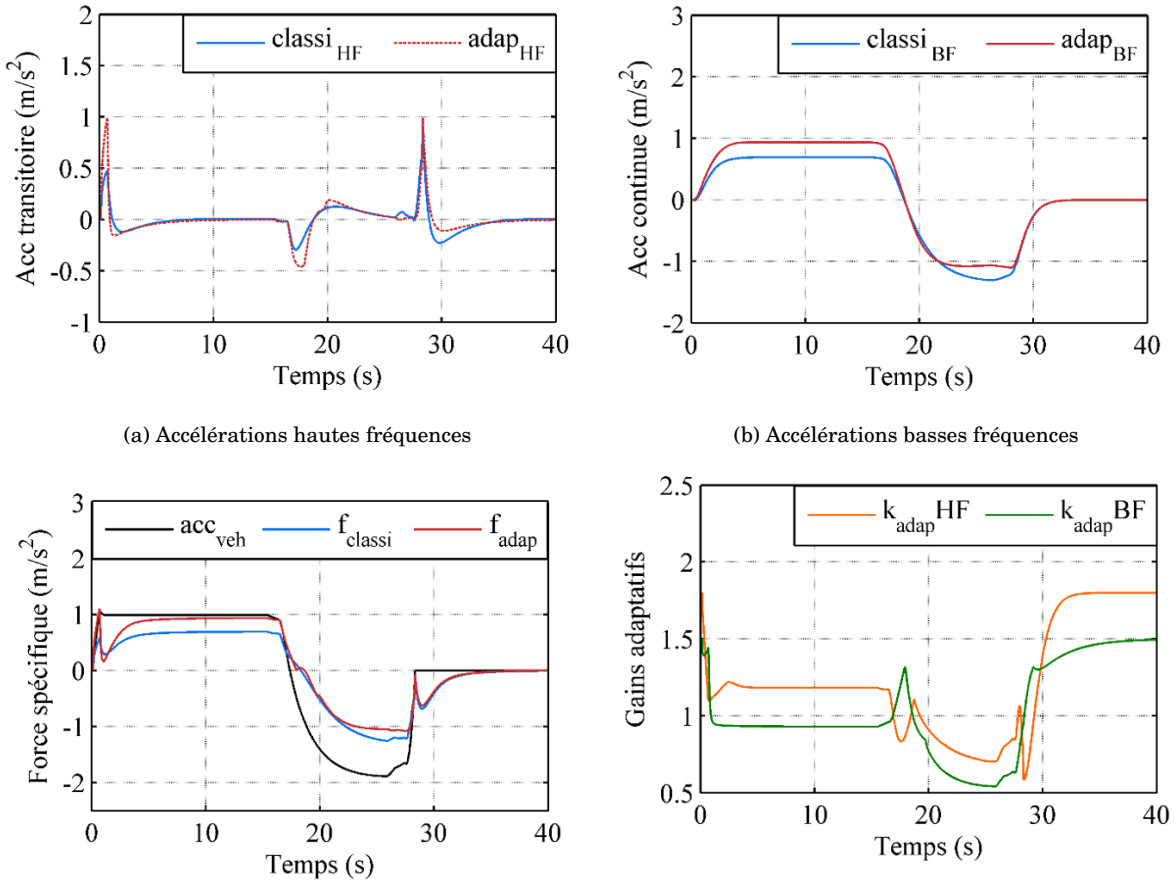
$$\dot{P}_i = -\alpha_i \frac{\partial J}{\partial P_i} \tag{10}$$

Avec, $\alpha > 0$ le pas d'adaptation qui définit la stabilité de la méthode et $\frac{\partial J}{\partial P_i}$ representant la sensibilité de la fonction J aux changements du paramètre P_i. Parfois, il est nécessaire de faire varier le paramètre α_i afin de stabiliser le système et améliorer la qualité du signal de sortie.

Afin d'évaluer les avantages de cette stratégie par rapport à la stratégie classique, nous comparons la réponse des filtres passe-haut de l'algorithme classique et la réponse de ces mêmes filtres avec l'adaptation du gain statique en temps réel dans le même scénario de simulation. La figure 25 montre cette comparaison, la réponse totale du système exprimée en termes de force spécifique, ainsi que la variation des gains statiques des accélérations hautes fréquences et basses fréquences. Nous observons que la stratégie adaptative permet un meilleur suivi de la force spécifique d'entrée.

3.2.1 Avantages et inconvénients

La commande adaptative fournit de nombreux avantages par rapport à la méthode classique : l'augmentation de la fidélité de conduite lorsque le simulateur est proche de sa position neutre, une souplesse de réglage des filtres pour les accélérations de translation et rotation, la définition d'une compromis entre le rendu de l'accélération et la limitation du mouvement de la plateforme [Dagdelen *et al.*, 2009], l'identification de paramètres en temps réel et une meilleure exploitation du potentiel de la plateforme. Néanmoins, l'algorithme adaptatif n'est pas utilisé parmi



(a) Accélérations hautes fréquences

(b) Accélérations basses fréquences

(c) Réponse totale du système : forces spécifiques

(d) Gain adaptatif des filtres hautes fréquences et basses fréquences

FIGURE 25 – Comparaison entre la réponse des filtres avec les algorithmes classique et adaptatif.

les algorithmes de restitution de mouvement, car la méthode d'optimisation des paramètres peut déstabiliser le système. Outre l'instabilité, la procédure de réglage est plus compliquée qu'avec la méthode classique. Par ailleurs, cette technique ne limite pas au maximum les contraintes de la plateforme, ce qui signifie que la plateforme peut dépasser les limites dans certaines conditions de conduite. En fait, ce modèle a été utilisé dans différents travaux de recherche pour les simulateurs de véhicules, de motocyclettes ou d'avions. Cependant, à notre connaissance, aucun simulateur à hautes performances n'utilise cette technique. Nous pensons que l'une des raisons est que les restrictions de la plateforme ne sont pas implicitement intégrées, ce qui peut entraîner, selon le scénario, un risque pour le système et pour le conducteur.

3.3 Commande optimale

Le première implémentation de la commande optimale a été proposée par [Kurosaki, 1978]. L'approche utilise une structure optimale quadratique pour définir la fonction de coût. Postérieurement, [Sivan *et al.*, 1982] utilisent la même approche de [Kurosaki, 1978] mais en intégrant le système vestibulaire dans la formulation de la fonction de coût afin de fournir une meilleure perception dans les simulateurs aéronautiques. Dans leurs propositions, l'algorithme optimal considère deux systèmes dynamiques : le système réel et le système simulé. Cette commande optimale minimise la fonction de coût non seulement en fonction de l'erreur de suivi, mais aussi en fonction du modèle de perception du mouvement linéaire [Ish–Shalom, 1982].

L'intégration du système vestibulaire est assez courante chez les développeurs de MCA [Cardullo *et al.*, 1999; Chen & Fu, 2010; Ish–Shalom, 1982]. Cependant, dans cette thèse, le système vestibulaire ne sera pas pris en compte dans la modélisation de la stratégie optimale, car l'idée dans ce chapitre est de comparer les capacités de cette stratégie avec les autres, qui eux n'intègrent pas le modèle de perception. Le système vestibulaire sera intégré plus tard à la stratégie que nous aurons retenue comme la plus appropriée pour la validation du véhicule autonome et les simulateurs à hautes performances.

Le schéma 26 illustre la structure de cette stratégie, Ici, le système vestibulaire est intégré dans la figure seulement à titre indicatif.

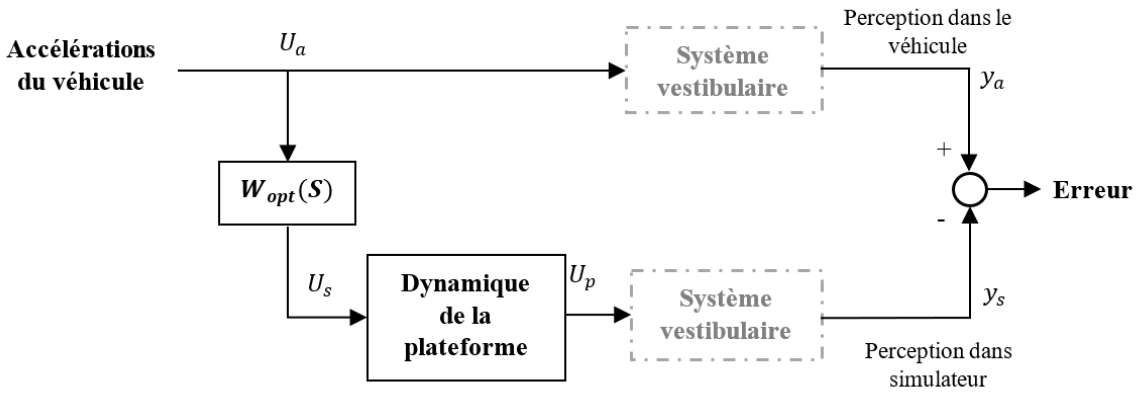


FIGURE 26 – Contrôle optimal.

La matrice du filtre optimal $W_{opt}(S)$ est le résultat d'un processus d'optimisation qui minimise l'erreur entre la sortie et l'entrée du système, comme représenté dans la figure 26. Pour trouver $W_{opt}(S)$, il faut d'abord définir le système complet. Pour cela, nous représentons une série de sous-systèmes qui décrivent le fonctionnement du simulateur dynamique. Parmi eux se trouvent :

- L'erreur entre les états du simulateur x_{pf} et du véhicule x_{veh} : $x_e = x_{pf} - x_{veh}$.
- Les états du simulateur tels que la vitesse, le déplacement et l'accélération linéaire et angulaire selon tous les DDL. Ils seront intégrés et pénalisés dans la fonction de coût de l'algorithme [Chen & Fu, 2010].
- Un modèle de bruit blanc : il fournit la référence du système hors ligne afin de reproduire plusieurs situations de conduite. Dans la théorie du contrôle, il est connu en tant que perturbation stochastique. L'entrée du véhicule U_a (mouvement du véhicule inconnu) est modélisée par un bruit gaussien coloré $n(t)$. Le bruit blanc filtré doit contenir des fréquences qui représentent les états de mouvement d'un simulateur de conduite [Ish–Shalom, 1982].

Le couplage de ces trois sous-systèmes est défini par :

$$\begin{aligned} \dot{X}(t) &= Ax(t) + Bu(t) + Hn(t) \\ Y(t) &= Cx(t) + Du(t) \end{aligned} \tag{11}$$

Où,

$$X = \begin{bmatrix} x_e \\ x_{pf} \\ x_n \end{bmatrix} \quad Y = \begin{bmatrix} \textit{accélération linéaire} \\ \textit{accélération angulaire} \end{bmatrix} \tag{12}$$

Avec, X le vecteur d'état contenant l'erreur du système x_e, les états de la plateforme x_{pf} et le bruit blanc filtré x_n ; $u(t)$ l'accélération linéaire et angulaire et $n(t)$ le bruit blanc. Les matrices A, B, H, C, D contiennent les variables correspondantes à chaque entité. La procédure pour obtenir le système d'état (11) est détaillée dans l'annexe A.

Pour le critère d'optimisation, nous utilisons une forme quadratique, c'est-à-dire, un polynôme homogène du deuxième degré. Celui-ci est représenté sous forme matricielle dans la fonction de coût suivante :

$$J = \int_0^\infty (e^T Q_e e + X_{pt}^T R_{pf} X_{pf} + u^T R_u u) \tag{13}$$

Dans l'intégrale, le premier terme contient l'erreur entre le simulateur et le véhicule réel, le deuxième terme contient toutes les variables d'état et le troisième terme définit le contrôle du système. Les matrices Q_e, R_{pf} et R_u sont les matrices de pondération respectives. Le contrôleur résultant du problème d'optimisation consiste en un retour des états du système complet dans l'équation (11), ainsi :

$$u = -KX \tag{14}$$

Les gains de rétroaction en régime permanent K dépendent de la solution d'une équation différentielle matricielle non linéaire, appelée équation de Riccati [Bellon, 2008]. Lorsque l'on modifie l'équation (14) [Chen & Fu, 2010], nous pouvons dériver l'équation suivante dans le domaine de Laplace :

$$u_a(S) = W_{opt}(S) u_{pf}(S) \tag{15}$$

Avec W_{opt} la matrice de la fonction de transfert optimisée qui relie les entrées du simulateur u_{pf} aux entrées du véhicule u_a.

3.3.1 Performance en tant que stratégie de contrôle de mouvement

Pour analyser la mise en œuvre de cette stratégie, nous avons implémenté la commande optimale pour les accélérations linéaires selon l'axe x. Nous avons constaté que son développement est plus complexe que pour les deux stratégies précédentes, étant donné que la stabilité et la performance du système sont liées aux matrices de pondération Q_e, R_{pf} et R_u. Pour choisir leurs valeurs, nous avons d'abord essayé la règle de Bryson, selon laquelle ces valeurs sont fixées à l'inverse du carré des valeurs maximales admissibles des états correspondants [Kurosaki, 1978]. Cependant, le contrôle obtenu avec ces valeurs a provoqué une instabilité du système. Nous avons donc opté pour la technique d'essai et erreur, qui nécessite une certaine expérience et connaissance du système, et qui malheureusement prend beaucoup de temps. Nonobstant, ce problème de réglage peut être amélioré avec d'autres méthodes. C'est le cas de [Asadi *et al.*, 2015b] qui proposent une méthode fondée sur les algorithmes génétiques afin de résoudre le problème d'optimisation linéaire quadratique (LQ). Leur objectif était de suivre la trajectoire, réduire les faux indices de mouvement, inclure le seuil de sensibilité humaine dans le *tilt coordination* et maximiser l'espace de travail de la plateforme.

La figure 27 compare les signaux obtenus par l'algorithme classique et l'algorithme optimal. Nous observons que la commande optimale ne suit pas bien les accélérations transitoires du signal d'entrée et que par rapport à la stratégie classique, elle a une performance inférieure.

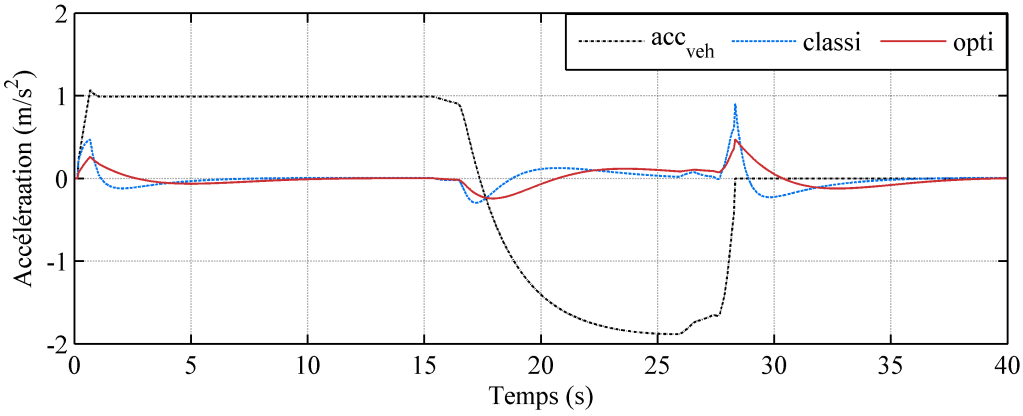


FIGURE 27 – Comparaison entre la restitution de mouvement de la courbe acc_{veh} par la stratégie classique (classi) et la stratégie optimale (opti).

Malgré les efforts de la commande optimale pour réduire l'erreur de perception des mouvements dans les simulateurs de conduite, la plupart des auteurs de la littérature concluent que les résultats ne sont pas satisfaisants pour atteindre une bonne perception sans ressentir de faux indices dûs aux limitations de l'espace de travail de la plateforme. Même si dans tous les MCA, les contraintes physiques sont les mêmes, dans la conception des stratégies vues précédemment, les faux

indices sont traités par les paramètres des filtres, ce qui n'est pas le cas de cette stratégie. D'ailleurs, le seuil de perception et la limite de vitesse du *tilt coordination* ne sont pas intégrés dans le processus d'optimisation. En général, la plupart des algorithmes optimaux ne considèrent pas de contraintes ni les limites physiques dans l'optimisation.

[Jamson, 2010] explique que l'algorithme optimal n'est pas très utilisé dans la simulation de conduite, en raison de l'insuffisance et de la complexité des modèles physiologiques de perception. Les avantages de l'algorithme optimal par rapport aux autres MCA ne justifient pas l'effort dans la mise en œuvre de cette stratégie étant donné l'inconvénient significatif de l'ajustement des poids dans la fonction de coût. De même, le paramétrage des coefficients de pondération est fait hors ligne et basé sur un profil d'accélération recréé par un bruit blanc, donc, ils ne sont pas toujours adaptés à une situation de conduite réelle. Certes, le bruit blanc permet un ajustement global pour plusieurs situations de conduite, mais cela ne veut pas dire qu'il représente le profil de conduite exact dans toutes les situations [Dagdelen *et al.*, 2009].

Contrairement aux travaux précédents, [Asadi *et al.*, 2014] comparent l'algorithme classique avec l'optimal ; les résultats montrent la capacité du MCA optimal à bien restituer le mouvement grâce à une meilleure performance, une haute fidélité des mouvements, une amélioration de la perception de mouvement des conducteurs et une meilleure exploitation de l'espace de travail de la plateforme. Cependant, malgré de multiples tentatives pour obtenir un bon paramétrage dans notre simulation, l'algorithme optimal a donné une performance inférieure.

Il est néanmoins possible d'améliorer cette stratégie avec d'autres MCA. C'est le cas de [Telban & Cardullo, 2005], qui développent un algorithme combinant à la fois l'algorithme adaptatif non linéaire et l'algorithme optimal. L'algorithme formule un problème de contrôle optimal avec une loi de contrôle non linéaire grâce à des filtres non linéaires ajustés en temps réel. Il incorpore un modèle de perception humaine qui intègre les sensations visuelles, vestibulaires et les interactions entre les stimuli.

3.4 Commande prédictive

La commande prédictive été proposée en 1960, mais ce n'est qu'en 1980 que les premières implémentations de cette commande, telles que la commande par matrice dynamique (DMC pour *Dynamic Matrix Control*) et la commande prédictive généralisée (GPC pour *Generalized Predictive Control*) sont sorties [Bemporad & Morari, 1999]. La commande prédictive a été conçue pour faire face aux problèmes de contrôle multi-variables et intègrent des contraintes dans les états du système.

La terminologie de commande prédictive n'est pas spécifique à une seule stratégie de commande, mais s'applique à une large gamme de méthodes de contrôle qui

utilisent explicitement un modèle du processus pour obtenir le signal optimisant une fonction de performance [Bordons & Camacho, 2007]. Parmi eux se trouvent [Marruedo, 2002] :

- **MAC :** *Model Algorithmic Control*, connue normalement sous le nom d'IDCOM (identification, commande). Il est utilisé pour les systèmes stables et pour déterminer la réponse impulsionnelle du système, donc sans horizon de contrôle.

- **DMC :** utilise la réponse à un échelon d'un modèle mathématique afin de modéliser le processus. Une extension de ce type est le QDMC - *Quadratic Dynamic Matrix Control* dans lequel le problème d'optimisation avec contraintes est résolu en utilisant la programmation dynamique.

- **PFC :** *Predictive Functional Control*, utilise un modèle d'espace d'état du processus et permet de considérer des modèles non linéaires, instables et multi-variables.

- **SMOC :** *Shemm Multivariable Optimizing Control*, considère les systèmes sous forme d'état et incorpore les modèles de perturbations et les restrictions dures et souples.

- **GPC :** *Generalized Predictive Control*, la prédiction de la sortie est fondée sur le modèle de CARIMA (*Controller Autoregressive and Integrated Moving-Averange*), qui représente le système lui-même. Ce modèle est optimal lorsque des perturbations du système ne sont pas stationnaires et permettre de contrôler des systèmes qui ne peuvent pas l'être avec le DMC.

La commande prédictive que nous utiliserons dans cette thèse regroupe certaines caractéristiques des commandes nommées précédemment, telles que l'utilisation d'un modèle formulé dans l'espace d'état, la considération des contraintes dures et souples [1] et la solution d'un problème d'optimisation quadratique. Dans cette thèse nous appellerons cette commande, commande prédictive par modèle ou MPC pour *Model predictive control*.

La commande par modèle prédictif est un cas particulier de la stratégie optimale dont le principe est d'utiliser un modèle mathématique précis du système pour prédire, dans un horizon de prédiction fini, le comportement futur du système. Ainsi, les éléments principaux dans la commande prédictive sont le modèle du processus, la prédiction et l'optimisation. La figure 28 montre la structure de la commande prédictive avec ces éléments principaux.

1. Les contraintes dures ne peuvent en aucun cas être violées, en revanche, les contraintes souples doivent être satisfaites si possible et sont assouplies lorsque les contraintes dures sont contradictoires. Les matrices de pondération sont présentes dans la fonction de coût pour pénaliser leur violation.

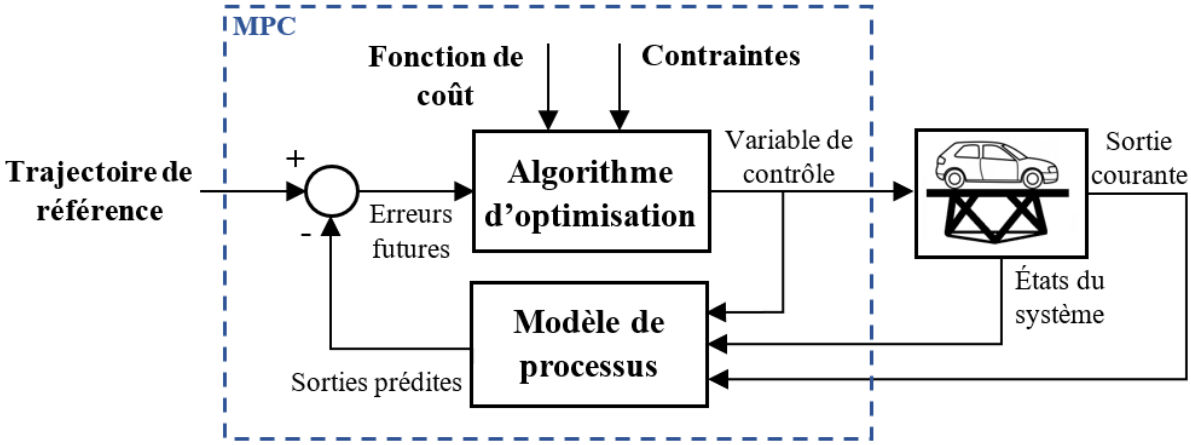


FIGURE 28 – Structure du contrôle prédictif.

3.4.1 Modèle mathématique du processus

Le modèle du système est constitué de tous les éléments nécessaires pour caractériser le comportement réel du système. Ce modèle peut être représenté, pour n'en citer que quelques-uns, par un modèle CARIMA, par le système dans l'espace d'état et par la réponse du système à une impulsion ou à un échelon. Pour obtenir le modèle mathématique du processus, nous utilisons l'estimation des états ou d'autres méthodes telles que les réseaux de neurones (partie 3.5.2) ou la logique floue (partie 3.5.1). Pour la prédiction, le contrôleur utilise un modèle approximatif du système, les valeurs précédentes et actuelles des variables afin de prédire le comportements des variables de sortie.

La dynamique du simulateur de conduite utilisée dans cette thèse a été réalisée par le fabricant de la plateforme, par conséquent nous n'y avons pas accès et elle est réputée idéale. C'est pourquoi, nous la considérons comme un gain unitaire. Cependant, nous utilisons un système linéaire invariant dans le temps, afin de représenter le modèle sous la forme d'un double intégrateur. Cela permet d'accéder à tous les états du simulateur x_{sim} tels que la position p, la vitesse linéaire v, l'angle θ et la vitesse angulaire ω des axes x, y et z de la plateforme. L'équation (16) représente le système pour l'un des axes du simulateur.

$$\begin{aligned} x_{sim}(k+1) &= \overbrace{\begin{bmatrix} 1 & t_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t_s \\ 0 & 0 & 0 & 1 \end{bmatrix}}^{A_{sim}} \overbrace{\begin{bmatrix} p(k) \\ v(k) \\ \theta(k) \\ \omega(k) \end{bmatrix}}^{x_{sim}} + \overbrace{\begin{bmatrix} \frac{t_s^2}{2} & 0 \\ t_s & 0 \\ 0 & \frac{t_s^2}{2} \\ 0 & t_s \end{bmatrix}}^{B_{sim}} \overbrace{\begin{bmatrix} u_{lin} \\ u_{rot} \end{bmatrix}}^{u} \\ y_{sim}(k) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{lin} \\ u_{rot} \end{bmatrix} \end{aligned} \tag{16}$$

Dans ce document, nous traitons l'optimisation selon son degré de liberté (DDL), c'est-à-dire, le nombre de variables de décision qui sont prises en compte pour

l'optimisation du MPC :

- **1 DDL :** chaque axe de la plateforme x, y, z, θ, φ et ψ est restitué de manière indépendante et parallèle en temps réel. La fonction de coût devient donc la différence entre les accélérations de référence et les accélérations en sortie, tout en considérant les contraintes physiques du simulateur et perceptives du conducteur. Dans ce cas nous utiliserons le système (16).

FIGURE 29 – Optimisation par degré de liberté.

- **2 DDL :** les accélérations et les angles d'inclinaison sont considérés comme variables de décision dans un même problème d'optimisation comme illustré dans la figure 30, et la sortie du système devient la force spécifique. La fonction de coût cherche donc à minimiser l'erreur entre la force spécifique f du modèle du véhicule et la force spécifique du simulateur avec l'intégration des contraintes.

FIGURE 30 – Optimisation de la force spécifique avec deux degrés de liberté.

Afin de mettre en œuvre ce type d'optimisation et prendre compte de la force spécifique du simulateur, la sortie y_{sim} du système (16) doit être modifiée comme suit :

$$y_{sim}(k) = \begin{bmatrix} 0 & 0 & g & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ \dot{x}(k) \\ \theta(k) \\ \dot{\theta}(k) \end{bmatrix} + \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} u_{lin}(k) \\ u_{rot}(k) \end{bmatrix} \tag{17}$$

Le modèle du système peut en même temps intégrer le modèle du système vestibulaire, afin de minimiser l'erreur de perception de mouvement entre le simulateur et

le conducteur. En utilisant l'axe x comme exemple, nous prenons le modèle de la force spécifique perçue par les otolithes $\hat{f}_x$ dans l'équation (1) et nous remplaçons la force spécifique d'entrée f_x dans l'équation (5).

$$\hat{f}_x = \frac{G_{oto}(\tau_{aoto}s+1)}{(\tau_{Loto}s+1)(\tau_{soto}s+1)}\left[u_{lin}(s) + \frac{g}{s^2}u_{rot}(s)\right] \tag{18}$$

En utilisant la forme canonique observable, le modèle des organes otolithiques est représenté dans un espace d'état par :

$$\begin{aligned} \dot{x_{oto}} &= \overbrace{\begin{bmatrix} -T_{3oto} & 1 & 0 & 0 \\ -T_{4oto} & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}^{A_{oto}} x_{oto} + \overbrace{\begin{bmatrix} T_{1oto} & 0 \\ T_{2oto} & 0 \\ 0 & gT_{1oto} \\ 0 & gT_{2oto} \end{bmatrix}}^{B_{oto}} \begin{bmatrix} u_{lin} \\ u_{rot} \end{bmatrix} \\ y_{oto} &= \overbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}}^{C_{oto}} x_{oto} \end{aligned} \tag{19}$$

Où,

$$T_{1oto} = \frac{G_{oto}\tau_{aoto}}{\tau_{Loto}\tau_{soto}}, \quad T_{2oto} = \frac{G_{oto}}{\tau_{Loto}\tau_{soto}}, \quad T_{3oto} = \frac{\tau_{Loto}+\tau_{soto}}{\tau_{Loto}\tau_{soto}}, \quad T_{4oto} = \frac{1}{\tau_{Loto}\tau_{soto}}$$

Afin de minimiser l'erreur de perception entre le simulateur et le conducteur tout en maintenant la plateforme à l'intérieur des limites physiques, nous unifions le système de la force spécifique perçue (19) et le système du simulateur (16). Ce système est utilisé dans le MCA de cette thèse et fait référence au modèle de la technique de *tilt coordination* :

$$\begin{aligned} \dot{X_{force}} &= \begin{bmatrix} A_{oto} & 0 \\ 0 & A_{sim} \end{bmatrix} \begin{bmatrix} x_{oto} \\ x_{sim} \end{bmatrix} + \begin{bmatrix} B_{oto} \\ B_{sim} \end{bmatrix} \begin{bmatrix} u_{lin} \\ u_{rot} \end{bmatrix} \\ Y_{force} &= \begin{bmatrix} C_{oto} & 0 \end{bmatrix} \begin{bmatrix} x_{oto} \\ x_{sim} \end{bmatrix} \end{aligned} \tag{20}$$

3.4.2 Prédiction

Étant donné que les systèmes (16), (17) ou (20) se présentent sous la forme d'un espace d'état, pour des raisons pratiques, dans cette partie, ils seront représentés par le même système discret invariant dans le temps (LTI) :

$$\begin{aligned} x_m(k+1) &= A_m x_m(k) + B_m u(k) \\ y(k) &= C_m x_m(k) + D_m u(k) \end{aligned} \tag{21}$$

Avec le vecteur d'état $x_m \in \mathbb{R}^s$, la commande $u_k \in \mathbb{R}^l$, la sortie $y_k \in \mathbb{R}^t$ et les matrices A, B, C, D de dimensions compatibles. La plateforme de conduite est soumise à certaines limites qui forcent la commande du système à travailler dans ces frontières. Les limites peuvent être physiques, dynamiques ou perceptives, mais les signaux du système ne doivent jamais dépasser de telles restrictions. Normalement, les contraintes sont exprimées comme des ensembles X et U qui contiennent les états du système et les actions du contrôle à chaque instant. Sans contraintes dans l'algorithme de mouvement, l'accélération du simulateur peut changer brusquement ou être de signe contraire au véhicule, ce qui donne une mauvaise perception de mouvement au conducteur [Fang & Kemeny, 2014]. Ce système est soumis respectivement aux conditions limites suivantes :

$$\begin{aligned} x_{min} \leq x(k) \leq x_{max} \\ y_{min} \leq y(k) \leq y_{max} \\ \Delta u_{min} \leq \Delta u(k) \leq \Delta u_{max} \end{aligned} \tag{22}$$

Dans une situation générale de suivi de trajectoire, nous souhaitons que l'erreur en régime permanent soit égale à zéro. C'est pourquoi, nous augmentons le modèle du système avec un intégrateur afin d'assurer une poursuite sans décalage dans la loi de contrôle [Rossiter, 2003]. L'entrée devient un incrément de contrôle $\Delta u(k)$ plutôt que la commande $u(k)$ elle-même :

$$\overbrace{\begin{bmatrix} x_m(k+1) \\ u(k) \end{bmatrix}}^{x_{k+1}} = \overbrace{\begin{bmatrix} A_m & B_m \\ 0_m & I_m \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} x_m(k) \\ u(k-1) \end{bmatrix}}^{x_k} + \overbrace{\begin{bmatrix} B_m \\ I_m \end{bmatrix}}^{B} \Delta u_k$$

$$y_k = \overbrace{\begin{bmatrix} C_m & D_m \end{bmatrix}}^{C} \begin{bmatrix} x_m(k) \\ u(k-1) \end{bmatrix} + D\Delta u_k$$

Avant de prédire les états, il convient de définir deux horizons de travail : un horizon de prédiction N_p $(1 \leq N_p)$ qui représente le temps pendant lequel les réponses du système sont évaluées et un horizon de contrôle N_u $(0 < N_u \leq N_p)$ qui détermine le temps pendant lequel les actions de contrôle peuvent varier. Les actions appliquées au système dans l'horizon de prédiction sont appelées variables de décision [Marruedo, 2002].

Dans la phase d'optimisation, le signal de référence et les contraintes sont spécifiés sur l'horizon de prédiction afin de calculer les mouvements futurs de la variable manipulée sur un horizon de contrôle.

Les vecteurs de la sortie prédite Y, des états prédits X et de la commande future ΔU sont créés de manière itérative et sont définis par :

$$\begin{aligned} X &= \begin{bmatrix} x(k+1) & x(k+2) & \cdots & x\left(k+N_p\right) \end{bmatrix}^T \\ Y &= \begin{bmatrix} y(k+1) & y(k+2) & \cdots & y\left(k+N_p\right) \end{bmatrix}^T \\ \Delta U &= \begin{bmatrix} \Delta u(k) & \Delta u(k+1) & \cdots & \Delta u\left(k+N_u-1\right) \end{bmatrix}^T \end{aligned} \tag{23}$$

En termes d'états actuels et d'incréments de contrôle futurs, les prédictions prennent la forme suivante :

$$\begin{aligned} X &= F_x x(k) + G_x \Delta U \\ Y &= F x(k) + G \Delta U \end{aligned} \tag{24}$$

Où,

$$G_x = \begin{bmatrix} B & D & 0 & \dots & 0 \\ AB & B & D & \dots & 0 \\ A^2B & AB & B & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A^{N_p-1}B & A^{N_p-2}B & A^{N_p-3}B & \dots & A^{N_p-N_u}B \end{bmatrix}, \quad F_x = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{N_p} \end{bmatrix}$$

et

$$F = C_x F_x, \quad G = C_x G_x, \quad C_x \triangleq diag\{C, \cdots, C\}$$

3.4.3 Optimisation

Étant donné la manière dont les actions de contrôle sont appliquées au système, la commande prédictive utilise une stratégie en horizon glissant [Bordons & Camacho, 2007] : à chaque pas d'échantillonnage, une séquence de contrôle ΔU en boucle ouverte est calculée en optimisant une fonction de coût sur un horizon prédictif fini N_p, dans les limites des contraintes du système (22). De l'optimisation ΔU, seul le premier élément, c'est-à-dire $\Delta u(k)$, est appliqué au processus à l'instant k, ce qui fait un système en boucle fermée. À l'instant $k+1$, une compensation des erreurs de modélisation sera réalisée et un nouveau problème d'optimisation sera résolu sur un horizon glissant comme illustré dans la figure 31 et résumé dans l'algorithme 1.

1 Définir N_p, N_u, k, $\delta(j)$, $\lambda(j)$ et $q(j)$ dans l'équation (25) ;
2 Résoudre le problème de programmation quadratique en calculant les variables de décision optimales ΔU ;
3 Prendre le premier élément de la séquence ΔU ;
4 Calculer l'entrée de la commande optimale $u(k) = \Delta U[1] + u(k-1)$;
5 Appliquer $u(k)$ comme entrée au système ;
6 $k \leftarrow k+1$;
7 Aller à 2.

Algorithme 1 : Algorithme du contrôle prédictif.

La fonction de coût à minimiser est de la forme :

$$\begin{aligned} J = \sum_{j=1}^{N_p} \delta(j) \left\| r\left(k+j|k\right) - y(k+j) \right\|_2^2 + q(j) \left\| x\left(k+j|k\right) \right\|_2^2 \\ + \sum_{j=1}^{N_u} \lambda(j) \left\| \Delta u(k+j-1) \right\|_2^2 \end{aligned} \tag{25}$$

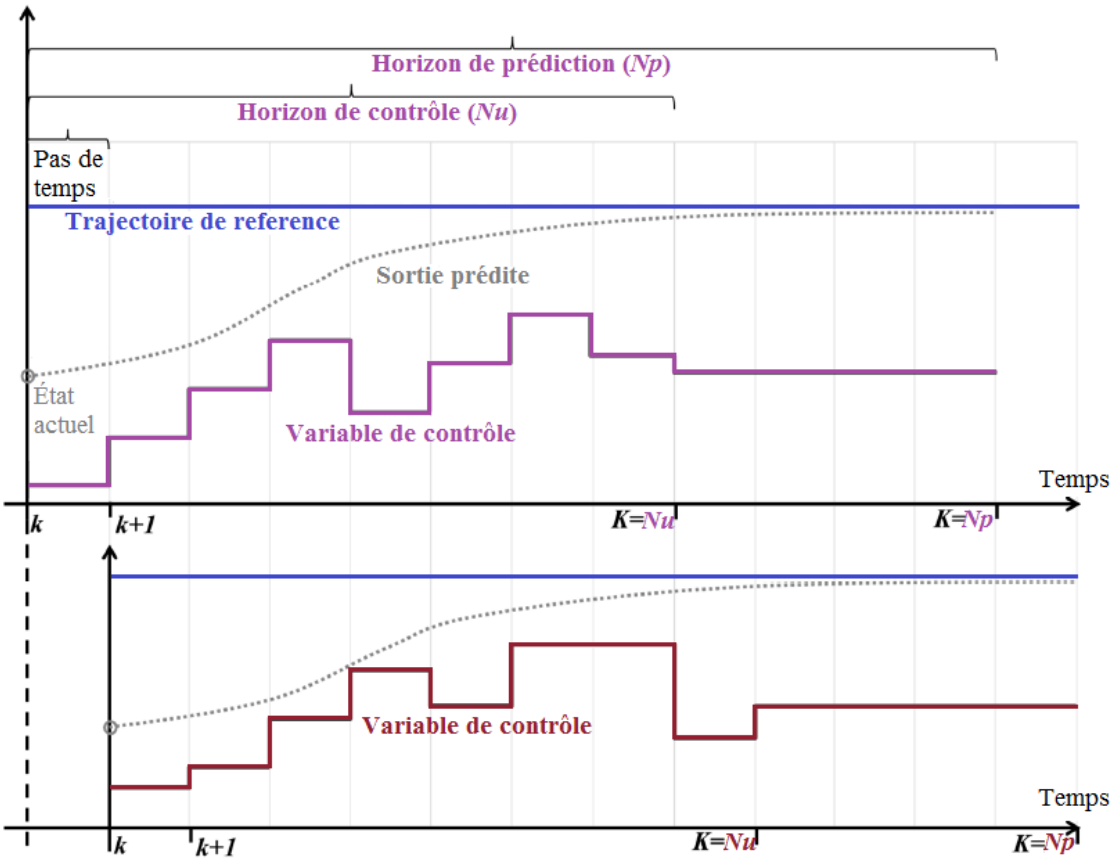


FIGURE 31 – Principe de fonctionnement du MPC en horizon glissant.

Avec J un objectif scalaire soumis aux contraintes définies dans l'équation (22). Cette fonction est composée de trois termes qui définissent l'importance des erreurs du système, des changements de contrôle et des états du système. Ainsi, $\delta(j)$, $\lambda(j)$ et $q(j)$ sont les paramètres de pondération respectifs. $r\,(k+j|k)$ indique la trajectoire de référence à l'instant d'échantillonnage k et $y(k+j)$ la sortie prédite. Les valeurs N_p et N_u sont des paramètres de réglage qui déterminent les horizons de prédiction et de contrôle dans la fonction de coût. Dans la forme matricielle, nous obtenons :

$$J = (Y - R_s)^T Q_\delta (Y - R_s) + \Delta U^T Q_\lambda \Delta U + X^T Q_q X \tag{26}$$

La trajectoire de référence R_s reste constante sur l'horizon de prédiction N_p. Dans la plupart des cas il est difficile d'avoir la référence future disponible, car elle dépend du comportement du conducteur. Ainsi, $R_s^T = r(k) \times [1 \quad 1 \quad \ldots \quad 1]_{1\times N_p}$.

En remplaçant à partir de l'équation (24), la sortie et les états prédits dans l'équation de J et en éliminant tous les termes qui ne dépendent pas de la variable de décision vectorielle ΔU, la fonction de coût obtenue est :

$$J = \Delta U^T \left(G^T Q_\delta G + Q_\lambda + {G_x}^T Q_q G_x\right) \Delta U + 2\left((Fx\,(k) - R_s)^T Q_\delta G + (F_x x\,(k))^T Q_q G_x\right) \Delta U$$

Enfin, le problème d'optimisation soumis à contraintes demeure :

$$\begin{aligned} &\text{minimiser} \quad J = \frac{1}{2}\Delta U^T H \Delta U + f^T \Delta U \\ &\text{soumis à} \quad A_c \Delta U \leq b \end{aligned} \tag{27}$$

Avec

$$\Delta U \in \mathbb{R}^n, \quad f \in \mathbb{R}^n, \quad b \in \mathbb{R}^m, \quad A_c \in \mathbb{R}^{m \times n}$$

La matrice H est symétrique définie positive $n \times n$, c'est-à-dire que toutes les valeurs propres de H sont strictement positives et, par conséquent la fonction de coût J a un minimum global unique. L'équation $A_c \Delta U \leq b$ contient toutes les contraintes dures d'inégalité linéaires de l'équation (22).

L'optimisation peut être calculée hors-ligne/*offline* ou en temps réel/*online*. Dans le cas *online*, on parle de MPC implicite dont le problème d'optimisation quadratique est résolu à chaque pas d'échantillonnage. Cela peut générer des conflits entre le temps de calcul et la solution optimale. Cette limite de calcul en temps réel peut détruire les propriétés de stabilité du MPC. Pour cette raison plusieurs chercheurs proposent une méthodologie dans laquelle les lois de contrôle optimal sont obtenues en utilisant une optimisation *offline*, on parle alors de MPC explicite. Dans celui-ci, les contraintes sont utilisées afin de diviser, hors ligne, l'espace d'état du système en ensembles ou régions polyédriques. Pour chacun de ces ensembles il existe une loi de contrôle qui est une fonction affine de l'état. Ainsi, la loi de contrôle est créée hors ligne et appliquée en ligne selon les états initiaux du système à chaque pas de temps [Beghi *et al.*, 2012; Fang & Kemeny, 2012a]. Néanmoins, cette méthode provoque certains problèmes ; l'inconvénient majeur est que le nombre de régions augmente de façon exponentielle avec l'horizon, le nombre des contraintes, la dimension des états et des entrées. Par conséquent, elle ne peut être appliquée de manière fiable qu'à de petits problèmes, où la dimension des états n'est pas, en moyenne, supérieure à 5 [Wang & Boyd, 2008]. De plus, à mesure que le nombre de régions augmente, la recherche et l'évaluation en temps réel devient plus complexe et plus lente [Bruschetta *et al.*, 2017]. En effet, dans [Fang & Kemeny, 2012a] il est montré que cette technique avec des ensembles invariants est indésirable pour le problème de suivi de trajectoire utilisant une optimisation à 2 DDL, car le calcul de l'ensemble admissible peut ne pas converger.

C'est pourquoi, nous avons choisi une stratégie MPC implicite dont le problème quadratique est résolu à chaque instant d'échantillonnage afin de trouver la séquence de contrôle ΔU. Les problèmes de temps de calcul et de stabilité que ce type de contrôle entraîne seront traités dans le chapitre suivant.

Maintenant, notre objectif est d'analyser la performance de cette stratégie en simulation. Dans un premier temps, nous comparons la réponse des accélérations linéaires selon l'axe x, en utilisant une optimisation implicite à 1 DDL avec le système (16) et à 2 DDL avec le système (17). Les résultats sont présentés dans la figure 32. Nous observons que le comportement de la sortie varie considérablement, surtout dans la plage 25-30 secondes. En effet, dans l'optimisation à 1 DDL, les

angles d'inclinaison ne sont pas pris en compte et le signal suit complètement la référence, alors que dans celle à 2 DDL, les angles d'inclinaison apportent un complément basse fréquence $g\theta$. Leur retard doit être compensé avec un pic d'accélération pour suivre la force spécifique.

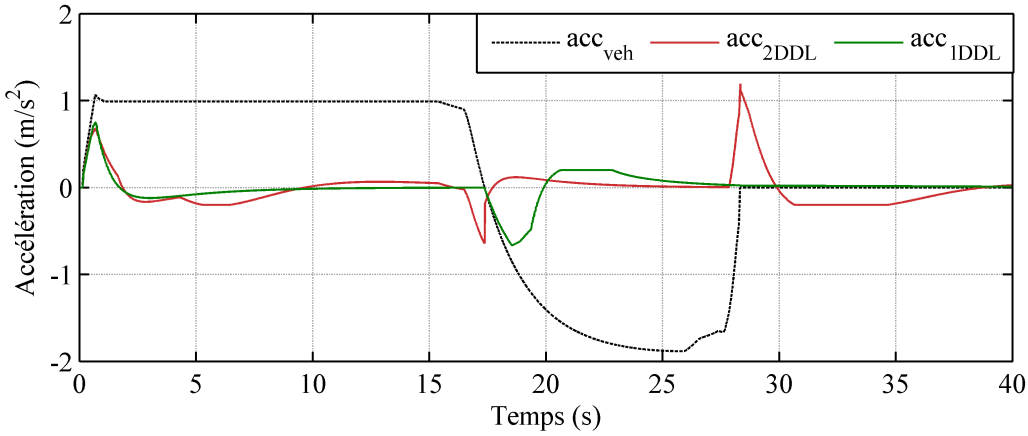


FIGURE 32 – Comparaison des accélérations linéaires selon l'axe x entre l'optimisation à 1 DDL et à 2 DDL.

Les composantes d'accélération à basses et hautes fréquences utilisant une optimisation à 2 DDL sont montrées dans la figure 33. Nous pouvons observer la compensation obtenue entre les deux fréquences de sorte que la force spécifique du simulateur suit la force spécifique du véhicule.

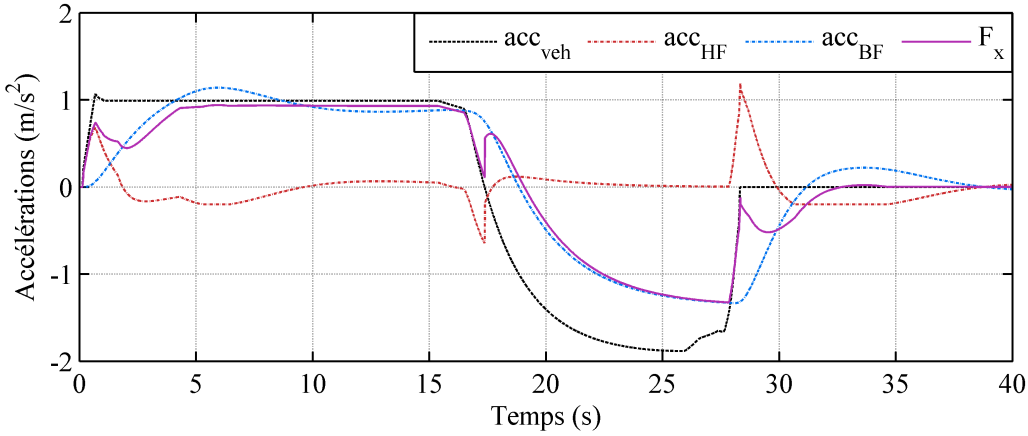


FIGURE 33 – Composantes basses (BF) et hautes (HF) fréquences de l'accélération, en utilisant le MPC afin de restituer la force spécifique selon x.

3.4.4 Avantages et inconvénients

Le principal avantage du MPC est la possibilité d'utiliser les contraintes d'entrée, de sortie et d'état du système dans le problème d'optimisation. Le MPC est la seule stratégie de contrôle générique qui n'utilise pas de filtres et qui gère efficacement les contraintes dans les états du système en offrant une meilleure utilisation de l'espace de travail de la plateforme.

En comparaison avec les autres stratégies, le MPC utilise une réalimentation pour compenser les effets des perturbations, gérer les contraintes et les non-linéarités dans le modèle. Également, avec ce type de contrôle, on peut utiliser un système « entrées multiples, sorties multiples » ou MIMO pour *Multiple-Input Multiple-Output*, sans augmenter la difficulté du problème.

Bien que cette stratégie présente des avantages précieux en termes de sécurité, il existe certains inconvénients qui doivent être résolus avant de l'utiliser dans les systèmes en temps réel. Il faut d'abord choisir le modèle mathématique basé sur la dynamique du processus réel, car avec un mauvais modèle, les prédictions accumulent des erreurs et le résultat n'est pas approprié. Un autre aspect important est la stabilité du MPC en boucles ouverte et fermée. En effet, l'intégration de contraintes dans le processus d'optimisation fait que le système est non linéaire en boucle fermée même si la dynamique du système est linéaire. Ces restrictions empêchent dans certains cas de trouver une solution au problème d'optimisation, ce qui peut entraîner une instabilité du système. De plus, le calcul pour résoudre le problème d'optimisation est lourd et nécessite un effort de calcul en temps réel considérable.

3.5 Autres techniques de contrôle avancées pour la restitution de mouvement

Jusqu'ici, nous avons vu quatre stratégies couramment utilisées pour la restitution de mouvement : le contrôle classique, adaptatif, optimal et prédictif. Néanmoins, l'un des objectifs principaux de cette thèse est de définir, parmi les stratégies de contrôle existantes, le MCA le plus approprié pour la validation des prestations ADAS ou du véhicule autonome. C'est pourquoi, dans cette partie nous analyserons d'autres techniques de contrôle avancé comme le contrôle flou et le contrôle neuronal, afin de définir leur potentiel en tant que MCA pour un simulateur dynamique à hautes performances. Nous verrons que chaque contrôle apporte des variations dans le suivi de la trajectoire, le modèle dynamique du système ou la perception du mouvement dans le simulateur.

3.5.1 Commande par logique floue

La théorie floue a été développée par le professeur Lotfi A. Zadeh en 1965 comme une extension de la théorie classique en permettant de représenter mathématiquement l'incertitude et l'imprécision [Regalado Crisanto, 1999]. En 1974, Ebrahim Mandami a démontré l'applicabilité de la logique floue dans le contrôle commande en réalisant le premier système de contrôle flou. Le contrôle flou permet, à partir d'un ensemble de règles linguistiques, de générer des connaissances qui vont donner au système la capacité à prendre des décisions sur son fonctionnement. Ces règles sont fondées sur des données des connaissances et expériences fournies par l'humain.

La théorie floue devient utile dans le contrôle lorsque le modèle mathématique du processus est difficile ou le processus implique l'interaction humaine et de l'expérience. La logique floue est l'une des meilleures suggestions lorsque les systèmes ont un comportement stochastique extrême et présentent différents types de variables d'entrée [Asadi *et al.*, 2015a].

Pour appréhender le fonctionnement de la logique floue, il est nécessaire de comprendre les termes suivants :

- Fonction d'appartenance : définit le degré auquel n'importe quelle variable est comprise partiellement dans un ensemble flou. Elles peuvent être représentées de différentes manières, comme triangulaire, trapézoïdal, singleton, etc.
- Fuzzification : procédé mathématique dans lequel une variable mesurée du processus devient une valeur du domaine flou.
- Mécanisme d'inférence : cycle de calcul qui émule la prise de décision de l'expert dans l'interprétation et l'application de connaissances sur le contrôle du système [1].
- Règles floues : représentent la connaissance du système sous la forme d'un ensemble de règles. Ils sont du type : SI l'ensemble de conditions est satisfait, ALORS, un ensemble de conséquences sont déduites.
- Défuzzification : processus mathématique qui convertit les ensembles flous en valeurs réelles pour le système.

Le schéma 34 illustre la boucle de contrôle en utilisant de logique floue. Les valeurs sont d'abord traitées par le processus de fuzzification. Puis, le mécanisme d'inférence évalue les règles en temps réel et prend une décision. Finalement, l'interface de défuzzification transforme les conclusions de mécanisme d'inférence en valeurs d'entrée pour le processus.

1. Dans le contrôle, l'inférence de Mamdani est la plus utilisée, car son calcul est relativement facile et adapté pour des applications en temps réel [Bahrami *et al.*, 2013].

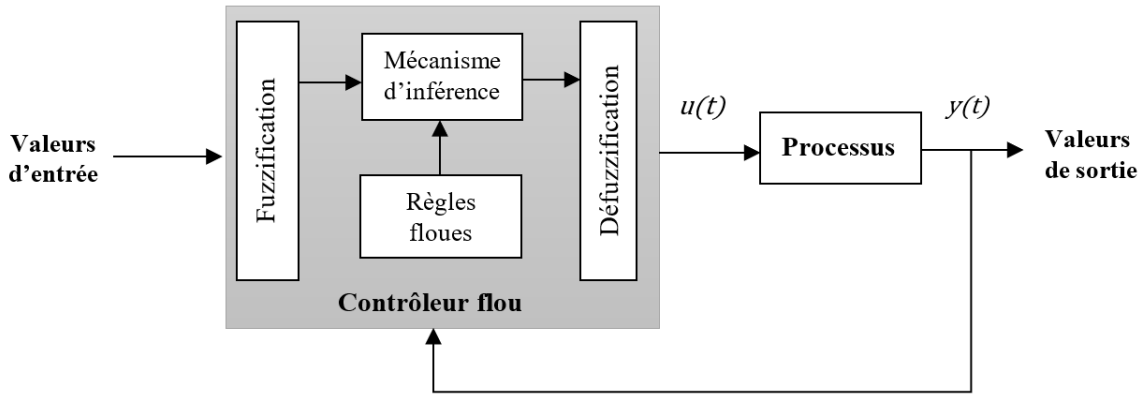


FIGURE 34 – Architecture du contrôle flou [Passino *et al.*, 1998].

La logique floue dans le contrôle plateforme

D'après la littérature, seulement quelques expérimentations ont intégré la logique floue dans les algorithmes de contrôle de mouvement. Ceux-ci se fondent principalement sur la recherche du modèle mathématique du système, à savoir, la cinématique inverse de la plateforme. Par exemple, [I-Fang Chung *et al.*, 1999] ont développé un contrôle flou pour contrôler le positionnement d'une plateforme de mouvement à 6 DDL de type Stewart. Pour commencer, ils ont développé la cinématique inverse du système hydraulique pour trouver le modèle mathématique basé sur la longueur de chaque actionneur par rapport à la plateforme fixe.

La logique floue est utilisée du fait qu'elle s'applique effectivement à un système dans lequel les relations entrées-sorties ne sont pas clairement identifiées. Pour cette raison, [Fellah *et al.*, 2013] l'utilisent pour modifier les paramètres des filtres dans l'algorithme classique. Ils utilisent en entrée la limite de déplacement, la limite de l'angle et la force spécifique à basse fréquence et en sortie les fréquences de coupures des filtres passe-haut et passe-bas. Pour créer les fonctions d'appartenance et postérieurement les règles floues, ils utilisent une optimisation par ensembles de particules, une technique qui ajuste les paramètres des fonctions d'appartenance. En suivant le même principe, [Asadi *et al.*, 2015a] proposent un filtre de *washout* adaptatif basé sur un réglage de la fréquence de coupure dans tous les filtres de la commande classique en utilisant la logique floue. La partie floue produit un signal plus précis qui minimise l'erreur de perception du mouvement entre le simulateur et le véhicule. Les entrées pour le contrôle flou sont l'erreur de perception et la limite de déplacement/angle de la plateforme, la sortie est la fréquence de coupure de chaque filtre dans l'algorithme classique.

[Tang *et al.*, 2016] proposent un algorithme qui contient un facteur de dimensionnement non linéaire, basé sur un réglage flou en temps réel, pour compenser le gain des accélérations d'entrées. Cette méthode est appliquée dans le *tilt coordination* et leur objectif est de permettre un mouvement continu, tout en respectant les seuils de perception. Le contrôle flou repose sur le déplacement, la limitation de l'espace

de travail et l'erreur perçue de la force spécifique. Il est appliqué dans l'algorithme classique uniquement pour les filtres impliqués dans les accélérations de translations. Par rapport à l'algorithme classique, la méthode a montré une meilleure performance, une fidélité accrue et une réduction de l'erreur de perception humaine.

À partir des informations citées précédemment, nous pouvons déduire que la logique floue est principalement utilisée dans le MCA afin de faire varier les valeurs des filtres de l'algorithme classique, en d'autres termes, cette technique peut être comparée à un algorithme de type adaptatif qui cherche à réduire l'erreur de perception du mouvement en temps réel. Bien que, par rapport à la commande classique, le contrôle flou améliore le suivi de la trajectoire et réduit l'erreur de perception, tout comme la stratégie classique, il n'est pas possible d'assurer le respect des limites physiques de la plateforme ni des seuils de perception humaine. En outre, afin de régler correctement les paramètres des filtres, il est nécessaire avoir des connaissances sur le comportement du système. Ce qui rend, par exemple, cette technique beaucoup plus complexe que la commande adaptative. C'est pourquoi nous avons décidé de ne pas l'appliquer à notre système.

3.5.2 Contrôle neuronal

En 1957, Frank Rosenblatt a développé le plus ancien réseau de neurones utilisé dans le contrôle, plus précisément dans les modèles d'identification. Les réseaux de neurones représentent un système de traitement de l'information, capable d'apprendre à partir de l'expérience. Sa structure s'approche du modèle artificiel du cerveau humain [Matich, 2001].

Ces réseaux peuvent apprendre à différencier les modèles à travers des exemples et la pratique grâce à l'apprentissage [1]. De cette façon, les réseaux neuronaux peuvent être utilisés comme contrôleur général, dans lequel, à partir de certaines références, le réseau génère le contrôle correspondant. En revanche, cette méthode peut générer des réponses non désirées puisqu'il n'y a aucun contrôle de ce qui se passe dans le réseau en temps réel et donc, le système peut devenir instable. De plus, pour générer toutes les données utilisées dans le processus d'apprentissage, le système doit être contrôlé à l'avance.

Les réseaux neuronaux peuvent faire partie d'une structure de contrôle, par exemple, en tant que fonctions d'approximation. C'est-à-dire, lorsque nous avons une fonction inconnue, mais que nous connaissons sa réponse à diverses entrées, nous pouvons ajuster les paramètres du réseau afin de produire la réponse de la fonction inconnue [Hagan *et al.*, 2002], tel que montré par exemple dans la figure 35 pour le MPC.

1. Processus par lequel un réseau de neurones peut modifier les poids du réseau afin de réduire une certaine erreur quadratique. Ce processus permet au réseau, à partir d'un ensemble des données, d'être capable de calculer la sortie correspondant au vecteur d'entrées.

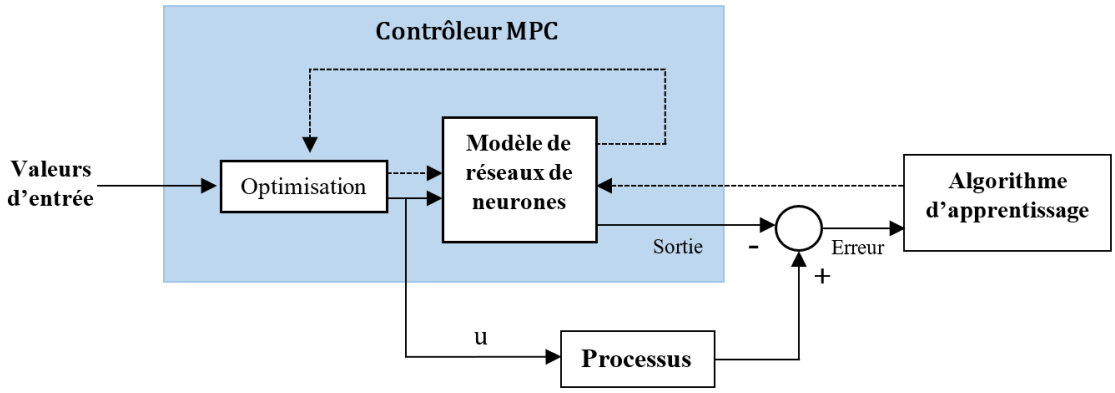


FIGURE 35 – Mise en oeuvre de réseaux neuronaux en tant que méthode d'identification dans la commande MPC.

Comme nous l'avons déjà indiqué dans la partie 3.4.1, cette approche n'est pas nécessaire pour notre problème, puisque nous disposons d'un modèle mathématique approprié qui définit le comportement de notre simulateur. Cependant, les réseaux neuronaux peuvent être utilisés, par exemple, pour générer la référence future dans le cadre du problème d'optimisation MPC. Dans ce cas nous ne faisons pas référence à un contrôle neuronal mais plutôt à une amélioration de la stratégie prédictive grâce aux réseaux neuronaux. C'est pourquoi, le contrôle neuronal ne sera pas considéré comme un MCA dans cette thèse.

De nombreuses variantes de contrôle sont apparues [Miunske *et al.*, 2019; Telban, 2002; Wang & Fu, 2004] dans le but d'améliorer les stratégies déjà existantes telles que les stratégies classique, adaptative, prédictive et optimale. Tout comme la logique floue ou le contrôle neuronal, ces variantes servent comme support et ne sont pas considérées comme un MCA en tant que tel. Par conséquent, elles ne seront pas détaillées dans le présent document.

3.6 Comparaison des algorithmes de contrôle de plateforme

D'après la littérature, les algorithmes classique, adaptatif et optimal sont les plus utilisés dans les différents simulateurs. Initialement, ces trois algorithmes ont été développés pour le domaine aéronautique, mais au fil du temps, ils ont été implémentés pour les simulateurs de conduite. Le dernier algorithme conçu spécifiquement pour les simulateurs de conduite est le contrôle prédictif, qui constitue aujourd'hui l'axe majeur de recherche pour les MCA.

Nous rappelons que parmi les algorithmes de restitution de mouvement, l'algorithme classique est le premier et l'un des plus utilisés dans les simulateurs de conduite. En termes de simplicité de conception et d'implémentation, il est vraisem-

blablement le plus approprié pour les simulateurs à faible coût. Leur paramétrage est relativement facile et fournit des résultats satisfaisants avec un temps court d'exécution. [Nehaoua *et al.*, 2006] a montré que cet algorithme est meilleur que l'algorithme adaptatif et optimal en ce qui concerne la sensation perçue par les conducteurs et la simplicité. Malheureusement, il génère des faux indices de mouvement, en particulier lorsque de nombreuses variations apparaissent dans l'accélération telle que le freinage ou le début d'une accélération. Le principal inconvénient est que les paramètres des filtres sont fixes pendant la simulation, ce qui entraîne une inflexibilité et une inefficacité à s'adapter aux situations qui produisent une mauvaise utilisation de l'espace disponible pour la plateforme de mouvement.

En revanche, l'algorithme adaptatif ajuste les paramètres en temps réel. Cependant, il présente une difficulté dans l'estimation de la fonction de coût et les valeurs initiales de ces paramètres dans l'optimisation. Sans compter que, à cause de l'optimisation, le système peut être soumis à certaines instabilités qui peuvent entraîner le non-respect des limites. De plus, ni l'approche classique ni l'approche adaptative ne tiennent compte de la perception humaine de mouvement par les modèles du système vestibulaire.

La commande optimale se présente alors comme une alternative qui permet d'intégrer le système humain de perception de mouvement, ce qui permet de réduire l'erreur entre la sensation de mouvement dans le simulateur et dans le véhicule grâce à une optimisation hors ligne. Toutefois, elle n'est pas très efficace en tant que MCA du fait que la référence est fondée sur un bruit blanc qui ne correspond pas toujours au profil d'accélération du véhicule [Nahon *et al.*, 1992]. Avec les expérimentations que nous avons réalisées en simulation, nous avons constaté que le processus d'optimisation est également difficile à paramétrer et en outre, les contraintes des états du système ne sont ajustées que dans les poids de pondération de la fonction de coût. Cela signifie que le système ne restreint pas fortement les limites physiques de la plateforme dans toutes les situations de conduite.

Il est bien entendu que l'un des plus grands problèmes de la stratégie adaptative et optimale est que le respect des contraintes n'est pas assuré dans la conception du contrôle. La stratégie classique empêche la plateforme de sortir de son enveloppe de travail, cependant, les performances du système sont réduites si le signal de référence ne correspond pas à celui choisi pour concevoir les filtres. La commande prédictive apparaît alors comme une méthode de contrôle idéale pour notre problématique, car elle permet à la fois de respecter les contraintes et de maximiser la performance du système.

Au début, le MPC a été utilisé pour sa grande capacité à respecter les restrictions [Dagdelen *et al.*, 2004]. Plus tard, étant donné la structure de ce contrôle, le système vestibulaire a été inclus afin de minimiser l'erreur de perception entre le modèle du véhicule et le simulateur de conduite. Cela nous amène à penser que la meilleure stratégie de contrôle pour les simulateurs dynamiques à hautes performances est la stratégie de contrôle prédictif, car elle maximise l'espace de travail de la plateforme sans pour autant dépasser ses limites physiques et augmente la perception du

mouvement.

Afin de valider cette hypothèse, nous avons comparé en simulation, les stratégies classique, adaptative, optimale et prédictive. Aucune des stratégies n'a été modifiée, c'est-à-dire qu'elles conservent les paramètres indiqués tout au long de ce chapitre. En revanche, nous avons généré un scénario de conduite autonome de 100 secondes. Dans celui-ci, le véhicule autonome devait suivre la trajectoire représentée par la ligne en pointillé dans la figure 36. Le terrain et le modèle autonome ont été générés par le logiciel de simulation SCANeR Studio.

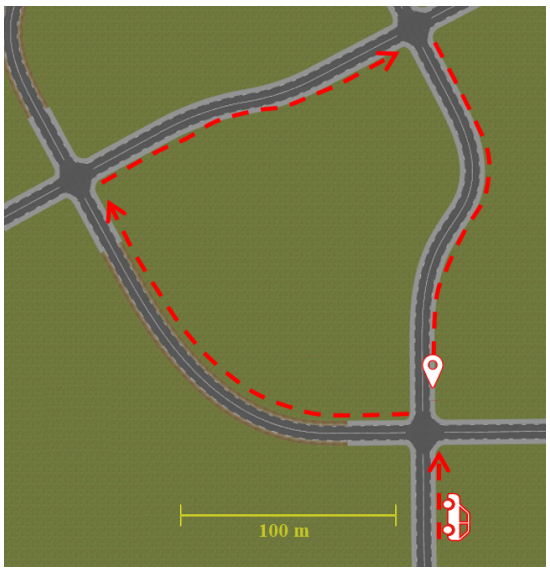


FIGURE 36 – Terrain et trajectoire utilisés pour l'essai de simulation de conduite autonome.

La figure 37 montre la force spécifique produite selon l'axe x par les quatre stratégies. Bien que la stratégie classique essaie de restaurer la force spécifique, sa performance est bien inférieure à celle de la stratégie adaptative ou prédictive. Dans l'intervalle 45-50 secondes, nous observons que la stratégie classique provoque une impulsion importante et de signe contraire à la référence, ce qui génère des faux indices au conducteur et favorise le mal du simulateur. Outre la faible amplitude, la commande classique n'utilise que la moitié de l'espace de travail, comme l'indique la figure 38.

Par rapport à la stratégie classique, la stratégie adaptative génère plus d'amplitude de force spécifique, en particulier pour les accélérations à haute fréquence, par exemple, dans l'intervalle de 10 à 20 secondes, nous observons que la sortie dépasse le signal de référence. Certes, une accélération supérieure peut affecter négativement la perception de mouvement du conducteur dans le simulateur. D'après la courbe de déplacement, cette commande ne maximise pas l'espace de travail par rapport à la prédictive ou l'optimale.

Concernant la stratégie optimale, nous observons une mauvaise suivie de trajectoire de la force spécifique, tant pour les accélérations à haute fréquence que pour les accélérations à basse fréquence.

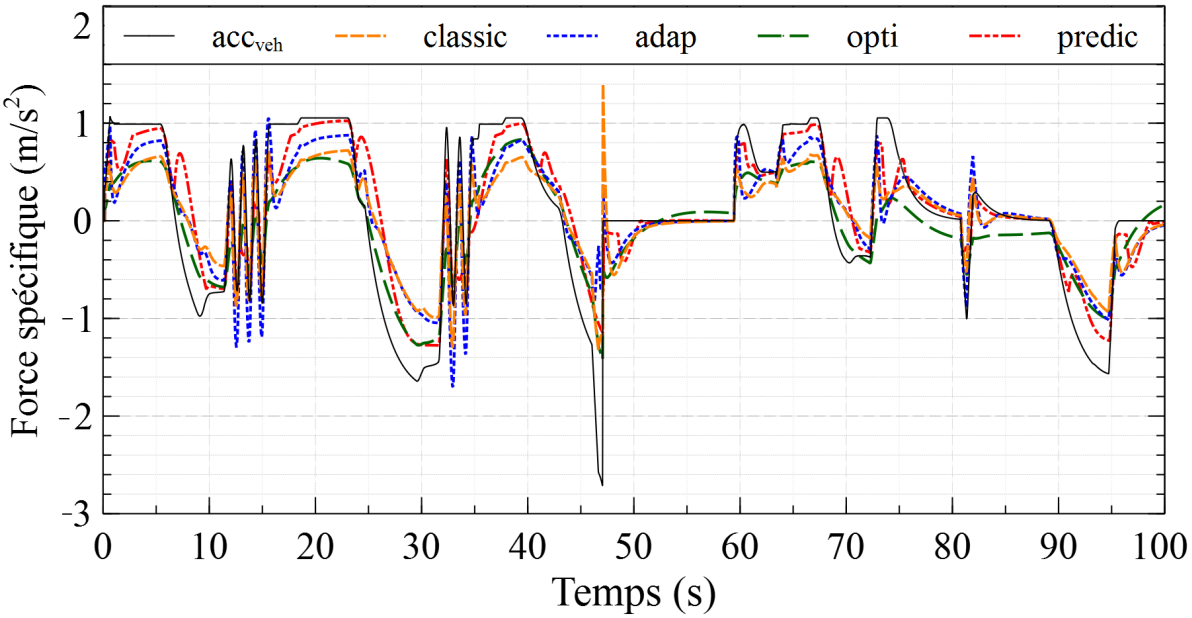


FIGURE 37 – Restitution de la force spécifique pour chacune des stratégies de contrôle.

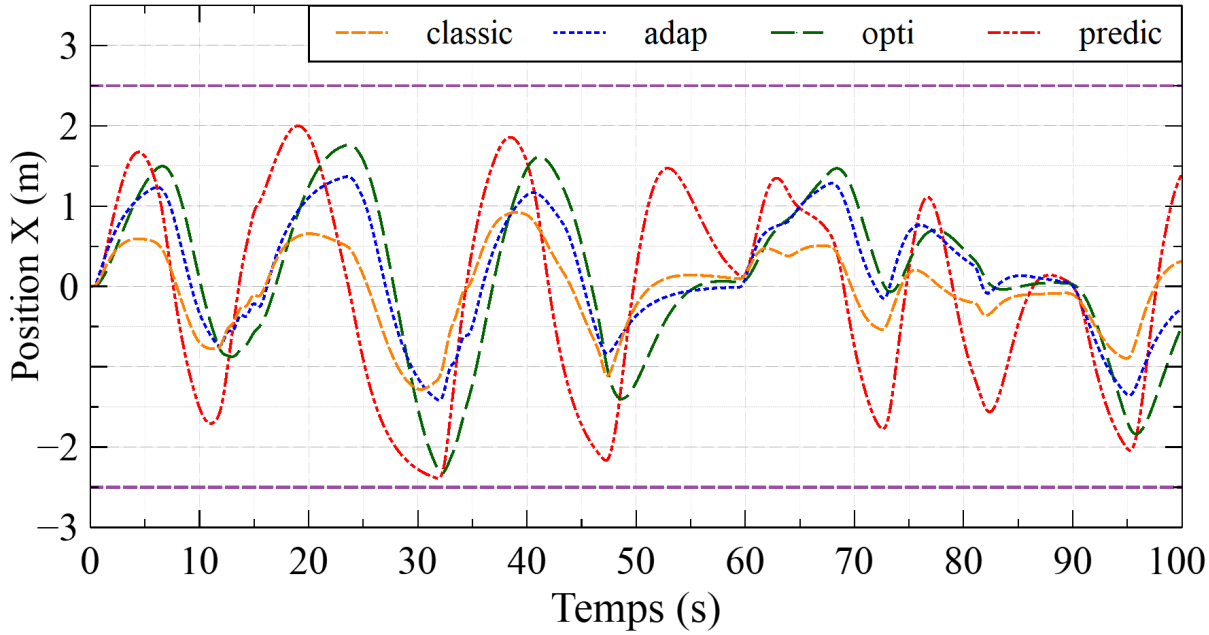


FIGURE 38 – Déplacement de la plateforme selon l'axe x.

En comparaison avec les autres commandes, la stratégie prédictive est celle qui réussit le mieux à suivre la trajectoire de la force spécifique et qui maximise l'espace de travail sans dépasser, à aucun moment, les limites du simulateur. Cependant, nous pouvons observer de petits retards lorsque qu'il y a des freinages, c'est-à-dire lorsque l'accélération passe de positive à négative. Ces retards sont causés par la limite de vitesse de l'angle d'inclinaison utilisé dans la technique de *tilt coordination*. Bien entendu, ils peuvent être évités en modifiant les paramètres de la fonction de

coût de cette stratégie.

Le tableau 7 présente un résumé des avantages et des inconvénients pour chacune des stratégies.

Stratégie	Avantages	Inconvénients
Classique	- Relativement facile - La plus utilisée	- Les paramètres des filtres sont fixés au pire cas de conduite - N'optimise pas l'espace de travail de la plateforme - N'assure pas le respect des contraintes
Adaptative	- L'identification des paramètres est fait en temps réel - L'adaptation en ligne fait une meilleure exploitation de l'espace de travail	- Procédure de réglage compliquée - N'inclut pas les contraintes dans la fonction de coût
Optimale	- Utilisation des modèles du système vestibulaire - Optimisation	- Paramétrage fait hors simulation et basé sur un signal de bruit
Prédictive	- Utilisation des modèles du système vestibulaire - Optimisation explicite et implicite - Espace de travail maximisé - Gestion des contraintes des états dans l'optimisation - Prédiction de la trajectoire (anticipation)	- Difficulté à assurer la faisabilité et la stabilité en bouclée fermée - Temps de calcul important

TABLEAU 7 – Avantages et inconvénients des différents stratégies : classique, adaptative, optimale et prédictive.

Dans cette étude, nous n'avons pas seulement comparé la stratégie classique à la stratégie prédictive, mais aussi à la stratégie adaptative et à la stratégie optimale. Les résultats de la simulation confirment ceux de la littérature. Nous choisirons la stratégie prédictive en tant que MCA pour le contrôle des simulateurs de conduite à hautes performances, car elle permet d'améliorer la perception de mouvement dans le simulateur grâce à l'inclusion du système vestibulaire, de produire un contrôleur optimal et de respecter, pour tous les types d'entrée, les restrictions physiques et de perception.

3.6.1 Axes d'amélioration de la commande choisie

L'objectif de cette partie est d'identifier les axes d'amélioration de la stratégie MPC dans le cadre des simulateurs à hautes performances et pour la validation du véhicule autonome. Nous faisons un récapitulatif des travaux qui, à notre connaissance, ont été développés pour utiliser la commande prédictive dans la restitution du mouvement.

La commande prédictive a été proposée pour la première fois pour le simulateur ULTIMATE de Renault par [Dagdelen *et al.*, 2004]. Elle prenait en compte le modèle dynamique, l'ensemble des contraintes physiques et un seuil de perception de l'accélération. Le même auteur a proposé en 2009, un MPC avec un système linéaire discret qui outre le modèle dynamique de la plateforme, contient le modèle de perception du système vestibulaire soumis à des contraintes convexes sur le contrôle et les états. Ce contrôle essaie de suivre la trajectoire du véhicule, mais lorsque la plateforme arrive à ses limites, le MPC est désactivé et l'accélération est fixée, selon une loi linéaire, en-dessous des seuils de perception. De plus, il regroupe en parallèle l'effet du *whashout* et du *tilt coordination* lorsque l'accélération du véhicule est inférieure au seuil de détection [Dagdelen *et al.*, 2009]. [Correia Augusto & Leal Loureiro, 2009] ont développé un contrôle plateforme pour le simulateur Chalmers basé sur la stratégie MPC afin que les limitations de l'espace de travail soient respectées et les signaux de perception soient mieux reproduits. Dans leurs travaux, le modèle du véhicule est basé sur le modèle d'une bicyclette.

[Baseggio *et al.*, 2011] appliquent le MPC à un simulateur à 6 DDL en utilisant le système vestibulaire et le système dynamique. Comparé au classique, le MPC est plus performant dans l'exploitation de l'espace de travail et fournit moins de faux indices. Le *washout* se fait en pondérant la position dans la fonction objective. [Beghi *et al.*, 2012] utilisent le MPC pour traiter la technique de *tilt coordination* et les contraintes de l'espace de travail dans une procédure d'optimisation facilement ajustable et robuste sans la mise en place de filtres. [Garrett & Best, 2013] présente un algorithme basé sur le MPC pour une plateforme Stewart à 6 DDL. Leur proposition, contrairement aux autres algorithmes, utilise les positions et les vitesses des actionneurs comme contraintes.

Dans la plupart des MPC, la référence future du système reste constante dans l'horizon de prédiction, car la présence d'un humain dans la boucle introduit des comportements potentiellement imprévisibles. Cependant, lorsque les scénarios sont prédéfinis, la présence de signaux répétitifs aide à définir des trajectoires de référence appropriées. Ainsi, [Bruschetta *et al.*, 2016] appliquent un MPC qui peut varier en temps réel d'une référence prédite selon le comportement du conducteur, à une référence constante. La première consiste en une stratégie *Look-Ahead* (LA), c'est-à-dire un horizon de prédiction suffisamment long avec une référence variable et connue. La décision de changer d'une référence à l'autre dépend de l'erreur entre la référence prédéfinie et la référence actuelle du système. Ils concluent qu'une stratégie LA est capable d'optimiser l'utilisation de l'espace de travail en fonction des événements à venir [Maran *et al.*, 2016].

[Fang & Kemeny, 2012a] proposent une condition de stabilité limite dans le MPC qui correspond à un ensemble invariant dans l'espace de travail basé sur l'algorithme explicite. Puis ils appliquent cette condition de stabilité à un MPC implicite [Fang & Kemeny, 2016]. Les auteurs ont proposé en 2017 un algorithme d'optimisation pour une simulateur à 9 DDL (intégrant une table de lacet en plus) [Fang *et al.*, 2017].

[Mohammadi *et al.*, 2016] proposent une méthode de pondération qui varie exponentiellement afin de stabiliser l'algorithme MPC avec un horizon de prédiction court et un ajustement optimisé des poids dans la fonction de coût. Les auteurs ont utilisé ensuite un algorithme génétique pour obtenir des horizons de contrôle et de prédiction plus performants. Cette méthode intègre la minimisation de plusieurs termes tels que l'erreur de sensation, le déplacement et le temps de calcul [Mohammadi *et al.*, 2018].

L'état de l'art précédent nous permet de constater que des efforts considérables ont été faits dans la réduction du temps de calcul, dans la stabilité de la commande prédictive et dans l'intégration du modèle de perception comme processus principal. C'est pourquoi les chapitres suivants proposeront quelques pistes d'amélioration, en se concentrant principalement sur ces trois aspects. Notre objectif est de rendre plus performante et adaptée la stratégie de contrôle MPC pour les simulateurs à hautes capacités dynamiques qui seront utilisés pour la validation des prestations ADAS et du véhicule autonome.

Chapitre 4

Optimisation de la commande prédictive par réseaux de neurones récurrents

> « Every problem is an optimization problem. »
>
> Stephen Boyd

Dans le chapitre précédent, nous avons montré que l'algorithme par commande prédictive (ou MPC) est l'algorithme le plus pertinent pour les nouveaux simulateurs à hautes performances. De plus, le MPC facilite la perception du mouvement des conducteurs lors de tests de validation de nouvelles technologies de conduite ou de la conduite autonome, car il permet une meilleure utilisation de l'espace de travail et minimise l'erreur de perception du mouvement humain. Cependant, tout comme les autres stratégies de restitution de mouvement, la commande prédictive possède des caractéristiques qui doivent être traitées avant de la mettre en œuvre dans les simulateurs en temps réel.

Nous étudions, dans ce chapitre, un élément important de cette stratégie de contrôle : l'optimisation. Notre intérêt ne se réduit pas seulement à expliquer les techniques utilisées à cette fin, mais à montrer une méthode qui peut améliorer la praticité et la rapidité de l'optimisation dans le MPC.

La première partie du chapitre présente les critères nécessaires pour résoudre le problème d'optimisation et garantir la stabilité en boucle fermée du système. Puis, une nouvelle approche d'optimisation nommée « réseaux des neurones récurrents » est présentée. Pour cette dernière, nous décrivons précisément la méthode proposée et les différentes approches afin de trouver celle qui s'accommode le mieux aux algorithmes de contrôle de mouvement. Nous évaluons chaque approche en simulation afin de monter ses performances, ses avantages, ainsi que les problématiques rencontrées. Enfin, dans la dernière partie nous discutons les résultats obtenus et

les perspectives de cette approche dans les simulateurs de conduite dynamique.

4.1 Faisabilité dans la commande prédictive

Dans le MPC, un problème d'optimisation en boucle ouverte est résolu à chaque pas d'échantillonnage. Cette solution doit être garantie tout au long de l'horizon de prédiction, sinon la loi de commande appliquée peut déstabiliser les états du système et endommager le simulateur ou, pire encore, le conducteur. La solution du problème d'optimisation est exprimée en variables de décision ΔU indépendantes. Elles sont appliquées au modèle dynamique qui représente le comportement réel du système et développé dans la partie 3.4.1.

Dans les techniques de régulation optimales à horizon infini (LQR pour *Linear Quadratic Regulator)*, la stabilité en boucle fermée est facile à atteindre et à maintenir pendant la simulation. En effet, si le problème d'optimisation est faisable, l'on peut prouver que la fonction de coût à l'horizon infini est monotone et décroissante, par conséquent, elle est interprétée en tant que fonction de Liapounov et garantit donc la stabilité en boucle fermée du système. En revanche, l'analyse de stabilité des contrôleurs à horizon fini comme le MPC est une tâche difficile qui ne garantit pas la stabilité dans tous les scénarios de conduite.

Certes, lorsqu'il n'y a que des contraintes dans l'entrée du système, une simple saturation peut assurer la solution du problème d'optimisation. Toutefois, lorsqu'il y a des contraintes d'état, la régulation du système devient plus difficile, car le contrôleur doit trouver un compromis entre la performance souhaitée du système et les contraintes. Ainsi, même si le système est stable en boucle ouverte, l'infaisabilité du problème d'optimisation peut provoquer une instabilité du système en boucle fermée. C'est pourquoi la stabilité doit être abordée dans la formulation du problème d'optimisation du MPC. Nous évaluerons les conditions qui garantissent la stabilité par la faisabilité du problème d'optimisation sous contraintes, puis nous présenterons notre proposition de la méthode d'optimisation en utilisant ces conditions.

Afin de mettre en œuvre différents tests de simulation, dans cette partie, nous implémenterons la stratégie MPC telle que présentée dans la partie 3.4. Pour ce faire, nous utiliserons le modèle de la force spécifique en intégrant le système vestibulaire de la plateforme. Ce système a été décrit dans la partie 3.4.1 et est représenté par le système dans l'espace d'état de l'équation (20). Il faut signaler qu'afin d'obtenir les accélérations linéaires et angulaires pour ce système, une optimisation à 2 DDL est utilisée.

4.1.1 Paramétrage

Il est évident que la solution du problème d'optimisation dépend directement de sa formulation. C'est pourquoi, nous analysons les paramètres qui interviennent dans la mise en œuvre du MPC en temps réel, qui augmentent la fidélité de mouvements et qui maximisent l'espace de travail de la plateforme. Le réglage de l'algorithme consiste à choisir les poids de pondération, les horizons de prédiction et de contrôle, le temps d'échantillonnage, la référence future du système et les facteurs d'échelle pour obtenir les performances souhaitées du système.

Horizons de prédiction et contrôle

La commande prédictive est un type de contrôle optimal qui n'utilise pas un $N_p = N_u = \infty$ dans la fonction de coût (25), puisqu'il s'agit d'une stratégie à horizon fini. Par conséquent, afin d'obtenir une loi de contrôle MPC avec les mêmes performances que la stratégie optimale, il est nécessaire de définir un N_u et un N_p suffisamment long [Rossiter, 2003]. Pour l'horizon de contrôle, nous avons choisi $N_u = 3$ parce qu'à partir de cette valeur, la performance de la loi de contrôle ne varie pas. Au contraire, une valeur plus élevée consomme plus de temps de calcul.

Lorsque la plateforme est proche de ses limites, celle-ci doit s'arrêter avec une valeur inférieure au seuil du mouvement humain afin de respecter l'enveloppe de travail. Ce seuil est conçu pour que le conducteur ne perçoive pas de faux indices de mouvement. Les détails de ce choix et de cette technique ont été précisés dans la partie 2.3.5.

D'après la littérature [Fang *et al.*, 2017], un temps de prédiction entre 7 et 14 secondes est nécessaire pour que le simulateur freine avant d'arriver à ses limites (tableau 2). Ainsi, si nous considérons un temps de prédiction minimum de 7 secondes et une fréquence de contrôle nécessaire de 100 Hz [1], l'horizon de prédiction devient $N_p = 700$. Cette période de prédiction est appliquée au système dans la prédiction des états (partie 3.4.2) et aux contraintes (22), ce qui représente un coût de calcul important dans l'optimisation. Ce coût empêche de résoudre le problème d'optimisation en temps réel. Afin d'atteindre ce temps de prédiction en temps réel, nous utilisons une fréquence d'échantillonnage ts non-uniforme le long de l'horizon de prédiction.

La méthode consiste à appliquer deux fréquences différentes lors de la discrétisation du système et des contraintes. La première période d'échantillonnage ts_1 est la même que la fréquence de contrôle (0,01s) et sera fixée dans les 10 premiers pas de temps. Pour la seconde période d'échantillonnage ts_2, nous avons fait un test en simulation pour choisir le meilleur compromis entre la performance et le temps de calcul. Ainsi, nous avons fait varier le nombre de pas dans l'horizon de prédiction N_p et ts_2 afin d'obtenir le temps de prédiction requis de 7,1 secondes qui est dans la

1. La fréquence minimale requise pour une simulation en temps réel est de 100 Hz.

gamme des valeurs nécessaires pour que le simulateur freine avant d'arriver à ses limites. Il faut signaler que si ts_2 augmente, la valeur de N_p diminue ainsi que le temps de calcul. La figure 39 illustre la distribution de ts_1 et ts_2 pour un horizon de prédiction donné et le tableau 8 résume les différentes conditions de simulation.

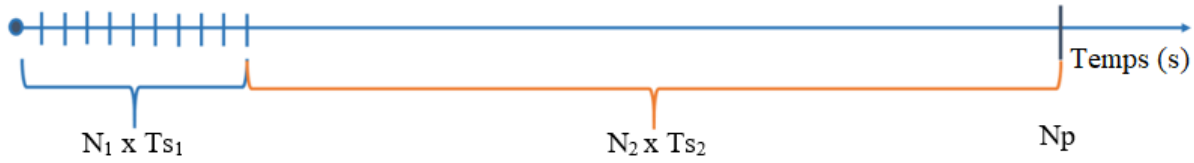


FIGURE 39 – Temps d'échantillonnage non uniforme.

Il faut signaler que cette méthode a été inspirée des travaux effectués par [Maran *et al.*, 2015], dans lequel un temps d'échantillonnage différent est appliqué au problème d'optimisation dans les variables de décision ΔU afin de réduire le temps de calcul de l'optimisation. En revanche, dans notre méthode, les temps d'échantillonnage sont appliqués dans la prédiction des états et les contraintes du système.

ts_1*(N_1)	ts_2*(N_2)	**Dimension de N_p**	**Temps de prédiction (s)**
0,01*(10)	0,5*(140)	150	7,1
0,01*(10)	0,1*(70)	80	7,1
0,01*(10)	0,15*(47)	57	7,1
0,01*(10)	0,2*(35)	35	7,1

TABLEAU 8 – Fréquences d'échantillonnage différentes pour un même temps de prédiction.

Afin de simuler les différentes ts_2 et horizons de prédiction N_p, une accélération de référence acc_{veh} avec des composantes à hautes et à basses fréquences a été choisie comme indiqué dans la figure 40. Cette figure montre les réponses du système en boucle fermée en utilisant différentes fréquences dans l'horizon de prédiction non uniforme. Nous observons que plus la valeur de ts_2 est faible, plus le suivi du signal d'entrée est important. Cependant, un ts_2 petit génère un coût de calcul élevé à cause de la valeur de N_p. En tenant compte du temps de calcul et de la performance de la stratégie de contrôle, nous avons choisi un $ts_2 = 0,1$ avec $N_p = 80$, car il est suffisamment petit pour permettre une mise en œuvre du MPC en temps réel avec un bon suivi de la trajectoire. Nous constatons également que dans aucune des configurations la référence n'est entièrement suivie, à cause des fortes restrictions imposées aux états.

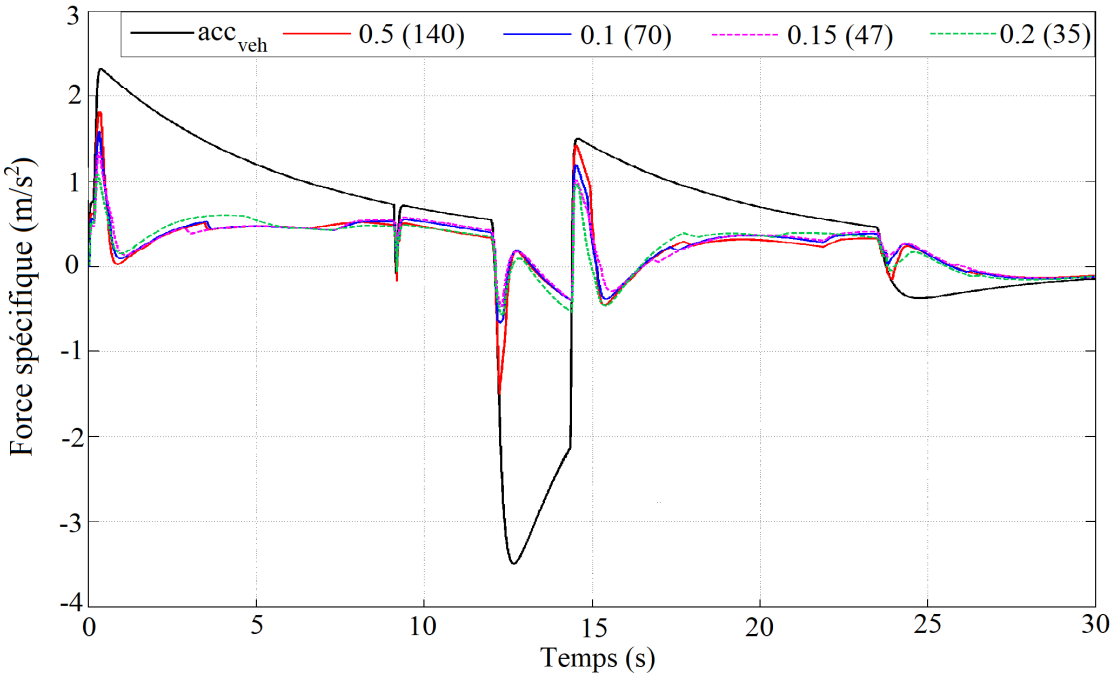


FIGURE 40 – Réponses en boucle fermée avec différentes fréquences dans l'horizon de prédiction non uniforme (les valeurs correspondent à $t_{s2} * N_2$ dans le tableau 8).

Réglage des poids

Bien que nous ayons déjà défini l'horizon de contrôle et de prédiction, ces deux conditions ne sont pas suffisantes pour assurer la faisabilité du problème de minimisation de l'équation (25) en temps réel. En fait, il reste à choisir les poids des matrices de pondération Q_δ, Q_λ et Q_q qui définiront la loi de contrôle et par conséquent, le comportement du système en boucle fermée.

La plupart des auteurs choisissent les matrices de pondération de manière intuitive aux moyens d'essais et d'erreurs [Beghi *et al.*, 2012; Correia Augusto & Leal Loureiro, 2009; Garrett & Best, 2013]. Pour ce faire, des valeurs constantes ou des séquences exponentielles sont utilisées [Mohammadi *et al.*, 2016]. Ainsi, si les poids de pondération de l'erreur ont une valeur élevée au début de l'horizon de prédiction, le contrôle sera strict. Au contraire, si les poids sont importants à la fin de l'horizon de prédiction, le contrôle sera plus fluide [Bordons & Camacho, 2007]. D'autres auteurs utilisent des méthodes d'optimisation pour déterminer les poids qui donnent une meilleure performance du système selon une référence donnée [Mohammadi *et al.*, 2018]. Bien que cette dernière offre de nombreux avantages en ce qui concerne les performances du système, notre objectif n'est pas de fixer une référence future dans la prédiction puisque le comportement du conducteur est inconnu. C'est pourquoi nous avons décidé de choisir les pondérations en utilisant la méthode de l'essai-erreur.

Un autre paramètre de réglage important du MPC est la mise à l'échelle du signal d'entrée. Comme nous l'avons mentionné dans la section 3.1, une reproduction 1 :

1 n'est pas toujours possible étant donné l'espace de travail limité du simulateur. Cependant, comme il est dit dans [Bruschetta *et al.*, 2016], le choix de la valeur de mise à l'échelle est un sujet de débat important dans la littérature, principalement en raison de la présence du conducteur dans la boucle de contrôle. En outre, la mise à l'échelle dépend fortement de l'utilisation du simulateur : soit pour tester une nouvelle technologie, soit pour un scénario de conduite spécifique. Par conséquent, nous nous sommes basés sur l'étude réalisée par [Berthoz *et al.*, 2013] afin de choisir le facteur d'échelle du signal de référence. Dans celle-ci, 65 participants indiquent une préférence pour les facteurs d'échelle inférieurs à l'unité et compris dans une large gamme de valeurs acceptables entre 0,4 et 0,75. En dehors de ces valeurs, les performances de conduite se dégradent considérablement.

4.1.2 Méthodes pour assurer la faisabilité

Le choix des horizons de contrôle et de prédiction, les poids de pondération et la mise à l'échelle du système ne sont pas suffisants pour assurer la faisabilité du problème d'optimisation dans la commande prédictive. Ainsi, dans cette partie nous indiquerons d'autres méthodes qui visent à favoriser la faisabilité du problème d'optimisation. Ces méthodes interviennent dans le processus d'optimisation et ont une relation étroite avec le paramétrage du MPC.

État de l'art

Comme nous l'avons vu précédemment, le MPC a été appliqué pour la première fois par [Dagdelen *et al.*, 2004] dans le simulateur ULTIMATE. Dans ces travaux, ils utilisent une procédure explicite proposée dans [Gutman & Cwikel, 1987] afin de trouver un ensemble invariant d'états initiaux à partir desquels le système atteint un point d'équilibre dans un horizon de prédiction fini. Ce problème utilise une condition de contrôle de type tout ou rien (bang-bang), dans lequel le système en boucle fermée pourra atteindre la stabilité des états à l'intérieur de ses limites. Dans [Fang & Kemeny, 2012a], une stratégie explicite pour une optimisation à 1 DDL est mise en œuvre, les auteurs présentent différentes techniques de stabilité classiques dans le MPC, mais ceux-ci ne sont pas comparés objectivement. Ils proposent une nouvelle condition de stabilité fondée sur une loi de freinage. Puis [Fang & Kemeny, 2016] mettent en œuvre la même condition de stabilité dans un problème d'optimisation à 2 DDL en utilisant une stratégie implicite.

[Garrett & Best, 2013] utilisent une méthode à barrière [1] pour résoudre le problème d'optimisation. Dans cette méthode, les contraintes d'inégalité, qui initialement sont traitées comme des contraintes dures, sont transformées en contraintes souples

1. Les méthodes de pénalité intérieures ou à barrières utilisent une fonction de pénalité qui forme une barrière infinie le long de la frontière du domaine réalisable, garantissant la faisabilité dans l'ensemble de l'optimisation [Sareni, 1999].

de la fonction de coût et sont pondérées par un poids p. Pour garantir une solution à chaque période d'échantillonnage, l'algorithme d'optimisation commence par une grande valeur de p qui restreint autant que possible les contraintes du système. Puis, la valeur p est réduite d'un facteur de 10 et le processus d'optimisation continue jusqu'à ce que la simulation soit terminée. [Maran *et al.*, 2016] proposent une stratégie qui augmente l'horizon de prédiction sans augmenter le coût de calcul pour atteindre un horizon de prédiction de 8 secondes permettant d'assurer la faisabilité des contraintes. D'autres auteurs qui utilisent le MPC pour la restitution de mouvement ne mettent pas en évidence les conditions pour lesquelles le problème d'optimisation est faisable à chaque pas de temps [Augusto, 2009; Baseggio *et al.*, 2011; Beghi *et al.*, 2012]. Ces algorithmes choisissent des solutions fondées uniquement sur la pondération des poids dans la fonction de coût.

De cet état de l'art, nous pouvons conclure qu'il existe de nombreuses techniques pour résoudre les problèmes d'infaisabilité du MPC. Cependant, à notre connaissance, aucun auteur n'a comparé numériquement les différentes stratégies de faisabilité. C'est pourquoi, dans cette partie, nous évaluerons objectivement les méthodes les plus utilisées pour assurer une loi de contrôle qui stabilise le système tout au long de l'horizon de prédiction.

Comparaison des stratégies

Comme indiqué précédemment, cette partie vise à expliquer certaines des stratégies appliquées au MPC qui renforcent la faisabilité et éventuellement assurent la stabilité du système en boucle fermée. Parmi toutes les conditions de stabilité proposées dans la littérature, nous avons choisi les quatre les plus utilisées dans le cadre de la restitution de mouvement :

- **Contrainte de l'état terminal égal à zéro :** c'est la première méthode qui garantit la stabilité dans le MPC non linéaire [Kwon & Pearson, 1977] à cause des contraintes dans les états. Avec cette condition, les derniers états prédits sont forcés à être égaux à zéro en tant que contraintes dures, c'est-à-dire qu'en 7,1 secondes (temps de prédiction choisi), la position, la vitesse et l'accélération du simulateur doivent avoir une valeur nulle.

$$x_{sim}(t + N_p) = 0 \tag{28}$$

- **Ensemble invariant :** en fonction du temps de prédiction, la condition précédente peut limiter les mouvements du simulateur. Toutefois, elle peut être relaxée en utilisant un ensemble de contraintes pour le problème d'optimisation. En effet, il existe un ensemble possible d'états initiaux dont les contraintes sont respectées le long de la trajectoire du système. L'ensemble est polyédrique, c'est-à-dire défini par les inégalités linéaires du système (22).

$$X_f = \begin{cases} p_k \leq p_{max} - \frac{v_k^2}{2a_s}, & v_k \geq 0 \\ p_k \leq \frac{v_k^2}{2a_s} - p_{max}, & v_k < 0 \end{cases} \tag{29}$$

Où v_k, p_k et a_s sont respectivement les limites en vitesse, position et accélération. La même relation est mise en œuvre pour les contraintes de rotation. La région X_F vise à changer les contraintes d'états à chaque pas de temps en suivant une condition de *washout* proposée par [Dagdelen *et al.*, 2004].

- **Loi de freinage :** Cette condition de stabilité a été proposée par [Fang & Kemeny, 2012a]. Ils proposent une loi de contrôle stable le long de l'horizon de prédiction à l'aide d'une loi de freinage qui s'applique une fois que la plateforme s'approche de ses limites. Cette loi peut être comparée à un filtre adaptatif qui permet à la plateforme de revenir à sa position neutre avec un certain seuil d'accélération. La condition se résume comme suit :

$$p_{min} \leq p_k + c_v T v_k + c_a \frac{T^2 ulin_k}{2} \leq p_{max} \tag{30}$$

$$\theta_{min} \leq p_k + c_v T \dot{\theta} + c_a \frac{T^2 urot_k}{2} \leq \theta_{max} \tag{31}$$

L'équation cinématique de l'accélération ou décélération constante, montre que la position finale p_f en fonction du temps t, de la position initiale p_i, de la vitesse initiale v_i et de l'accélération a est représentée par :

$$p_f = p_i + v_i t + \frac{1}{2} a t^2 \tag{32}$$

Nous pouvons observer que cette fonction est presque identique à la condition de stabilité proposée par [Fang & Kemeny, 2012a], à la différence que les équations (30) et (31), trois autres coefficients sont ajoutés : c_v, c_a et T. Ces coefficients agissent sur la performance de la loi de contrôle en matière de suivi de la trajectoire de référence. Le système représente un système du deuxième ordre, avec la pulsation propre $\omega_0 = \sqrt{\frac{2}{c_a T^2}}$ et le coefficient d'amortissement $\xi = \frac{c_v}{c_a T \omega_0} = \frac{c_v}{c_a}\sqrt{\frac{c_a}{2}}$. Le temps T est considéré également comme un paramètre de conception et se réfère au temps que prend la plateforme pour freiner complètement. Les valeurs c_v, c_a et T servent à contrôler le comportement du système en veillant à ce qu'il ne dépasse pas ses limites physiques et perceptives. Le choix de ces paramètres du filtre est un compromis entre la fidélité de restitution et les limites physiques de la plateforme. Nous modifierons uniquement les paramètres T et C_v, car ils affectent directement la fréquence et l'amortissement, réduisant ainsi le nombre de variables à pondérer.

Comparé aux filtres passe-haut implémentés dans les algorithmes classiques de MCA, cette représentation du système intègre les états initiaux x_0 et v_0. Ils constituent les dernières valeurs des états du simulateur à partir desquels le processus de freinage commence. La différence entre cette technique et l'ensemble invariant est, qu'ici, les contraintes sont appliquées directement aux variables de décision.

- **Matrice de pondération terminale « P » :** afin d'expliquer en détail cette condition, nous considérons le problème de contrôle optimal avec un horizon

de prédiction infini sans contraintes suivant :

$$J_k = \frac{1}{2}\sum_{j=1}^{N_p-1} x_k^T Q x_k + \Delta u_k R \Delta u_k + \frac{1}{2} x_{Np}^T P x_{Np} \tag{33}$$

Ici, P est la solution positive et unique de l'équation algébrique discrète de Riccati :

$$P = A^T P - K^T (R + B^T P B) K + Q \tag{34}$$

$K = (R + B^T P B)^{-1} B^T P A$ est le gain du contrôle. L'utilisation de P en tant que matrice de pondération de l'état terminal pour l'horizon de prédiction à temps fini implique que les fonctions de coût sont de type Liapounov en garantissant la stabilité nominale en boucle fermée. Ici, le contrôleur à horizon glissant avec un horizon de contrôle et de prédiction suffisamment grand se comporte comme un contrôle optimal LQR [Bordons & Camacho, 2007]. En présence de contraintes, le MPC devient non linéaire et le contrôleur $\Delta U = Kx$ ne satisfait pas nécessairement les limites de la plateforme. Cependant, bien que le système soit non-linéaire, la preuve de stabilité de Liapounov s'applique en présence de contraintes si la solution est réalisable sur tout l'horizon de prédiction. Dans ce cas, le choix des matrices de pondération Q et R sont celles qui détermineront les performances du système en boucle fermée.

Afin de comparer ces quatre conditions de stabilité et de faisabilité de la commande prédictive, nous avons développé quatre MCA différentes fondées sur la commande prédictive sur le logiciel MATLAB/Simulink [1].

Simulation

Pour générer différents scénarios de conduite, c'est-à-dire différents signaux de référence, nous avons utilisé le logiciel de simulation de conduite SCANeR studio. Nous avons choisi des manœuvres générales qui représentent la force spécifique longitudinale et latérale sur une large gamme de fréquences, c'est-à-dire, des signaux contenant des composantes à hautes et basses fréquences. Ces signaux ont été générés par un modèle de véhicule virtuel en mode autonome qui suivait la trajectoire de trois scénarios différents avec une durée de 50 secondes chacun. Ces signaux étaient la référence en accélération des quatre MCA. Chaque MCA avait une condition de faisabilité différente.

Le premier scénario Sce_1 est une simulation urbaine. Il consiste en une intersection à laquelle le véhicule doit s'arrêter puis tourner à gauche. Le deuxième scénario Sce_2 est une simulation routière : le véhicule autonome roule à vitesse constante en suivant une voiture, la dépasse puis se rabat à droite. Dans le dernier scénario Sce_3, la voiture suit une accélération constante de $2m/s^2$ pendant 10 secondes puis s'arrête complètement jusqu'à une vitesse nulle. Notre objectif est de comparer

1. https://fr.mathworks.com/products/simulink.html

chacune des quatre conditions de stabilité avec chacun des scénarios. Pour ce faire, nous comparons le nombre d'infaisabilités fournies par chaque stratégie pendant les 50 secondes de simulation.

Le problème d'optimisation a été résolu avec l'outil open-source qpOASES [Ferreau *et al.*, 2014], car celui-ci fournit le nombre d'infaisabilités dues aux contraintes. En outre, lorsqu'il n'y a pas de solution, l'outil fournit une solution approximative afin de ne pas arrêter la simulation.

Le tableau 9 indique le nombre d'infaisabilités pour les quatre stratégies présentées plus haut et pour chaque scénario décrit précédemment. Nous avons inclus une stratégie qui ne contient aucune des conditions de stabilité, nommée la stratégie « normale ».

Stratégies	**Infaisabilités**		
	$Sce1$	$Sce2$	$Sce3$
État terminal égal à zéro (S_1)	0	8	1
Ensemble invariant*	0	112	4
Ensemble invariant avec la matrice P (S_2)	0	17	3
Loi de freinage (S_3)	0	0	0
Normale*	12	1212	345
Normale avec la matrice P (S_4)	0	18	2

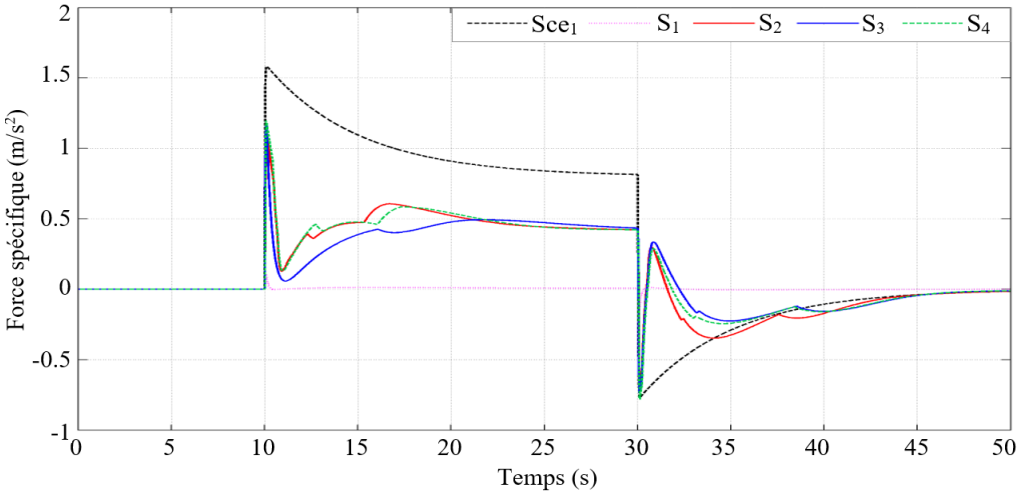
TABLEAU 9 – Nombre d'infaisabilités par stratégie et par scénario.

Dans le tableau 9, nous avons marqué les conditions « Ensemble invariant » et « Normale » d'un astérisque, indiquant que ces deux conditions ne seront pas comparées dans l'analyse des courbes, car dans Sce_2 la plateforme a été déstabilisée et a dépassé les limites en position et vitesse. Pour consolider les deux conditions, nous avons rajouté la condition de la matrice terminale P.

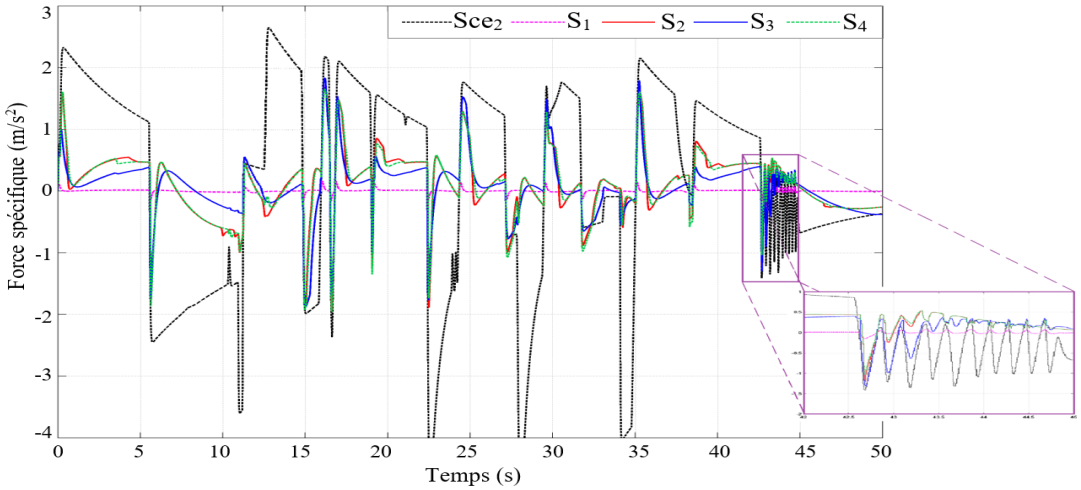
Les résultats indiquent l'importance d'ajouter ces conditions de faisabilité dans la commande prédictive. En effet, parmi toutes les stratégies évaluées, la stratégie « Normale » est la stratégie la plus indésirable, car la plateforme dépasse ses limites physiques à cause du grand nombre d'infaisabilités pendant les simulations.

Nous avons analysé les performances de ces stratégies par rapport à la minimisation de l'erreur de suivi de la trajectoire et par rapport au gain d'exploitation de l'espace de travail du simulateur. La figure 41 illustre le suivi de trajectoire des stratégies S_1, S_2, S_3 et S_4 dans Sce_1 (figure 41a), Sce_2 (figure 41b) et Sce_3 (figure 41c).

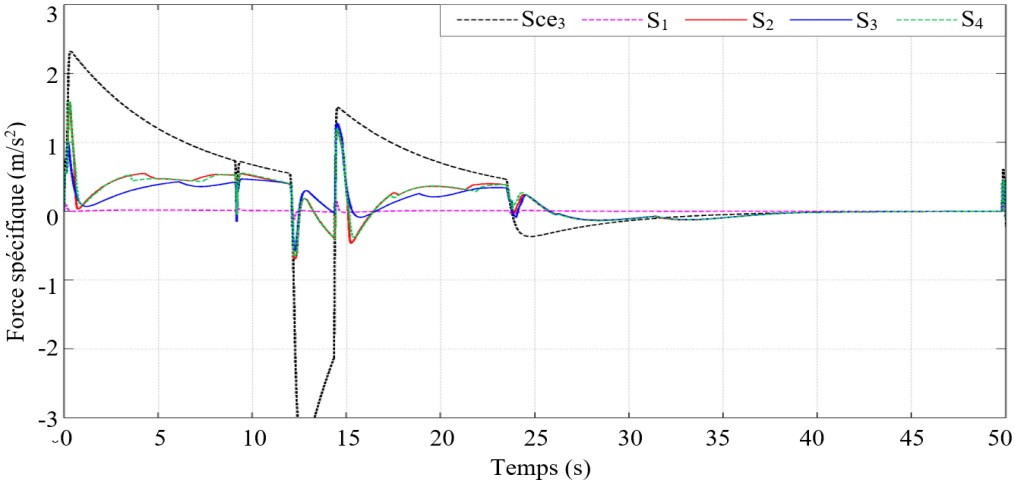
Concernant le nombre d'infaisabilités, l'ajout de la matrice P terminale pour les conditions invariantes (S_2) et les contraintes normales (S_4) augmente le nombre de solutions au problème d'optimisation, ce qui permet au logiciel qpOASES de donner une solution approximative. Cependant, aucune de ces stratégies n'assure



(a) Scénario 1



(b) Scénario 2



(c) Scénario 3

FIGURE 41 – Comparaison des sorties du simulateur avec les différentes stratégies dans les trois scénarios.

une solution sur tout l'horizon de prédiction, donc le système peut devenir instable en boucle fermée.

Les stratégies S_2 et S_4 ont pratiquement les mêmes performances puisqu'elles partagent la condition de la matrice terminale P. Cela indique que l'ensemble invariant présenté dans ces travaux n'apporte pas d'avantages si les états du système ont déjà été correctement pénalisés avec les matrices de pondération P à la fin de l'horizon de prédiction.

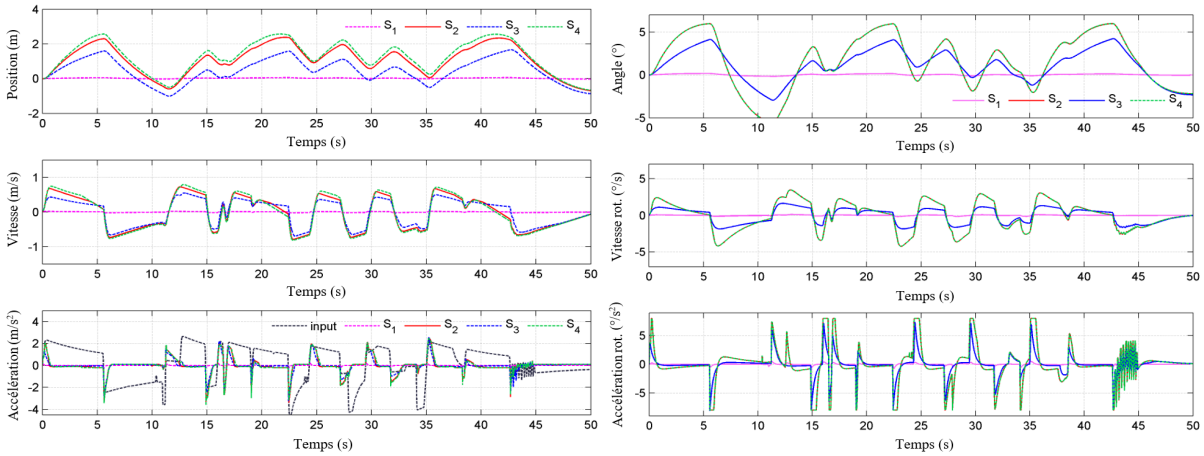
Quel que soit le scénario de conduite, la stratégie qui force les états à zéro (S_1) à la fin de l'horizon de prédiction entraîne une erreur non négligeable entre la référence acc_{veh} et la sortie du système S_1 dans la figure 41. L'erreur de suivi de trajectoire est due aux limites de la perception humaine qui interdisent de fortes variations de l'accélération. Avec cette condition, la faisabilité de l'algorithme n'est pas garantie, car la région des états admissibles du système est très restreinte à la fin de l'horizon de prédiction N_p.

La stratégie avec la loi de freinage (S_3) est la seule à trouver une solution dans tout l'horizon de prédiction pour les trois scénarios de conduite. Par conséquent, avec cette stratégie, la fonction de coût qui décrit le problème d'optimisation est une fonction de Liapounov qui assure la stabilité en boucle fermée du système.

Si nous prenons en compte les résultats du tableau 9 et le suivi de la trajectoire de la figure 41, nous constatons qu'un compromis doit être fait entre les conditions de faisabilité et la performance de la loi de contrôle. Afin d'éclaircir cette hypothèse, nous prenons comme exemple le scénario 2 : la figure 42 illustre les états longitudinaux et angulaires de la plateforme tels que la position, la vitesse et l'accélération du simulateur. Nous observons que la stratégie avec la loi de freinage ne maximise pas complètement l'espace de travail des rails (figure 42a) ou de l'hexapode (figure 42b). En outre, cette condition ne fournit pas la réponse la plus optimale pour le contrôle et le suivi de trajectoire (figure 41b). Cependant, c'est la seule stratégie qui assure la stabilité en boucle fermée en utilisant les critères de faisabilité. Cela dit, la stabilité du système dépend du choix des paramètres de la loi de freinage. Ces paramètres ont été choisis par la méthode d'essai-erreur : nous avons sélectionné les paramètres qui minimisent l'erreur de perception et qui génèrent une solution tout au long de l'horizon de prédiction.

4.1.3 Discussion

Dans cette partie, nous traitons les problèmes de faisabilité de la commande prédictive dans les algorithmes de restitution de mouvement. Pour cela, différentes alternatives ont été proposées. La première a été le paramétrage de la stratégie de contrôle. Une étude a été réalisée sur la valeur que devraient avoir l'horizon de contrôle, l'horizon de prédiction, ainsi que les matrices de poids de la fonction de coût. De nombreux auteurs de la littérature n'utilisent que ce paramétrage pour assurer la stabilité du système. Toutefois, la stabilité du système en boucle fermée



(a) Mouvements longitudinaux

(b) Mouvements angulaires

FIGURE 42 – Comparaison des mouvements : position, vitesse et accélération dans le scénario 2.

n'est pas garantie pour tous les scénarios de conduite. Par conséquent, les paramètres doivent être constamment modifiés. Cela peut devenir une tâche fastidieuse si seule la méthode d'essai et d'erreur est utilisée.

C'est pourquoi, dans la deuxième partie, nous avons évalué les différentes conditions utilisées dans la littérature pour valider la stabilité du système, en résolvant le problème de contrôle à chaque étape de l'échantillonnage. Pour ce faire, quatre techniques ont été comparées : les états égaux à zéro à la fin de l'horizon de prédiction, les ensembles invariants, la loi de freinage et la matrice terminale de pondération P inspirée de la commande optimale.

Les résultats de la simulation ont démontré qu'il est nécessaire de trouver un équilibre entre la maximisation de l'espace de travail du simulateur et la faisabilité du problème d'optimisation. Les différentes stratégies comparées dans ce chapitre montrent qu'il est recommandé, sinon nécessaire, de faire varier les contraintes dures des états à chaque étape du problème d'optimisation. Si cela n'est pas possible, il est au moins recommandé d'appliquer un poids fort à la fin de l'horizon de prédiction, c'est-à-dire, la matrice P. D'après les résultats, nous constatons que seule une condition de loi de freinage appliquée directement dans les variables de décision fournit les conditions nécessaires pour garantir la faisabilité et la stabilité en boucle fermée pour tous les scénarios de conduite. Il est important de noter que lorsque cette loi de freinage est utilisée, la performance des variables de contrôle peut être modifiée par les paramètres adaptatifs c_v, c_a et T contenus dans cette condition. Ces paramètres peuvent être adaptés afin d'obtenir un profit optimal de l'espace de travail et d'assurer la stabilité du système même dans les situations de conduite les plus critiques.

Après avoir vu les éléments nécessaires pour stabiliser notre système en boucle fermée, nous allons proposer dans la prochaine partie une nouvelle méthode d'optimisation qui pourrait remplacer l'outil qpOASES, afin de rendre le problème

d'optimisation plus facile à mettre en œuvre.

4.2 Application de l'optimisation avec des réseaux neuronaux récurrents

Comme nous l'avons présenté dans la partie 3.4.3, l'optimisation du MPC sera fondée sur la méthode implicite dont le problème (25) doit être résolu en temps réel à chaque pas d'échantillonnage. Dans les MCA basés sur le MPC, la dynamique du système varie rapidement et la solution numérique requise en temps réel est souvent considérée comme un obstacle pour les outils de résolution de problèmes d'optimisation. Cette solution au problème d'optimisation en temps réel a été largement étudiée [Goodwin *et al.*, 2006]. Les méthodes traditionnelles pour résoudre les problèmes de programmation quadratique impliquent normalement un processus itératif qui nécessite un temps de calcul limitant leur utilisation en temps réel.

La recherche de méthodes plus rapides, efficaces et robustes pour trouver une solution au problème d'optimisation fournit une approche à de nouvelles formes d'optimisation. Une alternative prometteuse consiste à utiliser les réseaux neuronaux récurrents artificiels (RNR) comme méthode de calcul de l'optimisation dynamique.

4.2.1 Introduction aux réseaux de neurones

L'unité centrale d'un réseau de neurones est le neurone lui-même, qui est semblable à un neurone biologique, mais qui en fait est un simple dispositif de calcul qui transforme la somme pondérée d'un vecteur en une réponse unique. La structure d'un neurone artificiel est définie par ses connexions. Cette structure est illustrée dans la figure 43.

Chaque neurone calcule la somme de toutes ses entrées et transfère la valeur résultante à travers une fonction d'activation. Pour comprendre le type de réseau qui sera utilisé pour résoudre le problème d'optimisation de notre système, il est nécessaire de comprendre deux concepts qui caractérisent le modèle neuronal : l'apprentissage et l'architecture du réseau.

Dans l'apprentissage, les poids des connexions entre les neurones sont modifiés selon une règle d'optimisation définie jusqu'à atteindre leur valeur finale. L'apprentissage d'un réseau de neurones peut être supervisé, par renforcement, ou non supervisé. Dans cette thèse, nous utilisons un réseau de neurones avec un apprentissage non supervisé : les données d'entrée ne sont ni classées ni étiquetées et permettent à l'algorithme d'agir sur ces informations sans guidage préalable.

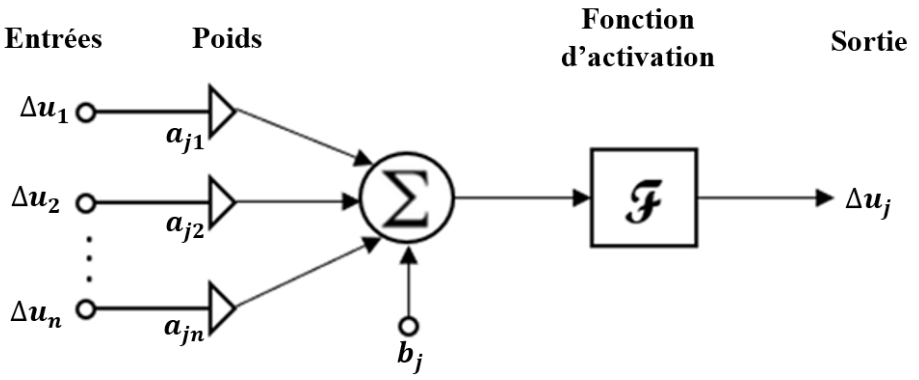


FIGURE 43 – Structure d'un neurone artificiel.

L'architecture d'un réseau est représentée par sa topologie, c'est-à-dire le nombre de couches, le nombre de neurones par couche, le degré de connectivité et le type de connexion entre les neurones [Sammut & Webb, 2010]. Selon leur topologie, les réseaux de neurones peuvent être classés en catégories statiques ou dynamiques.

- **Réseaux simples ou statiques :** dans ces réseaux, la sortie est constante et n'est fonction que de l'entrée du réseau. Ils sont strictement *feedforward*, sans connexions en amont, dans lequel l'information ne se propage que vers l'avant.

- **Réseaux dynamiques ou récurrents :** ce sont des réseaux qui contiennent des retards dans le domaine discret ou des intégrateurs dans le domaine continu. Dans les réseaux dynamiques, la sortie dépend non seulement de l'entrée actuelle du réseau, mais aussi des entrées, sorties ou états précédents du réseau [Demuth *et al.*, 2014] ; pour cette raison, il est dit qu'ils possèdent une mémoire. Un réseau récurrent est un réseau avec rétroaction (*feedback*). Ils sont potentiellement plus puissants et plus complexes que les réseaux statiques. L'une des considérations qui les rendent si avantageux en termes d'application, est qu'un réseau dynamique est équivalent à un système dynamique comme nous l'observons dans la figure 44.

  Durant la phase d'apprentissage, le réseau de neurones récurrent doit apprendre les relations complexes entre les entrées et les sorties, afin de déterminer les coefficients synaptiques les plus adaptés pour représenter les données des connexions entre les neurones.

Ce dernier correspond au type de structure utilisé pour résoudre notre problème d'optimisation. En effet, les réseaux de neurones récurrents permettent de construire un grand nombre d'algorithmes dédiés à la solution de problèmes d'optimisation sous la forme d'un système différentiel dans lequel des équations différentielles décrivent l'algorithmes d'apprentissage non supervisé du RNR. Ceux-ci sont résolus simultanément par un circuit associé qui est constitué d'unités de traitement (ou

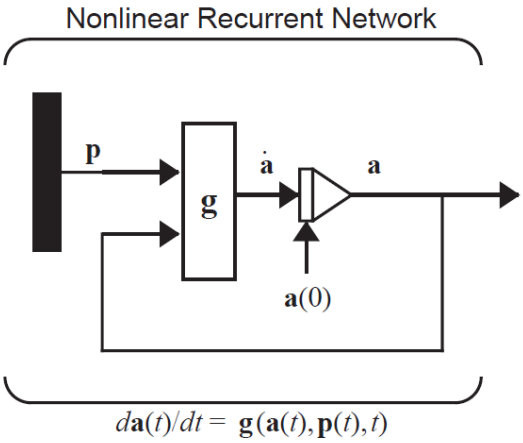


FIGURE 44 – Réseau de neurones récurrent, non linéaire, à temps continu. Tiré de [Demuth *et al.*, 2014].

neurones) hautement interconnectées entre elles. Cela donne un avantage en ce qui concerne la capacité de traitement parallèle du réseau et la vitesse de convergence [Liu & Wang, 2006]. Les capacités des réseaux neuronaux récurrents, telles que le parallélisme élevé, l'adaptabilité et la mise en œuvre de circuits basés sur les circuits embarqués sont reconnus par les chercheurs et sont utilisés dans de nombreux domaines d'application différents, tels que notamment : la conception de systèmes de contrôle, la mémoire associative, la programmation mathématique, le traitement du signal et des images. Ces capacités potentielles nous ont motivés à proposer une solution au problème de l'optimisation du MPC pour les algorithmes de restitution du mouvement, fondée sur les réseaux neuronaux récurrents.

4.2.2 État de l'art spécifique

Les réseaux de neurones artificiels ont été utilisés pour gérer les solutions en temps réel pour la science et pour les applications d'ingénierie. Les réseaux neuronaux sont étudiés depuis de nombreuses années pour leur capacité à résoudre les problèmes d'optimisation complexes. Dans ce domaine, le premier modèle de réseau neuronal récurrent a été proposé par [Hopfield & Tank, 1985] pour une implémention avec des circuits analogiques. Leur RNR contient des neurones dynamiques hautement interconnectés gouvernés par des équations différentielles non linéaires. Ces types de modèles récurrents sont instables à cause de leur dynamique (discrète ou continue). Cependant, [Tank & Hopfield, 1986] ont appliqué la théorie de stabilité de Liapounov à l'analyse des réseaux récurrents. Ainsi, ils ont formulé un problème de programmation linéaire sous forme d'un réseau de neurones en boucle fermée. Au début, le modèle a été utilisé comme une mémoire associative qui a servi dans de nombreuses applications telles que le rétablissement et l'association de motifs, l'élimination de bruit, l'amélioration du contraste, le traitement d'images, entre autres. Toutefois, ce modèle a été rapidement mis en œuvre comme un outil

pour résoudre les problèmes d'optimisation. Leur modèle est basé sur un circuit électrique, où chaque neurone est représenté par un amplificateur opérationnel et un réseau associé formé par un condensateur et différentes résistances qui sont les paramètres du RNR [Cochocki & Unbehauen, 1993]. C'est pourquoi ils ont introduit un modèle continu pour les problèmes d'optimisation sans contraintes basé sur l'approche de la fonction énergétique [Tank & Hopfield, 1986]. Cette fonction énergétique est une fonction de Liapounov pour laquelle le réseau est stable sous la forme d'une fonction quadratique [Demuth *et al.*, 2014]. Leur réseau ne donnait pas de solution au problème de programmation original, mais a inspiré d'autres chercheurs pour améliorer les circuits basés sur des architectures neuronales.

Ainsi, [Kennedy & Chua, 1988] ont amélioré le réseau de [Tank & Hopfield, 1986] en développant un réseau de neurones récurrents pour résoudre les problèmes de programmation non linéaire en utilisant les méthodes du gradient et de fonction de pénalité. Les points d'équilibre du réseau correspondent aux solutions optimales approximatives qui satisfont les conditions de Karush-Kuhn-Tucker (KKT) [1]. Comme ils utilisent des paramètres de pénalisation pour les contraintes des états, leur RNR ne donne qu'une solution approximative au problème d'optimisation original, car les points d'équilibre stables ne sont les solutions des problèmes d'optimisation que lorsque les paramètres de pénalité sont infinis [Chen *et al.*, 1991; Lillo *et al.*, 1993; Mladenov & Maratos, 2000]. En effet, d'après la littérature, plusieurs auteurs n'utilisent pas l'implémentation des paramètres de pénalisation dans la fonction d'énergie comme c'est le cas de [Rodriguez-Vazquez *et al.*, 1988].

[Zhang & Constantinides, 1992] ont proposé la programmation de Lagrange d'un réseau neuronal basé sur la théorie des multiplicateurs de Lagrange pour la programmation sous contraintes. En revanche, leurs RNR sont limités à la dérivation des équations différentielles. [Xia, 1996] a amélioré le réseau de neurones en créant une méthode qui résout les problèmes primal et dual simultanément, capable d'obtenir les solutions exactes. Nous introduisons ces problèmes dans la partie 4.3.2. Ensuite, [Wang, 1997] a proposé un réseau de neurones récurrents basé sur la méthode duale, afin de résoudre directement le problème de programmation quadratique. Plusieurs auteurs ont suivi cette approche et ont construit des réseaux similaires [Cheng *et al.*, 2007; Liu & Wang, 2006; Wang & Xia, 1999]. D'autres ont préféré implémenter une topologie discrète pour la création d'équations différentielles. Les méthodes sont résumées dans le tableau 10.

1. Conditions nécessaires du premier ordre pour qu'une solution au problème d'optimisation non linéaire soit optimale.

RNR	Caractéristiques	Références
Fonction de pénalité	mise en œuvre facile, solutions optimales approximatives seulement	[Maa & Shanblatt, 1992], [Mladenov & Maratos, 2000], [Cheng *et al.*, 2009]
Lagrange	gère les contraintes d'égalité	[Zhang & Constantinides, 1992],[Bouzerdoum & Pattison, 1993],[Costantini *et al.*, 2008]
Projection	sans paramètre de pénalisation	[Xia & Feng, 2005],[Cheng *et al.*, 2009], [Liu & Wang, 2015]
Primal-Dual	sans paramètre de pénalisation	[Xia *et al.*, 2005]
Dual	sans paramètre de pénalisation, complexité de calcul réduite	[Xia, 1996], [Liu & Wang, 2006], [Hu & Wang, 2008]
Temps discret	adapté pour la mise en œuvre avec matériel informatique	[Perez-Ilzarbe, 2013], [Liu & Cao, 2011]

TABLEAU 10 – Différentes approches d'optimisation par réseaux de neurones récurrents.

4.3 Formulation du problème d'optimisation en termes de réseaux de neurones récurrents

Comme nous l'avons vu dans la partie 3.4.3, le problème d'optimisation de la commande prédictive (25) est quadratique avec des contraintes linéaires sur les variables de décision ΔU. Afin de représenter ce problème en termes de réseaux de neurones récurrents, ce problème d'optimisation doit être transformé en une fonction d'énergie de Liapounov du réseau de neurones récurrents [Hopfield & Tank, 1985]. Pour ce faire, il est nécessaire d'appliquer la méthode du gradient au problème d'optimisation quadratique. Puis, le problème est représenté mathématiquement par un ensemble d'équations différentielles ordinaires asymptotiquement stables. Ce système représentera l'architecture du réseau de neurones récurrents.

De cette approche, il a été montré qu'il suffit d'une condition initiale et d'une fonction d'énergie de type Liapounov pour que la solution de l'équation différentielle représentée par le réseau neuronal converge vers les solutions optimales de notre problème (25). Cette fonction de Liapounov définie la stabilité de la trajectoire du système dynamique, et lorsque le temps augmente, la valeur de la fonction diminue. Cela permet de résoudre des problèmes d'optimisation en temps réel grâce aux opérations parallèles des neurones et à leur rapide taux de convergence [Medsker & Jain, 2001].

Il existe deux approches pour la formulation de problèmes d'optimisation en termes de réseau de neurones récurrents : les paramètres de pénalisation et la dynamique du système [Xia & Wang, 1998]. Dans la première approche se trouvent les méthodes dans lesquelles le problème sous contraintes dures devient un problème sans contraintes ou restrictions souples grâce aux paramètres de pénalisation, notamment la méthode de pénalité pour les contraintes d'inégalité et la méthode des multiplicateurs de Lagrange pour les contraintes d'égalité. Dans la seconde approche, le RNR est un ensemble d'équations différentielles pour lesquelles les points d'équilibre correspondent à la solution du problème de programmation, tels que la méthode duale et de projection. Parmi ces deux approches, la plus facile et la plus utilisée est la méthode de pénalité. Compte tenu des difficultés connues associées à son implémentation, les paramètres de pénalisation peuvent varier et donner de bons résultats. C'est pourquoi, elle est la première approche à être testée dans notre modèle.

À des fins pratiques, dans cette partie, notre problème d'optimisation sera représenté par la fonction de coût suivante :

$$\begin{aligned} &\text{minimiser} \quad f(x) = \frac{1}{2}x^T H x + f^T x \\ &\text{soumis à} \quad g_i(x) = A_c x - b \leq 0, \quad i = 1, 2...m. \end{aligned} \tag{35}$$

Où $f(x)$ est une fonction fortement convexe puisque H est une matrice définie positive. Dans cette fonction, $g(x) \in \mathbb{R}^m$ sont les contraintes d'inégalité du système. Nous soulignons que l'équation (35) est en effet le problème d'optimisation du MPC de l'équation (25). Cependant, les variables de décision ΔU ont été remplacées par x et seront appelées dans cette partie les états du réseau de neurones récurrents. Ces états doivent satisfaire des conditions supplémentaires présentées sous la forme de contraintes d'inégalité, afin que le problème d'optimisation soit une fonction d'énergie de type Liapounov pour le réseau de neurones récurrents [Cochocki & Unbehauen, 1993].

4.3.1 Méthode de pénalisation

En utilisant la méthode de pénalité, la fonction $f(x)$ est adaptée afin d'intégrer les contraintes d'inégalité $g_i(x)$. Ainsi, le problème d'optimisation sous contraintes (35) devient un problème d'optimisation sans contraintes en ajoutant un terme de pénalité k aux contraintes $g_i(x)$ dans la fonction de coût $f(x)$. Ceci permet d'obtenir rapidement une solution très proche de la solution optimale [Luenberger *et al.*, 1984].

$$\begin{aligned} &E(x,k) = f(x) + \frac{k}{2}\sum_{i=1}^{m}\left({g_i}^+(x)\right)^2 \\ &{g_i}^+(x) = \max\{0, g_i(x)\} \end{aligned} \tag{36}$$

Où $k > 0$ est le paramètre de pénalisation constante et ${g_i}^+(x)$ est une fonction de pénalité continue positive égale à zéro au point où toutes les contraintes sont

satisfaites [Luenberger *et al.*, 1984]. La fonction différentielle $E(x,k)$ est normalement nommée la fonction d'énergie du réseau neuronal. Dans cette fonction, chaque contrainte non-satisfaite influencera l'état x en attribuant une pénalité multipliée par le paramètre k. Celui-ci doit être assez grand afin de fournir une solution dans la région de faisabilité et précise. Idéalement, quand la valeur de k tend vers l'infini, la solution du problème (36) converge vers la solution du problème avec contraintes (35). Toutefois, dans la pratique, cela reste difficile à réaliser puisque la fonction d'énergie est souvent très mal conditionnée et le système devient alors un système rigide. Par conséquent, avec cette méthode, les solutions sont une approximation au problème d'optimisation réel [Maa & Shanblatt, 1992]. Aussi, si la valeur de k est petite, les contraintes pourront être violées.

D'après la littérature [Cochocki & Unbehauen, 1993; Rodriguez-Vazquez *et al.*, 1988], il est souhaitable de créer une fonction d'activation S_i pour la fonction de pénalité. Autrement dit, si une contrainte est active ($S_i = 1$), la fonction de pénalité est considérée dans l'optimisation, et si une contrainte n'est pas active ($S_i = 0$), la fonction d'énergie $E(x,k)$ ne prend alors pas en compte la fonction de pénalité et elle est résolue comme un problème d'optimisation sans contraintes.

$$E(x,k) = f(x) + \frac{k}{2}\sum_{i=1}^{m} S_i[g_i(x)]^2 \tag{37}$$

Afin de trouver la valeur minimale de la fonction énergétique $E(x,k)$ en utilisant les réseaux des neurones récurrents, le problème d'optimisation sans contraintes est transformé en un système associé d'équations différentielles ordinaires du premier ordre en utilisant la méthode de descente du gradient [Cochocki & Unbehauen, 1993] :

$$\frac{dx}{dt} = -\mu\nabla_x E(x,k) \tag{38}$$

Les trajectoires des états $x(t)$ se déplacent dans la direction de la descente du gradient et convergent vers un minimum global qui est le point d'équilibre de l'équation (38). Cela ne se produit que lorsque la fonction énergétique $E(x,k)$ est une fonction de Liapounov. L'équation dynamique pour un neurone j s'écrit :

$$\begin{gathered}\frac{dx_j}{dt} = -\mu\left(\frac{\partial f(x)}{\partial x_j} + k\sum_{i=1}^{m} S_i g_i(x)\frac{\partial g_i(x)}{\partial x_i}\right), (j = 1,2...N_u), (i = 1,2...m), k > 0, \mu > 0 \\ \mu = diag\left(\mu_1, \mu_2, \ldots, \mu_{N_u}\right), \quad \mu_j = \frac{1}{\tau}, \; S_i = \begin{cases} 0, & \text{si } g_i(x) \leq 0 \\ 1, & \text{autrement} \end{cases}\end{gathered} \tag{39}$$

Avec un paramètre d'apprentissage μ, la constante de temps d'intégration τ, le nombre de contraintes d'inégalités m et l'horizon de contrôle N_u. Le choix du paramètre μ est crucial pour assurer la stabilité du système et la vitesse de convergence vers un point d'équilibre. De même, le choix de μ est basé sur un principe d'essai et d'erreur : un taux d'apprentissage plus élevé sera rapide, mais s'il est trop important, l'algorithme deviendra instable et le système (39) ne convergera pas. Cela dépend également de la valeur de τ et de l'état initial du réseau [Lillo *et al.*, 1993].

Le système d'équations (39) représente un réseau neuronal récurrent en tant que système dynamique d'équations différentielles. Après remplacement des fonctions par leurs index respectifs l'on obtient :

$$\frac{dx_j}{dt} = -\mu_j \left(\left(f_j + \sum_{i=1}^{N_u} h_{ji} x_i \right) + k \sum_{i=1}^{m} S_i a_{ij} g_i(x) \right) \tag{40}$$

Mise en œuvre

Afin de mettre en œuvre cette nouvelle approche d'optimisation dans le simulateur de conduite dynamique, nous utilisons le logiciel Simulink-Matlab. Comme indiqué dans la partie précédente, le réseau neuronal récurrent (RNR) est considéré comme un ensemble d'équations différentielles (38) qui représentent toutes les trajectoires des états x du réseau. Ce système a été reproduit dans Simulink à l'aide de différentes composantes mathématiques telles que les gains, les multiplicateurs, les intégrateurs, les additionneurs et autres. L'annexe B montre un exemple d'optimisation en utilisant cette méthode sur Simulink.

Le système des RNR a été simulé en tant qu'algorithme d'optimisation dans la boucle de contrôle prédictif comme le montre la figure 45. Pour ce faire, nous avons utilisé une optimisation à 1 DDL (voir partie 3.4.3).

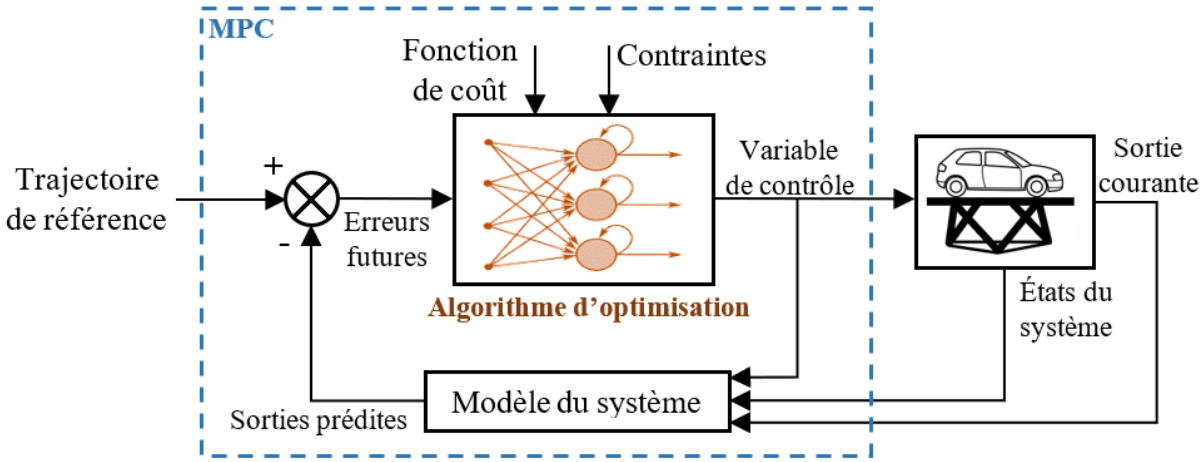


FIGURE 45 – Commande prédictive avec l'algorithme d'optimisation basé sur les réseaux de neurones récurrents.

Le problème d'optimisation est représenté par un circuit embarqué d'équations différentielles qui sont résolues à chaque pas de temps par une méthode numérique propre à Simulink. Contrairement aux méthodes d'optimisation existantes, avec les RNR nous n'avons pas à programmer un algorithme d'optimisation, mais seulement à le représenter et à le faire évoluer avec une méthode numérique jusqu'à son point d'équilibre, qui devient la solution du problème d'optimisation. Nous avons constaté que la méthode montre une convergence pour toutes les trajectoires de l'état x vers un point optimal. Néanmoins, cela n'était possible qu'avec des méthodes numériques

avec des pas d'échantillonnage variables et pour des systèmes rigides. Ainsi, lors de l'implémentation des méthodes numériques à pas de temps fixe disponibles, la méthode RNR n'a pas fonctionné complètement puisque la convergence des états dans l'équation (39) n'était pas assurée pour l'ensemble de la simulation. Cela soulève un problème pour la mise en œuvre en temps réel, car le programme doit avoir un pas d'échantillonnage fixe.

Pour déterminer le pas minimal pour que le RNR trouve une solution, c'est-à-dire, un point d'équilibre pour chaque état x à chaque temps d'échantillonnage, nous avons analysé le pas utilisé par la méthode numérique à pas variable de Simulink. La figure 46 montre que la taille minimale du pas pour capturer la dynamique avec précision est très petite ($\sim 10^{-8}$) et ne peut, par conséquent, être appliquée dans une application en temps réel.

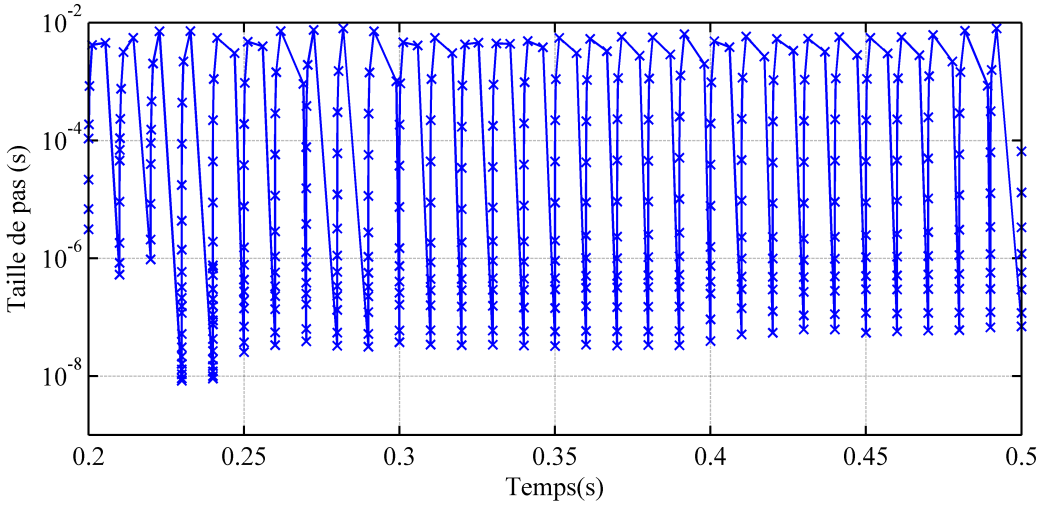


FIGURE 46 – Pas de temps du logiciel de résolution pour un modèle rigide.

Comme le montre la figure 46, le système d'équations différentielles est rigide, c'est-à-dire que le système a une dynamique continue qui varie à la fois lentement et rapidement. Par conséquent, la méthode numérique de Simulink doit utiliser des pas de temps petits pour obtenir une solution qui converge au point d'équilibre du système. L'une des raisons pour lesquelles les équations différentielles sont rigides est que le système est mal conditionné : $cond(H) >> 1$. Cette condition signifie que le rapport entre la plus grande et la plus petite valeur propre de la matrice H est très élevé. Dans ce cas, le gradient ne trouve pas la meilleure façon de descendre au minimum et doit prendre plusieurs pas de temps pour le faire. Cependant, la fonction énergétique ne dépend pas seulement de la fonction de coût originale (35), mais aussi de la fonction de pénalité (36) qui est mise à jour à chaque étape d'échantillonnage en fonction des contraintes du système $g(x)$ et des états x, ce qui rend difficile le pré-conditionnement du système basé uniquement sur la matrice H de $f(x)$. En effet, le paramètre de pénalité k agit comme un ressort amortisseur du système : plus il est important, plus il est rigide et donc, plus il est difficile pour le système de trouver une solution.

Pour atténuer le mauvais conditionnement et la rigidité des équations différen-

tielles, une stratégie de régulation est mise en œuvre. Nous avons transformé la topologie du réseau comme le montre la figure 47.

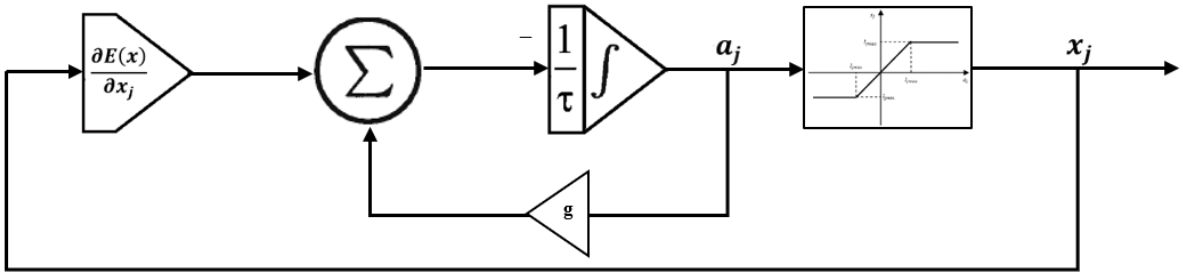


FIGURE 47 – Schéma du réseau neuronal récurrent proposé.

L'architecture consiste en un intégrateur à temps continu qui peut être considéré comme un neurone spécial dépendant du temps [Cochocki & Unbehauen, 1993], des blocs opérationnels, un gain g et une fonction de saturation. Nous pouvons constater que les deux derniers éléments ne font pas partie du système de RNR original (38). Le gain g représente le paramètre de réglage qui aide à gérer les oscillations pour chaque trajectoire d'état x. En outre, pour s'assurer que toutes les contraintes de l'équation (35) sont satisfaites, et étayer l'analyse de convergence faite par [Costantini *et al.*, 2008], nous convertissons les contraintes du système $g(x)$ en une fonction linéaire de saturation illustrée dans la figure 48 qui garantit que les variables de décision ne dépassent pas ses limites :

$$x_j = f\left(a_j\right) = \begin{cases} l_{jmin}, & a_j \leq l_{jmin} \\ a_j, & a_j \in [l_{jmin}, l_{jmax}] \\ l_{jmax}, & a_j \geq l_{jmax} \end{cases} \quad , \quad j = 1, 2, ..., N_u \tag{41}$$

Les valeurs de l_{max} et l_{min} dépendent des contraintes d'inégalités $g(x)$. Ces contraintes suivent la loi de freinage énoncée dans la partie 4.1.2. Nous avons choisi ce type de saturation, car c'est la seule qui peut garantir la faisabilité du problème d'optimisation tout au long de l'horizon de prédiction. Dans ce cas, la fonction énergétique est Liapounov, et, par conséquent, les états du réseau neuronal convergent vers les points d'équilibre associés aux solutions du problème d'optimisation.

Grâce à la fonction (41) le RNR trouvera une séquence de contrôle $\Delta U = x$ pour laquelle le gradient de la fonction de coût est nul. Lorsque la plateforme dynamique s'approchera des limites physiques, la fonction de saturation empêchera le gradient de trouver une solution instable et limitera la valeur au maximum ou au minimum du gradient comme indiqué dans l'équation (42).

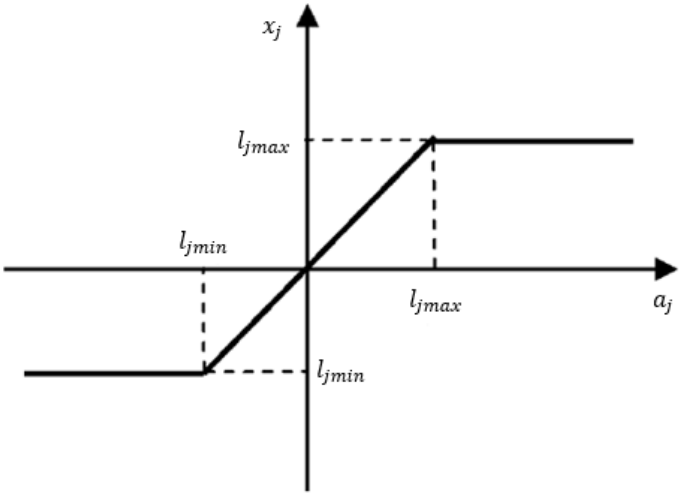


FIGURE 48 – Fonction dynamique de saturation $f\left(a_j\right)$.

$$\begin{aligned} x_j = l_{max}, \quad & \frac{\partial E(x)}{\partial x_j} \leq 0 \\ l_{min} < x_j < l_{max} \quad & \frac{\partial E(x)}{\partial x_j} = 0 \\ x_j = l_{min}, \quad & \frac{\partial E(x)}{\partial x_j} \geq 0. \end{aligned} \tag{42}$$

Le système dynamique proposé décrivant ce comportement est détaillé dans l'équation d'état suivante :

$$\frac{dx}{dt} = -\mu\left(\nabla_x E(x) + g.a\right) \tag{43}$$

Comme la matrice Hessienne H de la fonction $f(x)$ dans (35) est définie positive, chaque point d'équilibre du système (43) correspond à une solution globale optimale de cette fonction.

Simulation

Afin de montrer la performance de la méthode fondée sur les réseaux neuronaux récurrents en tant qu'algorithme d'optimisation, un scénario de simulation de conduite a été généré en utilisant le logiciel SCANeR Studio. L'idée est d'analyser le suivi de référence et le respect des limites du simulateur.

Le scénario de simulation est une simple manœuvre qui consiste à rester à une position initiale pendant quatre secondes puis à suivre un véhicule dont la vitesse est de 30 km/h. Le scénario a une durée de 18 secondes, temps suffisant pour évaluer le comportement du système. L'accélération linéaire selon l'axe x de ce scénario est illustrée dans la figure 49. Dans Simulink, nous avons sélectionné une

méthode numérique implicite (ode14x, *extrapolation*) à pas de calcul fixe, car il offre la stabilité nécessaire pour les comportements oscillatoires, évitant ainsi des coûts de calcul inutiles. Le temps d'échantillonnage global du système RNR a été fixé à $1ms$, et un minimum de 3 itérations est nécessaire pour que le réseau converge vers une solution optimale. Le système de la plateforme est échantillonné avec un pas de $10ms$, ce qui est le minimum requis pour une application en temps réel (fréquence de contrôle de 100 Hz).

D'abord, nous avons effectué une comparaison avec une autre méthode d'optimisation afin de valider la précision de la valeur optimale à chaque pas de temps. Aussi, une fonction quadprog-Matlab avec une méthode d'ensemble actif a été utilisée afin de reproduire les solutions de notre problème d'optimisation et les comparer avec les résultats du RNR. Nous avons constaté que la réponse donnée par le programme quadprog-Matlab coïncidait avec les états d'équilibre du réseau de neurones. Cela confirme l'applicabilité de la RNR pour résoudre le problème d'optimisation du MCA fondée sur le MPC.

Ensuite, nous avons analysé le temps moyen que prend la fonction de quadprog pour résoudre le problème d'optimisation. Cette analyse a été réalisée en utilisant la méthode de Matlab-Simulink : *model profiler*. Nous avons constaté que pour notre problème à 1 DDL, la fonction quadprog prend au moins $3ms$ pour trouver une solution. Puis nous avons analysé le temps pris par chaque bloc de notre RNR (figure 47), pour résoudre son opération, et nous avons trouvé un temps total de $0,235ms$. Comme indiqué ci-dessus, un minimum de 3 itérations est nécessaire pour trouver une solution optimale, par conséquent, le temps nécessaire pour résoudre le problème d'optimisation quadratique avec le RNR est de $\sim 0,7ms$. Cette comparaison montre le potentiel de l'approche proposée pour résoudre des problèmes de programmation quadratique avec un plus grand nombre d'états et en particulier le problème d'optimisation du MPC en tant qu'algorithme de restitution de mouvement en temps réel.

La figure 49 montre les signaux d'accélération selon l'axe x de la plateforme pendant le scénario de simulation. Nous observons que la plateforme suit la référence en accélération jusqu'à ce qu'elle approche de ses limites physiques. Ensuite, les contraintes sont appliquées de sorte que l'optimisation puisse trouver une solution à chaque instant d'échantillonnage. Le comportement de l'accélération dépend des paramètres de la loi de freinage mentionnés dans la partie 4.1.2 sur la faisabilité de la commande prédictive.

La complexité de calcul de la méthode montré ici est réduite grâce à que la fonction de pénalité peut être remplacée en utilisant une fonction d'activation saturée pour les contraintes d'inégalité et un paramètre de réglage approprié. Par conséquent, le nombre de neurones et d'opérations logiques utilisés pour résoudre le problème de programmation quadratique est réduit.

Dans cette première expérimentation, nous avons évalué l'efficacité du réseau neuronal basé sur la méthode de pénalité pour résoudre un problème d'optimisation à 1 DDL. Cela signifie que chaque axe de mouvement de la plateforme doit être

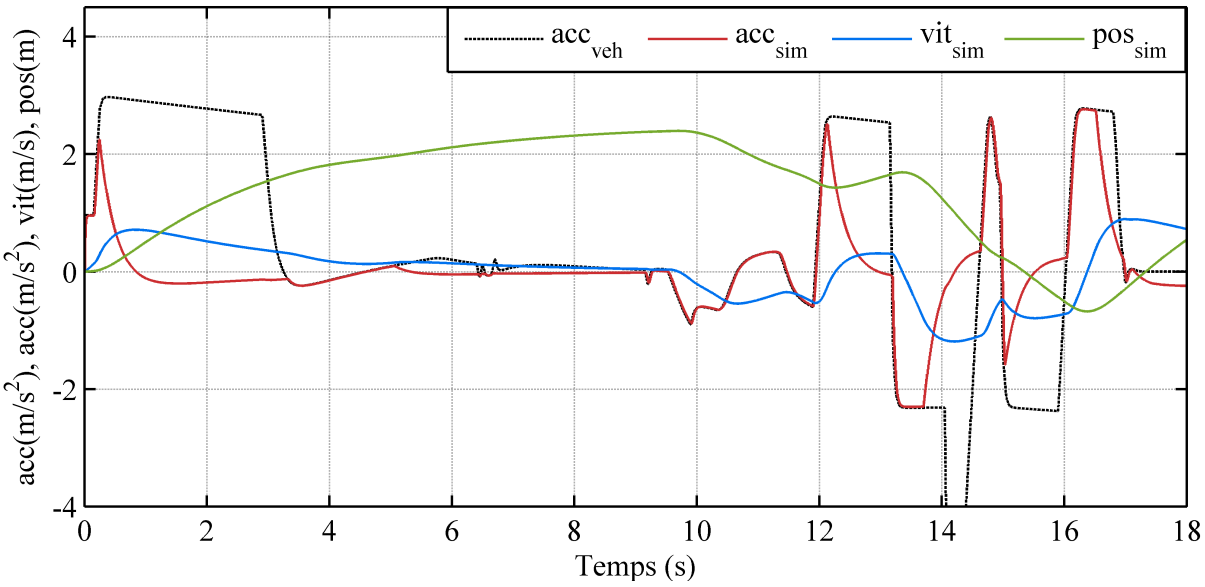


FIGURE 49 – Rendu de l'accélération longitudinale en utilisant uniquement les rails x du simulateur.

pris indépendamment. Cependant, l'un des objectifs de cette thèse est de pouvoir intégrer la perception de mouvement du conducteur dans la boucle de contrôle. C'est pourquoi nous cherchons à obtenir non seulement les accélérations de chaque axe de mouvement, mais aussi la force spécifique ressentie par le conducteur. En outre, avec la fonction de saturation adaptative et une stratégie à 1 DDL, le MPC peut être proposé en tant que système LTI sans contraintes. Dans ce cas le MPC n'a aucun avantage pratique, car il existe de nombreuses autres méthodes qui conviennent au moins aussi bien [Bemporad & Morari, 1999].

Ainsi, notre deuxième expérimentation consiste à vérifier la stabilité et la performance du réseau pour une optimisation à 2 DDL, c'est-à-dire, optimiser en même temps les accélérations linéaires et l'angle d'inclinaison à partir de la force spécifique du système $f = g\theta + acc_{lin}$. Il faut rappeler que la force spécifique ressentie par le conducteur du simulateur est tirée du modèle mathématique des organes otolithiques. Ce modèle a été expliqué en détail dans la partie 3.4.1.

Contrairement au problème d'optimisation à 1 DDL, le modèle mathématique utilisé pour la formulation du MPC est passé de 3 états (position, vitesse et accélération) à 8 états différents, dont 4 états qui proviennent du système vestibulaire (x_{oto} de l'équation (19) et 4 états qui dépendent de la plateforme : position et vitesse linéaire et angulaire. Quant au problème d'optimisation, il est de la même forme que l'équation (35). Néanmoins, le nombre de restrictions et de variables de décision augmente.

En utilisant la même topologie de réseau de neurones (figure 47), nous avons constaté que le système n'était pas stable, c'est-à-dire que le réseau ne convergeait pas vers une solution qui minimisait le problème d'optimisation (35). Cela est dû à la fonction de saturation qui dépend des restrictions $g(x) : Ax \leq b$. Dans cette fonction,

la matrice A doit être une matrice carrée de sorte qu'il y ait le même nombre de contraintes que de variables de décision ou états du réseau x. Toutefois, les états et les contraintes n'ont pas la même dimension ; la matrice A dépend de l'horizon de prédiction N_p et les états du réseau dépendent de l'horizon de contrôle N_u ($N_p > N_u$). Pour une optimisation à 1 DDL, seules les premières restrictions $g(1 : N_u)$ ont été considérées, et cela a suffi pour que le système soit stable. Cependant, lorsqu'il y a deux variables de décision, un N_u aussi court n'est pas suffisant pour que le réseau soit stable.

Afin de résoudre ce problème, deux solutions ont été envisagées : prendre $N_u = N_p$ ou appliquer une autre topologie de réseau qui prend en compte toutes les contraintes. Dans la première option, la matrice A est carrée. Cependant, le temps de calcul du problème d'optimisation augmente considérablement, c'est pourquoi nous avons opté pour la seconde solution. Nous avons vu précédemment (tableau 10) d'autres types de réseaux de neurones applicables aux méthodes d'optimisation quadratique sous contraintes. Le réseau neuronal de Lagrange souffre d'un mauvais traitement des contraintes d'inégalité et est plus large que la méthode de pénalité. Le réseau neuronal primal-dual a habituellement une structure à deux couches qui résout le problème dual simultanément en minimisant l'écart de dualité avec les conditions de KKT et la méthode du gradient. Cela génère des équations plus complexes qui contiennent des termes non linéaires d'ordre supérieur [Cheng *et al.*, 2007]. En effet, les méthodes primal-dual et de projection ont une dynamique et une structure plus compliquée par rapport à la méthode duale [Zhang & Wang, 2002]. C'est pourquoi nous appliquons, dans la partie suivante, la méthode duale pour résoudre le problème d'optimisation à 2 DDL.

4.3.2 Méthode duale

En optimisation, il est commun d'implémenter le principe de dualité [Wolfe, 1961]. Celui-ci permet d'établir, à partir d'un problème d'optimisation, deux problèmes de programmation différents : le problème primal et le problème dual. Les deux problèmes sont construits à partir des mêmes coefficients de coût et de contraintes, mais de façon que si l'un de ces problèmes est une minimisation, l'autre est une maximisation [Luenberger *et al.*, 1984]. Pour notre système, le problème dual est le problème d'optimisation (44) : il est construit à partir du primal (35), mais en intégrant les contraintes en tant que variables de décision du problème principal. Ces variables duales sont interprétées comme des multiplicateurs de Lagrange associés à la solution primale.

Pour appliquer ce principe de dualité, il est nécessaire que le problème de programmation original soit convexe. Dans notre cas, le problème mathématique primal (35) est carré et convexe. Par conséquent, le minimum de cette fonction correspond au maximum de la fonction dans le problème dual [Fletcher, 2000]. En utilisant la formulation de [Wolfe, 1961], nous transformons le problème primal en le problème

dual suivant :

$$\begin{aligned} &\text{maximiser} \quad f(x) = \frac{1}{2}x^T H x + f^T x - \lambda^T (A_c^T - b) \\ &\text{soumis à} \quad Hx + f - A_c\lambda = 0,\ \lambda \leq 0 \end{aligned} \tag{44}$$

Avec λ le vecteur des variables de décision duales. Dans ce problème d'optimisation dual il existe une contrainte duale pour chaque variable de décision primale et une variable de décision duale pour chaque contrainte primale. Cette formulation alternative permet de résoudre les problèmes linéaires qui possèdent plus de contraintes que de variables, comme dans notre cas. En effet, le coût de calcul pour résoudre les problèmes de programmation linéaire est fonction du nombre de lignes dans la matrice A_c et non du nombre de colonnes. Ainsi, en appliquant la dualité au problème (35), l'on obtient un autre problème linéaire dans lequel le nombre de lignes est inférieur au nombre de colonnes dans les contraintes.

Implémentation

Nous trouvons dans la littérature différents problèmes d'optimisation utilisant ce type de réseau de neurones. Le premier réseau neuronal basé sur la méthode de dualité [Wang, 1997] a été développé pour réduire la complexité des réseaux précédents tels que le primal ou de projection et afin d'augmenter l'efficacité de calcul. De nombreux autres réseaux basés sur cette méthode ont été créés par la suite pour produire une convergence globale et exponentielle [Wang & Xia, 1999], pour inclure toutes sortes de restrictions linéaires [Zhang & Wang, 2002], pour réduire la topologie du réseau [Liu & Wang, 2006] ou pour générer des réseaux neuronaux de manière discrète [Perez-Ilzarbe, 2013]. Pour notre système, nous avons effectué une synthèse de ces articles pour créer un réseau neuronal récurrent dual.

Afin de créer la topologie du RNR dual, il faut suivre certaines étapes. La première consiste à définir les conditions de KKT de notre problème d'optimisation dual (44) :

$$\begin{aligned} &Hx + f - A_c^T\lambda = 0 \\ &A_c x = g(A_c x - \lambda) \end{aligned} \tag{45}$$

La fonction $g(\beta)$ est une fonction de saturation inspirée de la loi de freinage (partie 4.1.2) qui assure la faisabilité et la convergence globale du réseau de neurones. Elle dépend des contraintes limites $A_c x \leq b$ de la fonction (35).

$$g(\beta) = \begin{cases} l_{min}, & \beta < l_{min} \\ \beta, & l_{min} \leq \beta \leq l_{max} \\ l_{max}, & \beta > l_{max} \end{cases} \tag{46}$$

À partir des équations (45), (46) et basé sur la formulation de [Liu & Wang, 2006]

la dynamique du réseau neuronal dual est :

$$\begin{aligned} &\text{Équation d'état :} \quad \frac{d\lambda}{dt} = \alpha(-A_c x + g(A_c x - \lambda)), \\ &\text{Équation de sortie :} \quad x = H^{-1}(A_c^T \lambda - f). \end{aligned} \tag{47}$$

Avec λ le vecteur d'état du système et α le paramètre de convergence du réseau neuronal. Afin de simuler le réseau ci-dessus, la même courbe d'accélération de la figure 49 a été utilisée. Par rapport à la formulation du MPC, seul l'algorithme d'optimisation est changé en une méthode à 2 DDL avec réseaux neuronaux récurrents duaux, mais la stratégie de contrôle du MCA ne varie pas. La dynamique de ce réseau est représentée à l'aide de composants de base telles que des sommes, des multiplicateurs et des intégrateurs du logiciel Simulink. Elle est illustrée dans la figure 50.

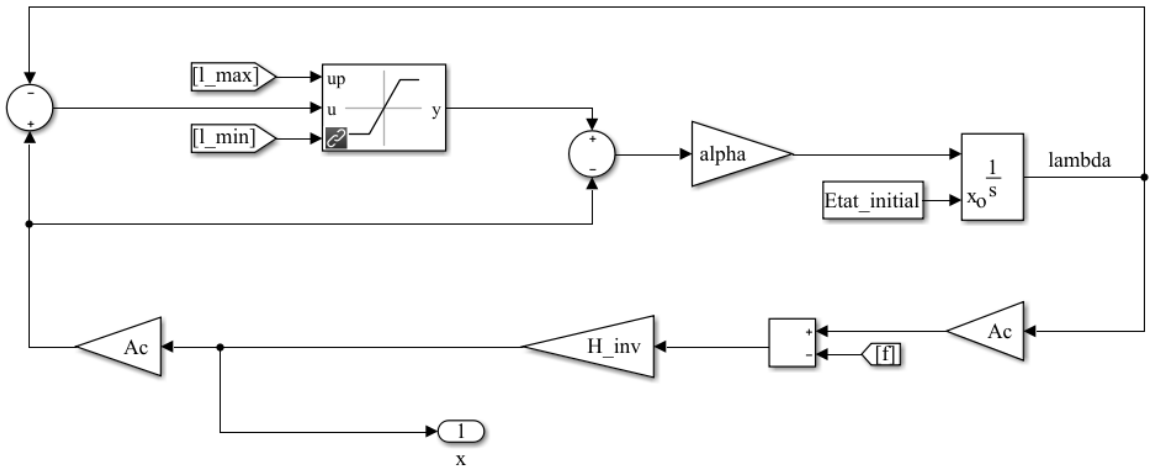
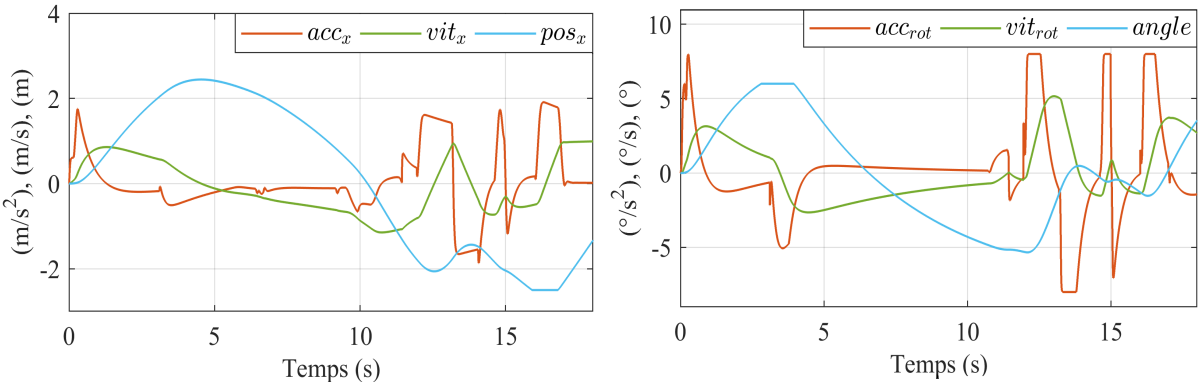


FIGURE 50 – Schéma fonctionnel du réseau neuronal dual.

Le réseau (47) a été implémenté en utilisant un algorithme numérique pour les systèmes rigides à pas d'échantillonnage variable, car malgré plusieurs méthodes de régulation mises en œuvre, il n'a pas été possible de stabiliser le réseau et de le faire converger vers un point optimal en peu de temps en utilisant les méthodes disponibles à pas fixe. Par conséquent, ce réseau neuronal ne peut pas être mis en œuvre dans un simulateur de conduite dynamique en temps réel. En effet, pour l'intégrer dans le système d'exploitation du simulateur en temps réel, le MCA doit être compilé en utilisant une période d'échantillonnage fixe, ce qui n'est pas le cas avec une optimisation fondée sur cette topologie neuronale duale. Néanmoins, la méthode de pénalisation montrée dans la partie 4.3.1 peut être intégrée dans le simulateur dynamique, mais seule une optimisation à 1 DDL peut être effectuée. Malheureusement, cette optimisation n'est pas optimale pour toutes les situations de conduite, en particulier lorsque le scénario de conduite possède des périodes d'accélération continue. Dans la prochaine partie, nous indiquerons quelques perspectives pour la méthode des réseaux de neurones récurrents en tant qu'optimisation du MCA basé sur le MPC.

En ce qui concerne la méthode duale à pas variable, nous avons voulu montrer

son efficacité en termes de respect des limites physiques de la plateforme et dans la réduction de l'erreur de perception de la force spécifique. Dans la figure 51, nous observons qu'avec la réponse du réseau, le système respecte les restrictions physiques du simulateur, tant pour les rails dans la figure 51a que pour l'hexapode dans la figure 51b.



(a) Mouvement linéaire

(b) Mouvement angulaire

FIGURE 51 – Déplacement, vitesse et accélération de la plateforme.

La performance de cette stratégie en termes de réduction de l'erreur entre la force spécifique du simulateur et la force spécifique perçue par le conducteur est illustrée dans la figure 52. Nous observons que les accélérations transitoires sont suivies correctement alors que les accélérations continues fournies avec l'angle d'inclinaison (technique de *tilt coordination*) ne sont pas assez rapides et génèrent un retard dans le signal final. Ce retard est dû au seuil de perception en vitesse, position et accélération angulaire. Toutefois, il est important de noter que la performance de la stratégie de contrôle du mouvement ne dépend pas seulement de l'optimisation, mais aussi des paramètres de pondération de la fonction de coût ou de la fonction d'énergie dans le cas des réseaux neuronaux.

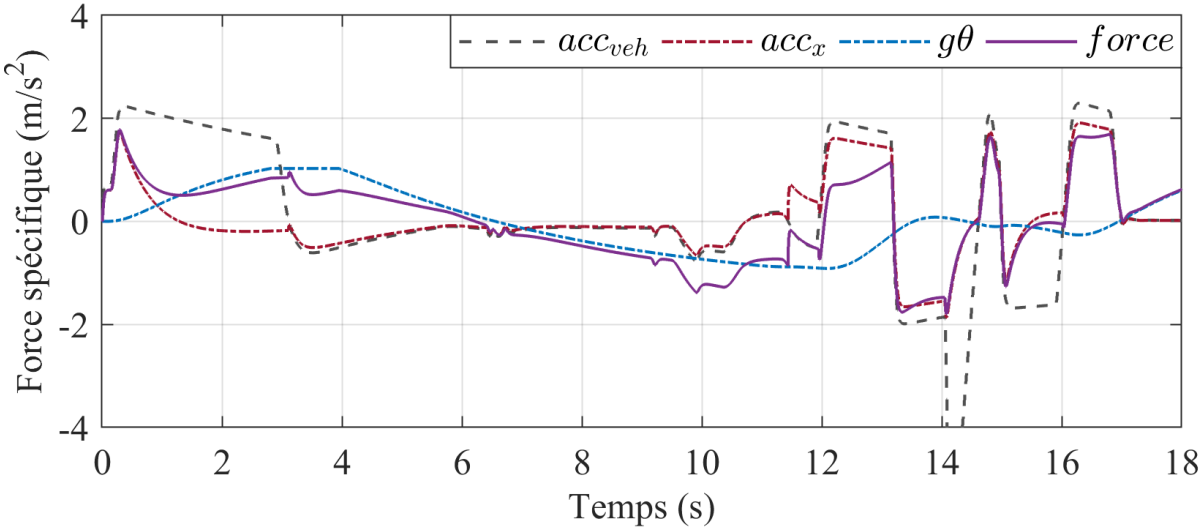


FIGURE 52 – Force spécifique restituée pour la plateforme.

4.3.3 Limitations et perspectives

Malgré la bonne performance de cette méthode d'optimisation pour le contrôle prédictif, ce type d'optimisation n'a pas pu être mis en œuvre dans le simulateur réel à hautes performances en temps réel. Il présente quelques inconvénients qui le rendent inapproprié pour les algorithmes de contrôle de mouvement MCA. Ces inconvénients sont énumérés ci-dessous :

- Il est difficile de trouver le coefficient de convergence α pour que le réseau neuronal soit stable et converge vers son point d'équilibre, qui correspond à la solution du problème d'optimisation.
- Afin d'intégrer l'algorithme en temps réel, il est nécessaire d'utiliser des méthodes de régulation supplémentaires pour éviter que le système soit rigide ou que le gradient n'explose [1].
- Plusieurs pas d'échantillonnage sont nécessaires pour que le système d'équations différentielles, représenté par le RNR, trouve son point d'équilibre.
- La faisabilité d'une solution pour le problème d'optimisation doit être garantie tout au long de l'horizon de prédiction. Si ce n'est pas le cas, la fonction d'énergie n'est pas Liapounov et le problème ne peut pas être représenté par un réseau de neurones.

Malgré ces difficultés, les réseaux de neurones présentent des caractéristiques qui sont loin d'être négligeables et qui ont été à l'origine de l'engouement pour leur utilisation en tant qu'algorithmes d'optimisation pour le MPC. Ils possèdent : une dynamique simple en termes de mise en œuvre, une convergence globale (exponentielle) vers des solutions optimales [Zhang, 2013], une adaptabilité à différents problèmes et types de contraintes et une topologie qui permet une parallélisation des opérations. Malheureusement, ces caractéristiques ne sont pas suffisantes pour les mettre en œuvre dans notre système, car comme indiqué auparavant, il est nécessaire que le problème d'optimisation ait une solution avec un temps d'échantillonnage fixe.

Dans les faits, le simulateur dynamique utilise une plateforme d'exploitation en temps réel qui exige une fréquence d'échantillonnage fixe de 0,008 secondes pour tous les modules de simulation, y compris le système d'optimisation. Par conséquent, le réseau neuronal récurrent doit respecter ce temps d'échantillonnage afin d'être intégré à la plateforme en temps réel. Cette fréquence n'a pas été atteinte avec une méthode à étapes fixes pour le RNR dual et c'est pour cette raison que cette méthode d'optimisation n'a pas été intégrée dans la plateforme. De plus, bien que la méthode de pénalité puisse être modifiée pour cette fréquence, cette méthode n'est

1. Dans les réseaux neuronaux récurrents, les erreurs de gradient peuvent s'accumuler à chaque pas d'échantillonnage et entraîner des gradients très importants. Par conséquent le réseau peut devenir instable et empêcher sa convergence.

utilisée que pour une optimisation à 1 DDL, ce qui réduit son utilité par rapport aux méthodes d'optimisation quadratiques existantes.

L'une des alternatives et des améliorations futures de ce type de réseau est de l'intégrer comme un système d'optimisation embarqué à partir de portes de FPGA comme cela a été fait dans plusieurs algorithmes MPC [Jerez *et al.*, 2011; McInerney *et al.*, 2018]. Cependant, cette option n'a pas été étudiée dans cette thèse, car elle s'éloigne du cadre d'étude de la thèse qui est l'amélioration des techniques de contrôle du mouvement et non pas seulement celle de l'algorithme d'optimisation.

4.4 Conclusion

La première partie de ce chapitre a permis d'acquérir les connaissances nécessaires sur la faisabilité de la commande prédictive afin de garantir la stabilité en boucle fermée du simulateur. L'analyse de la faisabilité des MCA a permis de définir les paramètres et conditions qui sont indispensables pour obtenir une bonne performance du système, tout en respectant les contraintes physiques de la plateforme. Nous avons conclu que seule une saturation, fondée sur une loi de freinage, peut assurer la solution au problème d'optimisation tout au long de l'horizon de prédiction. Il a été démontré que la performance du système est affectée par les différentes méthodes de faisabilité implémentées ; ainsi, si la condition de la loi de freinage est très restrictive, l'on peut utiliser une pondération terminale fondée sur la théorie de la commande optimale. Néanmoins, la faisabilité de la solution n'est pas garantie pour tous les scénarios de conduite.

Grâce à cette analyse, nous avons proposé, dans la seconde partie du chapitre, une nouvelle méthode d'optimisation sous la forme des réseaux de neurones récurrents. En effet, pour représenter le problème d'optimisation par un circuit intégré il faut démontrer que la fonction d'énergie représentant le circuit est une fonction de Liapounov, c'est-à-dire, il faut assurer la faisabilité et stabilité en boucle fermée du système à chaque pas de temps. Ainsi, les points d'équilibre du réseau correspondent aux solutions optimales du problème d'optimisation original. L'une des caractéristiques majeures qui nous a motivé à utiliser les réseaux de neurones récurrents pour l'optimisation est qu'ils sont exécutés en continu plutôt qu'itérativement comme pour la plupart des algorithmes numériques. Par conséquent, sa caractéristique de calcul parallèle permet aux RNR de résoudre plus efficacement les problèmes en temps réel tels que le MPC implicite.

Afin d'adapter cette méthode à notre problème, nous avons simulé numériquement le RNR par deux approches différentes : la méthode de pénalisation et la méthode duale. La première utilise des paramètres de pénalisation pour les contraintes, mais ces paramètres ont été remplacés par une fonction de saturation et une méthode de régulation, afin de permettre une simulation avec un pas d'échantillonnage fixe. La méthode a été mise en œuvre pour chaque axe de mouvement indépendamment, c'est-à-dire une optimisation à 1 DDL. Néanmoins, cette méthode n'a pas été

suffisante pour une optimisation à 2 DDL, car la fonction de saturation n'est conçue que pour des contraintes jusqu'à l'horizon de contrôle et non pas de prédiction. Par conséquent, nous avons mis en œuvre une méthode d'optimisation duale qui permet d'intégrer toutes les contraintes du système dans la fonction de saturation. Avec la méthode de RNR dual, le problème de programmation quadratique possède une solution pour les accélérations linéaires et angulaires, mais malheureusement, seule une méthode numérique à pas variable a permis de stabiliser le système. Cela veut dire que telle que la méthode a été présentée dans ces travaux, elle n'est pas applicable aux algorithmes de restitution de mouvement en temps réel.

En effet, nous avons essayé d'intégrer une méthode innovante dans les algorithmes de restitution de mouvement afin de rendre l'algorithme d'optimisation plus rapide et plus performant pour les nouvelles plateformes de simulation de conduite à hautes performances. Malheureusement, la méthode proposée fondée sur les réseaux de neurones récurrents possède plusieurs inconvénients qui la rendent inadaptée pour ce type d'application. Comme nous l'avons dit auparavant, cette méthode peut être améliorée ou intégrée au système d'exploitation en tant que logiciel embarqué avec des composantes analogiques qui peuvent représenter le système en continu. Toutefois, afin de continuer avec notre sujet de recherche, nous laissons les améliorations de cette méthode en tant que perspectives et nous nous concentrons sur le renforcement du MPC à partir du modèle mathématique afin d'améliorer la perception du mouvement dans le simulateur en situation de conduite active et en mode autonome.

Chapitre 5

Évaluation du modèle perceptif dans le simulateur de conduite dynamique

Comme nous l'avons vu dans la partie 3.4, le contrôle prédictif est composé de trois éléments principaux : le modèle, la prédiction et l'optimisation. Parmi ceux-ci, seuls le modèle et l'optimisation peuvent être modifiés, car la prédiction du comportement dynamique du système est essentiellement une série d'opérations matricielles qui dépendent du type de modèle. Dans le chapitre précédent, nous avons abordé l'optimisation du MPC, c'est pourquoi, l'objectif principal de ce chapitre est d'évaluer le modèle mathématique de l'algorithme de mouvement, afin d'améliorer l'immersion du conducteur, la perception du mouvement et l'utilisation de l'espace de travail dans un simulateur de conduite à hautes performances. L'influence du modèle mathématique sera évaluée en simulation dans un mode de conduite autonome et un mode de conduite par un conducteur.

La première partie de ce chapitre expose le contexte de l'expérimentation, les objectifs ainsi qu'un état de l'art des différents types de modèles utilisés en MPC pour la restitution de mouvement. Nous modéliserons chacun de ces modèles, puis nous comparerons en simulation leurs différences en termes d'espace de travail et de suivi de trajectoire.

Bien qu'une restitution fidèle des mouvements dans le simulateur soit un critère important dans la validation d'une stratégie, il ne faut pas laisser de côté l'élément principal : le conducteur. C'est pourquoi, nous avons utilisé un questionnaire qui évalue le comportement de conduite auto-déclaré du conducteur. Cette information sera utilisée pour définir l'influence du comportement de conduite dans le choix d'un MCA. Ainsi, dans la seconde partie de ce chapitre, nous présenterons les conditions expérimentales d'une étude qui vise à étudier le rôle du modèle de perception du mouvement dans les MCA.

La troisième partie sera consacrée à l'analyse des résultats subjectifs et objectifs

de l'expérimentation. En effet, la comparaison des différents modèles et du type de comportement de conduite sera étayée par la corrélation des mesures de performance obtenues lors de l'expérimentation dans le simulateur. Les résultats seront accompagnés d'une discussion et d'une conclusion générale à la fin du chapitre.

5.1 Introduction et contexte

Plusieurs actions ont été menées dans la conception des MCA, pour augmenter le réalisme de la simulation et réduire le mal du simulateur induit en raison du conflit entre les sous-systèmes sensoriels. L'une des approches visant à résoudre ces problèmes a été d'intégrer la perception de mouvement humaine dans la conception des commandes de la plateforme. À cet égard, les techniques fondées sur l'optimisation, telles que la commande optimale [Asadi *et al.*, 2014] ou la commande par modèle prédictif (MPC) [Fang & Kemeny, 2014], intègrent le modèle mathématique du système vestibulaire dans la conception du contrôle, afin d'améliorer la perception du mouvement des conducteurs. Tout comme nous l'avons présenté dans la partie 2.3, le modèle de perception humaine du mouvement est fondé sur le modèle mathématique du système vestibulaire. Ce dernier est le principal organe humain capable de percevoir les accélérations linéaires et angulaires.

La littérature fournit plusieurs modèles de système vestibulaire [Momani & Cardullo, 2018]. Cependant, lors de la conception des MCA, seuls deux modèles sont communément utilisés dans un contexte de « modèle mathématique » et de « théorie du contrôle ». Le premier est le modèle des canaux semi-circulaires proposé par [Young & Oman, 1969] et des organes otolithiques proposé par [Young & Meiry, 1968]. Ce modèle a été utilisé dans de nombreuses études : [Asadi *et al.*, 2015a; Augusto, 2009; Chen & Fu, 2010; Fang & Kemeny, 2014; Garrett & Best, 2013; Lamprecht *et al.*, 2019; Reid & Nahon, 1985; Zacharias, 1978]. Le second modèle, proposé par [Telban & Cardullo, 2005], intègre également les modèles des canaux semi-circulaires et des organes otolithiques, mais avec des valeurs de constante de temps différentes. Il a été conçu plus récemment et il est l'un de plus utilisé parmi les MCA [Asadi *et al.*, 2017; Baseggio *et al.*, 2011; Beghi *et al.*, 2012; Bruschetta *et al.*, 2019; Mohammadi *et al.*, 2016; Zaychik *et al.*, 2009]. Bien que la plupart des chercheurs suggèrent l'utilisation d'un modèle vestibulaire précis afin d'améliorer la qualité de la restitution du mouvement, l'intégration d'un tel modèle dans un MCA sans aucun test expérimental n'est pas évidente. De fait, certains auteurs continuent à concevoir des MCA sans tenir compte d'aucun modèle de perception humaine [Dagdelen *et al.*, 2009; Fang *et al.*, 2017]. De plus, l'intégration du modèle de perception du mouvement rend la conception des lois de commande plus complexe et augmente le temps d'optimisation à mesure qu'augmente le nombre d'états dans le système.

Toutefois, à notre connaissance, aucune comparaison n'a été faite entre l'intégration d'un modèle de perception du mouvement et la simple utilisation de la dynamique de la plateforme dans le MCA pour un simulateur de conduite. De plus, les deux

modèles vestibulaires présentés ci-dessus n'ont pas été comparés dans des tests expérimentaux. Par conséquent, ce chapitre propose d'évaluer trois stratégies de mouvement qui dépendent directement du modèle mathématique utilisé dans la conception du MPC implicite. La première stratégie intègre le modèle dynamique de la plateforme sans aucun modèle de perception humaine (16), la seconde intègre le modèle de [Young & Oman, 1969] et [Young & Meiry, 1968] tandis que la troisième utilise le modèle de [Telban & Cardullo, 2005]. Tout au long de ce chapitre ils seront appelés M0, M1 et M2, respectivement.

Lorsque l'on compare la restitution du mouvement dans les simulateurs de conduite, on a tendance à se concentrer sur la différence des phases, l'amplitude des fréquences d'entrée et de sortie et la maximisation de l'espace de travail de la plateforme. Même si de telles considérations sont très importantes pour le contrôle du système, il faut également considérer l'opinion des conducteurs. Ainsi, nous allons étudier le rôle du modèle de perception du mouvement dans le MCA, du point de vue du contrôle mais aussi du point de vue de l'utilisateur.

Ce chapitre traite également de l'influence du comportement du conducteur sur la préférence pour chacune des stratégies. Puisque tous les conducteurs ne se comportent pas de la même façon, nous faisons l'hypothèse que le comportement de conduite réel dans une situation de conduite a un impact sur la perception des mouvements dans le simulateur de conduite. Par conséquent, nous analysons subjectivement et objectivement l'influence que le comportement de conduite des sujets peut avoir sur la préférence pour chacune des stratégies M0, M1 ou M2.

Pour évaluer le comportement de conduite, plusieurs questionnaires d'auto-évaluation ont été utilisés au cours des dernières décennies. Ceux-ci ont pour objectif d'évaluer les décisions et les actions des conducteurs sur la route, comme indicateurs de leur comportement de conduite habituel. Parmi les auto-évaluations existantes, nous citons :

- Le questionnaire sur le comportement des conducteurs (DBQ pour *Driving Behaviour Questionnaire* [Reason *et al.*, 1990]), conçu pour étudier et classer les comportements de conduite aberrantes.
- L'inventaire des comportements de conduite (DBI pour *Driving Behaviour Inventory* [Gulian *et al.*, 1989]), mis au point pour étudier les ordres de grandeur du stress des conducteurs.
- Le questionnaire sur le style de conduite (DSQ pour *Driving Style Questionnaire* [French *et al.*, 1993]), proposé pour évaluer les styles de prise de décision.
- L'inventaire des compétences des conducteurs (DSI pour *Driver Skill Inventory* [Lajunen & Summala, 1995]), utilisé pour comparer les capacités de conduite auto-déclarées avec les compétences générales des conducteurs.
- L'inventaire multidimensionnel des styles de conduite (MDSI pour *Multidimensional Driving Style Inventory* [Taubman-Ben-Ari *et al.*, 2004]), qui

présente une auto-évaluation complète et multidimensionnelle du comportement au volant.

Le comportement du conducteur peut également être évalué de manière objective, en prélevant les données du conducteur, telles que la vitesse ou la position latérale directement du véhicule pendant la conduite [Espié *et al.*, 2013]. Cependant, pour cela, chacun des participants devrait conduire le simulateur pendant une longue durée et il serait plus difficile de classer leur type de comportement de conduite. C'est pourquoi, dans cette étude, nous considérerons les questionnaires DBQ et MDSI pour définir le comportement de conduite auto-déclaré de chaque participant. À notre connaissance, ce sont les seuls qui décrivent, de façon générale, le comportement de conduite en se fondant uniquement sur les habitudes comportementales des conducteurs.

Le DBQ est l'un des questionnaires les plus utilisés en ce qui concerne les comportements de conduite, mais il tient compte uniquement des comportements de conduite aberrants. La version originale du DBQ [Reason *et al.*, 1990], ainsi que d'autres versions modifiées, ont été utilisées pour évaluer différentes caractéristiques telles que la localisation géographique [Bener *et al.*, 2008; Guého, 2014; Smorti & Guarnieri, 2016; Stanojević *et al.*, 2018; Stephens & Fitzharris, 2016], l'âge ou le sexe [Martinussen, 2013], l'influence culturelle [Lajunen *et al.*, 2004]. Les conditions des scénarios de conduite [Zhao, 2012] ont même été utilisées dans des études liées aux simulateurs de conduite [Helman & Reed, 2015]. D'autre part, le MDSI est plus récent et moins utilisé que le DBQ. Cependant, il classe le comportement de conduite d'une manière beaucoup plus simple et compréhensible. Il a été conçu à partir de mesures d'auto-évaluation précédentes pour créer une conceptualisation unique et multidimensionnelle des styles de conduite. Bien que les mesures du questionnaire soient subjectives, certains auteurs ont montré qu'elles étaient de bons indicateurs du comportement de conduite sur la route : [Hooft van Huysduynen *et al.*, 2018; Kaye *et al.*, 2018; Taubman *et al.*, 2016]. C'est pourquoi, nous évaluerons de manière psychométrique (cohérence interne et fiabilité) une version commune des deux questionnaires afin d'aborder le comportement de conduite dans le simulateur de conduite et corréler ce comportement avec la préférence des différents stratégies M0, M1 et M2.

Toutefois, la relation entre la préférence du modèle mathématique MCA et le comportement de conduite est fondée uniquement sur des données subjectives. Afin donc de compléter cette étude avec une analyse objective, nous utilisons les mesures recueillies après avoir conduit le simulateur. Les recherches développées par [Kaye *et al.*, 2018] sur la base de 20 études différentes montrent qu'il existe des similitudes et une corrélation positive entre le comportement des conducteurs et les mesures objectives. Parmi les nombreux indicateurs de performance des conducteurs existants que l'on peut trouver dans la littérature, nous considérerons ceux qui se sont avérés avoir une influence significative sur la validation du comportement de conduite auto-déclaré et la restitution des mouvements dans un simulateur de conduite. Ainsi, la moyenne et l'écart-type de la vitesse [Helman & Reed, 2015; Taubman Ben-Ari & Skvirsky, 2016] ; la moyenne et l'écart-type de la

position latérale [Hooft van Huysduynen *et al.*, 2018] et la moyenne des à-coups (*jerk*) [Haycock & Grant, 2007] seront pris en tant qu'indicateurs de performance dans le simulateur de conduite.

5.1.1 Présentation des différents modèles

Cette partie décrit les modèles utilisés dans les stratégies M0, M1 et M2. Tout d'abord, il faut préciser que tous les modèles sont conçus pour une optimisation à 2 DDL comme expliqué dans la partie 3.4.1. Le modèle M0 est représenté par le système (17) et les modèles M1 et M2 par le système (20). Même si ces deux derniers partagent le même système d'état, leurs paramètres leur confèrent un comportement en fréquence très différent. Pour les organes otolithiques, ce comportement est détaillé dans la partie 2.3.5 et illustré par le diagramme de *Bode* de la figure 13. Le comportement pour les canaux semi-circulaires est décrit dans la partie 2.3.6 et illustré par la figure 14.

Les trois stratégies intègrent la même conception de commande pour les accélérations linéaires et angulaires. La figure 53 illustre la structure complète du MPC pour les accélérations selon l'axe x, dans lequel les signaux d'entrée sont des accélérations linéaires et angulaires envoyées à partir des commandes du cockpit du simulateur.

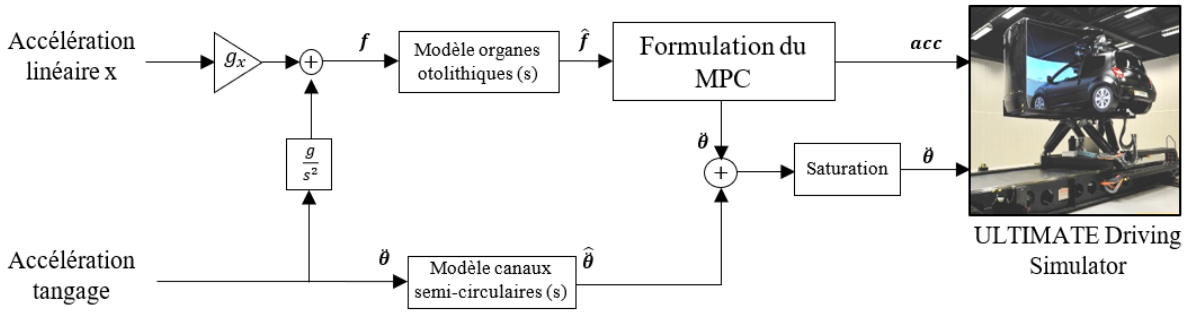


FIGURE 53 – Conception de la commande pour la restitution des mouvements du simulateur de conduite.

Étant donné que le suivi complet d'une référence n'est pas forcément la meilleure solution pour la perception de mouvement dans les simulateurs, les signaux d'entrée ont été mis à l'échelle en fonction de chaque stratégie. Par conséquent, nous avons sélectionné un gain de $g_x = 0,6$ pour mettre à l'échelle le signal de force spécifique f. La même valeur a été implémentée selon l'axe y, car elle fournit un équilibre entre la perception visuelle et vestibulaire [Greenberg *et al.*, 2003].

Comme la stratégie M0 n'intègre pas le système vestibulaire, les blocs qui représentent les fonctions de transfert pour les organes otolithiques et les canaux semi-circulaires sont inclus uniquement dans les configurations de M1 et M2. Ainsi, lorsque nous utilisons le modèle M0, ces blocs sont remplacés par un gain unitaire. Après cette phase de mise à l'échelle des accélérations linéaires et angulaires, une

optimisation à 2 DDL fondée sur la commande prédictive implicite est effectuée. Le résultat donne les valeurs optimales des accélérations linéaires et angulaires de la technique de *tilt coordination* (voir partie 2.3.8).

Pour réaliser un test utilisateur, nous avons décidé de ne pas inclure le modèle mathématique des canaux semi-circulaires, représenté dans l'équation (2), directement dans la structure du MPC car celui-ci génère du bruit dans les signaux de rotation et que nous privilégions l'optimisation de la force spécifique par la technique de *tilt coordination*. D'ailleurs, dans la plupart des scénarios de conduite, les angles résultants de l'accélération continue, grâce à l'apport de la composante de la gravité $g\theta$, remplissent les limites d'inclinaison de la plateforme. C'est pourquoi, le modèle de canaux semi-circulaires est utilisé uniquement comme un filtre des accélérations angulaires qui proviennent directement du modèle du véhicule. Ainsi, le bloc de saturation représente un filtre passe-haut qui sature le signal d'entrée de l'accélération angulaire. Ce signal est généré par le conducteur dans le simulateur, puis adapté au comportement du modèle du véhicule. La saturation est nécessaire afin de ne pas dépasser les limites physiques de la plateforme en termes d'angles, vitesses et accélérations angulaires.

5.1.2 Réglage préliminaire pour l'expérimentation

Le simulateur utilisé dans cette expérimentation est l'ULTIMATE de Renault (voir figure 6a). Il utilise une plateforme de développement (PDS pour *Plateform developement system*), qui permet d'intégrer tous les modèles nécessaires pour le bon fonctionnement du simulateur. La PDS est implémentée grâce à RT-LAB [1]. Comme nous l'avons expliqué dans le chapitre précédent, nous ne pouvons pas encore implémenter les réseaux des neurones récurrents dans le simulateur de conduite ULTIMATE. Toutefois, afin de résoudre le problème de programmation quadratique (48), l'outil open-source qpOASES est implémenté [Ferreau *et al.*, 2014]. Il applique la méthode des ensembles actifs et fournit un package en *C*++ prêt à l'emploi et capable de trouver l'action de contrôle optimale ΔU en temps réel. Cette méthode d'optimisation a été intégrée et adaptée à notre contrôle de plateforme, afin de qu'elle puisse être compilable sous Windows et sous Linux.

Les trois stratégies M0, M1 et M2 définissent un contrôleur MPC qui résout un problème d'optimisation spécifique pour obtenir la trajectoire de référence désirée (voir partie 3.4 pour une explication détaillée). Néanmoins, la fonction de coût varie d'une stratégie à l'autre. Dans la configuration M0, l'objectif est de suivre fidèlement les accélérations envoyées du conducteur vers la plateforme dynamique, alors que dans les configurations M1 et M2, l'objectif est orienté vers la minimisation de l'erreur de perception de mouvement entre la plateforme et le modèle du véhicule

1. Logiciel de simulation en temps réel produit par OPAL-RT qui offre la flexibilité et l'évolutivité nécessaires pour réaliser des applications de simulation en temps réel. `https://www.opal-rt.com/fr/logiciel-rt-lab/`

réel. Dans les trois configurations, le problème d'optimisation est quadratique :

$$\begin{aligned} &J = (Y-R)^T Q_\delta (Y-R) + \Delta U^T Q_\lambda \Delta U + X^T Q_q X \\ &s.t. \quad A_c \Delta U \leq b \end{aligned} \tag{48}$$

Où $Q_\delta > 0$, $Q_\lambda > 0$ et $Q_q > 0$ sont des matrices de pondération symétriques de dimensions compatibles pour l'erreur de suivi, le contrôle et les états futurs, respectivement. Les valeurs de ces matrices de pondération ont été choisies avec la méthode d'essai-erreur pour les trois stratégies, afin d'améliorer les performances du système et assurer la faisabilité du contrôle. Fondé sur la structure du problème d'optimisation de l'équation (25), nous présentons les valeurs de pondération dans le tableau 11.

MCA	λ		q			δ
	Δu_{rot}	Δu_{lin}	x, $\dot{x}$, θ, ω	u_{rot}	u_{lin}	$Erreur$
M0	300	1	0	0,1	0,1	50/j
M1	100	0,2	0	0,1	0,01	80/j
M2	100	10	0	0,1	0,1	50/j

La valeur de j varie selon l'horizon de prédiction Np, comme présenté dans l'équation (25) .

TABLEAU 11 – Paramètres de pondération des stratégies M0, M1 et M2.

La matrice R représente la trajectoire de référence qui reste constante sur tout l'horizon de prédiction. Elle prend la valeur de l'accélération linéaire de la plateforme pour M0 et la valeur de la force spécifique perçue pour M1 et M2. Les matrices Y et X représentent la sortie future et les états futurs du système, respectivement. Leurs valeurs changent selon le type de configuration, puisqu'elles dépendent directement du modèle mathématique. ΔU est la variable de décision et représente le taux de variation de l'accélération angulaire et linéaire. L'équation $A_c \Delta U \leq b$ représente les contraintes d'inégalité linéaires indiquées dans le système d'équations (22).

En plus de l'algorithme d'optimisation, plusieurs considérations doivent être prises en compte pour déployer le MPC en temps réel, c'est à dire avec une fréquence de contrôle souhaitée de 125 Hz et sans présenter aucun risque pour la plateforme ou l'utilisateur lui-même. Tout d'abord, nous avons discrétisé les systèmes de M0, de M1 et de M2 de façon non-linéaire, comme présenté dans la partie 4.1.1, afin d'élargir l'horizon de prédiction. À cet égard, nous utilisons également différentes matrices de pondération le long de l'horizon de prédiction.

En ce qui concerne la faisabilité de la solution pour les différentes stratégies, nous avons sélectionné deux approches différentes pour assurer la stabilité en boucle fermée. La première consiste à introduire une matrice de pénalité terminale et la seconde à appliquer une condition de stabilité supplémentaire proposée par [Fang *et al.*, 2017]. En effet, en présence de contraintes d'état, le MPC devient difficile à contrôler et la stabilité en boucle fermée dépendra de la solution optimale. Ces méthodes ont été expliquées et évaluées dans la partie 4.1.

5.1.3 Restitution de mouvement selon le type de modèle

Afin d'analyser les trois stratégies en termes de fidélité de restitution de mouvement dans le simulateur ULTIMATE, nous avons utilisé les données recueillies après avoir conduit le simulateur en mode autonome. Pour cela, nous avons utilisé le scénario de terrain présenté dans la figure 56a. Nous détaillerons le choix de cette trajectoire et le modèle de conduite autonome dans la partie 5.2.3. Le résultat est un signal de 250 secondes, contenant des accélérations transitoires et continues. Ce signal d'entrée est référencé par f_x dans la figure 54 ; il représente le comportement du modèle du véhicule autonome selon l'axe x.

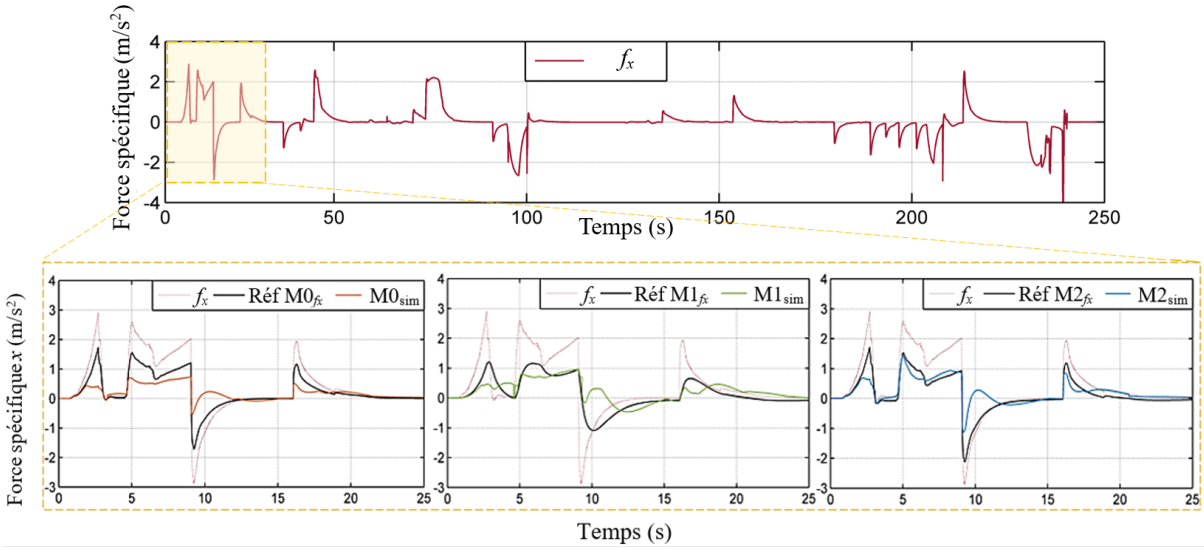


FIGURE 54 – Restitution du mouvement selon le modèle mathématique du MPC : l'image du haut représente l'accélération du modèle autonome selon l'axe x ; les figures du bas sont respectivement, de gauche à droite, le signal de poursuite sans modèle de perception M0, avec le modèle M1 et avec le modèle M2.

Les courbes de la figure 54 montrent la restitution du mouvement dans une fenêtre de 25 secondes pour faciliter la compréhension du suivi pour chaque configuration M0, M1 et M2. La force spécifique f_x est le signal d'entrée général pour toutes les configurations. Néanmoins, celle-ci est mise à l'échelle de manière différente selon le modèle, c'est-à-dire, pour les configurations M1 et M2, f_x passe à travers un filtre qui correspond au modèle des organes otolithiques de M1 et M2, respectivement. D'autre part, dans la configuration M0, f_x reste le même signal avec un gain inférieur g_x, comme illustré dans le schéma 53.

Tout au long de ce chapitre, nous comparerons la fidélité du mouvement pour les accélérations linéaires et angulaires. Pour ce faire, nous observerons les accélérations transitoires et continues. Les premières seront observées dans les hautes fréquences telles que les freinages ou les démarrages. Les deuxièmes se référeront aux accélérations générées par la composante de gravité grâce à la technique de *tilt*

coordination et correspondront aux accélérations basses fréquences.

Lorsqu'un MCA est analysé par rapport à la dynamique du véhicule réel, l'on espère que, dans le meilleur des cas, le module des accélérations linéaires et angulaires du simulateur sera égal à l'unité et que la phase sera nulle. Cependant, comme le simulateur se déplace dans les limites physiques du système de mouvement et selon les limites perceptives imposées dans l'optimisation, une transformation des accélérations $1:1$ sur toute la gamme de fréquences de conduite n'est pas possible.

En effet, nous pouvons observer dans la figure 54 qu'aucune des trois stratégies ne suit fidèlement la force spécifique en entrée. Néanmoins, nous observons que dans les cas de M0 et M2, la trajectoire suit un comportement proche de f_x, tout en sachant que M2 restitue beaucoup plus d'accélérations transitoires que M0. La stratégie M1 ne restitue pas bien les accélérations transitoires et investit beaucoup d'effort dans les accélérations continues. Ce fait génère un retard de phase plus important dans la restitution du mouvement, car la vitesse angulaire maximale a une limite imposée de $4°/s$. Cette valeur a été choisie en fonction de l'étude développée par [Fang *et al.*, 2014].

Toutes les limites utilisées dans cette étude sont présentées dans le tableau 12.

Limites	Position	Vitesse	Accélération
Hexapode x	$0,28m$	$0,7m/s$	$7,5m/s^2$
Hexapode y	$0,26m$	$0,7m/s$	$7,5m/s^2$
Hexapode z	$0,20m$	$0,4m/s$	$5,0m/s^2$
X rail	$2,5m$	$1,9m/s$	$3,2m/s^2$
Y rail	$2,5m$	$2,8m/s$	$4,5m/s^2$
Tilt technique/Roulis	$5,5°$	$4°/s$	$8°/s^2$
Tilt technique/Tangage	$5°$	$4°/s$	$8°/s^2$

TABLEAU 12 – Contraintes de la plateforme et de perception du mouvement utilisées dans l'expérimentation.

Un autre objectif du MCA pour les simulateurs à hautes performances est de tirer profit de l'espace de travail disponible. Ainsi, la figure 55 montre le déplacement des rails x et y lors de l'utilisation des trois différents MCA pendant la simulation.

Sur ce graphique, nous pouvons voir que les stratégies M0 et M2 utilisent toutes les parties de l'espace de travail, alors que la stratégie M1 n'utilise que la moitié de l'enveloppe de travail disponible. En effet, le modèle M1 filtre les hautes fréquences qui sont principalement restituées par les rails du simulateur.

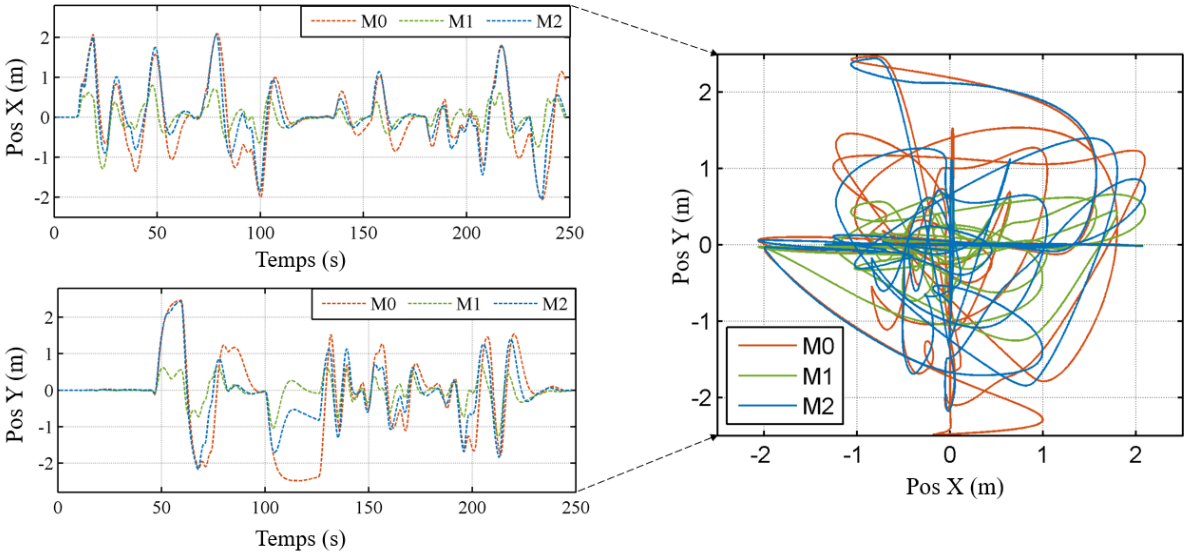


FIGURE 55 – Position de la plateforme selon les axes x et y.

5.2 Conditions expérimentales

Dans cette partie, nous présentons tous les éléments qui ont été nécessaires pour réaliser l'expérimentation dans le simulateur de conduite. Nous décrivons toutes les conditions nécessaires du simulateur dynamique, du contrôle de la plateforme, des modèles dynamiques, de l'environnement virtuel, de la procédure expérimentale et de tous les questionnaires utilisés dans cette étude.

5.2.1 Matériel

Les études se sont déroulées au centre de réalité virtuelle et de simulation immersive (DEA-TDV) chez Renault, France. Le simulateur à hautes performances utilisé est l'ULTIMATE (voir partie 1.4.5 pour une description détaillée).

Pour cette expérience, nous avons recruté 41 collaborateurs de Renault : 8 femmes et 33 hommes, âgés de 23 à 61 ans, avec une moyenne d'âge de 36,6 (écart-type : 11,5 ans). Nous n'avons pas utilisé des critères d'inclusion particulière pour les participants. Tous les sujets avaient le permis de conduire et ont signé un formulaire de consentement. Trois participants n'ont pas pu terminer l'expérimentation car ils ont ressenti des symptômes de mal du simulateur pendant la simulation. Par conséquent, 38 participants ont été inclus dans l'analyse des données. Les participants ont été séparés en deux groupes selon le mode de conduite. Le premier groupe nommé « **Libre** » était composé de 19 participants qui ont passé les trois MCA avec le contrôle total du véhicule, c'est-à-dire qu'ils conduisaient activement le simulateur. Le second groupe nommé « **AD** » était composé de 19 participants qui ont fait le test en utilisant un modèle de véhicule simulant la conduite autonome.

5.2.2 Procédure expérimentale

L'expérimentation a duré environ une heure par sujet et s'est déroulée en plusieurs étapes. La première phase a consisté à remplir le questionnaire sur le comportement de conduite auto-déclaré (voir partie 5.2.5) et quelques informations sur les sujets eux-mêmes, telles que l'âge et le sexe. Dans la deuxième étape, les sujets ont été installés dans le simulateur et des consignes de sécurité ont été données. Les participants ont été invités à prêter attention aux différentes situations qui pourraient affecter la perception du mouvement pendant la simulation (voir figure 56b). Les participants ont commencé par une phase de familiarisation qui a duré entre 150 et 200 secondes. Ils ont ensuite conduit le scénario décrit dans la partie 5.2.3 avec chacune des stratégies : M0, M1 et M2.

Comme indiqué précédemment, nous avons séparé les 38 participants en deux groupes différents, afin de comparer la conduite autonome et la conduite active. Les participants ont été répartis de manière aléatoire. Chaque sujet a conduit le simulateur quatre fois, incluant la phase de familiarisation et les trois MCA. Ceux-ci ont été présentés aux sujets dans un ordre aléatoire afin d'éviter des effets d'apprentissage. Pour le groupe Libre, il y avait des panneaux de circulation et des instructions verbales dans la phase de familiarisation indiquant la limite de vitesse et le parcours à suivre.

Entre chaque conduite, les participants ont rempli un questionnaire sur les indices de mouvement (voir la partie 5.2.6), afin d'évaluer, entre autres, la perception dans le simulateur pour chaque MCA. Après les quatre passages sur le simulateur, les sujets de chaque groupe ont dû classer par ordre de préférence les trois stratégies en indiquant d'abord la meilleure.

5.2.3 Scénario de conduite

Pour générer le scénario de conduite, le terrain et les situations de conduite ont été analysées auparavant afin d'évaluer leur pertinence dans la restitution du mouvement. Le terrain a été spécialement conçu pour générer des situations dans une ville, une autoroute, des sections d'insertion et de longs virages. Les situations de conduite on été conçues pour générer des accélérations transitoires et continues selon les axes de mouvement x et y. La génération du scénario a été réalisée avec le logiciel de simulation de conduite SCANeR Studio d'AVSimulation. Ce logiciel a également servi à créer les cas d'usages pour le mode véhicule autonome et le trafic utilisé. Le terrain est représenté dans la figure 56a.

Le premier segment se déroule en ville et la vitesse est limitée à 50 km/h. L'environnement virtuel de cette partie de route est représenté dans la figure 56b. Le deuxième segment correspond à une section d'insertion sur une autoroute avec une vitesse limitée à 50 km/h. Le troisième segment est une autoroute limitée à 100 km/h. Le quatrième segment est une sortie d'autoroute limitée à 50 km/h. Le

(a) Vue de dessus du terrain utilisé pour la simulation de conduite

(b) Capture d'écran de la vue des conducteurs du scénario virtuel dans la première situation

FIGURE 56 – Scénario utilisé dans le test utilisateur.

dernier segment correspond à une route urbaine avec une limite de vitesse de 70 km/h représentant une condition de type slalom grâce à quelques obstacles mis en place sur la route, afin de reproduire les accélérations latérales transitoires selon l'axe y. Des panneaux de signalisation sont présents tout au long du parcours afin d'indiquer la limite de vitesse respective pour chaque segment. La distance parcourue en simulation par chaque participant et pour chaque MCA est d'environ 2,8 km. Chaque segment a comme objectif de produire divers indices de mouvement dans le simulateur, en particulier des accélérations transitoires (haute fréquence) et continues (basse fréquence) selon les axes x et y comme suit :

Type A : accélérations x transitoires, par exemple, les situations Stop and Go et les passages pour piétons, cas d'utilisation en ville (premier segment).

Type B : accélérations continues selon l'axe x, par exemple, cas d'utilisation sur autoroute libre (troisième segment).

Type C : accélérations transitoires selon l'axe y, par exemple, cas d'utilisation en slalom (cinquième segment).

Typel D : accélérations y continues, par exemple, longs virages dans les sections d'insertion et de sortie (deuxième et quatrième segment).

5.2.4 Modèle du véhicule autonome

L'un des enjeux dans cette expérimentation est de comparer les MCA lorsque le sujet a le contrôle du véhicule (mode Libre) et lorsqu'il ne l'a pas (mode AD). En effet, le comportement du véhicule autonome peut être complètement différent du comportement habituel du conducteur. Cet effet peut affecter les réponses subjectives des sujets, car ils peuvent confondre la conduite autonome et la restitution des mouvements, au moment de remplir les questionnaires. C'est pourquoi il est important que le modèle du véhicule autonome soit comparable à une conduite réelle. Le comportement du véhicule autonome mis en œuvre pour le mode AD a été conçu sur la forme d'un script capable de traiter, en toute sécurité, les décisions de planification et les différents cas d'usages. Cependant, la plupart des actions et

comportements dynamiques de ce véhicule autonome ont été fournis par le modèle de trafic du logiciel de simulation SCANeR Studio auquel nous n'avons pas accès.

Le comportement de conduite du modèle autonome a été comparé aux signaux d'accélération d'un participant, choisi aléatoirement, après avoir conduit le scénario de la figure 56 dans le simulateur. Les résultats sont présentés dans les figures 57 et 58. Dans celui-ci, les signaux de type A se situent principalement au début de la simulation. Ils sont illustrés dans la zone comprise entre 0-50 secondes de la figure 57. Les signaux de type B sont illustrés dans la section entre 75-90 secondes dans la même figure. Les signaux de type C sont représentés par un scénario générique de type slalom, afin de générer une trajectoire sinusoïdale avec une accélération latérale maximale de $1m/s^2$. Ce cas d'utilisation est représenté sur la figure 58 entre 130 et 180 secondes. Le signal de type D, pour des accélérations continues selon l'axe y est montré dans la figure 58 entre 40-60 secondes et entre 100-175 secondes.

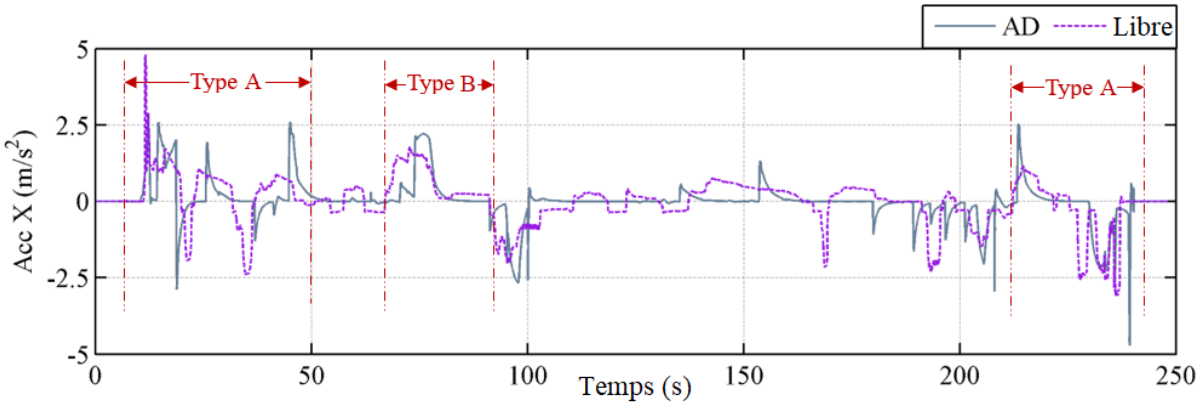


FIGURE 57 – Comparaison des accélérations selon l'axe longitudinal x entre le mode de conduite AD et le mode de conduite Libre, après avoir effectué le test.

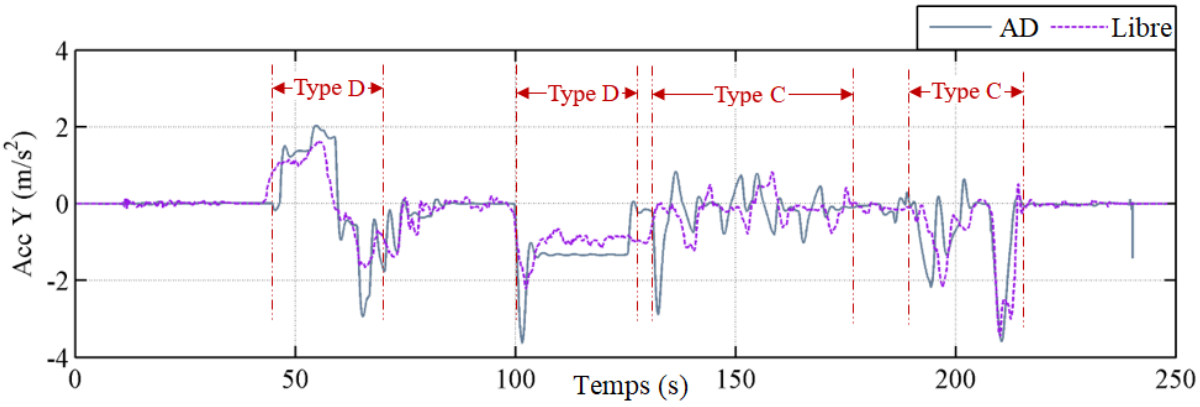


FIGURE 58 – Comparaison des accélérations latérales y entre le mode AD et le mode Libre.

La figure 58 montre que les accélérations latérales suivent presque la même trajectoire et les mêmes fréquences dans les modes Libre et AD. En effet, les signaux dépendent principalement de la trajectoire imposée par la morphologie du terrain et

pas nécessairement par le conducteur ou par le modèle autonome. En revanche, les signaux le long de l'axe x, représentés dans la figure 57, sont légèrement différents, surtout pour les accélérations transitoires positives. Cet effet dépend d'une part de l'anticipation et du style de conduite des conducteurs, et d'autre part du modèle dynamique du véhicule. La présence d'un décalage dans les deux figures s'explique par les différents profils de vitesse dans le scénario inhérents aux différents modes de conduite : « Libre » et « AD ».

5.2.5 Questionnaire d'auto-évaluation sur le comportement en situation de conduite

Des questionnaires sur les comportements de conduite auto-déclarés ont été mis en place afin de comprendre si la préférence du sujet concernant la restitution de mouvement était influencée par son comportement de conduite réel ou non. Pour cette étude, nous avons choisi d'utiliser deux des plus utilisés, le DBQ [Reason *et al.*, 1990] et le MDSI [Taubman-Ben-Ari *et al.*, 2004] afin d'aborder le comportement de conduite dans le simulateur de conduite.

Sur la base des questions existantes du DBQ, une version modifiée de ce questionnaire a été produite avec 28 éléments représentés dans le tableau 17. Ceux-ci ont été sélectionnés après avoir analysé les études précédentes : [Gabaude *et al.*, 2010; Guého, 2014; Martinussen, 2013; Mattsson, 2012; Zhao, 2012]. Le critère de sélection a été le coefficient de pondération : plus il était élevé, plus l'élément était important. Les 28 items sont tous présents dans des versions courtes du DBQ, telles que [Lajunen *et al.*, 2004] et [Reimer *et al.*, 2005], qui sont orientées pour des populations spécifiques. Ces questionnaires incluent les échelles de caractérisation présentées dans [Reason *et al.*, 1990] telles que les infractions, les erreurs et les défaillances ou manquements.

Le MDSI d'origine contient 44 questions pour traiter quatre styles de conduite différents : conduite anxieuse ou dissociative, conduite patiente et prudente, conduite imprudente et conduite en colère et hostile. Afin de réduire le nombre de questions du formulaire MDSI d'origine, nous avons sélectionné 27 questions dans le tableau 16 provenant d'études antérieures : [Taubman-Ben-Ari *et al.*, 2004], [Poó *et al.*, 2013], [Holman & Havarneanu, 2015]. Les 27 éléments choisis ne sont pas redondants et ont été considérés comme les plus appropriés pour évaluer le style de conduite en fonction des actions des conducteurs. Cette modification a dû être faite car la disponibilité du simulateur et le temps par participant pour compléter toute l'expérience était d'une heure.

Le questionnaire final obtenu est un formulaire composé de 47 questions (voir annexe D.1) issues des questionnaires DBQ et MDSI. Pour ce questionnaire modifié, les sujets ont indiqué la fréquence à laquelle ils se sont engagés dans un comportement spécifique en reposant leur jugement sur leur expérience personnelle de conduite. Chaque sujet devait indiquer, pour chaque question, la fréquence de ces

comportements sur une échelle de six points, où 1= jamais à 6= presque tout le temps.

5.2.6 Questionnaire sur la restitution de mouvement dans le simulateur

Après chaque passage sur le simulateur, les participants ont rempli l'index des échelles d'évaluation du mouvement de [Sullivan, 1985] qui évalue les différentes caractéristiques suivantes du MCA : douceur, perception du mouvement, retard, amplitude et inconfort, pour chacune des quatre situations : accélération/freinage (type A), accélération continue (type B), slalom (type C) et longs virages (type D). Pour chaque situation, les sujets devaient donner une note de 1 à 7 selon le tableau 13 (voir annexe D.2).

Caractéristique	**Score**	
	1 $\longrightarrow$	7
Douceur/Lissage	Extrêmement lisse, comparable à une plateforme fixe	Extrêmement brusque, limite de tolérance
Perception du mouvement	Totalement juste, comme dans une vrai véhicule	Complètement inversée
Amplitude	Aucun mouvement n'a été ressenti	Au moins deux fois plus fort que prévu
Retard	Aucun	Excessif, inacceptable
Inconfort	Aucun	Impossible de continuer le test
Désorientation	Aucun	Impossible d'effectuer des manœuvres

TABLEAU 13 – Échelle d'aide pour noter subjectivement de 1 à 7 les différentes caractéristiques du MCA. Tiré de [Sullivan, 1985].

Dans la dernière partie du questionnaire, les participants ont noté toutes les configuration du MCA en utilisant une échelle de type *Likert* à sept points (1 : « pas du tout d'accord », 7 : « tout à fait d'accord ») [Berthoz *et al.*, 2013]. Jusqu'à quatre affirmations devaient être notées : « J'ai oublié le simulateur », « Le mouvement était réaliste », « J'avais l'impression de conduire » et « Je conduisais comme d'habitude » (en mode conduite libre seulement).

Après que les participants ont conduit le simulateur 4 fois : familiarisation, stratégies M0, M1 et M2, les participants ont été invités à classer les stratégies M0, M1 et M2 selon le type de préférence.

5.2.7 Acquisition de données

L'analyse des comportements de conduite auto-déclarés a été principalement fondée sur le jugement subjectif des conducteurs. Cependant, la méthode subjective par elle-même n'est pas suffisamment transparente et conduit à des résultats ambigus dans l'analyse du type de conduite. Par conséquent, il existe des repères dynamiques dans les simulateurs de conduite qui peuvent compléter de façon objective les résultats subjectifs. D'ailleurs, nous voulons évaluer l'influence du style de conduite sur la préférence de l'algorithme de restitution ; M0, M1 ou M2. À cet égard, nous avons recueilli plusieurs données des participants qui ont conduit le simulateur en mode Libre pour chacun des MCA jugés.

Comme expliqué précédemment, le terrain utilisé pour le scénario de test a été divisé en cinq segments de route (voir figure 56a). Ainsi, les données ont été recueillies pour chacun des segments. Nous tenons à signaler que les connexions entre chaque segment n'ont pas été prises en compte dans la collecte des données.

Les mesures ou indicateurs de conduite retenus par participant étaient la moyenne des à-coups (dérivée de l'accélération, plus communément appelé *jerk*) en m/s^3, la vitesse en km/h, la position latérale en mètres, ainsi que l'écart-type pour la vitesse et la position latérale. Ces paramètres ont été choisis en fonction de leur pertinence dans le comportement de conduite d'après des études antérieures : [Haycock & Grant, 2007; Helman & Reed, 2015; Hooft van Huysduynen *et al.*, 2018; Taubman *et al.*, 2016]. Nous avons d'abord pris les données de conduite de tous les sujets après avoir conduit le simulateur avec les trois stratégies. Puis, nous avons effectué la moyenne des mesures pour chacun des MCA, dans chacun des cinq segments de route.

Le *jerk* a été calculé comme le taux de changement de l'accélération. La valeur du *jerk* x ou y a été prise en fonction du segment, car elle dépend du signal de restitution du mouvement. Par exemple, dans les segments de route n°1 et n°3, les situations ont produit principalement des accélérations linéaires x alors que dans les autres segments de route, les situations ont produit des accélérations linéaires y.

La position latérale a été fournie par le logiciel de simulation SCANeR Studio comme suit : lorsque le véhicule se trouvait au milieu de la voie, la position latérale était égale à zéro, complètement à droite, égale à 1,5 m et complètement à gauche, égale à -1,5 m. Nous n'avons considéré que les valeurs absolues pour évaluer la moyenne. Pour cet indicateur, le dernier segment de route qui représente la condition de slalom (n°5) n'a pas été pris en compte, car le véhicule change constamment de voie. De ce fait, la comparaison du comportement des participants n'est pas pertinente.

5.3 Analyse expérimentale

L'analyse a consisté en plusieurs étapes principalement divisées en une analyse pour les données objectives et une analyse pour les données subjectives. La figure 59 illustre la manière dont les analyses de cette étude ont été menées. L'analyse pour les données subjectives intègre les questionnaires sur la restitution de mouvement et sur le comportement en situation de conduite. L'analyse pour les données objectives concerne la collecte des données des participants et l'évaluation des indicateurs de perception de mouvement pour chacun d'entre eux.

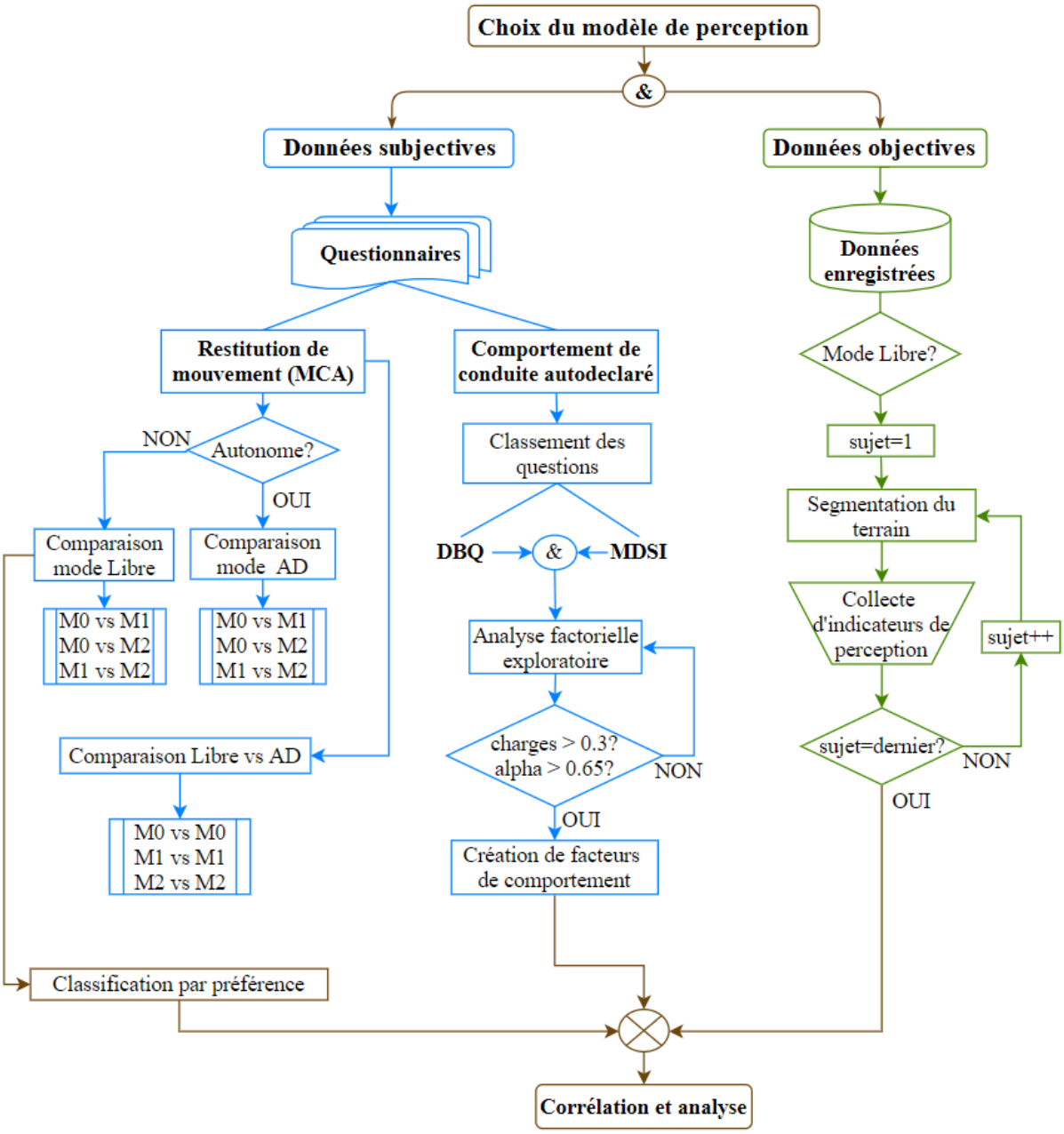


FIGURE 59 – Description générale du cadre d'analyse.

Pour le questionnaire MCA, deux tests différents ont été réalisés. Le premier a consisté à comparer les caractéristiques du MCA (voir partie 5.2.6) entre les trois configurations M0, M1 et M2. Chaque caractéristique de chaque stratégie a été regroupée au sein d'une paire, par exemple pour le retard : M0 avec M1 ; M1 avec M2 et M0 avec M2. Cette analyse a été effectuée dans les deux modes de conduite, AD et libre. Le second a consisté à comparer, par MCA, les caractéristiques entre les modes AD et de conduite libre, par exemple pour le retard : M0-Libre avec M0-AD ; M1-Libre avec M1-AD et M2-Libre avec M2-AD. Nous avons réalisé ces analyses pour chaque type de signal A, B, C et D comme expliqué dans la partie 5.2.3.

Pour le questionnaire sur le comportement au volant, une analyse a été effectuée pour classer séparément les 28 éléments du DBQ et les 27 éléments du MDSI dans des groupes spécifiques appelés facteurs.

Afin de pouvoir comparer les données de simulation des 19 sujets qui ont conduit activement le simulateur et leur comportement de conduite, nous les avons séparés en groupes de préférence MCA, c'est-à-dire les participants qui ont préféré la stratégie M0 (groupe P0), les participants qui ont préféré la stratégie M1 (groupe P1) et les participants qui ont préféré la stratégie M2 (groupe P2). Cela a permis de mieux classer chacun des indicateurs de mouvement par groupe, puis d'analyser la corrélation entre les données des groupes et leur comportement de conduite auto-déclaré.

En ce qui concerne l'analyse objective, nous avons recueilli les données de simulation pour chacun des participants ayant conduit le simulateur en mode Libre. Puis, pour chaque groupe P0, P1 et P2, des tests de corrélation ont été effectués entre le comportement de conduite déclaré par les conducteurs et les indicateurs de mouvement de conduite.

5.3.1 Comparaison selon le type de conduite

Comme nous l'avons décrit précédemment, après chaque passage dans le simulateur, le participant remplissait un formulaire qui évalue les différentes caractéristiques du MCA : lissage, perception, retard, amplitude, inconfort. Cette évaluation est effectuée pour chacune des quatre situations : Type A, Type B, Type C et Type D. Nous avons comparé les configurations M0, M1 et M2 deux à deux.

Pour chaque paire, nous avons comparé tous les caractéristiques dans chaque situation. Dans le tableau 14 nous montrons un exemple pour la caractéristique « Retard » dans la situation de longs virages.

La normalité de la distribution des données pour toutes les caractéristiques du MCA a été vérifiée en utilisant le test de Shapiro-Wilk. Aucune distribution ne suit une loi normale. Par conséquent, nous avons utilisé des tests de Wilcoxon non paramétriques appariés avec un niveau de significativité de 5%, afin de déterminer les différences significatives entre toutes les caractéristiques des MCA pour chaque

Situation	Caractéristique	Comparaison de stratégies
Longs virages : Signal de type C	Retard	M0 *contre* M1
	Retard	M1 *contre* M2
	Retard	M0 *contre* M2

TABLEAU 14 – Exemple de comparaison selon le type de conduite pour la caractéristique « Retard » dans les longs virages.

stratégie : M0, M1, M2. Dans ce test, les populations à comparer proviennent d'observations répétées sur le même sujet.

Trois tests de Wilcoxon séparés (correspondant aux trois MCA) ont été effectués pour chaque caractéristique. Nous avons donc appliqué une correction de Bonferroni conduisant à un niveau de significativité de : $p < 0.05/3 = 0.016$. Nous avons examiné séparément le groupe de 19 personnes pour le mode de conduite autonome (AD) et le groupe de 19 personnes pour le mode de conduite active (Libre).

Mode de conduite active « Libre »

Afin de comparer les stratégies de contrôle en mode Libre, un total de 84 tests bidirectionnels de Wilcoxon ont été effectués. Le même nombre de tests a été effectué en mode autonome. Par conséquent, pour une meilleure lisibilité et ne pas surcharger cette partie d'analyses, tous les résultats et les figures présentés à partir de maintenant correspondent uniquement aux données qui sont statistiquement différentes. La figure 60 résume les moyennes et les écarts-types des caractéristiques qui sont statistiquement différentes en mode de conduite active.

Les tests ont montré que les accélérations continues selon les deux axes x (type B) et y (type D) présentent des différences significatives au niveau du retard. Selon l'axe x, le retard de M1 est supérieur à celui généré par M2 ($Z = 2.18, p = 0.014$), selon l'axe y, le retard de M0 est moins important que celui de M1 ($Z = 2.18, p = 0.015$). Dans les deux cas, la stratégie M1 présente un retard plus important par rapport aux deux autres stratégies. Nous pouvons expliquer ce résultat en analysant la tâche de suivi de M1 dans l'optimisation du MPC : en appliquant le modèle des organes otolithiques de M1 les accélérations à haute fréquence sont filtrées. Par conséquent, lorsque l'on minimise l'erreur de perception de la force spécifique $f = a_{lin} + g\theta$, les accélérations angulaires et donc les angles d'inclinaison ont plus d'importance que les accélérations linéaires, car celles-ci ont été filtrées avant d'entrer dans la boucle d'optimisation. Nonobstant, le seuil de vitesse angulaire, imposé pour générer un effet d'accélération et non un effet de rotation, génère un retard lorsque les angles d'inclinaison sont importants. Le retard de M1 était aussi plus élevé dans le slalom (Type C) par rapport à M0 ($Z = 2.39, p = 0.009$). Dans le même cas d'utilisation, M2 présente un retard plus important que la stratégie M0 ($Z = 2.16, p = 0.015$).

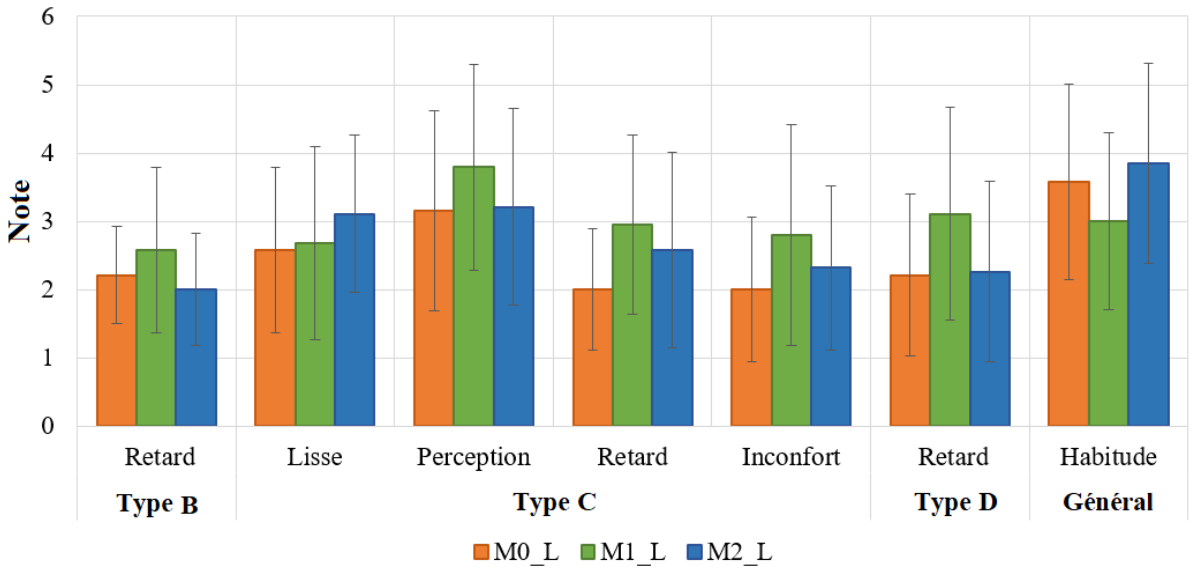


FIGURE 60 – Résultats pour les caractéristiques MCA avec une différence statistiquement significative dans le mode de conduite libre.

Pour les autres caractéristiques, dans le cas du slalom (Type C), nous avons trouvé plusieurs différences entre les configurations : la stratégie M0 était plus lisse que M2 ($Z = 2.31, p = 0.011$) ; la perception du mouvement était presque inversée en utilisant M1 au lieu de M2 ($Z = 2.35, p = 0.009$), et l'inconfort était inférieur avec M0 par rapport à M1 ($Z = 2.35, p = 0.009$). En ce qui concerne la douceur (caractéristique dite lisse dans la figure), les résultats des tests sont en accord avec la performance de suivi pour la stratégie M2 dans laquelle les accélérations transitoires sont plus importantes par rapport aux stratégies M0 et M1. Ces composantes hautes fréquences du mouvement sont importantes dans l'estimation de la vitesse du véhicule. Sans elles, les conducteurs pourraient ressentir un mouvement différent de celui auquel ils s'attendaient, tel qu'observé avec la stratégie M1.

Un autre résultat indique que les participants ont eu tendance à donner une note plus élevée à l'évaluation « conduire comme d'habitude » en utilisant la configuration M2 plutôt que M1 ($Z = 2.14, p = 0.017$). Nous précisons que même si la valeur p est plus élevée que le niveau de significativité avec la correction de Bonferroni ($p = 0.016$), nous avons trouvé pertinente d'indiquer le résultat. En effet, ce résultat est conforme à l'absence de restitution des accélérations transitoires avec la stratégie M1.

Un résultat inattendu est qu'aucune différence significative n'a été trouvée en ce qui concerne les accélérations transitoires le long de l'axe x (Type A), même si la restitution de mouvement pour ce type de signal était considérablement différente pour les trois configurations comme le montre la figure 54.

Mode de conduite autonome « AD »

La figure 61 montre les résultats pour lesquels une différence significative a été trouvée, obtenus lors de la comparaison des caractéristiques de chaque type de signal, pour chacune des stratégies.

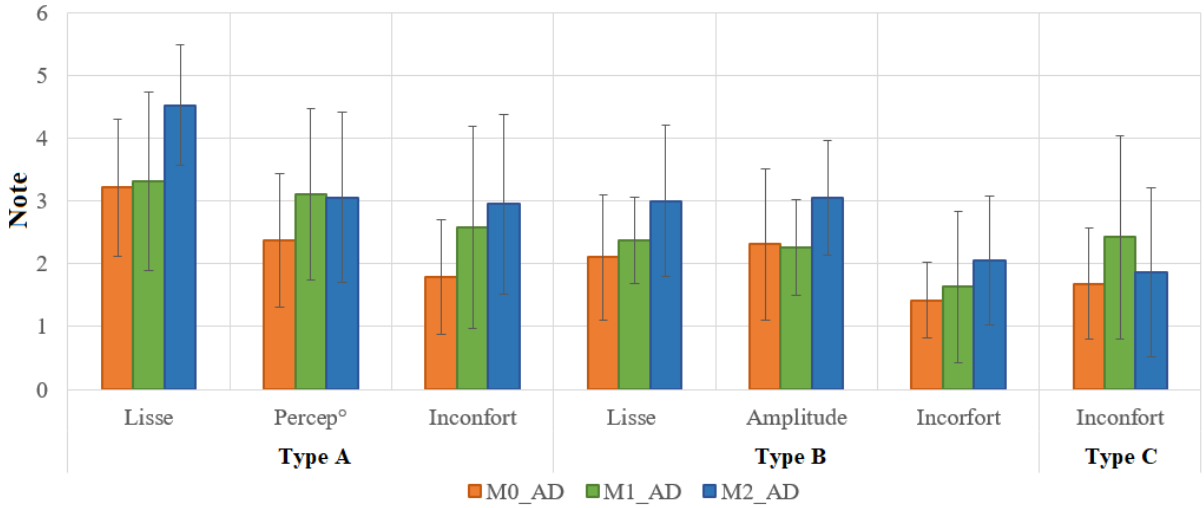


FIGURE 61 – Résultats pour les caractéristiques MCA avec une différence statistiquement significative dans le mode de conduite autonome (AD).

Par rapport au mode de conduite libre, nous observons plusieurs différences entre les caractéristiques, notamment pour les accélérations selon l'axe x. Pour les accélérations haute fréquence, la stratégie M2 est moins douce que M0 ($Z = 3.32, p = 0.0004$) et M1 ($Z = 2.92, p = 0.002$); la perception du mouvement est meilleure avec la stratégie M0 par opposition aux stratégies incluant un modèle vestibulaire, M1 ($Z = 2.07, p = 0.018$) et M2 ($Z = 2.24, p = 0.001$); l'inconfort est plus grand avec M1 ($Z = 2.31, p = 0.010$) et M2 ($Z = 2.76, p = 0.003$) par rapport à M0, mais il n'y pas de différence significative entre M1 et M2. Pour les accélérations à basse fréquence, M0 est encore plus lisse ($Z = 2.68, p = 0.004$) et plus confortable ($Z = 2.18, p = 0.014$) que M2, et la valeur de l'amplitude de M2 est plus grande que celle de M0 ($Z = 2.15, p = 0.015$) et M1 ($Z = 2.62, p = 0.004$). Ces résultats montrent que la stratégie M2 est plus agressive que les stratégies M0 et M1 pour les basses et hautes fréquences, ce qui est cohérent avec le suivi de mouvement de M2 pour les accélérations transitoires, mais pas pour les basses fréquences. Cela peut signifier que les participants n'ont pas fait de distinction entre les deux types de fréquences, ce qui rend leur opinion biaisée pour les hautes fréquences. Par conséquent, en général, les participants ont estimé que la restitution du mouvement M2 selon l'axe x est plus saccadée que la conduite normale. Dans l'ensemble, la stratégie M0 a été la plus appréciée pour la restitution des mouvements selon l'axe x, car elle a obtenu de meilleurs résultats en matière de confort et de perception des mouvements par rapport aux stratégies utilisant un modèle vestibulaire dans la conception de la boucle de contrôle. Une seule différence a été constatée selon l'axe y : l'inconfort est plus grand pour les accélérations à haute fréquence avec M1 plutôt qu'avec M0 ($Z = 2.12, p = 0.016$).

Pour résumer, les participants ont préféré les caractéristiques fournies par la stratégie M0 plutôt que les stratégies intégrant un modèle de système vestibulaire (M1 et M2). Cela peut s'expliquer par le modèle de conduite autonome utilisé dans cette expérimentation. Comme expliqué précédemment dans la partie 5.2.4, le modèle autonome a été programmé pour réagir à chaque cas d'utilisation. Néanmoins, leur profil d'accélération dépend du modèle de trafic fourni par le logiciel SCANeR Studio ainsi que de la dynamique du modèle de véhicule, auxquels nous n'avons pas accès. Les accélérations transitoires ont été perçues de manière agressive et, par conséquent, le MCA M0 a obtenu de meilleurs résultats que les deux autres stratégies, car il établit un équilibre correct entre les accélérations à haute et à basse fréquences.

5.3.2 Comparaison entre le mode « AD » et « Libre »

Dans cette partie, nous évaluons les caractéristiques de perception entre les groupes des sujets qui ont conduit le simulateur en mode AD et en mode Libre. Pour ce faire, des tests non paramétriques de Mann-Whitney ont été utilisés. Les résultats significatifs des tests sont présentés dans le tableau 15.

Nous observons que pour tous les types d'accélération, il existe, au moins, des différences dans les caractéristiques : perception du mouvement, désorientation et retard. Pour ces caractéristiques, les sujets ont eu tendance à noter plus sévèrement le comportement du simulateur avec le mode de conduite Libre qu'avec le mode de conduite AD. Une explication possible est que les sujets en mode Libre ont comparé leur conduite habituelle sur route avec le mouvement du simulateur : les sujets ne connaissent pas le comportement d'un véhicule autonome, puisqu'ils n'existent pas encore, par conséquent, dans le simulateur, ils évaluent la conduite autonome elle-même et non pas le mouvement du simulateur. Cet effet est particulièrement visible lorsque l'on considère, par exemple, la stratégie M0 : en mode AD, M0 est mieux apprécié que les deux autres stratégies concernant la perception et l'orientation du mouvement, ce qui n'a pas été le cas en mode de conduite Libre. Une autre observation qui appuie cette hypothèse est la différence dans la caractéristique lisse/douceur pour la configuration M2 : en mode AD, cette configuration a été jugée comme étant agressive, tandis qu'en conduite libre, la restitution du mouvement avec M2 était cohérent avec les accélérations continues et transitoires de manière générale, les participants ont préféré cette stratégie à M1. Afin d'observer plus clairement les résultats, nous indiquons dans la figure 62 les moyennes et les écart-types de chaque caractéristique qui présentent une différence significative entre les deux groupes.

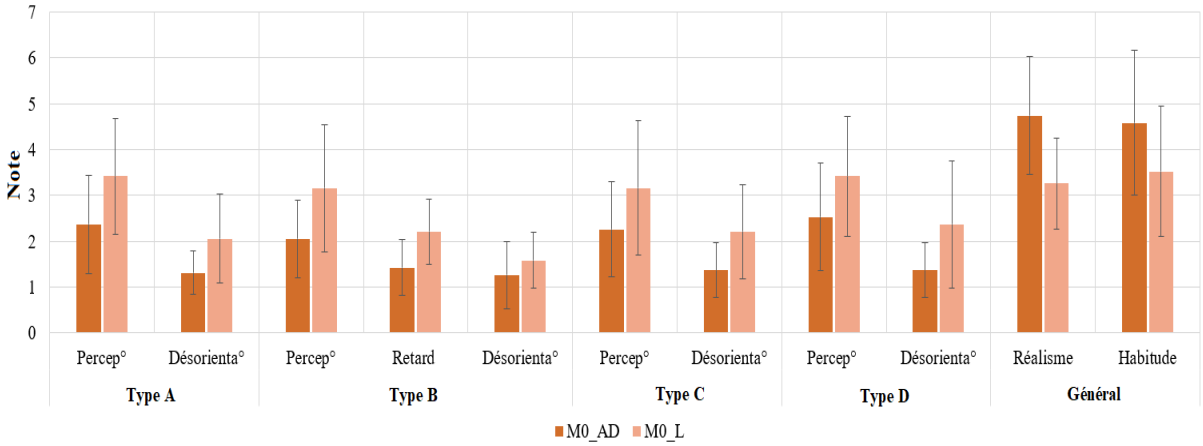
Situation/ mouvement	**Caractéristique**	**AD**	**vs.**	**Libre**	**(U, p-value)**
Type A	Percep° mouvement	M0	<	M0	(94.5, 0.005)
	Désorientation	M0	<	M0	(99, 0.004)
	Retard	M1	<	M1	(110.5, 0.017)
	Douceur	M2	>	M2	(259.5, 0.008)
Type B	Percep° mouvement	M0	<	M0	(95, 0.017)
	Retard	M0	<	M0	(76, <0.001)
	Désorientation	M0	<	M0	(118, 0.014)
	Retard	M1	<	M1	(109, 0.015)
	Douceur	M2	>	M2	(270, 0.003)
Type C	Percep° mouvement	M0	<	M0	(114.5, 0.022)
	Désorientation	M0	<	M0	(92, 0.002)
	Retard	M1	<	M1	(101, 0.009)
	Désorientation	M2	<	M2	(116.5, 0.021)
Type D	Percep° mouvement	M0	<	M0	(111, 0.017)
	Désorientation	M0	<	M0	(97.5, 0.004)
	Retard	M1	<	M1	(104.5, 0.011)
Général	Réalisme	M0	>	M0	(292.5, <0.001)
	J'ai l'impression de conduire	M0	>	M0	(253, 0.016)
	Réalisme	M1	>	M1	(262, 0.008)

TABLEAU 15 – Comparaison entre les modes de conduite autonome et libre à l'aide des tests de Mann-Whitney.

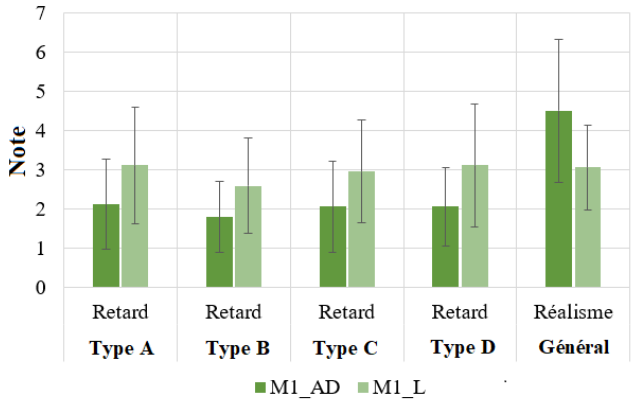
5.3.3 Division des groupes selon le type de MCA préféré

Dans cette partie, nous ne tenons compte que des 19 participants qui ont conduit le simulateur en mode Libre. En effet, l'objectif est d'analyser si le type de comportement de conduite auto-évalué a une relation avec le MCA qu'ils ont choisi dans le simulateur de conduite. Cette comparaison sera évaluée au même titre que la façon dont ils ont conduit le simulateur. Comme précisé précédemment, nous avons classé les sujets du mode Libre par groupes de préférence MCA : P0 pour le groupe des sujets qui ont préféré M0, P1 pour M1 et P2 pour M2.

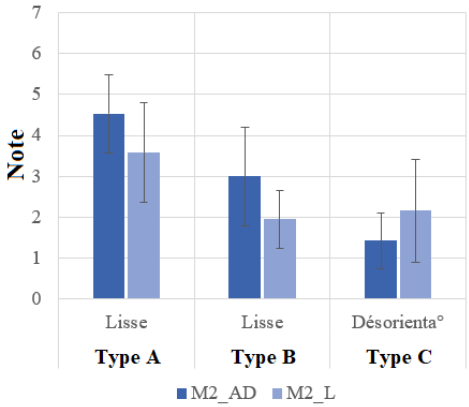
D'après la dernière partie du questionnaire présenté dans la partie 5.2.6, sept participants ont préféré la stratégie M0 qui ne contient que le modèle dynamique du simulateur (16) et douze participants ont préféré la stratégie M2 qui, en plus de la dynamique de la plateforme, intègre le modèle de perception humaine proposé



(a) Stratégie M0



(b) Stratégie M1



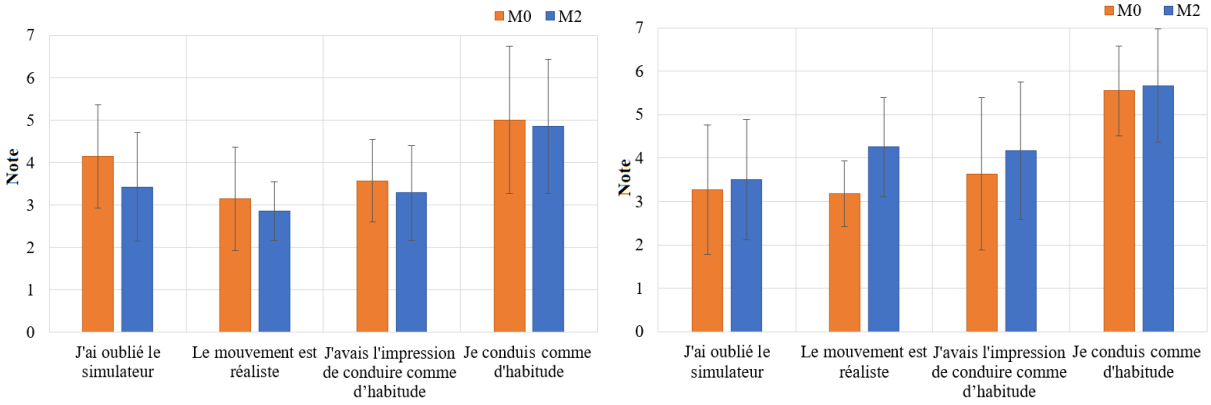
(c) Stratégie M2

FIGURE 62 – Moyennes et écart types pour les caractéristiques MCA statistiquement significativement différentes entre les modes de conduite AD et Libre.

par [Telban & Cardullo, 2005]. Aucun participant n'a choisi la stratégie M1. De fait, nous ne conservons que deux groupes, P0 et P2. Pour chacun des groupes, les données collectées sont les données du scénario de test et les scores des formulaires MCA.

Les données des groupes P0 et P2 ne suivent pas une distribution normale, par conséquent, des tests non paramétriques appariés de Wilcoxon avec un niveau de significativité de 5% ont été utilisés pour comparer les différences entre les deux groupes. Les moyennes et les écarts-types des scores du formulaire MCA pour chaque groupe sont présentés dans la figure 63.

En ce qui concerne seulement les énoncés du MCA qui ont obtenu un score statistiquement différent, le test a indiqué que le groupe P0 a oublié davantage le simulateur de conduite lorsqu'il a utilisé la stratégie M0 que la stratégie M2 ($Z = 2.731, p = 0.018$). Pour les participants du groupe P2, le test a révélé que les énoncés « J'ai l'impression de conduire » ($Z = 2.135, p = 0.032$) et « Je conduis comme d'habitude » ($Z = 2.098, p = 0.036$) étaient significativement plus élevés lorsqu'ils utilisaient la stratégie M2 que lorsqu'ils utilisaient la stratégie M0, c'est-à-dire que



(a) Groupe P0

(b) Groupe P2

FIGURE 63 – Évaluation subjective du questionnaire MCA en fonction de la stratégie préférée par les sujets.

leur comportement n'était pas influencé par le fait d'être dans un simulateur de conduite et non dans une vraie voiture. Enfin, pour le même groupe, le test a indiqué que la médiane des scores de « réalisme de mouvement » avec la stratégie M2 était statistiquement plus élevée que la médiane du même énoncé avec la stratégie M0 ($Z = 2.315, p = 0.006$). Ce qui est cohérent avec la répartition faite des participants selon le type de stratégie préférée.

Dans les parties suivantes, ces résultats seront corrélés avec les données recueillies auprès de chaque participant et seront comparés avec le comportement de ces derniers en situation de conduite.

5.3.4 Classification du type de conduite auto-déclaré

Comme nous l'avons présenté dans la partie 5.2.5, au début de l'expérimentation, les sujets ont remplit un questionnaire sur le comportement de conduite, composé de 47 questions (voir annexe D.1), pour déterminer une relation entre le type de comportement et la préférence du MCA. Afin de réduire le nombre de variables (47) du questionnaire et d'identifier clairement les facteurs comportementaux, il est nécessaire de catégoriser chaque variable (de chaque questionnaire) dans un groupe spécifique appelé facteurs ou dans notre cas d'étude, styles. Il existe deux grandes méthodes pour déterminer les différents facteurs : l'analyse par composantes principales (ACP) et l'analyse factorielle (AF). La première est une transformation linéaire des variables originales et la seconde est un modèle statistique de mesure d'une variabilité des données. Nous choisissons la méthode d'analyse factorielle, car étant donné la nature subjective du questionnaire, nous considérons qu'il peut exister des erreurs dans les réponses des formulaires.

Le premier travail de MDSI réalisé par [Taubman-Ben-Ari *et al.*, 2004] suggère une analyse factorielle exploratoire (AFE), afin d'identifier les relations entre les

variables (questions) et déterminer si ces relations peuvent être expliquées par un nombre réduit de variables et classées en différents facteurs. Tout comme d'autres auteurs l'ont fait [Holman & Havarneanu, 2015], nous avons appliqué l'AFE au questionnaire sur le comportement de conduite. Nous avons effectué une AFE pour les questions du DBQ et une AFE pour les questions du MDSI.

Les techniques utilisées pour déterminer le nombre de facteurs dans les deux questionnaires étaient l'analyse parallèle [1], le critère de Kaiser [2] et l'inspection visuelle des diagramme d'éboulis (*scree plots*). Nous avons décidé de n'interpréter que les questions ayant une valeur de charge ou poids [3] (*factor loading*) absolue de 0,30 ou plus afin d'obtenir une bonne interprétabilité des facteurs compte tenu de la taille du test, de la population de sujets, et du nombre de questions. Cette valeur est soulignée par une majorité de chercheurs [Osborne *et al.*, 2008]. Par ailleurs, ni les items du DBQ ni ceux du MDSI n'ont eu des valeurs de charge inférieures à 0,35. Puis, nous avons vérifié la cohérence du DBQ et du MDSI en utilisant l'alpha de Cronbach α, comme estimation de la fiabilité interne, pour les facteurs obtenus par l'AFE.

MDSI : inventaire multidimensionnel des styles de conduite

Une première AFE a révélé de mauvais résultats puisque la valeur de charge de chaque variable dans son facteur respectif était inférieure à 0,30. De plus, le coefficient alpha de Cronbach de fiabilité n'était pas satisfaisant pour la moitié des facteurs car il était inférieur à 0,7. Par conséquent, une deuxième AFE a été effectuée pour regrouper les variables dans de nouveaux facteurs. Cette fois, nous avons supprimé la question 39, car sa valeur de charge était inférieure à 0,3 avec son facteur correspondant.

La deuxième analyse a donné une distribution de 26 éléments avec 5 facteurs (styles de conduite) expliquant 50 % de la variance commune totale du MDSI modifié : dissociatif, attentif, agressif, impatient et risqué. Le coefficient de concordance de Tucker qui en résulte est de 0,84. Cela correspond à une bonne similitude entre les différents styles de conduite ou facteurs.

Le tableau 16 résume pour chaque style de conduite : l'élément de la question, une brève description, les valeurs de charge (poids) des questions, le coefficient α par style et la moyenne μ et l'écart-type σ pour toutes les questions.

1. Technique de simulation de Monte Carlo qui aide à déterminer le nombre de facteurs à retenir dans l'analyse des composantes principales et des facteurs exploratoires [Ledesma & Valero-Mora, 2007].
2. Retient les facteurs ayant des valeurs propres supérieures à 1.
3. Indique dans quelle mesure le facteur influence la question mesurée.

Facteurs et éléments	**Poids**	μ (σ)
Style dissociatif ($\alpha = 0.84$)		
Q1 Vous essayez de démarrer au feu en 3ème	0.54	1.55 (0.78)
Q5 Vous freinez brutalement pour éviter une collision	0.53	2.05 (0.82)
Q6 Vous allumez les essuie-glaces au lieu des phares	0.66	1.48 (0.93)
Q7 Vous êtes inquiet(e) de conduire par mauvais temps	0.71	2.13 (1.09)
Q19 Vous ne faites pas attention à la présence d'un piéton qui attend à un passage piéton	0.45	1.95 (0.71)
Q20 Vous vous sentez mal à l'aise en conduisant	0.62	1.45 (0.86)
Q21 En reculant, vous percutez quelque chose que vous n'avez pas vu	0.65	1.68 (0.69)
Q36 Conduire vous frustre	0.57	1.85 (1.21)
Q37 Vous planifiez si mal le trajet que vous vous retrouvez dans un embouteillage que vous auriez pu éviter	0.42	2.35 (1.09)
Q43 Vous sous-estimez la vitesse d'un véhicule approchant lorsque vous dépassez	0.56	2.08 (1.07)
Style attentif ($\alpha = 0.71$)		
Q11 Vous vous sentez détendu (e) en conduisant	0.41	5.13 (1.02)
Q22 Vous conduisez prudemment	0.71	4.75 (1.08)
Q33 Vous avez le contrôle de la conduite	0.35	5.33 (0.83)
Q34 Vous conduisez à une vitesse inférieure à la limite	0.59	3.85 (1.75)
Q38 Vous attendez si vous n'avez pas la priorité.	0.4	5.05 (1.36)
Q42 « Mieux vaut prévenir que guérir »	0.51	4.58 (1.41)
Q46 Vous êtes toujours prêt à réagir aux manœuvres inattendues des autres conducteurs	0.49	4.85 (1.09)
Style agressif($\alpha = 0.78$)		
Q41 Vous aimez prendre des risques en conduisant	0.72	1.93 (0.99)
Q47 Vous appréciez la conduite dangereuse	0.83	2.15 (1.21)
Style impatient ($\alpha = 0.6$)		
Q2 Vous êtes impatient aux heures de pointe	0.45	3.28 (1.09)
Q35 Dans un embouteillage, lorsque la file d'à côté avance, vous essayez de vous y insérer dès que possible	0.78	2.33 (0.92)
Style risqué ($\alpha = 0.68$)		
Q12 Vous passez alors que le feu vient de passer au rouge	0.37	1.25 (0.49)
Q16 Vous lisez vos messages en conduisant	0.68	1.98 (1.21)
Q30 Vous faites des activités relaxantes en conduisant	0.40	1.97 (1.05)
Q40 Manger, se coiffer ou maquiller en conduisant	0.42	1.90 (1.11)
Q44 Vous écrivez des messages en conduisant	0.85	1.73 (0.96)
Q39 Vous réfléchissez à des moyens de traverser la circulation rapidement *		

*élément éliminé afin de calculer la charge souhaitée et l'alpha de Cronbach (α).

TABLEAU 16 – Modèle factoriel et poids des coefficients du MDSI modifié.

DBQ : questionnaire sur le comportement des conducteurs

Parmi les 47 questions du questionnaire de comportement de conduite auto-déclaré, 28 questions forment le DBQ modifié, soit 19 questions de moins que le DBQ d'origine [Reason *et al.*, 1990].

De même que pour le MDSI, nous avons effectué un AFE afin de réduire le nombre d'éléments en conservant le maximum d'informations. Le point de coupure de chaque facteur est de 0,35 étant donné le nombre de questions et la taille de la population. Une première AFE donne un coefficient de Tucker inférieur à 0,7. Par conséquent, les questions dont la charge est inférieure au seuil (Q10, Q13, Q29) ont été supprimées et une nouvelle analyse a été effectuée. Cette dernière aboutit à une distribution de 25 éléments avec 4 facteurs expliquant 47% de la variance commune totale du DBQ-modifié. Le coefficient de Tucker obtenu est de 0,88, ce qui assure une similitude équitable entre la structure et les facteurs.

Selon la méthodologie de [Reason *et al.*, 1990], la classification des facteurs dépend du type de comportement en situation de conduite et du degré de risque pour les autres conducteurs. Nous retenons les trois grandes catégories proposées dans la revue de la littérature présentée ci-dessus : les erreurs, les défaillances et les infractions. Dans notre cas, les défaillances et les infractions ont été divisées en deux sous-catégories qui font référence au niveau de risque A ou B. Cette classification a été définie en fonction du nombre de facteurs obtenus par l'AFE et s'est inspirée d'études antérieures [Guého, 2014; Lajunen *et al.*, 2004; Martinussen, 2013]. Le niveau de risque A indique un risque plus élevé, c'est-à-dire un risque certain pour les autres usagers de la route. Le niveau B signifie une faible possibilité de risque pour les autres conducteurs. La fiabilité interne des infractions pour le risque B étant faible ($\alpha = 0,5$), par conséquent nous avons unifié les infractions dans un seul facteur ($\alpha = 0,68$).

Le tableau 17 montre pour chacun des 25 éléments de la nouvelle version du DBQ : l'élément de la question, une brève description, les valeurs de charge (poids) des questions, le coefficient α par facteur et la moyenne et l'écart-type pour toutes les questions.

Facteurs et éléments	**Poids**	μ (σ)
Erreurs ($\alpha = 0.79$)		
Q1 Vous essayez de démarrer au feu en 3ème	0.52	1.55 (0.78)
Q4 Vous oubliez où vous vous êtes garé dans un parking	0.59	2.30 (1.18)
Q6 Vous allumez les essuie-glaces au lieu des phares	0.79	1.48 (0.93)
Q21 En reculant, vous percutez quelque chose	0.51	1.68 (0.69)
Q24 Vous vous insérez mal dans un rond-point	0.61	1.73 (0.75)
Q43 Vous sous-estimez la vitesse d'un véhicule	0.66	2.08 (1.07)
Manquements : A ($\alpha = 0.76$)		
Q5 Vous freinez brutalement pour éviter une collision	0.49	2.05 (0.81)
Q8 Vous avez oublié le trajet que vous venez de parcourir	0.4	2.08 (0.83)
Q18 Vous avez une aversion pour une catégorie particulière d'usager	0.56	1.53 (0.72)
Q19 Vous ignorez la présence d'un piéton	0.48	1.95 (0.71)
Q26 Vous oubliez de contrôler rétroviseurs avant de sortir	0.76	1.45 (0.60)
Q31 Vous percutez presque le véhicule devant vous	0.72	1.78 (0.95)
Manquements : B ($\alpha = 0.7$)		
Q9 Vous manquez la sortie et vous devez faire un détour	-0.58	2.03 (0.83)
Q25 Mal lire les panneaux et se tromper au rond-point	-0.62	2.05 (1.01)
Q37 Vous planifiez si mal votre trajet que vous vous retrouvez dans un embouteillage que vous auriez pu éviter	-0.39	2.35 (1.10)
Q45 En ayant l'intention d'aller à une destination, vous vous retrouvez sur la route de votre destination habituelle	-0.43	2.30 (1.24)
Infractions : risque A ($\alpha = 0.7$)		
Q3 Vous collez le véhicule devant vous pour lui dire d'aller plus vite ou de changer de voie	0.63	2.08 (1.10)
Q23 Vous doublez un véhicule lent par la droite	0.64	2.35 (1.17)
Q32 Intentionnellement, vous roulez trop près des autres	0.66	2.05 (1.15)
Q14 Délibérément, vous ne respectez pas la limitation de vitesse tard la nuit ou très tôt le matin	0.39	2.95 (1.50)
Q28 Vous ignorez les panneaux de priorité et vous évitez de justesse une collision avec un véhicule prioritaire	0.35	1.35 (0.53)
Infractions : risque B ($\alpha = 0.5$)		
Q12 Vous passez alors que le feu vient de passer au rouge	0.62	1.25 (0.49)
Q15 En tournant, vous manquez de renverser un cycliste	0.5	1.23 (0.42)
Q17 Boire de l'alcool et conduire	0.45	1.53 (0.68)
Q27 Vous doublez un véhicule qui signale un virage à gauche	0.38	1.20 (0.46)
Q10 Vous oubliez le rapport de vitesse*		
Q13 Vous suivez un conducteur de façon malveillante *		
Q29 Vous êtes impliqué(e) dans une « course non officielle » avec d'autres conducteurs*		

TABLEAU 17 – Modèle factoriel et poids des composants du DBQ modifié.

Comparaison entre MDSI et DBQ

L'analyse du MDSI et DBQ nous permet de constater qu'il existe des facteurs très similaires entre les deux questionnaires. De fait, l'un des problèmes principaux dans la réalisation de cette expérience a été de trouver le questionnaire le plus pertinent pour évaluer le comportement des conducteurs en situation de conduite, puisque dans la littérature il existe de nombreux questionnaires similaires, comme le montre la partie 5.2.5.

C'est pourquoi, dans le but d'évaluer la relation entre les deux questionnaires, des corrélations de Pearson ont été calculées entre les cinq styles de conduite : dissociatif, prudent ou attentif, agressif, impatient et risqué, et les cinq facteurs du DBQ : erreurs, manquements A, manquements B, risques d'infraction A et risques d'infraction B. D'autres corrélations ont été calculées pour le groupe d'infractions (A et B) ensemble, car ce facteur a une plus grande fiabilité interne que chaque facteur d'infraction séparé.

Les corrélations ont été calculées avec la moyenne des scores aux questions (voir le tableau 18) et pour tous les sujets qui ont répondu au questionnaire sur le comportement de conduite.

FACTEURS	**Erreurs**	**Manq. A**	**Manq. B**	**Infrac° A**	**Infrac° B**	**Infrac° (AB)**
Dissociatif	***0.87 (<.001)***	***0.58 (<.001)***	***0.473 (.003)***	-0.089 (.594)	0.033 (.844)	-0.064 (.700)
Attentif	***-0.329 (.043)***	***-0.396 (.013)***	0.109 (.513)	-0.203 (.221)	0.110 (.512)	-0.134 (.421)
Agressif	-0.005 (.981)	0.143 (.391)	0.000 (.998)	***0.55 (<.001)***	***0.345 (.034)***	***0.60 (<.001)***
Impatient	0.114 (.490)	-0.096 (.565)	-0.080 (.631)	0.258 (.118)	0.043 (.118)	0.237 (.151)
Risqué	-0.056 (.742)	0.074 (.656)	0.000 (.997)	0.171 (.304)	0.239 (.305)	0.235 (.155)

Note : Les entrées du tableau sont les coefficients de corrélation (r) avec les p-valeurs entre parenthèses.

TABLEAU 18 – Corrélations entre DBQ et MDSI.

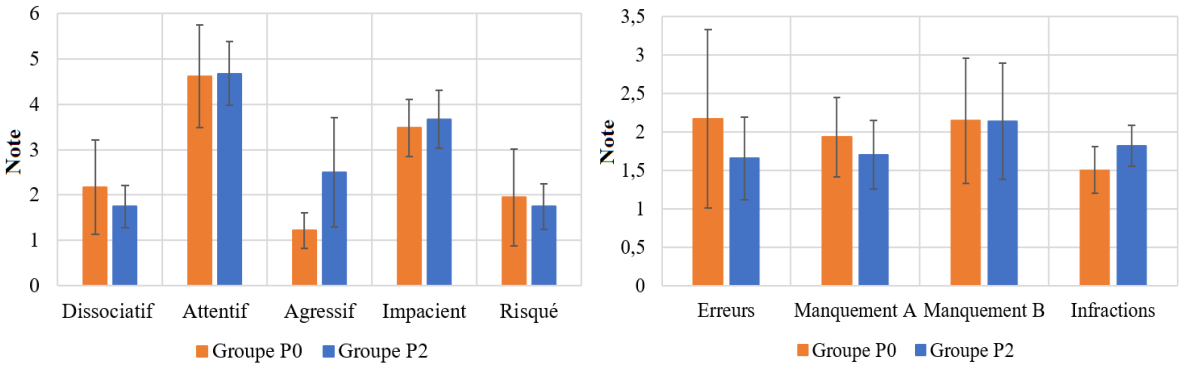
Des corrélations fortement positives ont été observées pour deux facteurs de conduite : le premier est le style dissociatif avec les erreurs et les manquements, et le second est le style de conduite agressif avec les infractions. Le style attentif ou prudent est corrélé modérément négativement avec les erreurs. Idem pour les manquements avec les risques B. Nous pouvons également observer que les styles impatient, risqué ou imprudent ne sont pas corrélés avec les facteurs du DBQ.

Les résultats montrent qu'il serait idéal d'unifier et d'améliorer encore les ques-

tionnaires DBQ et MDSI pour faciliter la recherche sur le comportement réel de conduite sur route au moins pour les éléments qui font partie des facteurs tels que le dissociatif, l'attentif, les erreurs et les infractions.

5.3.5 Relation entre le comportement de conduite et la préférence de mouvement de la plate-forme

Afin d'évaluer le rapport entre le type de comportement de conduite et le choix du MCA des participants, nous avons comparé le comportement auto-déclaré entre les groupes P0 et P2 en réalisant des tests de Wilcoxon Mann-Whitney. La figure 64 montre les valeurs moyennes et l'écart-type des différents groupes pour chaque questionnaire sur le comportement de conduite.



(a) Scores du MDSI

(b) Scores du DBQ

FIGURE 64 – Résultats des groupes P0 et P2 selon les types des facteurs des questionnaires MDSI et DBQ.

D'après les résultats, le groupe P2 a des scores significativement plus élevés dans le style de conduite agressif ($U = 10, p = 0,005$) et pour le facteur infractions ($U = 16, p = 0,046$) que le groupe P0. Aucun des autres facteurs restants n'est significativement différent entre les groupes P0 et P2. Ce résultat sera corrélé avec les indicateurs de conduite de mouvement dans la partie suivante.

5.3.6 Relation objective entre les questionnaires auto-déclarés et le comportement de conduite sur simulateur

L'un des objectifs principaux de cette étude est de fournir une compréhension plus approfondie du comportement de conduite et de sa relation avec la préférence du modèle de perception du mouvement dans les simulateurs de conduite, l'idée

étant que le comportement de conduite auto-déclaré des conducteurs fournisse des informations permettant un meilleur paramétrage de la stratégie de restitution du mouvement. Ainsi, si les connaissances préalables du comportement des conducteurs sont disponibles, les MCA peuvent être améliorés et adaptés en fonction des désirs des conducteurs, rendant la simulation plus immersive, réaliste et agréable pour chaque conducteur. Pour ce faire, nous souhaitons compléter l'analyse subjective de cette étude faite dans la partie précédente avec une analyse objective par la corrélation des mesures de performance obtenues lors de l'expérimentation.

D'après les résultats précédents (voir partie 5.3.5), nous pouvons constater que le comportement de conduite des groupes P0 et P2 est différent pour le style agressif du MDSI et pour le facteur infraction du DBQ. En effet, les résultats des corrélations de Pearson ont indiqué qu'il y avait une association positive significative entre ces deux facteurs ($r = 0.597, p < 0.01$). C'est pourquoi, dans cette partie, nous considérerons uniquement ces deux facteurs dans la comparaison du comportement de conduite et l'analyse objective des résultats entre les groupes P0 et P2. Les autres facteurs ne sont pas pris en compte car il n'y a pas de différences significatives entre les groupes de préférences. Ainsi, en utilisant la méthodologie expliquée dans la partie 5.2.7, nous avons calculé les valeurs minimales, maximales, moyennes et les écart-types (E-T) de tous les indicateurs de mouvement de conduite (vitesse, *jerk* et position latérale) pour chaque segment de route de la figure 56a. Ces indicateurs sont énumérés dans le tableau 19.

	Indicateurs	**Min**	**Max**	**Moyenne**	**E-T**
Segment 1	*Jerk* x	-224,23	198,65	0,09	0,02
	Vitesse	0,00	66,65	25,46	3,01
	Position latérale	0,00	0,56	0,14	0,07
Segment 2	*Jerk* y	-9,95	15,91	-0,08	0,10
	Vitesse	30,11	79,20	51,73	5,63
	Position latérale	0,00	1,65	0,41	0,13
Segment 3	*Jerk* x	-17,86	16,35	-0,04	0,04
	Vitesse	34,35	120,20	81,00	7,00
	Position latérale	0,00	1,15	0,27	0,14
Segment 4	*Jerk* y	-9,25	17,92	0,03	0,03
	Vitesse	41,63	97,39	61,66	9,39
	Position latérale	0,00	1,75	0,41	0,24
Segment 5	*Jerk* y	-19,12	54,81	0,01	0,03
	Vitesse	1,17	112,33	74,77	8,86

TABLEAU 19 – Moyenne, écart-type (E-T), minimum et maximum des indicateurs de mouvement.

Notre hypothèse est que si le comportement auto-déclaré des participants a une influence sur le MCA qu'ils choisissent, les indicateurs de mouvement auront une

corrélation positive avec les deux facteurs pour le groupe P2, car la stratégie M2 est celle qui fournit le plus ce type de comportement (voir partie 5.1.3). Au contraire, les indicateurs du groupe P0 n'auront pas une corrélation positive avec ces deux facteurs.

Le tableau 20 donne un aperçu des corrélations de Pearson entre les réponses des questionnaires de comportement pour les facteurs agressif et infractions, et les indicateurs de mouvement pour les groupes P0 et P2. Ces corrélations ont été établies pour chaque segment de route.

Nous observons que pour le groupe P0, aucune des mesures présentées n'a une corrélation significative avec le style agressif et seul l'écart-type de la position latérale présente une corrélation significative et fortement négative avec le facteur d'infraction ($r = -0.805, p = 0.029$). Ce résultat était attendu puisque le premier segment de route ne dure que 300 mètres et présente plusieurs manœuvres de démarrage et de freinage. En ce qui concerne le groupe P2, les corrélations de Pearson ont montré plusieurs relations fortes entre les facteurs et les mesures. En comparant le style agressif, la moyenne du *jerk* selon l'axe x sur le segment de route 1 ($r = 0.648, p = 0.031$), l'écart-type du *jerk* selon l'axe y sur les segments de route 2 ($r = 0.810, p = 0.003$), 4 ($r = 0.658, p = 0.028$) et 5 ($r = 0.597, p = 0.049$), et la vitesse moyenne sur les segments de route 2 ($r = 0.618, p = 0.043$) et 4 ($r = 0.743, p = 0.009$) présentent une corrélation fortement positive. En ce qui concerne les infractions, deux mesures ont une corrélation fortement positive avec ce facteur : la vitesse moyenne sur le segment 4 ($r = 0.669, p = 0.024$) et l'écart type de vitesse ($r = 0.631, p = 0.037$) et du jerk ($r = 0.618, p = 0.043$) sur le segment 5. Ce résultat confirme qu'il existe une différence entre les groupes P0 et P2 en fonction du comportement de conduite auto-déclaré par les conducteurs. Comme nous l'avions prévu dans l'hypothèse, les indicateurs du groupe P2 ont une forte corrélation positive avec les deux facteurs alors que le groupe P0 n'a aucune corrélation positive avec ces deux facteurs.

5.4 Discussion

L'étude présentée dans ce chapitre décrit une expérimentation de conduite dans un simulateur de conduite dynamique. Trois MCA fondés sur la commande prédictive (MPC) ont été implémentés et comparés grâce à une étude réalisée dans un simulateur de conduite dynamique à hautes performances. Les stratégies dépendaient du modèle mathématique utilisé dans la conception de la boucle de contrôle : la première, nommée M0, correspondait au modèle dynamique de la plateforme. Les deux autres stratégies intégraient en outre les modèles de perception du mouvement de [Young & Meiry, 1968]-[Young & Oman, 1969] et de [Telban & Cardullo, 2005], nommées respectivement M1 et M2.

Dans une premier temps nous avons comparé, subjectivement, les trois stratégies selon le mode de conduite : mode de conduite autonome « AD » ou mode de conduite

	Agressif Groupe P0	**Infractions Groupe P0**	**Agressif Groupe P2**	**Infractions Groupe P2**
Segment 1				
Moyenne *jerk* x	-0.242 (.600)	-0.230 (.620)	***0.648 (.031)***	0.534 (.090)
E-T *jerk* x	-0.291 (.527)	0.126 (.789)	0.405 (.216)	0.001 (.998)
Moyenne vitesse	-0.093 (.843)	-0.096 (.837)	0.215 (.525)	-0.026 (.939)
E-T vitesse	-0.537 (.214)	-0.572 (.180)	0.132 (.698)	0.398 (.225)
Moyenne pos. lat.	-0.340 (.456)	-0.739 (.058)	-0.102 (.764)	-0.409 (.212)
E-T position latérale	-0.352 (.438)	***-0.805 (.029)***	-0.162 (.635)	-0.332 (.319)
Segment 2				
Moyenne *jerk* y	0.615 (.142)	0.578 (.174)	-0.08 (.816)	-0.477 (.138)
E-T *jerk* y	-0.330 (.470)	-0.710 (.074)	***0.810 (.003)***	0.468 (.146)
Moyenne vitesse	-0.573 (.179)	-0.598 (.156)	***0.618 (.043)***	0.550 (.080)
E-T vitesse	-0.061 (.896)	-0.492 (.262)	0.195 (.566)	0.191 (.573)
Moyenne pos. lat.	0.338 (.458)	0.372 (.411)	0.209 (.538)	-0.506 (.113)
E-T position latérale	0.220 (.636)	0.480 (.276)	0.235 (.486)	-0.592 (.055)
Segment 3				
Moyenne *jerk* x	0.180 (.699)	0.140 (.764)	0.497 (.120)	-0.154 (.651)
E-T *jerk* x	-0.335 (.463)	-0.631(.128)	0.354 (.286)	0.273 (.416)
Moyenne Vitesse	-0.370 (.414)	-0.140 (.765)	0.596 (.053)	0.567 (.069)
E-T vitesse	-0.222 (.632)	-0.311 (.497)	0.101 (.768)	0.244 (.470)
Moyenne pos. lat.	0.275 (.550)	0.144 (.758)	-0.395 (.230)	-0.115 (.736)
E-T position latérale	0.244 (.598)	-0.003 (.994)	-0.175 (.607)	-0.22 (.516)
Segment 4				
Moyenne *jerk* y	0.094 (.841)	0.337 (.460)	0.466 (.149)	0.202 (.551)
E-T *jerk* y	-0.356 (.434)	-0.555 (.195)	***0.658 (.028)***	0.404 (.218)
Moyenne Vitesse	-0.272 (.555)	-0.154 (.741)	***0.743 (.009)***	***0.669 (.024)***
E-T vitesse	-0.086 (.854)	-0.606 (.149)	0.483 (.132)	0.345 (.298)
Moyenne pos. lat.	-0.295 (.521)	0.649 (.115)	0.413 (.207)	-0.168 (.620)
E-T position latérale	-0.100 (.832)	0.648 (.116)	0.315 (.345)	-0.246 (.466)
Segment 5				
Moyenne *jerk* y	-0.355 (.435)	-0.387 (.391)	0.121 (.722)	-0.379 (.251)
E-T *jerk* y	-0.342 (.453)	-0.620 (.138)	***0.597 (.049)***	***0.618 (.043)***
Moyenne Vitesse	-0.493 (.261)	-0.491 (.263)	0.548 (.081)	0.360 (.277)
E-T vitesse	0.573 (.179)	0.659 (.107)	0.415 (.204)	***0.631 (.037)***

Note : Les entrées du tableau sont les coefficients de corrélation de Pearson (r) avec la p-valeur entre parenthèses.

TABLEAU 20 – Corrélations pour les groupes P0 et P2 entre les indicateurs de performance et les facteurs présentant une différence statistique significative dans le questionnaire du comportement de conduite : style agressif et infractions.

active « Libre ». Ensuite nous avons comparé ces deux modes entre eux. D'autre part, nous avons analysé les questionnaires de comportement de conduite auto-déclaré pour ainsi analyser son impact dans la préférence parmi les trois stratégies de contrôle. Dans le but d'adapter les MCA en fonction du type de conduite du

conducteur, nous avons comparé ces résultats avec les indicateurs de performance obtenus de l'étude utilisateur. Ceci nous a permis de valider les styles de conduite auto-déclaré avec le comportement de conduite associé dans le simulateur.

5.4.1 Contrôle de restitution du mouvement fondé sur le modèle prédictif

En mode de conduite libre, la stratégie M2 a montré un comportement plus approprié dans le mouvement latéral par rapport aux deux autres stratégies. Celles-ci étaient significativement trop souples, inhibant ainsi la perception du mouvement, surtout dans le cas du slalom. De plus, la stratégie M1 a eu une note plus élevée que celle de M0 pour la caractéristique inconfort et plus basse que M2 pour la perception de mouvement. Outre ces résultats statistiques, les participants ont montré une difficulté à s'adapter à la stratégie M1. Par conséquent, ils n'ont pas pu reproduire leur comportement de conduite réel dans le simulateur. Néanmoins, nous ne pouvons pas supposer que M2 est le meilleur MCA en ce qui concerne la perception du mouvement en conduite active, car il est nécessaire de réaliser plus d'expérimentations dans les simulateurs de conduite, afin de creuser d'avantage l'impact direct du modèle du système vestibulaire de la stratégie M2 par rapport au modèle dynamique de la plateforme M0. De fait, la stratégie de contrôle prédictif dépend non seulement du modèle mathématique mais aussi des horizons de contrôle et de prédiction, des paramètres de pondération de la fonction de coût et des conditions de stabilité.

En ce qui concerne le mode de conduite autonome, nous avons constaté que les participants préféraient les indices de mouvement fournis par la stratégie M0 car ils procuraient moins d'inconfort, de retard et une meilleure immersion dans la simulation que l'utilisation d'un modèle de système vestibulaire. Cependant, nous considérons que la qualification des stratégies a été affectée par le profil d'accélération du véhicule autonome, qui présentait une réponse de grande amplitude concernant les accélérations transitoires, surtout selon l'axe x. En ce sens, la configuration M2 a été considérée comme trop agressive en mode autonome car elle restitue plus de hautes fréquences par rapport à M0 et M1. En revanche, la stratégie M1, n'a pas été qualifiée d'agressive, mais les participants ont remarqué une gêne et une mauvaise perception du mouvement lors de son utilisation.

Indépendamment du mode de conduite, nous avons constaté que l'attribut de retard était moins important pour la stratégie M0. Une raison possible est que l'intégration de filtres supplémentaires lors de l'inclusion du système vestibulaire pour les stratégies M1 et M2 génère un retard supplémentaire dans la restitution du mouvement. Lors de la comparaison des deux modes de conduite, plusieurs différences ont été constatées, notamment en ce qui concerne la stratégie M0. Cela indique qu'il est nécessaire de réévaluer le mode autonome et de comparer les types de conduite.

5.4.2 Influence du comportement de conduite dans la préférence du type de contrôle de plate-forme

Dans cette partie, l'objectif était d'analyser, du point de vue du conducteur, l'influence du modèle mathématique MCA et sa relation avec le comportement de conduite dans un simulateur de conduite. Une telle étude peut généralement être obtenue en utilisant uniquement une approche subjective. Toutefois, nous avons effectué une analyse plus approfondie, afin de comprendre les raisons qui ont amené les participants à choisir la stratégie qui leur convient le mieux. Ainsi, nous avons examiné objectivement les comportements de conduite auto-déclarés et les indicateurs de données réelles obtenus après avoir conduit le simulateur avec les différentes stratégies. Il est important de rappeler que pour cette partie de l'étude, seules les données des participants qui ont conduit le simulateur en mode libre ont été prises en compte.

Selon la préférence de la stratégie de conduite, nous avons formé deux groupes de sujets : ceux qui ont préféré conduire le simulateur avec la stratégie M0, nommés groupe P0 et ceux qui ont préféré la stratégie M2, nommés groupe P2. Aucun participant n'a préféré la stratégie M1.

En comparant subjectivement les stratégies M0 et M2, les résultats ont montré que la préférence entre les deux stratégies peut découler de différentes raisons. Le groupe P0 a obtenu un score plus élevé pour l'affirmation « J'ai oublié le simulateur », tandis que le groupe P2 a obtenu un score plus élevé pour les autres énoncés tels que « J'avais l'impression de conduire », « Je conduisais comme d'habitude » et « Le mouvement était réaliste ». Ce résultat peut être expliqué en observant le comportement de M0 en tant que contrôle de restitution du mouvement : les accélérations transitoires sont plus douces que celles produites par la stratégie M2, produisant ainsi un effet de mouvement confortable mais légèrement irréel.

En examinant l'influence du comportement de conduite sur le choix du MCA par les participants, nous avons évalué une version modifiée des formulaires MDSI et DBQ. Les éléments des échelles originales du DBQ et du MDSI ont révélé une structure de cinq composantes chacune. Pour le DBQ, les facteurs sont interprétés comme des erreurs, des défaillances de risque A, des défaillances de risque B et des infractions. Cette structure confirme la classification originale [Reason *et al.*, 1990] et la distinction entre chaque facteur telle qu'observée dans d'autres études [Guého, 2014; Martinussen, 2013]. Pour le MDSI, la structure révèle cinq styles interprétés comme : anxieux ou dissociatif, attentif, hostile ou agressif, impatient et risqué. Ces 5 facteurs ont été déterminés par une AFE et ils sont conformes à ceux rapportés dans la littérature [Holman & Havarneanu, 2015; Hooft van Huysduynen *et al.*, 2018; Poó *et al.*, 2013], sauf pour le style de réduction du stress car il est composé de nombreux éléments qui n'ont pas été pris en compte dans cette étude.

Les résultats permettent d'énoncer deux conclusions principales : premièrement,

les deux questionnaires ont plusieurs facteurs avec des corrélations fortement positives comme le style agressif avec les infractions et le style anxieux ou dissociatif avec des erreurs et des manquements. Le style attentif présente des corrélations modérément négatives avec les erreurs et les manquements de type A. Ce résultat reflète l'importance de créer un formulaire unifié pour faciliter et généraliser la compréhension du comportement en situation de conduite. La deuxième constatation indique que seuls les infractions et les facteurs agressifs peuvent être considérés comme des approximations potentielles pour la conception des MCA, c'est-à-dire que les participants qui préféraient conduire le simulateur avec la stratégie M2 ont obtenu des résultats plus élevés pour les deux facteurs que les participants qui préféraient la stratégie M0.

Pour compléter les résultats précédents, nous avons examiné objectivement la relation entre les deux facteurs déclarés par les participants, c'est-à-dire le comportement agressif et les infractions, et différents indicateurs de conduite comme le *jerk*, la vitesse et la position latérale, grâce aux données des participants ayant conduit le simulateur avec les stratégies M0 et M2. L'analyse objective a été effectuée pour les deux groupes, P0 et P2. Concernant le groupe P0, aucune corrélation positive n'a été trouvée entre les facteurs et les mesures du simulateur de conduite. Cependant, plusieurs indicateurs du groupe P2 sont corrélés positivement avec ces facteurs. Dans le style agressif, le *jerk* sur le premier segment de route était fortement positivement corrélé. Cela signifie que les conducteurs qui ont eu des scores plus élevés sur l'échelle du style de conduite agressif ont eu un *jerk* moyen plus élevé lorsqu'ils étaient confrontés à des situations d'accélérations à haute fréquence telles que des situations de démarrage et de freinage. De plus, la variation du *jerk* sur touts les segments impliquant un mouvement selon l'axe y était corrélé positivement avec ce type de comportement. Le même groupe avait une vitesse moyenne plus élevée sur les segments de route qui correspondaient à des sections d'insertion et de sortie sur l'autoroute. Enfin, concernant les infractions, une corrélation fortement positive a été observée pour le groupe P2 avec la vitesse moyenne dans les situations avec accélérations latérales à basses fréquences, et l'écart-type de la vitesse et du *jerk* dans les situations avec accélérations latérales à hautes fréquences.

Ce résultat est néanmoins conforme aux études réalisées par [Zhao, 2012] et [Helman & Reed, 2015], dans lesquelles une corrélation positive significative existe entre la vitesse moyenne et le facteur d'infraction. Ce facteur est corrélé positivement avec un style de conduite plus agressif. En outre, nous avons vu que, comme suggéré dans [Kruk, 2017; Murphey *et al.*, 2009], la conduite agressive est généralement caractérisée par une vitesse plus élevée, une accélération plus forte et une conduite saccadée en général. Contrairement aux résultats trouvés dans [Hooft van Huysduynen *et al.*, 2018], le rôle de la position latérale moyenne et son écart-type en tant qu'indicateurs pertinents pour évaluer le comportement au volant n'a pas été confirmé dans cette étude.

De cette expérience, nous pouvons tirer la conclusion que le type de comportement de conduite auto-déclaré peut avoir un impact sur le type de stratégie de restitution

de mouvement que le conducteur choisit. En effet, les conducteurs ayant le type de comportement de conduite auto-déclaré agressive ou d'infractions préféreront que les accélérations linéaires et latérales à hautes fréquences soient fortement restituées par le simulateur. Toutefois, nous tenons à souligner que de nombreuses autres études sont encore nécessaires pour définir précisément le comportement de conduite de chaque conducteur ainsi que l'impact d'autres caractéristiques du MCA, distinctes du modèle de perception, dans le choix de la stratégie de contrôle (voir annexe C).

5.4.3 Limites

L'une des limites de cette étude est la disparité entre les sujets qui se manifeste dans les écarts-types pour la majorité des situations. Cela signifie qu'une partie de la population avait des opinions très différentes pour chaque stratégie de mouvement. Cependant, ce résultat était attendu compte tenu de la subjectivité de l'expérience. Une autre considération importante est l'influence de l'adaptation des conducteurs au simulateur de conduite. Malheureusement, compte tenu de la disponibilité du simulateur et des participants, les trois stratégies ont été présentées de façon consécutive et aléatoire. De plus, l'adaptation au simulateur et au scénario d'essai peut influencer la perception du mouvement des conducteurs et donc leurs réponses subjectives au questionnaire.

La taille modérée de la population et la faible représentativité des sujets peuvent limiter la généralisation de nos conclusions. Le nombre de participants peut être responsable des faibles valeurs de fiabilité interne de certains facteurs de comportement de conduite et contribuer ainsi à des corrélations plus faibles en ce qui concerne les mesures objectives. Il serait donc utile de reproduire nos résultats avec une population plus importante et plus représentative. Malgré ce fait, cette étude fournit une base prometteuse pour adapter et valider le type de modèle à utiliser, en tenant compte du comportement des conducteurs.

Concernant les questionnaires de conduite auto-déclaré, la présente étude confirme l'importance d'utiliser le retour d'information du conducteur pour la conception des MCA. Notamment dans les cas où seul le comportement humain est évalué dans certaines situations de conduite. Néanmoins, dans le cas où le comportement dynamique d'un système est évalué, il est préférable que le simulateur ait un comportement aussi proche que possible du véhicule réel.

Les résultats fournissent des preuves supplémentaires en montrant l'intérêt des questionnaires MDSI ou DBQ en tant qu'un outil utile et fiable pour identifier le comportement de conduite, comme l'ont également observé des études antérieures : [Kaye *et al.*, 2018; Lajunen & Summala, 2003; Taubman *et al.*, 2016; Zhao, 2012]. De plus, ces résultats suggèrent qu'un questionnaire commun des infractions et des comportements agressifs pourrait également être utilisée pour évaluer les caractéristiques du MCA dans des scénarios de conduite spécifiques qui nécessitent

des mouvements plus importants, comme par exemple la situation de slalom ou de freinage. Nous pensons que l'absence de cadre conceptuel commun est l'une des difficultés à aborder pour comprendre le comportement de conduite sur la route. Toutefois, cette étude met en lumière la nécessité d'adapter les questions sur le comportement de conduite, et plus précisément, les questions qui aident à différencier un type de comportement. En outre, elle met en évidence la nécessité d'adapter, dès la conception, les stratégies de restitution de mouvement en fonction du type de comportement en conduite du conducteur.

D'après les résultats de cette étude, nous avons constaté que la préférence pour un type de stratégie peut affecter le confort de conduite pendant la simulation. Ce confort peut être produit, consciemment ou inconsciemment, par le mal du simulateur. Nous faisons l'hypothèse que si la stratégie de contrôle est choisie en tenant compte du comportement de conduite auto-déclaré du conducteur, le mal du simulateur peut être réduit. Cependant, celui-ci n'a pas été évalué dans nos expérimentations en raison des contraintes de temps imposées par la disponibilité du simulateur.

5.5 Conclusion

Après avoir choisi la stratégie de contrôle prédictif pour les simulateurs à haute capacité dynamique dans le chapitre 3, nous nous sommes intéressés à l'amélioration de cette stratégie pour la validation des prestations ADAS et véhicules autonomes. Pour cela, nous considérons qu'il est essentiel de choisir le bon modèle de perception en appliquant tous les concepts présentés dans les chapitres précédents. En effet, l'une des difficultés dans le contrôle MPC est le choix du modèle utilisé lors de la prédiction d'états et l'optimisation dont dépend la performance et le temps de calcul de l'algorithme.

Ainsi, dans ce chapitre, nous avons déployé et analysé trois types de contrôle de mouvement qui dépendent du modèle mathématique : la stratégie M0 avec le modèle dynamique de la plateforme (17), la stratégie M1 avec le système (16) et le modèle de perception de [Young & Oman, 1969]/[Young & Meiry, 1968], et la stratégie M2 avec le système (16) et le modèle de perception de [Telban & Cardullo, 2005]. Ces stratégies ont été évaluées pour une conduite active, ainsi que pour une conduite en mode autonome dans un simulateur de conduite à hautes performances.

Le modèle mis en œuvre dans le MCA fondé sur le MPC joue un rôle important dans la perception du mouvement par les conducteurs. Les résultats de cette expérimentation montrent que les sujets ont pu faire la différence entre l'utilisation de la dynamique du modèle uniquement et l'intégration d'un modèle de système vestibulaire. Néanmoins, aucune des stratégies de contrôle-plateforme ne fait l'unanimité. Cependant, nous pouvons conclure que, quel que soit le mode de conduite, la stratégie M1 est la moins préférée par les participants. Ce fait s'explique entre autres, par l'absence de restitution des accélérations transitoires. Une condition hautement

nécessaire dans les MCA puisque la perception de la vitesse, les conditions de la route et le comportement du véhicule dépendent de ce type de sensations.

Les résultats fournissent des preuves supplémentaires de la nécessité impérative de choisir un modèle mathématique approprié pour définir la tâche de suivi dans la conception des contrôles des MCA fondés sur le MPC. Il est aussi importante d'inclure le retour d'information du conducteur ainsi qu'une évaluation de son style de conduite dans la boucle de contrôle. Cela peut faciliter la conception du contrôle de mouvement, car il est difficile d'évaluer la perception de mouvement si seules les données théoriques et l'analyse du signal sont prises en compte. C'est pourquoi nous avons proposé, dans ce chapitre, d'utiliser le comportement de conduite auto-déclaré des conducteurs en tant que variable de paramétrage dans les algorithmes de restitution de mouvement. En effet, les questionnaires d'auto-évaluation peuvent être un outil précieux pour étudier et améliorer la perception de mouvement dans les simulateurs de conduite. Ils peuvent fournir des informations fiables sur les raisons qui motivent les conducteurs à choisir un MCA spécifique. La connaissance du comportement de conduite de chaque sujet peut fournir des simulations plus réalistes, plus agréables et plus immersives. De fait, les résultats montrent qu'il existe une corrélation positive entre le type de comportement de conduite et la préférence d'un type de MCA.

Les résultats montrent enfin qu'il serait idéal d'unifier et d'améliorer encore les questionnaires DBQ et MDSI pour faciliter la recherche sur le comportement réel de conduite sur route. Cependant, d'un point de vue pratique, notre expérimentation a montré que seuls les facteurs agressif et infractions servent à adapter les MCA en fonction du type de conduite.

Discussion Générale

Le travail présenté dans cette thèse s'est focalisé sur l'amélioration des algorithmes de contrôle de mouvement (MCA) pour la validation du véhicule autonome dans les simulateurs dynamiques à hautes performances. Nous nous sommes particulièrement intéressés à déterminer le MCA le plus approprié pour la simulation du véhicule autonome et à identifier les techniques permettant de le rendre plus performant en termes d'utilisation de l'espace de travail et de perception de mouvement.

Le premier chapitre a été consacré à la simulation de conduite, nous avons souligné l'importance de la simulation dans le développement des nouvelles techniques de conduite, du véhicule autonome ainsi que les avantages pour l'industrie automobile. Nous avons commencé avec un état de l'art des simulateurs de conduite avec une classification selon leur niveau de performance et leur structure. Cette étude nous a permis de montrer les stratégies de contrôle les plus utilisées selon le type de simulateur. Cependant, pour comprendre ces stratégies de contrôle, il a été nécessaire de connaître la manière dont l'être humain est capable de percevoir le mouvement. Ainsi, dans le deuxième chapitre, nous avons présenté les différents capteurs sensoriels de l'être humain : le système vestibulaire, le système visuel et la proprioception. À partir de cette étude, nous avons conclu que le système vestibulaire apporte les informations les plus pertinentes concernant la perception du mouvement. Les indices visuels, auditifs et haptiques sont tout aussi importants, mais compte tenu des signaux d'accélération, ils servent principalement à confirmer la sensation de mouvement ressentie par le système vestibulaire. Ainsi, afin d'intégrer la perception de mouvement dans les algorithmes de contrôle plateforme, nous nous sommes basés sur les modèles de perception du système vestibulaire et sur la technique de *tilt coordination*. Celle-ci permet de maximiser l'espace de travail de la plateforme et d'augmenter la sensation de mouvement produite par le simulateur. Néanmoins, pour qu'elle soit appliquée de la meilleure façon possible, des seuils de perception, principalement pour la vitesse de rotation, doivent être respectés. En outre, cette technique ne doit pas être utilisée lorsque les accélérations de référence sont surtout transitoires, car elle peut générer un retard ressenti par le conducteur.

Après avoir montré et justifié les techniques de perception utilisées dans les MCA, le troisième chapitre a présenté les techniques de contrôle elles-mêmes. Dans ce chapitre, les MCA existants (classique, optimal adaptatif et prédictif) et d'autres techniques de contrôle avancé (logique floue et réseaux neuronaux) ont été évalués. Le résultat de cette étude a montré que la technique la plus appropriée pour le

contrôle des plateformes à hautes performances dédiées à la validation du véhicule autonome est la commande prédictive par modèle (MPC). En effet, nous avons pu constater que c'est la seule stratégie de contrôle qui permet de tenir compte des contraintes physiques de la plateforme et des limites perceptives de l'être humain en temps réel, tout en maximisant l'espace de travail des grands simulateurs. De plus, cette stratégie permet d'anticiper la trajectoire du système et de l'intégrer dans la boucle d'optimisation, permettant ainsi un contrôle beaucoup plus précis et optimal qu'avec les autres stratégies. Dans la dernière partie de ce chapitre, nous avons montré un état de l'art du MPC dans les MCA afin de proposer des axes d'amélioration de cette stratégie. En effet, il existe deux aspects principaux de la stratégie de contrôle prédictif qui affectent directement la performance de la loi de contrôle qui en résulte : l'optimisation et le modèle mathématique.

Aussi, dans le quatrième chapitre nous avons commencé par traiter l'optimisation de la stratégie de contrôle prédictif. En tenant compte de la dimension du problème d'optimisation de notre système en termes d'états et de contraintes, nous avons décidé d'implémenter une optimisation en temps réel dite implicite. Ce problème d'optimisation quadratique résultant de la formulation du MPC est étudié depuis longtemps et il existe plusieurs logiciels qui permettent de le résoudre efficacement. Néanmoins, ces processus continuent à générer de nombreux problèmes en ce qui concerne le coût de calcul et la faisabilité du problème d'optimisation. En effet, lorsque le système a de nombreuses restrictions dans les états, le temps de calcul peut augmenter considérablement, ce qui empêche une mise en œuvre en temps réel. En outre, afin d'éviter de mettre en danger la plateforme ou le conducteur, en raison d'une instabilité du système, il est nécessaire de garantir sa faisabilité tout au long de l'horizon de prédiction.

C'est pourquoi nous avons présenté, dans le chapitre 4, une nouvelle méthode d'optimisation basée sur des circuits embarqués, communément appelés réseaux de neurones récurrents. Cette technique consiste à transformer le problème d'optimisation en un système d'équations différentielles de premier ordre par la méthode du gradient. Ensuite, ce système d'équations est représenté comme un réseau neuronal récurrent dont les points d'équilibre correspondent aux variables optimales du problème d'optimisation initial. Contrairement aux méthodes d'optimisation existantes, cette nouvelle méthode est facile à mettre en œuvre, dispose d'une capacité de calcul rapide grâce à sa topologie parallèle et, compte tenu des conditions de faisabilité déterminées auparavant, assure la stabilité du système en boucle fermée.

Pour mettre en œuvre cette stratégie dans le simulateur, deux types de réseaux de neurones ont été testés : le réseau neuronal récurrent basé sur la méthode de pénalisation et le réseau neuronal récurrent basé sur la théorie de dualité. Pour que les réseaux neuronaux soient stables, nous avons implémenté une fonction de saturation basée sur la théorie de faisabilité présentée au début du chapitre 4. Le RNR basé sur la pénalisation a été appliqué efficacement au simulateur avec une optimisation à 1 DDL, dans laquelle chaque DDL est optimisé indépendamment des autres. Néanmoins, pour une optimisation à 2 DDL la méthode de pénalisation n'est pas robuste et le système peut ne pas respecter les contraintes de sécurité. En

outre, si le MPC est utilisé avec une optimisation à 1 DDL et une référence future constante, l'implémentation de cette stratégie n'est pas indispensable, car elle peut être remplacée par une stratégie classique ou adaptative avec les paramètres de filtrage appropriés. C'est pourquoi il a été décidé d'adopter une stratégie RNR avec la méthode duale, car aucun paramètre de pénalité n'est nécessaire pour respecter les restrictions. Cette méthode nous a permis de résoudre le problème d'optimisation à 2 DDL. Cependant, pour une mise en œuvre en temps réel, il est nécessaire d'implémenter une méthode numérique à pas d'échantillonnage fixe, ce qui n'a pas été possible avec la méthode de RNR duale, car les états du réseau ne convergent vers une solution stable qu'en utilisant un pas d'échantillonnage variable. Par conséquent, la technique d'optimisation avec des réseaux neuronaux récurrents n'a pas pu être testée dans le simulateur de conduite dynamique. Nous avons montré dans la partie suivante les perspectives relatives à cette méthode d'optimisation.

Le cinquième et dernier chapitre évoque enfin le second axe d'amélioration de la stratégie de contrôle prédictif abordé dans cette thèse : le modèle mathématique du système. Dans un premier temps, nous avons étudié les différents types de modèles utilisés pour les MCA dans les simulateurs de conduite à hautes performances. Nous avons vu que la grande majorité des auteurs utilisent deux types de modèles : un modèle de système vestibulaire pour représenter la perception humaine du mouvement et un modèle mathématique de plateforme sous forme d'un gain unitaire ou d'un double intégrateur pour intégrer les états du système tels que la position, la vitesse et l'accélération. Dans ce chapitre, notre objectif était de connaître la contribution de l'implémentation de ces types de modèles sur la performance du simulateur, l'utilisation de l'espace de travail de la plateforme et sur la perception de mouvement, en conduisant le simulateur activement ou en utilisant un modèle de véhicule en mode autonome. Par ailleurs, nous voulions savoir si le type de comportement en conduite d'une personne avait un impact sur sa préférence pour un modèle. Pour cela, nous avons développé une expérimentation utilisateur dans le simulateur de conduite dynamique de Renault qui comparait trois modèles : deux avaient un modèle de perception différent et le troisième n'avait pas de modèle de perception. Avec cette étude nous avons constaté que les trois modèles provoquent des performances différentes dans la loi de contrôle pour les accélérations transitoires et continues selon les axes de mouvement x et y. Parmi les deux premiers modèles de perception, la grande majorité des participants a préféré le modèle qui génère le moins de retard de signal et celui qui reproduit plus d'accélérations transitoires que continues. Nous pouvons en conclure que dans la restitution du mouvement, les accélérations transitoires priment sur les accélérations continues générées par la technique de *tilt coordination*. De fait, cette technique doit être utilisée avec beaucoup de précautions dans une stratégie de type MPC : si la référence future est constante sur l'horizon de prédiction, il est recommandé de ne l'appliquer que lorsque les cas d'utilisation génèrent des accélérations continues, tels que des longs virages ou la conduite sur autoroute (sans changement de file). En revanche, si la référence future est connue, cette technique peut être appliquée pour les deux types d'accélération, puisque c'est le processus d'optimisation qui génère l'angle d'inclinaison basé sur la référence.

En comparant les MCA avec ou sans modèle de perception, nous concluons qu'il est nécessaire de faire plus d'expériences afin de définir le type de modèle à implémenter dans les algorithmes de restitution de mouvement dédiés à la validation du véhicule autonome. Cependant, il est à noter que ce n'est pas tant le modèle qui affecte la plateforme, mais le pourcentage d'accélération restaurée des accélérations transitoires et continues. En effet, lorsqu'un modèle de perception est inclus, celui-ci affecte le résultat de l'optimisation tout comme les matrices de pondération utilisées. Cette question sera abordée dans les perspectives.

Nous avons également conclu que pour les participants de cette expérimentation, le type de comportement de conduite auto-déclaré a un impact sur la préférence du type de contrôle de mouvement, en particulier pour ceux qui présentent un comportement de conduite plus agressif. En effet, si la personne conduit de manière hostile, elle a tendance à préférer un type de contrôle qui restitue surtout les accélérations transitoires. Ce type de comportement a été comparé avec leur comportement dans le simulateur et nous avons pu constater que ces participants conduisent avec plus de *jerk* et de vitesse dans la plupart des situations de conduite.

D'un point de vue purement technique, tous les algorithmes ont été implémentés sur le logiciel Matlab et Simulink en utilisant différentes fonctions écrites dans le langage C++. Pour la partie simulation, nous avons utilisé le logiciel SCANeR Studio. Les algorithmes ont été implémenté pour fonctionner en temps réel en utilisant le logiciel RT-LAB, puis mis en œuvre dans le simulateur dynamique à hautes performances de Renault.

Perspectives

Dans cette thèse, les différentes stratégies de contrôle utilisées dans les simulateurs à hautes performances ont été évaluées. Après étude de plusieurs travaux de recherche, il a été décidé de choisir la stratégie de contrôle prédictif et de déterminer, à partir de là, différentes pistes d'amélioration pour fournir de meilleures performances et une meilleure perception de mouvement. Grâce à ce travail, nous avons identifié de nombreux axes d'amélioration et de perspectives qui n'ont pas pu être évalués dans le cadre de ce travail, mais qui peuvent être abordés dans des travaux futurs :

Dans le cadre de la stratégie de contrôle prédictif, il existe de nombreuses techniques qui n'ont pas été abordées dans l'algorithme présenté dans cette thèse, mais qui ont été étudiées par d'autres chercheurs afin d'améliorer la performance de la stratégie de contrôle et accroître l'impression de réalisme durant l'expérience de conduite. Nous souhaitons dans le futur les implémenter dans notre algorithme de restitution de mouvement :

- Optimisation des paramètres de pondération : il est bien connu que la loi de contrôle dépend largement des paramètres de pondération de la fonction de

coût, ainsi que des horizons de contrôle et de prédiction [Mohammadi *et al.*, 2018]. L'optimisation de ces paramètres en amont permettrait de réduire le temps utilisé dans la formulation de la fonction de coût, car actuellement, leur valeur est définie par essai et erreur. En outre, elle permettrait de maximiser l'espace de travail de la plateforme et de minimiser l'erreur de suivi de trajectoire.

- Pré-positionnement : en utilisant les connaissances sur la trajectoire à suivre et le modèle du véhicule, la plateforme peut être placée, avant de lancer la simulation, en un point différent du point neutre, ce qui permet d'augmenter ainsi l'espace disponible de déplacement [Hansson & Stenbeck, 2014]. Pendant la simulation un pré-positionnement pourrait également être proposé pour anticiper les accélérations futures et positionner la plateforme à une position optimale avec une accélération inférieure au seuil de perception. La prédiction des accélérations pourrait être générée par exemple avec des méthodes d'apprentissage profond.

- Choisir une référence future variable : dans cette thèse, nous avons décidé de prendre une référence future constante afin d'avoir les mêmes conditions expérimentales lors de la comparaison des différentes stratégies ou méthodes mises en œuvre, car celle-ci fournit une bonne qualité du signal. Cependant, la connaissance de la trajectoire future du véhicule réduit le retard, améliore la correspondance avec la trajectoire de référence et augmente l'utilisation de l'espace de travail [Grottoli *et al.*, 2019]. De plus, non seulement la référence future peut être choisie en fonction de la trajectoire de la route, mais elle peut être adaptée en temps réel en fonction du comportement de conduite de la personne afin de gérer efficacement un éventuel comportement inattendu du conducteur [Bruschetta *et al.*, 2019].

- Appliquer le test objectif de restitution de mouvement (OMCT pour *Objective Motion Cueing Test*) : l'objectif de ce test est de compléter l'évaluation subjective du mouvement d'un simulateur par une mesure plus objective. Ce test a été développé avant tout pour les simulateurs de vol [Stroosma,Olaf *et al.*, 2013; Zaal *et al.*, 2017]. Toutefois, il constitue la base de la définition des critères d'évaluation objective des MCA dans les simulateurs de conduite dynamiques [Drag *et al.*, 2016].

Nous avons proposé dans cette thèse une méthode d'optimisation basée sur un réseau de neurones à temps continu décrit par un ensemble d'équations différentielles ordinaires qui permet de résoudre en temps réel des problèmes d'optimisation. Toutefois, des améliorations sont possibles et souhaitables afin de pouvoir intégrer cette technique dans les simulateurs de conduite en temps réel :

- Analyse de stabilité approfondie : les réseaux de neurones récurrents peuvent osciller, diverger vers l'infini ou suivre une forme chaotique si l'on n'assure pas sa stabilité ni sa convergence. Dans cette thèse, nous avons utilisé la méthode de Liapounov pour définir si la dynamique mathématique du système RNR

était stable. Cependant, ce système d'équations dépend de la formulation du MPC, et comme il existe des restrictions dans les états, nous avons réalisé une analyse de stabilité basée sur la faisabilité du problème de programmation quadratique sur l'ensemble de l'horizon de prédiction. Toutefois, dans les réseaux de neurones récurrents, la stabilité et le taux de convergence vers un point d'équilibre dépendent aussi du paramètre d'apprentissage et des paramètres initiaux du réseau neuronal. Dans cette thèse, ces paramètres ont été fixés et choisis par essai-erreur, cependant ils peuvent être adaptés en temps réel pour maximiser la vitesse de convergence sans perdre la stabilité du système.

- Implémenter une autre topologie de réseaux de neurones récurrents : dans cette thèse, deux types de méthodes de réseaux de neurones récurrents ont été évalués : la méthode de pénalité et la méthode duale. Cependant, il existe d'autres topologies qui peuvent être adaptées pour améliorer l'implémentation dans les logiciels en temps réel tels que les réseaux neuronaux discrets [Pérez-Ilzarbe, 2014].

- Réseaux de neurones dans un système embarqué : l'un des principaux avantages de l'utilisation des RNR est qu'ils peuvent évoluer en continu et non sous forme d'itération comme le font la plupart des systèmes. Cependant, lorsqu'ils sont appliqués sur calculateur, cette capacité est gaspillée. Aussi peut-on penser à un système embarqué et intégré au simulateur, qui résoudrait le problème de l'optimisation de manière parallèle et extrêmement rapide grâce aux composants analogiques.

Enfin, dans le dernier chapitre, nous avons évalué le modèle mathématique de la stratégie de contrôle et avons recherché une relation entre celui-ci et le comportement de conduite auto-déclaré. Pour cette partie, nous proposons également un ensemble de perspectives afin d'améliorer le réalisme de la simulation pour chaque conducteur :

- Unifier les questionnaires sur le comportement de conduite : nous avons utilisé deux des questionnaires les plus utilisés pour définir le comportement auto-déclaré d'une personne : le MDSI et le DBQ. Bien qu'ils comportent des éléments très similaires, ces questionnaires classifient les comportements en différentes et longues sous-catégories. L'idéal serait de créer un seul formulaire dédié à la simulation de conduite dans lequel le comportement du conducteur pourrait être classé en au moins 3 catégories : agressif, normal et passif.

- Créer un MPC adaptatif en fonction du type de comportement : la loi de contrôle résultante du MPC dépend principalement des paramètres de la fonction de coût, qui comprennent les poids de pondération, les contraintes et le modèle mathématique. Cependant, tous les conducteurs n'ont pas la même sensibilité ou ne perçoivent pas le mouvement de la même manière. C'est pourquoi, notre dernière étude nous amène à imaginer une stratégie de contrôle qui s'adapte au comportement de la personne, qui en plus de

maximiser l'espace de travail du simulateur ou de minimiser l'erreur de suivi de trajectoire, maximise également la perception du mouvement et l'impression de réalisme de la simulation à partir de son comportement de conduite. Nous pourrions imaginer que chaque personne qui utiliserait le simulateur remplisse un questionnaire sur son comportement de conduite et qu'en fonction de ce résultat, l'algorithme soit adapté : intégration d'un modèle de perception, augmentation des états du système pour contrôler directement le *jerk*, optimisation des paramètres de pondération, etc. À cette fin, nous pourrions aussi imaginer intégrer des algorithmes d'intelligence artificielle qui détectent en temps réel le comportement de conduite de la personne.

- Créer un banc de scénarios de test : les algorithmes de perception dépendent beaucoup des paramètres de pondération et du modèle de perception, mais ceux-ci dépendent à leur tour de la trajectoire générée à partir d'un scénario de simulation. Dans cette thèse, nous avons considéré différents cas d'utilisation qui nous ont permis de générer des accélérations transitoires et continues pour évaluer les différentes MCA proposées. Cependant, l'idéal serait de pouvoir définir un banc de scénarios de validation basés sur des données réelles et représentatives pour chaque cas d'utilisation, et ainsi pouvoir optimiser les paramètres de la stratégie de contrôle en fonction du type de référence.

- Évaluation du mal du simulateur selon le type de stratégie de mouvement : nous considérons que si la stratégie de restitution de mouvement est choisie selon le type de comportement auto-déclaré du conducteur, le mal du simulateur pourrait être réduit. Cette hypothèse doit être évaluée avec une étude expérimentale dans le simulateur de conduite.

- Analyse expérimentale de l'impact du *jerk* sur les algorithmes de restitution de mouvement : nous avons effectué une première étude en simulation concernant la pertinence du *jerk* dans les MCA (voir annexe C). Dans cette étude, nous avons considéré le *jerk* en tant que variable de décision dans l'optimisation du MPC, afin de le contraindre et d'évaluer son impact sur le suivi de la trajectoire. Cette première étude montre que le *jerk* a un effet dans la restitution de mouvement. Cependant, il est nécessaire de faire d'autres expérimentations pour analyser son impact sur la perception du mouvement du conducteur dans un simulateur de conduite.

- Évaluation continue en temps réel des stratégies de contrôle : l'évaluation continue en temps réel a déjà été appliquée dans certains simulateurs de conduite pour évaluer la charge de travail des conducteurs [Girard *et al.*, 2005] et les faux indices de mouvement [Cleij *et al.*, 2018; van Leeuwen *et al.*, 2018]. Nous proposons une méthode qui consiste en une évaluation continue et non intrusive de la stratégie de contrôle en temps réel, ce qui permettrait d'améliorer le paramétrage du MCA en fonction du type de comportement de conduite. Nous pourrions évaluer avec une meilleure précision certaines caractéristiques du MCA dans chacune des situations de conduite présente dans une simulation, telles que la perception du mouvement, le retard et l'amplitude. Une évaluation continue du MCA en temps réel permettrait

d'obtenir plus d'information que les méthodes actuelles qui ne fournissent que des mesures globales par test de simulation.

Nous estimons que toutes les idées présentées ci-dessus ont le potentiel de compléter le travail développé dans ce projet de recherche. Elles visent à améliorer la performance des stratégies de contrôle du mouvement et la perception du mouvement chez le conducteur, et à contribuer à la validation du véhicule autonome dans le simulateurs de conduite.

ANNEXES

Annexe A

Système d'état utilisé dans l'algorithme optimal

Cette annexe décrit la procédure utilisée afin de réaliser la commande optimale dans la partie 3.3.

Le modèle de la plateforme selon chaque axe de mouvement est représenté par :

$$x_{pf}(k+1) = \overbrace{\begin{bmatrix} 1 & t_s & \frac{t_s^2}{2} \\ 0 & 1 & t_s \\ 0 & 0 & 0 \end{bmatrix}}^{A_{pf}} \overbrace{\begin{bmatrix} p(k) \\ v(k) \\ acc(k) \end{bmatrix}}^{x_{pf}} + \overbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}^{B_{pf}} \overbrace{\begin{bmatrix} acc \end{bmatrix}}^{u}$$

$$y_{pf}(k) = \overbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}^{C_{pf}} x_{pf}$$

avec $t_s = 0,01s$.

Le modèle du bruit blanc $n(t)$ est représenté par :

$$\dot{x_n} = A_n x_n + B_n n$$
$$u_{veh} = x_n$$

Le modèle de l'erreur $x_e = x_{pf} - x_n$ de suivi de trajectoire est représenté par :

$$\dot{x_e} = A_{pf} x_e + B_{pf} u - B_{pf}^T u_{veh} e = C_{pf} x_e$$

Le modèle utilisé dans l'optimisation et qui regroupe l'erreur du système x_e, les états de la plateforme x_{pf} et le bruit blanc filtré x_n devient :

$$\dot{x} = \overbrace{\begin{bmatrix} A_{pf} & 0 & -B_{pf} \\ 0 & A_{pf} & 0 \\ 0 & 0 & A_n \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} x_e \\ x_{pf} \\ x_n \end{bmatrix}}^{x} + \overbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}^{B} u + \overbrace{\begin{bmatrix} 0 \\ 0 \\ B_n \end{bmatrix}}^{H} n$$

$$y_{=} \overbrace{\begin{bmatrix} C_{pf} & 0 & 0 \end{bmatrix}}^{C} x$$

Annexe B

Exemple de réseau neuronal récurrent

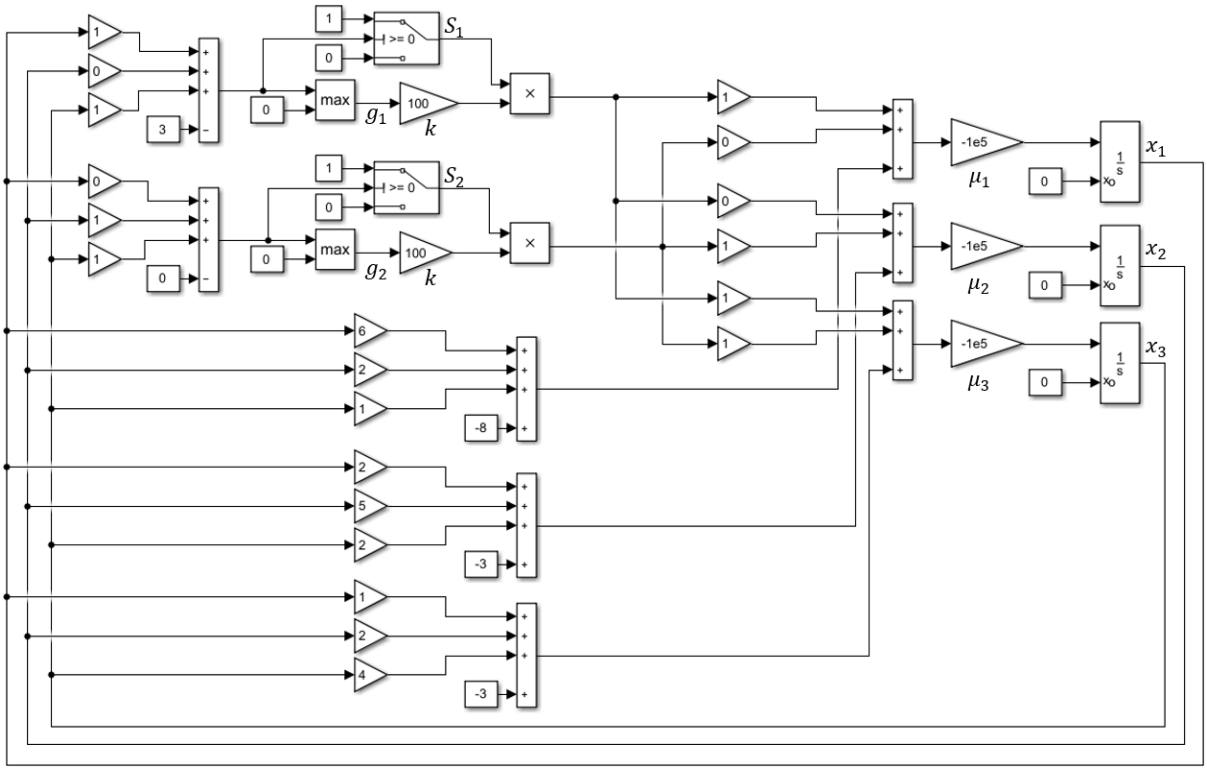
Le problème de programmation quadratique avec contraintes d'inégalité

$$\begin{aligned} &\text{minimiser} \quad f(x) = \frac{1}{2}x^T H x + f^T x \\ &\text{soumis à} \quad g_i(x) = A_c x - b \leq 0, \quad i = 1, 2...m. \end{aligned}$$

où,

$$H = \begin{bmatrix} 6 & 2 & 1 \\ 2 & 5 & 2 \\ 1 & 2 & 4 \end{bmatrix}, \quad f = \begin{bmatrix} -8 \\ -3 \\ -3 \end{bmatrix}, \quad A_c = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$$

est résolu par le réseau de neurones récurrents (39) suivant :



La solution au problème d'optimisation est : $x_1 = 1,379$, $x_2 = -0,276$ et $x_3 = 0,276$.

Annexe C

Influence des à-coups dans les algorithmes de contrôle de mouvement

Cette annexe fournit quelques informations sur l'importance de la prise en compte des saccades (*jerk*) dans la conception des algorithmes de restitution de mouvement.

Le jerk est défini comme le taux de variation de l'accélération ou de la décélération. Il représente la fluidité de la conduite, autrement dit un freinage brusque ou une accélération rapide ou une conduite irrégulière entraînent un *jerk* moyen plus important que chez les conducteurs qui décélèrent et accélèrent plus doucement ou maintiennent une vitesse plus stable [Hooft van Huysduynen *et al.*, 2018].

Dans l'expérimentation menée dans le chapitre 5, l'un de nos objectifs était de comparer différentes stratégies de mouvement. Lorsque nous avons comparé chacune des stratégies en termes de suivi de trajectoire dans la partie 5.1.3, nous avons remarqué que l'une des caractéristiques qui varient le plus est la contribution des à-coups. De plus, en comparant les indicateurs de mouvement avec le type de comportement auto-déclaré, nous nous sommes rendu compte que le *jerk* peut donner des informations importantes sur le comportement de conduite du conducteur [Kruk, 2017; Murphey *et al.*, 2009] et le type de contrôle qu'il préférera. C'est pourquoi nous considérons que le *jerk* doit être étudiée pour vérifier sa contribution aux stratégies de contrôle de mouvements.

D'après la littérature, le *jerk* a attiré davantage l'attention du secteur automobile : [Mehrotra & Mahapatra, 1997] soutiennent qu'un modèle de suivi basé sur le *jerk* offre des performances supérieures au modèle basé sur le suivi de l'accélération. Ils ont constaté que les modèles de suivi du *jerk* donnent de meilleures performances s'il s'agit de manœuvres dont l'accélération varie rapidement, tels que les freinages et les démarrages. En outre, le mouvement de secousse contribue à l'impression subjective de la force du mouvement et qu'un mouvement avec une accélération plus faible, mais un mouvement saccadé plus élevé pourrait créer une meilleure perception du mouvement [Haycock & Grant, 2007].

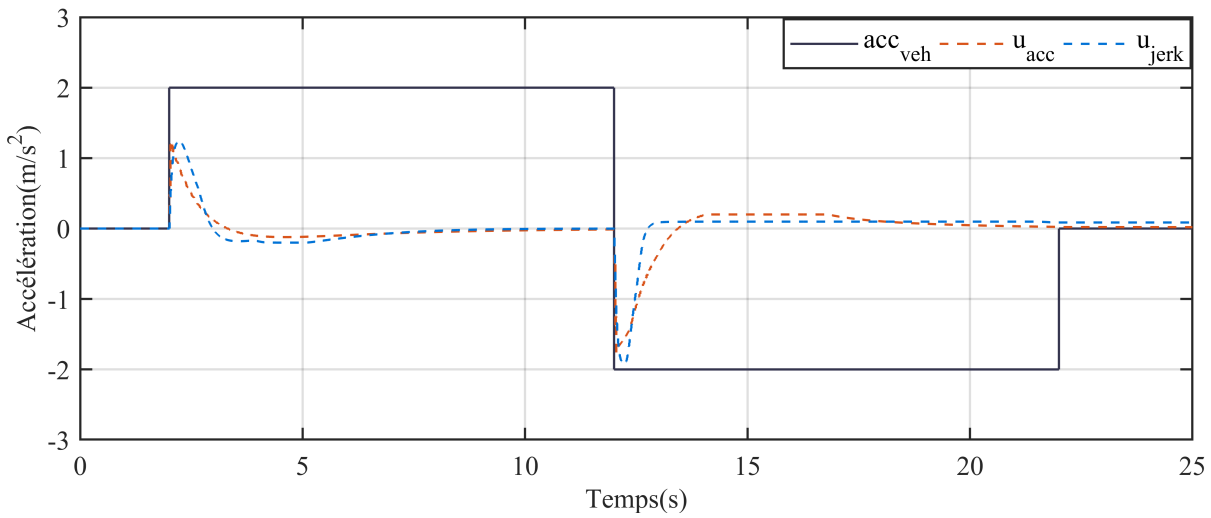
Pour évaluer la contribution de la secousse dans le contrôle du mouvement du simulateur, nous proposons un MCA basé sur le MPC qui, au lieu de contrôler les accélérations, il contrôle la secousse pour ajouter des contraintes au *jerk*. Pour

ce faire, nous comparons deux stratégies utilisant une optimisation à 1 DDL. La première utilise le modèle (16) contrôlé par l'accélération u_{acc}, la seconde utilise le modèle ci-dessous, contrôlé par le jerk u_{jerk} :

$$\dot{x_{sim}} = \overbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}^{A_{sim}} \overbrace{\begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix}}^{x_{sim}} + \overbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}^{B_{sim}} \overbrace{\begin{bmatrix} jerk \end{bmatrix}}^{u}$$

$$y_{sim} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x_{sim}$$

Afin de comparer les deux stratégies en simulation, nous implémentons un signal acc_{veh} avec une phase d'accélération de $2\ m/s^2$ pendant 10 secondes et une décélération continue de $-2\ m/s^2$. Le suivi de trajectoire des deux stratégies est montré dans la figure ci-dessous :



Nous pouvons remarquer que la stratégie contrôlée par le *jerk* u_{jerk} est plus lisse dans les accélérations transitoires, car il a été contraint dans l'optimisation. En effet, la valeur maximale du *jerk* avec cette stratégie est $0,209\ m/s^3$. Elle est inférieure à celle avec la stratégie contrôlée en accélération ($0,755\ m/s^3$). Bien que cette simulation montre la pertinence de la prise en compte du *jerk* dans les MCA, il est nécessaire de réaliser plus d'expérimentations dans le simulateur, afin d'analyser l'impact sur la perception du mouvement.

Annexe D

Questionnaires

D.1 Questionnaire sur le comportement du conducteur au volant

Sexe : ☐ F ☐ M **Âge :** ___

Pour chacune des questions suivantes, nous vous demandons d'indiquer la fréquence à laquelle vous avez eu ce type de comportement. Basez votre jugement par rapport à votre expérience de conduite personnelle. Ne passez donc pas trop de temps à réfléchir à chaque question. Donnez simplement votre impression aussi rapidement que possible en cochant la case qui vous paraît la plus appropriée. Nous vous demandons d'être aussi honnête que possible. Vos réponses sont anonymes.

1 : jamais **2 : rarement** **3 : parfois** **4 : assez souvent** **5 : fréquemment** **6 : presque toujours**

	1	2	3	4	5	6
1. Vous essayez de démarrer au feu en 2ème ou en 3ème par inadvertance	☐	☐	☐	☐	☐	☐
2. Vous êtes impatient aux heures de pointe	☐	☐	☐	☐	☐	☐
3. Vous collez le véhicule devant vous ou bien vous lui faites des appels de phares pour lui dire d'aller plus vite ou de changer de voie	☐	☐	☐	☐	☐	☐
4. Vous ne vous souvenez plus où vous avez garé votre véhicule dans un grand parking	☐	☐	☐	☐	☐	☐
5. Vous êtes distrait(e) et vous réalisez tardivement que le véhicule devant vous a ralenti. Vous devez freiner brutalement pour éviter la collision	☐	☐	☐	☐	☐	☐
6. Vous tentez de mettre les essuie-glaces, mais vous allumez les phares ou vice-versa	☐	☐	☐	☐	☐	☐
7. Vous êtes inquiet(e) quand vous conduisez par mauvais temps	☐	☐	☐	☐	☐	☐
8. Vous réalisez soudain que vous n'avez pas de souvenir du trajet que vous venez juste de parcourir	☐	☐	☐	☐	☐	☐
9. Vous manquez votre sortie et vous devez faire un long détour	☐	☐	☐	☐	☐	☐
10. Vous oubliez à quel rapport de vitesse vous roulez et vous devez vérifier avec votre main	☐	☐	☐	☐	☐	☐
11. Vous vous sentez détendu (e) en conduisant	☐	☐	☐	☐	☐	☐
12. Vous croisez les doigts et passez alors que le feu vient de passer au rouge	☐	☐	☐	☐	☐	☐
13. Vous êtes irrité(e) par le comportement d'un autre conducteur et vous décidez de lui donner la chasse afin de lui montrer ce que vous pensez	☐	☐	☐	☐	☐	☐
14. Délibérément, vous ne respectez pas la limitation de vitesse tard dans la nuit ou très tôt le matin	☐	☐	☐	☐	☐	☐
15. En tournant à droite, vous manquez de renverser un cycliste qui était sur votre droite	☐	☐	☐	☐	☐	☐

16. Vous lisez vos messages en conduisant	☐	☐	☐	☐	☐	☐
17. Vous prenez le volant même si vous réalisez que votre taux d'alcoolémie peut être supérieur à la limite légale	☐	☐	☐	☐	☐	☐
18. Vous avez une aversion pour une catégorie particulière d'usager de la route et vous indiquez votre hostilité par n'importe quel moyen	☐	☐	☐	☐	☐	☐
19. Perdu(e) dans vos pensées, vous ne faites pas attention à la présence d'un piéton qui attend à un passage piéton	☐	☐	☐	☐	☐	☐
20. Vous vous sentez mal à l'aise en conduisant	☐	☐	☐	☐	☐	☐
21. En reculant, vous percutez quelque chose que vous n'avez pas vu	☐	☐	☐	☐	☐	☐
22. Vous conduisez prudemment	☐	☐	☐	☐	☐	☐
23. Sur une voie rapide, vous doublez un véhicule lent par la droite	☐	☐	☐	☐	☐	☐
24. Vous vous insérez par une mauvaise voie dans un rond-point ou à l'approche d'une intersection	☐	☐	☐	☐	☐	☐
25. Vous lisez mal les panneaux et vous prenez la mauvaise sortie à un rond-point	☐	☐	☐	☐	☐	☐
26. Vous oubliez de contrôler dans vos rétroviseurs avant de sortir d'un stationnement, de changer de voie, de tourner…	☐	☐	☐	☐	☐	☐
27. Vous tentez de doubler un véhicule qui signale un virage à gauche	☐	☐	☐	☐	☐	☐
28. Vous ignorez les panneaux de priorité et vous évitez de justesse une collision avec un véhicule prioritaire	☐	☐	☐	☐	☐	☐
39. Vous vous trouvez impliqué(e) dans une « course non officielle » avec d'autres conducteurs	☐	☐	☐	☐	☐	☐
30. Vous faites des activités relaxantes en conduisant une voiture	☐	☐	☐	☐	☐	☐
31. Lorsque vous vous préparez à passer d'une route secondaire à une route principale, vous faites tellement attention à la circulation sur la route principale que vous avez failli heurter le véhicule devant vous.	☐	☐	☐	☐	☐	☐
32. Intentionnellement, vous roulez trop près d'autres conducteurs	☐	☐	☐	☐	☐	☐
33. Vous avez l'impression d'avoir le contrôle de la conduite	☐	☐	☐	☐	☐	☐
34. Sur une autoroute dégagée, vous conduisez habituellement à une vitesse égale ou légèrement inférieure à la limite de vitesse	☐	☐	☐	☐	☐	☐
35. Dans un embouteillage, lorsque les véhicules sur la voie à côté commencent à bouger, vous essayez de vous y insérer dès que possible	☐	☐	☐	☐	☐	☐
36. Conduire vous frustre	☐	☐	☐	☐	☐	☐
37. Vous planifiez si mal votre trajet que vous vous retrouvez dans un embouteillage que vous auriez pu éviter	☐	☐	☐	☐	☐	☐
38. Vous attendez patiemment lorsque vous n'avez pas la priorité.	☐	☐	☐	☐	☐	☐
39. Dans un embouteillage, vous réfléchissez à des moyens de traverser le traffic plus rapidement	☐	☐	☐	☐	☐	☐

	1	2	3	4	5	6
40. Vous vous coiffez/mangez/maquillez en conduisant	☐	☐	☐	☐	☐	☐
41. Vous aimez prendre des risques en conduisant	☐	☐	☐	☐	☐	☐
42. Vous fondez votre comportement sur la devise « mieux vaut prévenir que guérir »	☐	☐	☐	☐	☐	☐
43. Vous sous-estimez la vitesse d'un véhicule approchant lorsque vous dépassez	☐	☐	☐	☐	☐	☐
44. Vous écrivez des messages en conduisant	☐	☐	☐	☐	☐	☐
45. En ayant l'intention de vous rendre à une destination, vous vous retrouvez sur la route de votre destination habituelle	☐	☐	☐	☐	☐	☐
46. Vous êtes toujours prêt à réagir aux manœuvres inattendues des autres conducteurs	☐	☐	☐	☐	☐	☐
47. Vous appréciez l'excitation de la conduite dangereuse	☐	☐	☐	☐	☐	☐

D.2 Questionnaire sur la stratégie de restitution de mouvement

Selon les différentes situations, indiquer votre ressenti du mouvement du simulateur en écrivant un numéro de 1-7 dans la case appropriée.

1. **Dans les situations de démarrage ou freinage** (passage piéton, insertion autoroute, véhicule arrêté dans votre voie, dépassement) :

Caractéristique		**Score**	
	1	→	**7**
Douceur	Extrêmement lisse - comparable à l'immobilité	—	Extrêmement brusque / limite de tolérance
Perception de mouvement	Totalement juste, comme dans une vraie voiture	—	Complètement inversée
Amplitude	Aucun mouvement n'a été ressenti	—	Au moins deux fois plus fort que prévu
Retard	Aucun	—	Excessif, inacceptable
Inconfort	Aucun	—	Inacceptable
Désorientation	Aucun	—	Impossible de finir l'expérimentation

2. **Dans les longs virages** (la sortie du village et la sortie de l'autoroute)

Caractéristique		**Score**	
	1	→	**7**
Douceur	Extrêmement lisse - comparable à l'immobilité	—	Extrêmement brusque / limite de tolérance
Perception de mouvement	Totalement juste, comme dans une vraie voiture	—	Complètement inversée
Amplitude	Aucun mouvement n'a été ressenti	—	Au moins deux fois plus fort que prévu
Retard	Aucun	—	Excessif, inacceptable
Inconfort	Aucun	—	Inacceptable
Désorientation	Aucun	—	Impossible de finir l'expérimentation

3. **Dans la section des obstacles dans les voies (SLALOM)**

Caractéristique		**Score**	
	1	→	**7**
Douceur	Extrêmement lisse - comparable à l'immobilité	—	Extrêmement brusque / limite de tolérance

Perception de mouvement	Totalement juste, comme dans une vraie voiture	—	Complètement inversée
Amplitude	Aucun mouvement n'a été ressenti	—	Au moins deux fois plus fort que prévu
Retard	Aucun	—	Excessif, inacceptable
Inconfort	Aucun	—	Inacceptable
Désorientation	Aucun	—	Impossible de finir l'expérimentation

4. **Dans les parties des accélérations continues** (conduite en autoroute et dernier trajet)

Caractéristique		**Score**	
	1	→	**7**
Douceur	Extrêmement lisse - comparable à l'immobilité	—	Extrêmement brusque / limite de tolérance
Perception de mouvement	Totalement juste, comme dans une vraie voiture	—	Complètement inversée
Amplitude	Aucun mouvement n'a été ressenti	—	Au moins deux fois plus fort que prévu
Retard	Aucun	—	Excessif, inacceptable
Inconfort	Aucun	—	Inacceptable
Désorientation	Aucun	—	Impossible de finir l'expérimentation

Donnez une valeur à chaque affirmation (1 : fortement en désaccord / 7 : tout à fait d'accord)

5. **De façon générale…**

a) J'ai oublié le simulateur :	—
b) Le mouvement semblait réaliste	—
c) J'avais l'impression que la conduite est comme d'habitude :	—
d) J'ai conduit comme d'habitude (seulement si conduite libre) :	—

Commentaires : Cliquez ou appuyez ici pour entrer du texte.

Bibliographie

Anderson, James, Kalra, Nidhi, Stanley, Karlyn, Sorensen, Paul, Samaras, Constantine, & Oluwatola, Oluwatobi. 2016. *Autonomous Vehicle Technology : A Guide for Policymakers*. RAND Corporation. 9

Ariel, Dan, & Silvan, Raphael. 1984. False cue reduction in moving flight simulators. 58, 59

Arioui, Hichem, & Nehaoua, Lamri. 2013. Simulateur de conduite deux-roues : Outil de formation. 19

Arruñada, Fernando. 2015. Vestibular anatomy. 27, 33

Asadi, Houshyar, Mohamed, Shady, Nelson, Kyle, & Nahavandi, Saeid. 2014. A linear quadratic optimal motion cueing algorithm based on human perception. *ACRA 2014 : Proceedings of Australasian Conference on Robotics and Automation*, 1–9. 57, 65, 122

Asadi, Houshyar, Mohamed, Shady, & Nahavandi, Saeid. 2015a. Incorporating Human Perception With the Motion Washout Filter Using Fuzzy Logic Control. *IEEE/ASME Transactions on Mechatronics*, **20**(6), 3276–3284. 3, 37, 42, 76, 77, 122

Asadi, Houshyar, Mohamed, Shady, Nelson, Kyle, Nahavandi, Saeid, & Oladazimi, Maysam. 2015b. An optimal washout filter based on genetic algorithm compensators for improving simulator driver perception. *Pages 1–10 of : DSC 2015 : Proceedings of the Driving Simulation Conference & Exhibition*. [The Conference]. 64

Asadi, Houshyar, Mohamed, Shady, Lim, Chee Peng, & Nahavandi, Saeid. 2016. A review on otolith models in human perception. *Behavioural Brain Research*, **309**(Aug.), 67–76. 37

Asadi, Houshyar, Mohamed, Shady, lim, Chee Peng, & Nahavandi, Saeid. 2017. Robust optimal motion cueing algorithm based on the linear quadratic regulator method ans a genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics : Systems*, **47**(2), 238–254. 3, 37, 122

Augusto, Bruno Daniel Correia. 2009. Motion cueing in the Chalmers driving simulator : An optimization-based control approach. 3, 37, 42, 93, 122

Bahrami, Arian, Masoule, Mojtaba Tafaoli, & Bahrami, Mansour Nikkhah. 2013. Fuzzy Control of Piezoelectric Stewart Platform for Active Vibration Control Purpose. *Transaction on Control and Mechanical Systems*, **1**(8). 76

Baseggio, Mauro, Beghi, Alessandro, Bruschetta, Mattia, Maran, Fabio, & Minen, Diego. 2011. An MPC approach to the design of motion cueing algorithms for driving simulators. *Pages 692–697 of : 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. 3, 37, 84, 93, 122

Bataille, Alexandre, & Chapus, François. 2016. *La voiture autonome, c'est parti !* 11

Beghi, Alessandro, Bruschetta, Mattia, & Maran, Fabio. 2012. A real time implementation of MPC based Motion Cueing strategy for driving simulators. *Pages 6340–6345 of : 2012 IEEE 51st Annual Conference on Decision and Control (CDC),*. IEEE. 2, 3, 37, 73, 84, 91, 93, 122

Bellon, James. 2008. Riccati Equations in Optimal Control Theory. 63

Bemporad, Alberto, & Morari, Manfred. 1999. Robust model predictive control : A survey. *Pages 207–226 of :* Garulli, A., & Tesi, A. (eds), *Robustness in identification and control*, vol. 245. London : Springer London. 65, 112

Bener, Abdulbari, Özkan, Türker, & Lajunen, Timo. 2008. The Driver Behaviour Questionnaire in Arab Gulf countries : Qatar and United Arab Emirates. *Accident Analysis & Prevention*, **40**(4), 1411–1417. 124

Bengler, Klaus, Dietmayer, Klaus, Farber, Berthold, Maurer, Markus, Stiller, Christoph, & Winner, Hermann. 2014. Three Decades of Driver Assistance Systems : Review and Future Perspectives. *IEEE Intelligent Transportation Systems Magazine*, **6**(4), 6–22. 9

Berthoz, A., Bles, W., Bulthoff, H. H., Correia Gracio, B. J., Feenstra, P., Filliard, N., Huhne, R., Kemeny, A., Mayrhofer, M., Mulder, M., Nusseck, H. G., Pretto, P., Reymond, G., Schlusselberger, R., Schwandtner, J., Teufel, H., Vailleau, B., van Paassen, M. M., Vidal, M., & Wentink, M. 2013. Motion Scaling for High-Performance Driving Simulators. *IEEE Transactions on Human-Machine Systems*, **43**(3), 265–276. 92, 135

Blana, Eva. 1996. *A Survey of Driving Research Simulators Around the World*. ITS Working papers 481. University of Leeds. 15, 18, 20, 22

Bordons, C, & Camacho, EF. 2007. *Model predictive control*. Springer Verlag London Limited. 66, 71, 91, 95

Bouzerdoum, Abdesselam, & Pattison, Tim R. 1993. Neural network for quadratic optimization with bound constraints. *IEEE transactions on neural networks*, **4**(2), 293–304. 104

Brown, Bruce. 2017. *Evidence stacks up in favor of self-driving cars in 2016 NHTSA fatality report*. 9

Bruschetta, M., Maran, F., & Beghi, A. 2017. A fast implementation of MPC-based motion cueing algorithms for mid-size road vehicle motion simulators. *Vehicle System Dynamics*, **55**(6), 802–826. 73

Bruschetta, Mattia, Maran, Fabio, & Beghi, Alessandro. 2016. A Nonlinear, MPC-Based Motion Cueing Algorithm for a High-Performance, Nine-DOF Dynamic Simulator Platform. *IEEE Transactions on Control Systems Technology*. 84, 92

Bruschetta, Mattia, Cenedese, Carlo, & Beghi, Alessandro. 2019. A real-time, MPC-based Motion Cueing Algorithm with Look-Ahead and driver characterization. *Transportation Research Part F : Traffic Psychology and Behaviour*, **61**(Feb.), 38–52. 3, 37, 122, 165

Cardullo, F, Telban, R, & Houck, J. 1999. Motion cueing algorithms : A human centered approach. *Pages 1–11 of : 5th International Symposium on Aeronautical Sciences*. 52, 62

Castignani, Luca. 2018. Achieving Autonomous Driving with Simulation & Testing. *Engineering Reality Magazine*, **8**. 1

Chapron, Thomas, & Colinot, Jean-Pierre. 2007. The new psa peugeot-citroen advanced driving simulator overall design and motion cue algorithm. *In : Proceedings of Driving Simulation Conference*. 22

Chen, Jiahan, Shanblatt, Michael A, & Maa, Chia-Yiu. 1991. Improved neural networks for linear and nonlinear programming. *International Journal of Neural Systems*, **2**(04), 331–339. 103

Chen, LD, Papelis, Yiannis, Waston, G, & Solis, Dario. 2001. NADS at the University of IOWA : A tool for driving safety research. *In : Proceedings of the 1st human-centered transportation simulation conference*. 21

Chen, Sung-Hua, & Fu, Li-Chen. 2010. An optimal washout filter design for a motion platform with senseless and angular scaling maneuvers. *Pages 4295–4300 of : Proceedings of the 2010 American Control Conference*. IEEE. 3, 37, 62, 63, 122

Cheng, Long, Hou, Zeng-Guang, & Tan, Min. 2007. Constrained multi-variable generalized predictive control using a dual neural network. *Neural Computing and Applications*, **16**(6), 505–512. 103, 113

Cheng, Long, Hou, Zeng-Guang, Homma, Noriyasu, Tan, Min, & Gupta, Madam M. 2009. Solving convex optimization problems using recurrent neural networks in finite time. IEEE. 104

Cleij, D., Venrooij, J., Pretto, P., Katliar, M., Bülthoff, H.H., Steffen, D., Hoffmeyer, F.W., & Schöner, H.-P. 2019. Comparison between filter- and optimization-based motion cueing algorithms for driving simulation. *Transportation Research Part F : Traffic Psychology and Behaviour*, **61**(Feb.), 53–68. 2

Cleij, Diane, Venrooij, Joost, Pretto, Paolo, Pool, Daan M., Mulder, Max, & Bulthoff, Heinrich H. 2018. Continuous Subjective Rating of Perceived Motion Incongruence During Driving Simulation. *IEEE Transactions on Human-Machine Systems*, **48**(1), 17–29. 167

Cochocki, A, & Unbehauen, Rolf. 1993. *Neural networks for optimization and signal processing*. John Wiley & Sons, Inc. 103, 105, 106, 109

Colombet, Florent. 2010. *Facteur d'échelle visuelle par la restitution de la perception de vitesse en simulation de conduite automobile*. Ph.D. thesis, Ecole nationale supérieure d'arts et métiers - ENSAM. 30, 39, 46

Colombet, Florent, Fang, Zhou, & Kemeny, Andras. 2017. Tilt thresholds for acceleration rendering in driving simulation. *SIMULATION*, **93**(7), 595–603. 41

Conrad, Bjorn, & Schmidt, SF. 1970. Motion drive signals for piloted flight simulators. 52

Coren, Stanley. 2003. *Sensation and perception*. Wiley Online Library. 27

Correia Augusto, Bruno Daniel, & Leal Loureiro, Rui José. 2009. *Motion cueing in the Chalmers driving simulator-A model predictive control approach*. Ph.D. thesis, Chalmers University of Technology. 84, 91

Costantini, G., Perfetti, R., & Todisco, M. 2008. Quasi-Lagrangian Neural Network for Convex Quadratic Optimization. *IEEE Transactions on Neural Networks*, **19**(10), 1804–1809. 104, 109

Dagdelen, Mehmet. 2005. *Restitution des stimuli inertiels en simulation de conduite*. Ph.D. thesis, Paris, ENMP. 41

Dagdelen, Mehmet, Reymong, G., & Maïzi, Nadia. 2004. MPC Based motion cueing algorithm : developpement and application to the ULTIMATE driving simulator. DSC 2004 Europe. 2, 80, 84, 92, 94

Dagdelen, Mehmet, Berlioux, Jean-Charles, Panerai, Francesco, Reymond, Gilles, & Kemeny, Andras. 2006. Validation process of the ULTIMATE high-performance driving simulator. *In : Driving Simulation Conference, Europe*. 53

Dagdelen, Mehmet, Reymond, Gilles, Kemeny, Andras, Bordier, Marc, & Maïzi, Nadia. 2009. Model-based predictive motion cueing strategy for vehicle driving simulators. *Control Engineering Practice*, **17**(9), 995–1003. 3, 54, 60, 65, 84, 122

Demuth, Howard B, Beale, Mark H, De Jesus, Orlando, & Hagan, Martin T. 2014. *Neural network design*. Martin Hagan. 101, 102, 103, 205

Direction des routes et de la circulation routière. 1981. *Vision, exploration visuelle et sécurité routière*. 29, 30

Drag, Pawel, Styczeń, Krystyn, Kwiatkowska, Marlena, & Szczurek, Andrzej. 2016. A Review on the Direct and Indirect Methods for Solving Optimal Control Problems with Differential-Algebraic Constraints. *Pages 91–105 of :* Fidanova, Stefka (ed), *Recent Advances in Computational Optimization*, vol. 610. Cham : Springer International Publishing. 165

Drosdol, Johannes, & Panik, Ferdinand. 1985 (Feb.). The Daimler-Benz Driving Simulator A Tool for Vehicle Development. 15

Duh, Henry Been-Lirn, Parker, Donald E., Philips, James O., & Furness, Thomas A. 2004. "Conflicting" Motion Cues to the Visual and Vestibular Self- Motion Systems Around 0.06 Hz Evoke Simulator Sickness. *Human Factors : The Journal of the Human Factors and Ergonomics Society*, **46**(1), 142–153. 46

Egea-Caparrós, Amaro. 2001. El comportamiento humano en conducción : factores perceptivos, cognitivos y de respuesta. *Apuntes de Psicología Cognitiva Aplicada*. 29

Espié, Stéphane, Boubezoul, Abderrahmane, Aupetit, Samuel, & Bouaziz, Samir. 2013. Data collection and processing tools for naturalistic study of powered two-wheelers users' behaviours. *Accident Analysis & Prevention*, **58**(Sept.), 330–339. 124

Fang, Z., Tsushima, M., Kitahara, E., Machida, N., Wautier, D., & Kemeny, A. 2017. Motion cueing algorithm for high performance driving simulator using yaw table. *IFAC-PapersOnLine*, **50**(1), 15965–15970. 3, 85, 89, 122, 127

Fang, Zhou, & Kemeny, Andras. 2012a. Explicit MPC motion cueing algorithm for real-time driving simulator. *Pages 874–878 of : 7th International conference on Power Electronics and Motion Control Conference (IPEMC)*, vol. 2. IEEE. 2, 73, 85, 92, 94

Fang, Zhou, & Kemeny, Andras. 2012b. Motion cueing algorithms for a real-time automobile driving simulator. 2

Fang, Zhou, & Kemeny, Andras. 2014. Review and prospects of Renault's MPC based motion cueing algorithm for driving simulator. *In : Driving Simulation Conference, Paris*. 3, 23, 37, 44, 70, 122

Fang, Zhou, & Kemeny, Andras. 2016. An efficient Model Predictive Control-based motion cueing algorithm for the driving simulator. *SIMULATION*, **92**(11), 1025–1033. 57, 85, 92

Fang, Zhou, Colombet, Florent, Collinet, Jean-Christophe, Kemeny, Andras, & Center, Immersive Simulation. 2014. Roll tilt thresholds for 8 DOF driving simulators. *Pages 4–5 of : Proceedings of the Driving Simulation Conference Europe, Paris, France*. 129

Fellah, Khaled, Guiatni, Mohamed, & Morsly, Yacine. 2013. Fuzzy/PSO based washout filter for inertial stimuli restitution in flight simulation. 77

Ferreau, Hans Joachim, Kirches, Christian, Potschka, Andreas, Bock, Hans Georg, & Diehl, Moritz. 2014. qpOASES : a parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, **6**(4), 327–363. 96, 126

Fischer, Martin, Sehammar, Håkan, & Palmkvist, Göran. 2011. Applied Motion Cueing Strategies for Three Different Types of Motion Systems. *Journal of Computing and Information Science in Engineering*, **11**(4). 55, 58

Fletcher, R. 2000. *Practical methods of optimization*. 2nd ed edn. Chichester ; New York : Wiley. 113

French, D. J., West, R. J., Elander, J., & Wilding, J. M. 1993. Decision-making style, driving style, and self-reported involvement in road traffic accidents. *Ergonomics*, **36**(6), 627–644. 123

Friedland, Bernand, Ling, Chong-Kuan, & Hutton, Maurice F. 1973 (Oct.). *Quasi-optimum design of a six degree of freedom moving base simulator control system*. Tech. rept. NASA, Washington. 53, 57

Gabaude, C., Marquié, J.-C., & Obriot-Claudel, F. 2010. Self-regulatory driving behaviour in the elderly : relationships with aberrant driving behaviours and perceived abilities. *Le travail humain*, **73**(1), 31. 134

Garrett, Nikhil JI, & Best, Matt C. 2010. Driving simulator motion cueing algorithms – a survey of the state of the art. 2

Garrett, Nikhil JI, & Best, Matthew C. 2013. Model predictive driving simulator motion cueing algorithm with actuator-based constraints. *Vehicle System Dynamics*, **51**(8), 1151–1172. 2, 3, 37, 84, 91, 92, 122

Gerdes, J. Christian, & Thornton, Sarah M. 2015. Implementable Ethics for Autonomous Vehicles. *Pages 87–102 of :* Maurer, Markus, Gerdes, J. Christian, Lenz, Barbara, & Winner, Hermann (eds), *Autonomes Fahren : Technische, rechtliche und gesellschaftliche Aspekte*. Berlin : Springer Vieweg. 10

Girard, J M, Wilczyk, M, Barloy, Y, Simon, P, & Popieul, J C. 2005. Towards an On-line Assessment of Subjective Driver Workload. *North America*, 10. 167

Goodwin, Graham, Seron, María M, & De Doná, José A. 2006. *Constrained control and estimation : an optimisation approach*. Springer Science & Business Media. 3, 100

Grant, Peter, & Naseri, Amir. 2005. Actuator State Based Adaptive Motion Drive Algorithm. *North America*, 10. 59, 60

Greenberg, Jeff, Artz, Bruce, & Cathey, Larry. 2003. The effect of lateral motion cues during simulated driving. *Pages 8–10 of : Proceedings of Driving Simulation Conference, North America*. 125

Grottoli, Marco, Cleij, Diane, Pretto, Paolo, Lemmens, Yves, Happee, Riender, & Bülthoff, Heinrich H. 2019. Objective evaluation of prediction strategies for optimization-based motion cueing. *SIMULATION*, **95**(8), 707–724. 165

Gulian, E., Matthews, G., & Debney, L. M. 1989. Dimensions of driver stress. *Ergonomics*, 585–602. 123

Gutman, P.-O., & Cwikel, M. 1987. An algorithm to find maximal state constraint sets for discrete-time linear dynamical systems with bounded controls and states. *IEEE Transactions on Automatic Control*, **32**(3), 251–254. 92

Gutridge, C Jason. 2004. *Three degree-of-freedom simulator motion cueing using classical washout filters and acceleration feedback*. Ph.D. thesis, Citeseer. 53

Guého, Ludivine. 2014. French validation of a new version of the Driver Behavior Questionnaire (DBQ) for drivers of all ages and level of experiences. *Accident Analysis and Prevention*, **63**, 41–48. 124, 134, 148, 156

Hagan, Martin T, Demuth, Howard B, & Jesús, Orlando De. 2002. An introduction to the use of neural networks in control systems. *International Journal of Robust and Nonlinear Control*, **12**(11), 959–985. 78

Hansson, Patrik, & Stenbeck, Anders. 2014. Prepositioning of a Driving Simulator Motion System. 69. 165

Haycock, Bruce, & Grant, Peter R. 2007. The Influence of Jerk on Perceived Simulator Motion Strength. 125, 136, 175

Helman, S, & Reed, N. 2015. Validation of the driver behaviour questionnaire using behavioural data from an instrumented vehicle and high-fidelity driving simulator. *Accident Analysis and Prevention*, **75**, 245–251. 124, 136, 157

Hirsch, Pierro, & Quimper, Rémi. 2015. *Recommended Requirements for a Driving Simulator Visual System*. 15

Holman, Andrei C., & Havarneanu, Corneliu E. 2015. The Romanian version of the multidimensional driving style inventory : Psychometric properties and cultural specificities. *Transportation Research Part F : Traffic Psychology and Behaviour*, **35**(Nov.), 45–59. 134, 146, 156

Hooft van Huysduynen, Hanneke, Terken, Jacques, & Eggen, Berry. 2018. The relation between self-reported driving style and driving behaviour. A simulator study. *Transportation Research Part F : Traffic Psychology and Behaviour*, **56**(July), 245–255. 124, 125, 136, 156, 157, 175

Hopfield, John J, & Tank, David W. 1985. "Neural" computation of decisions in optimization problems. *Biological cybernetics*, **52**(3), 141–152. 102, 104

Hosman, Ruud, Cardullo, Frank, & Bos, Jelte. 2011. Visual-Vestibular interaction in motion perception. American Institute of Aeronautics and Astronautics. 47

Hu, Xiaolin, & Wang, Jun. 2008. An Improved Dual Neural Network for Solving a Class of Quadratic Programming Problems and Its -Winners-Take-All Application. *IEEE Transactions on Neural Networks*, **19**(12), 2022–2031. 104

I-Fang Chung, Hung-Hsiang Chang, & Chin-Teng Lin. 1999. Fuzzy control of a six-degree motion platform with stability analysis. vol. 1. IEEE. 77

Ioannou, Petros A, & Sun, Jing. 2012. *Robust adaptive control*. Courier Corporation. 59, 60

Ish–Shalom, Jehuda. 1982. *The design of optimal control motion for flight simulators*. PhD thesis, Massachusetts Institute of Technology. 19, 62, 63

Jacqué, Phillipe. 2016. *Premier feu vert réglementaire pour les véhicules autonomes*. 10

Jamson, Andrew Hamish John. 2010 (Nov.). *motion cueing in driving simulators for research applications*. thesis. The university of leeds. 65

Jerez, Juan L., Constantinides, George A., Kerrigan, Eric C., & Ling, Keck-Voon. 2011. Parallel MPC for Real-Time FPGA-based Implementation. *IFAC Proceedings Volumes*, **44**(1), 1338–1343. 118

Kalra, Nidhi, & Paddock, Susan M. 2016. *Driving to Safety : How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability ?* Tech. rept. RR-1478-RC. RAND Corporation. 17

Kandel, Eric R, Schwartz, James H, Jessell, Thomas M, Siegelbaum, Steven A, & Hudspeth, A James. 2000. *Principles of neural science*. Vol. 4. McGraw-hill New York. 34, 35, 45, 203

Kaye, Sherrie-Anne, Lewis, Ioni, & Freeman, James. 2018. Comparison of self-report and objective measures of driving behavior and road safety : A systematic review. *Journal of Safety Research*, **65**(June), 141–151. 124, 158

Kemeny, Andras. 1999. Simulation and perception. *In : Introduction de la Conférence sur la Simulation de Conduite*. 28, 45, 47

Kemeny, Andras. 2014. From driving simulation to virtual reality. *Pages 1–5 of : Proceedings of the 2014 Virtual Reality International Conference on - VRIC '14*. Laval, France : ACM Press. 14

Kemeny, Andras, & Panerai, Francesco. 2003. Evaluating perception in driving simulation experiments. *Trends in Cognitive Sciences*, **7**(1), 31–37. 15

Kennedy, Michael Peter, & Chua, Leon O. 1988. Neural networks for nonlinear programming. *IEEE Transactions on Circuits and Systems*, **35**(5), 554–562. 103

Khan, Sarah, & Chang, Richard. 2013. Anatomy of the vestibular system : a review. *NeuroRehabilitation*, **32**(3), 437–443. 33, 34, 35

Kim, Juno, Chung, Charles Y. L., Nakamura, Shinji, Palmisano, Stephen, & Khuu, Sieu K. 2015. The Oculus Rift : a cost-effective tool for studying visual-vestibular interactions in self-motion perception. *Frontiers in Psychology*, **6**(Mar.). 30

Klüver, Malte, Herrigel, Carolin, Preuß, Stephanie, Schöner, Hans-Peter, & Hecht, Heiko. 2015. Comparing the Incidence of Simulator Sickness in Five Different Driving Simulators. 8. 22

Kolb, Helga. 2003. How the retina works. *American scientist*, **91**(1), 28–35. 28

Kruk, Sabina. 2017. *Driving Style Recognition : Literature Review and Application of Machine Learning*. Tech. rept. 157, 175

Kurosaki, Masahiro. 1978. Optimal Washout for Control of a Moving Base Simulator. *IFAC Proceedings Volumes*, **11**(1), 1311–1318. 62, 64

Kwon, W., & Pearson, A. 1977. A modified quadratic cost problem and feedback stabilization of a linear system. *IEEE Transactions on Automatic Control*, **22**(5), 838–842. 93

Lajunen, Timo, & Summala, Heikki. 1995. Driving experience, personality, and skill and safety-motive dimensions in drivers' self-assessments. *Personality and Individual Differences*, **19**(3), 307–318. 123

Lajunen, Timo, & Summala, Heikki. 2003. Can we trust self-reports of driving? Effects of impression management on driver behaviour questionnaire responses. *Transportation Research Part F : Traffic Psychology and Behaviour*, **6**(2), 97–107. 158

Lajunen, Timo, Parker, Dianne, & Summala, Heikki. 2004. The Manchester Driver Behaviour Questionnaire : a cross-cultural study. *Accident Analysis & Prevention*, **36**(2), 231–238. 124, 134, 148

Lamprecht, Alexander, Steffen, Dennis, Haecker, Jens, & Graichen, Knut. 2019. Comparison between a Filter and an MPC-based MCA in a offline SImulator Study. *Pages 101–107 of : Proceedings of Driving Simulation Conference*. 3, 37, 122

Laurens, Jean, & Droulez, Jacques. 2008. Bayesian Modelling of Visuo-Vestibular Interactions. *Pages 279–300 of :* Bessière, Pierre, Laugier, Christian, & Siegwart, Roland (eds), *Probabilistic Reasoning and Decision Making in Sensory-Motor Systems*, vol. 46. Berlin, Heidelberg : Springer Berlin Heidelberg. 30

LaValle, Steven M. 2016. *The physiology of human vision*. 28

Ledesma, Rubén Daniel, & Valero-Mora, Pedro. 2007. Determining the Number of Factors to Retain in EFA : An easy-to-use computer program for carrying out Parallel Analysis,". *Practical Assessment, Research, and Evaluation*, **12**. 146

Lillo, Walter E, Loh, Mei Heng, Hui, Stefen, & Zak, Stanislaw H. 1993. On solving constrained optimization problems with neural networks : A penalty method approach. *IEEE Transactions on neural networks*, **4**(6), 931–940. 103, 106

Liu, Qingshan, & Cao, Jinde. 2011. Global exponential stability of discrete-time recurrent neural network for solving quadratic programming problems subject to linear constraints. *Neurocomputing*, **74**(17), 3494–3501. 104

Liu, Qingshan, & Wang, Jun. 2015. A Projection Neural Network for Constrained Quadratic Minimax Optimization. *IEEE Transactions on Neural Networks and Learning Systems*, **26**(11), 2891–2900. 104

Liu, Shubao, & Wang, Jun. 2006. A Simplified Dual Neural Network for Quadratic Programming With Its KWTA Application. *IEEE Transactions on Neural Networks*, **17**(6), 1500–1510. 102, 103, 104, 114

Luenberger, David G, Ye, Yinyu, & others. 1984. *Linear and nonlinear programming*. Vol. 2. Springer. 105, 106, 113

Maa, C-Y, & Shanblatt, Michael A. 1992. Linear and quadratic programming neural network analysis. *IEEE transactions on neural networks*, **3**(4), 580–594. 104, 106

Maran, Fabio, Bruschetta, Mattia, Beghi, Alessandro, & Minen, Diego. 2015. Improvement of an MPC-based Motion Cueing Algorithm with Time-Varying Prediction and Driver Behaviour Estimation. 90

Maran, Fabio, Bruschetta, Mattia, & Beghi, Alessandro. 2016. Study of a real-time, MPC based motion cueing procedure with time-varying prediction for different classes of drivers. IEEE. 84, 93

Marruedo, Daniel Limón. 2002. *Control predictivo de sistemas no lineales con restricciones : estabilidad y robustez*. Ph.D. thesis, Universidad de Sevilla. 66, 70

Martinussen, Laila M. 2013. Age, gender, mileage and the DBQ : The validity of the Driver Behavior Questionnaire in different driver groups. *Accident Analysis and Prevention*, **52**, 228–236. 124, 134, 148, 156

Mas, Arnaud. 2012 (July). *utilisation des simulateurs de conduite pour l'evaluation des systèmes d'aide à la conduite en situation d'urgence*. Ph.D. thesis, Arts et Metiers, Cluny. 7, 13, 16

Matich, Damián Jorge. 2001. Redes Neuronales : Conceptos básicos y aplicaciones. *Cátedra de Informática Aplicada a la Ingeniería de Procesos–Orientación I*. 78

Mattsson, Markus. 2012. Investigating the factorial invariance of the 28-item DBQ across genders and age groups : An Exploratory Structural Equation Modeling Study. *Accident Analysis & Prevention*, **48**(Sept.), 379–396. 134

McInerney, Ian, Constantinides, George A., & Kerrigan, Eric C. 2018. A Survey of the Implementation of Linear Model Predictive Control on FPGAs. *IFAC-PapersOnLine*, **51**(20), 381–387. 118

Medsker, LR, & Jain, LC. 2001. Recurrent neural networks. *Design and Applications*, **5**. 104

Mehrotra, K., & Mahapatra, P.R. 1997. A jerk model for tracking highly maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, **33**(4), 1094–1105. 175

Miunske, Tobias, Holzapfel, Christian, Baumgartner, Edwin, & Reuss, Hans-Christian. 2019. A new Approach for an Adaptive Linear Quadratic Regulated Motion Cueing Algorithm for an 8 DoF Full Motion Driving Simulator. *Pages 497–503 of : 2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada : IEEE. 79

Mladenov, Valeri M, & Maratos, N. 2000. Neural networks for solving constrained optimization problems. *Proc. of CSCC'00, Athens, Greece,(N. Mastorakis*. 103, 104

Mohammadi, Arash, Asadi, Houshyar, Mohamed, Shady, Nelson, Kyle, & Nahavandi, Saeid. 2016. MPC-based motion cueing algorithm with short prediction horizon using exponential weighting. *Pages 000521–000526 of : International Conference on Systems, Man and Cybernetics (SMC)*. IEEE. 3, 37, 85, 91, 122

Mohammadi, Arash, Asadi, Houshyar, Mohamed, Shady, Nelson, Kyle, & Nahavandi, Saeid. 2018. Optimizing Model Predictive Control horizons using Genetic Algorithm for Motion Cueing Algorithm. *Expert Systems with Applications*, **92**(Feb.), 73–81. 85, 91, 165

Mohellebi, Hakim. 2005. *Conception et réalisation de systèmes de restitution de mouvement et de retour haptique pour un simulateur de conduite à faible coût dédie à l'étude comportementale du conducteur*. Ph.D. thesis, D'évry-val d'essonne. 18, 30, 45, 46

Momani, Ahmad Q., & Cardullo, Frank M. 2018. A Review of the Recent Literature on the Mathematical Modeling of the Vestibular System. *In : AIAA Modeling and Simulation Technologies Conference*. Kissimmee, Florida : American Institute of Aeronautics and Astronautics. 37, 122

Murgovski, Nikolce. 2007. Vehicle Modelling and Washout Filter Tuning for the Chalmers Vehicle Simulator. 27

Murphey, Yi Lu, Milton, Robert, & Kiliaris, Leonidas. 2009. Driver's style classification using jerk analysis. *Pages 23–28 of : 2009 IEEE Workshop on Computational Intelligence in Vehicles and Vehicular Systems*. Nashville, TN, USA : IEEE. 157, 175

Nahon, MA, Reid, LD, & Kirdeikis, J. 1992. Adaptive simulator motion software with supervisory control. *Journal of Guidance, Control, and Dynamics*, **15**(2), 376–383. 53, 57, 58, 59, 80

Nahon, Meyer A., & Reid, Llotd D. 1990. simulation motion-drive algorithmes- a designer's perspective. 52

Nehaoua, L., Arioui, H., Espie, S., & Mohellebi, H. 2006. Motion cueing algorithms for small driving simulator. IEEE. 2, 51, 57, 58, 80

Nehaoua, Lamri. 2008 (Nov.). *Conception et réalisation d'une plateforme mécatronique dédiée à la simulation de conduite des véhicules deux-roues motorisés*. Ph.D. thesis, Université d'evry-val d'essonne. 55, 60

Nieuwenhuizen, Frank M., & Bulthoff, Heinrich H. 2013. The MPI CyberMotion Simulator : A Novel Research Platform to Investigate Human Control Behavior. *Journal of Computing Science and Engineering*, **7**(2), 122–131. 18

Nobuyuki, Kuge, Masanori, Kubota, & Kazuya, Itoh. 2002. A study on a motion algorithm based on a driver-centered approach. 22

Nordmark, Staffan, Hakan, Jansson, Goran, Palmkvist, & Hakan, SEhemmar. 2004. The new VTI driving simulator. Multi purpose moving base with high performance linear motion. *Pages 45–55 of : Driving Simulation Conference*. 15

Nouvelle France Industrielle, Comité de pilotage. 2015. *Plan NFI - Objectifs de recherche Nouvelle France Industrielle : Véhicule Autonome*. 9

Osborne, Jason W, Costello, Anna B., & Kellow, Thomas. 2008. Best practices in exploratory factor analysis. *Pages 86–99 of : Best practices in quantitative methods*. Thousand Oaks, CA : SAGE Publications. OCLC : 992457770. 146

Parrish, Russell V, Dieudonne, James E, & Martin Jr, Dennis J. 1975. Coordinated adaptive washout for motion simulators. *Journal of aircraft*, **12**(1), 44–50. 58, 59

Passino, Kevin M, Yurkovich, Stephen, & Reinfrank, Michael. 1998. *Fuzzy control*. Vol. 2725. Citeseer. 77, 204

Perez-Ilzarbe, M. J. 2013. New Discrete-Time Recurrent Neural Network Proposal for Quadratic Optimization With General Linear Constraints. *IEEE Transactions on Neural Networks and Learning Systems*, **24**(2), 322–328. 104, 114

Poó, Fernando M., Taubman-Ben-Ari, Orit, Ledesma, Rubén D., & Díaz-Lázaro, Carlos M. 2013. Reliability and validity of a Spanish-language version of the multidimensional driving style inventory. *Transportation Research Part F : Traffic Psychology and Behaviour*, **17**(Feb.), 75–87. 134, 156

Priot, Anne-Emmanuelle. 2012. *Perception du mouvemeent propre : contributiond des fifférents modalités sensorielles et apllications en ergonomie cognitive*. Ph.D. thesis, Faculte miste de medecine et de pharmacie de rouen. 31, 46

Puell Marin, Maria Cinta. 2006. *Optica fisiologica : El sistema optico del ojo y la vision binocular*. 28

Pérez-Ilzarbe, María José. 2014. Improvement of the convergence speed of a discrete-time recurrent neural network for quadratic optimization with general linear constraints. *Neurocomputing*, **144**(Nov.), 493–500. 166

Raven, Peter H. 2011. *Biologie*. Bruxelles : De Boeck. OCLC : 812905272. 34, 203

Reason, James, Manstead, Antony, Stradling, Stephen, Baxter, James, & Campbell, Karen. 1990. Errors and Violations on the Roads : a Real Distinction ? *Ergonomics*, **33**(10-11), 1315–1332. 123, 124, 134, 148, 156

Regalado Crisanto, Julio César. 1999. *Lógica difusa aplicada al control de voltaje y potencia reactiva en subestaciones de distribución*. Ph.D. thesis, Universidad De Piura, Lima, Peru. 76

Regalón, Orlando, Diez Rodriguez, Milagros, Rodriguez Diez, Vladimir, & Raul, Baez. 2012. *Aplicación de algoritmos de control clásico, adaptable y robusto a sistemas dinámicos de parámetros variables*. 57

Reid, LD, & Nahon, Meyer A. 1985. *Flight simulation motion-base drive algorithms : Part 1. Developing and testing equations*. Tech. rept. University of Toronto. 3, 33, 37, 39, 54, 122

Reimer, Bryan, D'Ambrosio, Lisa A., Gilbert, Jennifer, Coughlin, Joseph F., Biederman, Joseph, Surman, Craig, Fried, Ronna, & Aleardi, Megan. 2005. Behavior differences in drivers with attention deficit hyperactivity disorder : The driving behavior questionnaire. *Accident Analysis & Prevention*, **37**(6), 996–1004. 134

Reschka, Andreas. 2016. Safety Concept for Autonomous Vehicles. *Pages 473–496 of :* Maurer, Markus, Gerdes, J. Christian, Lenz, Barbara, & Winner, Hermann (eds), *Autonomous Driving*. Berlin, Heidelberg : Springer Berlin Heidelberg. 10

Reymond, Gilles. 2000. *Contribution respective des stimuli visuels, vestibulaires et proprioceptifs dans la perception du mouvement du conducteur*. Ph.D. thesis, Université Paris VI. 30, 39, 47

Rodriguez-Vazquez, A, Rueda, A, Huertas, JL, & Dominguez-Castro, R. 1988. Switched-capacitor neural networks for linear programming. *Electronics letters*, **24**(8), 496–498. 103, 106

Romano, Richard. 2003. Non-linear optimal tilt coordination for washout algorithms. *In : AIAA Modeling and Simulation Technologies Conference and Exhibit*. AIAAA. 56

Romano, Richard, Sadraei, Ehsan, & Markkula, Gustav. 2017. Rapid tuning of the classical motion cueing algorithm. *Pages 89–92 of : Driving Simulation Conference*. 21, 58

Rossiter, J. A. 2003. *Model-based predictive control : a practical approach*. Control series. Boca Raton : CRC Press. 70, 89

Rubio, Francisco Rodríguez, & Sánchez, Manuel Jesús López. 1996. *Control adaptativo y robusto*. Vol. 9. Universidad de Sevilla. 59

Sadraei, E., Romano, R., Advani, S., Jamson, A.H., Chappell, P., Markkula, G., Bean, A., & Boer, E.R. 2016. Understanding Cue Utility in Controlled Evasive Driving Manoeuvres : Optimizing Vestibular Cues for Simulator & Human Abilities. *IFAC-PapersOnLine*, **49**(19), 414–419. 58

Sammut, Claude, & Webb, Geoffrey I. (eds). 2010. *Encyclopedia of machine learning*. New York ; London : Springer. OCLC : ocn651073009. 101

Sareni, Bruno. 1999. *Méthodes d'optimisation multimodales associées à la modélisation numérique en électromagnétisme*. Ph.D. thesis, Ecole centrale de Lyon. 92

Schreurs, Miranda A., & Steuwer, Sibyl D. 2016. Autonomous Driving—Political, Legal, Social, and Sustainability Dimensions. *Pages 149–171 of :* Maurer, Markus, Gerdes, J. Christian, Lenz, Barbara, & Winner, Hermann (eds), *Autonomous Driving*. Berlin, Heidelberg : Springer Berlin Heidelberg. 7

Schwarz, Chris, Gates, Tad, & Papelis, Yiannis. 2003. Motion characteristics of the national advanced driving simulator. *In : Proceedings of the Driving Simulation Conference*. 21, 58

Scotto Di Cesare, C., Buloup, F., Mestre, D.R., & Bringoux, L. 2014. How do visual and postural cues combine for self-tilt perception during slow pitch rotations? *Acta Psychologica*, **153**(Nov.), 51–59. 44

Seehof, Carsten, Durak, Umut, & Duda, Holger. 2014. Objective Motion Cueing Test - Experiences of a New User. *In : AIAA Modeling and Simulation Technologies Conference*. Atlanta, GA : American Institute of Aeronautics and Astronautics. 22, 58

Sender, Elena. 2016. *Voiture autonome : piéton ou conducteur, qui devra-t-elle sauver ?* 9

Shuttleworth, Jennifer. 2019. *SAE Standards News : J3016 automated-driving graphic update*. 11

Sivan, Raphael, Ish-shalom, Jehuda, & Huang, Jen-Kuang. 1982. An optimal control approch to the design of moving flight simulators. 62

Slob, JJ. 2008. State-of-the-Art driving simulators, a literature survey. *DCT report*, **107**. 22, 51

Smorti, Martina, & Guarnieri, Silvia. 2016. Exploring the factor structure and psychometric properties of the Manchester Driver Behaior Questionnaire (DBQ) in an Italian sample. *TPM - Testing, Psychometrics, Methodology in Applied Psychology*, 185–202. 124

Stahl, Konrad, Leimbach, Klaus-Dieter, Meroth, Ansgar, & Zöllner, Raoul Daniel. 2015. A washout and a tilt coordination algorith for a hexapod plateform. 53

Stanojević, Predrag, Lajunen, Timo, Jovanović, Dragan, Sârbescu, Paul, & Kostadinov, Svilen. 2018. The driver behaviour questionnaire in South-East Europe countries : Bulgaria, Romania and Serbia. *Transportation Research Part F : Traffic Psychology and Behaviour*, **53**(Feb.), 24–33. 124

Stephens, A.N., & Fitzharris, M. 2016. Validation of the Driver Behaviour Questionnaire in a representative sample of drivers in Australia. *Accident Analysis & Prevention*, **86**(Jan.), 186–198. 124

Stewart, D. 1965. A Platform with Six Degrees of Freedom. *Proceedings of the Institution of Mechanical Engineers*, **180**(1), 371–386. 19

Strachan, Ian W. 2014. Visual and motion cues in the real word and in simulators. 33

Stratulat, Anca, Roussarie, Vincent, Vercher, Jean-Louis, & Bourdin, Christophe. 2011. Improving the realism in motion-based driving simulators by adapting tilt-translation technique to human perception. *Pages 47–50 of : 2011 IEEE Virtual Reality Conference*. Singapore, Singapore : IEEE. 22, 42, 58

Stroosma,Olaf, Mulder, Max, Van Paassen, & Advani, Sunjoo. 2013. Applying to objetive motion cueing test to a classical washout algorithm. Aug. 55, 165

Suikat, Reiner. 2005. The new dynamic driving simulator at DLR. *Pages 374–381 of : Driving Simulator Conference, Orlando, Florida*. 22

Sullivan, Robert Bryan. 1985. *The use of vestibular models in flight simulator motion washout systems : an experimental evaluation*. Ph.D. thesis, Massachusetts Institute of Technology. 42, 46, 135, 207

Tang, Zhiyong, Hu, Ma, Pei, Zhongcai, & Zhang, Jinhui. 2016. Adaptive motion cueing algorithm based on fuzzy tuning for improving human sensation. Aug. 77

Tank, Df, & Hopfield, JJ. 1986. Simple'neural'optimization networks : An A/D converter, signal decision circuit, and a linear programming circuit. *IEEE transactions on Circuits and Systems*, **33**(5), 533–541. 102, 103

Taubman, Orit, Eherenfreund, Ahinoam, & Prato, Carlo Giacomo. 2016. The value of self-report measures as indicators of driving behaviors among young drivers. 10. 124, 136, 158

Taubman Ben-Ari, Orit, & Skvirsky, Vera. 2016. The multidimensional driving style inventory a decade later : Review of the literature and re-evaluation of the scale. *Accident Analysis & Prevention*, **93**(Aug.), 179–188. 124

Taubman-Ben-Ari, Orit, Mikulincer, Mario, & Gillath, Omri. 2004. The multidimensional driving style inventory—scale construct and validation. *Accident Analysis & Prevention*, **36**(3), 323–332. 123, 134, 145

Telban, Robert, Cardullo, Frank, & Houck, Jacob. 1999. Developments in human centered cueing algorithms for control of flight simulator motion systems. American Institute of Aeronautics and Astronautics. 46

Telban, Robert J. 2002. A nonlinear motion cueing algorithm with a human perception model. *Energy, Simulation-training, Ocean Engineering and Instrumentation : Research Papers of the Link Foundation Fellows*, **2**, 97–127. 79

Telban, Robert J, & Cardullo, Frank M. 2005. *Motion cueing algorithm development : Human-centered linear and nonlinear approaches*. Technical Report NASA/CR-2005-213747. 3, 30, 31, 37, 39, 41, 44, 47, 48, 59, 60, 65, 122, 123, 144, 153, 159, 203, 204

Telban, Robert J, Wu, Weimin, Cardullo, Frank M, & Houck, Jacob A. 2000. Motion cueing algorithm development : Initial investigation and redesign of the algorithms. 40, 41, 57

Van der Steen, F. A. M., & Brockhoff, P. T. M. 2000. Induction and impairment of saturated yaw and surge vection. *Perception & Psychophysics*, **62**(1), 89–99. 31

Van der Steen, Franciscus Albertus Martinus. 1998. *Self-motion perception*. Ph.D. thesis, University of Delft. 31

van Leeuwen, T.D., Cleij, D., Pool, D.M., Mulder, M., & Bülthoff, H.H. 2018. Time-varying perceived motion mismatch due to motion scaling in curve driving simulation. *Transportation Research Part F : Traffic Psychology and Behaviour*, June. 167

van Wees, K.A.P.C. 2004. Vehicle safety regulations and ADAS : tensions between law and technology. IEEE. 7

Venrooij, J, Cleij, D, Katliar, M, Pretto, P, Bülthoff, HH, Steffen, D, Hoffmeyer, FW, & Schöner, HP. 2016. Comparison between filter-and optimization-based motion cueing in the Daimler Driving Simulator. 2

Wachenfeld, Walther, & Winner, Hermann. 2016. The Release of Autonomous Vehicles. *Pages 425–449 of :* Maurer, Markus, Gerdes, J. Christian, Lenz, Barbara, & Winner, Hermann (eds), *Autonomous Driving*. Berlin, Heidelberg : Springer Berlin Heidelberg. 9

Wang, Jun. 1997. Primal and dual assignment networks. *IEEE Transactions on Neural Networks*, **8**(3), 784–790. 103, 114

Wang, Jun, & Xia, Youshen. 1999. A dual neural network solving quadratic programming problems. *Pages 588–593 of : IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339)*, vol. 1. Washington, DC, USA : IEEE. 103, 114

Wang, Su-Chiun, & Fu, Li-Chen. 2004. Predictive washout filter design for vr-based motion simulator. *Pages 6291–6295 of : Systems, Man and Cybernetics, 2004 IEEE International Conference on*, vol. 7. IEEE. 39, 79

Wang, Yang, & Boyd, Stephen. 2008. Fast Model Predictive Control Using Online Optimization. *IFAC Proceedings Volumes*, **41**(2), 6974–6979. 73

Wautier, Didier. 2018. *Autonomous Vehicle's Digital Simulation Challenges*. 1, 14

Weib, Cornelius. 2006 (Nov.). *Control of a Dynaamic Driving Simulator : Time-Variant motion cueing algorithms and prepositioning*. Ph.D. thesis, DLR, Braunschweig. 44

Wilkinson, Mark. 2012. The National Advanced Driving Simulator (NADS) Description and Capabilities in Vision Related Research. 5. 21

Wolfe, Philip. 1961. A duality theorem for non-linear programming. *Quarterlyof applied mathematics*, **19**(3), 239–244. 113

Woon-Sung Lee, Jung-Ha Kim, & Jun-Hee Cho. 1998. A driving simulator as a virtual reality tool. vol. 1. IEEE. 16

Wynne, Rachael A., Beanland, Vanessa, & Salmon, Paul M. 2019. Systematic review of driving simulator validation studies. *Safety Science*, **117**(Aug.), 138–151. 1

Xia, Youshen. 1996. A new neural network for solving linear and quadratic programming problems. *IEEE transactions on neural networks*, **7**(6), 1544–1548. 103, 104

Xia, Youshen, & Feng, Gang. 2005. An improved neural network for convex quadratic optimization with application to real-time beamforming. *Neurocomputing*, **64**(Mar.), 359–374. 104

Xia, Youshen, & Wang, Jun. 1998. A general methodology for designing globally convergent optimization neural networks. *IEEE Transactions on Neural Networks*, **9**(6), 1331–1343. 105

Xia, Y.S., Feng, G., & Wang, J. 2005. A Primal-Dual Neural Network for Online Resolving Constrained Kinematic Redundancy in Robot Motion Control. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, **35**(1), 54–64. 104

Young, Laurence R. 1978. Visually Induced Motion in Flight Simulation. *AGARD Symposium on Flight Simulation.* 31

Young, Laurence R., & Meiry, Jacob L. 1968. A revised dynamic otolith model. *Aerospace Medicine*, **39 (6)**, 606–608. 36, 37, 39, 122, 123, 153, 159, 203

Young, Laurence R., & Oman, Charles M. 1969. Model for vestibular adaptation to horizontal rotation. *Aerospace Medicine*, **40**, 1076–1080. 36, 41, 122, 123, 153, 159

Zaal, P.M.T., Schroeder, J.A., & Chung, W.W. 2017. Objective motion cueing criteria investigation based on three flight tasks. *The Aeronautical Journal*, **121**(1236), 163–190. 165

Zacharias, G.L., & Young, L.R. 1981. Influence of combined visual and vestibular cues on human perception and control of horizontal rotation. *Experimental Brain Research*, **41**(2). 48

Zacharias, Greg L. 1978. *Motion cue models for pilot-vehicle analysis*. Technical report ADA061477. DTIC Document. 3, 37, 40, 122

Zaychik, Kirill, Cardullo, Frank, George, Gary, & Kelly, Lon. 2009. Evaluating Effectiveness of Modeling Motion System Feedback in the Enhanced Hess Structural Model of the Human Operator. *In : AIAA Conference of Modeling and Simulation Technologies*. Chicago, Illinois : American Institute of Aeronautics and Astronautics. 3, 37, 122

Zhang, Shengwei, & Constantinides, AG. 1992. Lagrange programming neural networks. *IEEE Transactions on Circuits and Systems II : Analog and Digital Signal Processing*, **39**(7), 441–452. 103, 104

Zhang, Xiang-Sun. 2013. *Neural networks in optimization*. Vol. 46. Springer Science & Business Media. 117

Zhang, Yunong, & Wang, Jun. 2002. A dual neural network for convex quadratic programming subject to linear equality and inequality constraints. *Physics Letters A*, **298**(4), 271–278. 113, 114

Zhao, Nan. 2012. An investigation of the relationship between the driving behavior questionnaire and objective measures of highway driving behavior. *Transportation research part F : traffic psychology and behaviour*, **15**(6), 676–685. 124, 134, 157, 158

Zywiol Jr, Harry J, & Romano, Richard. 2003 (Oct.). *Motion drive algorithms and simulator design to study motion effects on infantry soldiers*. Tech. rept. DTIC Document. 57

Table des figures

1 Organisation de la thèse. 5

2 Niveaux d'automatisme de la SAE. 11

3 Chaîne de simulation AD / ADAS de l'alliance Renault-Nissan. . . . 15

4 Simulateurs à structure série. 19

5 Simulateurs à structure parallèle. 20

6 Simulateurs dynamiques à hautes performances. 23

7 Physiologie de l'œil humain et ses différentes structures. 29

8 Schéma fonctionnel du modèle de perception visuelle. 31

9 Anatomie du système vestibulaire. Image tirée de [Kandel *et al.*, 2000]. 34

10 Agrandissement d'une coupe du saccule et de l'utricule. Image tirée de [Raven, 2011]. 34

11 L'ampoule d'un canal semi-circulaire en présence d'une accélération rotationnelle. Image tirée de [Kandel *et al.*, 2000]. 35

12 Réponse temporelle des organes otolithiques. 38

13 Réponses en fréquence du modèle Mo1, proposé par [Young & Meiry, 1968] et du modèle Mo2, proposé par [Telban & Cardullo, 2005]. . . . 39

14 Réponses en fréquence du modèle des canaux semi-circulaires Mc1 et Mc2. 42

15 Modèle du système vestibulaire soumis à un mouvement linéaire et rotationnel. 43

16 Technique de tilt-coordination : accélération selon l'axe x. 44

17 Modèle d'interaction vestibulaire et visuelle proposé par [Telban & Cardullo, 2005]. 47

18 Différence entre la perception du mouvement par le système vestibulaire, visuelle et l'ensemble de ces derniers avec un stimulus constant en vitesse de $4°/s$. 48

19 Intégration des stratégies de mouvement dans le contrôle de simulateurs dynamiques. 52

20 Structure en boucle ouverte du contrôle classique. 53

21 Accélération (à gauche) et déplacement (à droite) du simulateur en utilisant le filtre $F1_a$ en bleu et le filtre $F1_b$ en rouge. 55

22 Restitution de mouvement produit par l'hexapode grâce à la technique du *tilt coordination*. 56

23 Accélérations hautes fréquences, basses fréquences et force spécifique en utilisant l'algorithme classique selon l'axe de mouvement x. . . . 56

24 Contrôle adaptatif. 59

25 Comparaison entre la réponse des filtres avec les algorithmes classique et adaptatif. 61

26 Contrôle optimal. 62

27 Comparaison entre la restitution de mouvement de la courbe acc_{veh} par la stratégie classique (classi) et la stratégie optimale (opti). . . . 64

28 Structure du contrôle prédictif. 67

29 Optimisation par degré de liberté. 68

30 Optimisation de la force spécifique avec deux degrés de liberté. . . . 68

31 Principe de fonctionnement du MPC en horizon glissant. 72

32 Comparaison des accélérations linéaires selon l'axe x entre l'optimisation à 1 DDL et à 2 DDL. 74

33 Composantes basses (BF) et hautes (HF) fréquences de l'accélération, en utilisant le MPC afin de restituer la force spécifique selon x. . . . 74

34 Architecture du contrôle flou [Passino *et al.*, 1998]. 77

35 Mise en oeuvre de réseaux neuronaux en tant que méthode d'identification dans la commande MPC. 79

36 Terrain et trajectoire utilisés pour l'essai de simulation de conduite autonome. 81

37 Restitution de la force spécifique pour chacune des stratégies de contrôle. 82

38 Déplacement de la plateforme selon l'axe x. 82

39 Temps d'échantillonnage non uniforme. 90

40 Réponses en boucle fermée avec différentes fréquences dans l'horizon de prédiction non uniforme (les valeurs correspondent à $t_{s2} * N_2$ dans le tableau 8). 91

41 Comparaison des sorties du simulateur avec les différentes stratégies dans les trois scénarios. 97

42 Comparaison des mouvements : position, vitesse et accélération dans le scénario 2. 99

43 Structure d'un neurone artificiel. 101

44 Réseau de neurones récurrent, non linéaire, à temps continu. Tiré de [Demuth *et al.*, 2014]. 102

45 Commande prédictive avec l'algorithme d'optimisation basé sur les réseaux de neurones récurrents. 107

46 Pas de temps du logiciel de résolution pour un modèle rigide. 108

47 Schéma du réseau neuronal récurrent proposé. 109

48 Fonction dynamique de saturation $f(a_j)$. 110

49 Rendu de l'accélération longitudinale en utilisant uniquement les rails x du simulateur. 112

50 Schéma fonctionnel du réseau neuronal dual. 115

51 Déplacement, vitesse et accélération de la plateforme. 116

52 Force spécifique restituée pour la plateforme. 116

53 Conception de la commande pour la restitution des mouvements du simulateur de conduite. 125

54 Restitution du mouvement selon le modèle mathématique du MPC : l'image du haut représente l'accélération du modèle autonome selon l'axe x ; les figures du bas sont respectivement, de gauche à droite, le signal de poursuite sans modèle de perception M0, avec le modèle M1 et avec le modèle M2. 128

55 Position de la plateforme selon les axes x et y. 130

56 Scénario utilisé dans le test utilisateur. 132

57 Comparaison des accélérations selon l'axe longitudinal x entre le mode de conduite AD et le mode de conduite Libre, après avoir effectué le test. 133

58 Comparaison des accélérations latérales y entre le mode AD et le mode Libre. 133

59 Description générale du cadre d'analyse. 137

60 Résultats pour les caractéristiques MCA avec une différence statistiquement significative dans le mode de conduite libre. 140

61 Résultats pour les caractéristiques MCA avec une différence statistiquement significative dans le mode de conduite autonome (AD). . . . 141

62 Moyennes et écart types pour les caractéristiques MCA statistiquement significativement différentes entre les modes de conduite AD et Libre. 144

63 Évaluation subjective du questionnaire MCA en fonction de la stratégie préférée par les sujets. 145

64 Résultats des groupes P0 et P2 selon les types des facteurs des questionnaires MDSI et DBQ. 151

Liste des tableaux

1 Modes et niveaux de fonctionnement du véhicule autonome. 12

2 Limites physiques des différents simulateurs hybrides à hautes performances. 25

3 Paramètres des modèles du système vestibulaire. 37

4 Valeurs des seuils de perception pour les otolithes. 40

5 Paramètres du modèle de canaux semi-circulaires. 41

6 Paramètres utilisés dans l'algorithme classique 57

7 Avantages et inconvénients des différents stratégies : classique, adaptative, optimale et prédictive. 83

8 Fréquences d'échantillonnage différentes pour un même temps de prédiction. 90

9 Nombre d'infaisabilités par stratégie et par scénario. 96

10 Différentes approches d'optimisation par réseaux de neurones récurrents. 104

11 Paramètres de pondération des stratégies M0, M1 et M2. 127

12 Contraintes de la plateforme et de perception du mouvement utilisées dans l'expérimentation. 129

13 Échelle d'aide pour noter subjectivement de 1 à 7 les différentes caractéristiques du MCA. Tiré de [Sullivan, 1985]. 135

14 Exemple de comparaison selon le type de conduite pour la caractéristique « Retard » dans les longs virages. 139

15 Comparaison entre les modes de conduite autonome et libre à l'aide des tests de Mann-Whitney. 143

16 Modèle factoriel et poids des coefficients du MDSI modifié. 147

17 Modèle factoriel et poids des composants du DBQ modifié. 149

18 Corrélations entre DBQ et MDSI. 150

19 Moyenne, écart-type (E-T), minimum et maximum des indicateurs de mouvement. 152

20 Corrélations pour les groupes P0 et P2 entre les indicateurs de performance et les facteurs présentant une différence statistique significative dans le questionnaire du comportement de conduite : style agressif et infractions. 154

Carolina RENGIFO CADAVID

Contrôle plateforme pour la validation du véhicule autonome sur simulateur dynamique à hautes performances

Résumé : Les travaux présentés dans ce manuscrit s'inscrivent dans le contexte de la simulation de conduite et plus concrètement dans celui des simulateurs de conduite dynamique à hautes performances utilisés pour la validation des systèmes avancés et du véhicule autonome. Afin de répondre aux enjeux de performance et de perception du mouvement, nous présentons différentes approches d'amélioration des algorithmes de restitution de mouvement (MCA).
L'ensemble de nos études montre que la stratégie de contrôle prédictif est le meilleur choix pour contrôler les mouvements des nouveaux simulateurs à hautes performances. Cependant, dans ce MCA l'optimisation en temps réel et le modèle de perception doivent être garantis afin d'améliorer l'immersion du conducteur dans l'environnement virtuel. Nous avons donc comparé différentes techniques pour résoudre les problèmes d'optimisation sous contraintes et avons proposé une technique d'optimisation à partir de circuits intégrés, qui propose une solution intuitive et rapide au problème d'optimisation. Enfin, nous avons établi des recommandations de paramétrage des MCA en fonction du comportement de conduite auto-déclaré qui permet une meilleure perception du mouvement dans un simulateur de conduite, en conduite interactive et en mode autonome .

Mots clés : algorithmes de restitution du mouvement, commande prédictive, conduite autonome, simulateurs de conduite dynamiques, perception du mouvement humaine.

Abstract : The work presented in this manuscript takes part in the context of driving simulation and more specifically in the one of dynamic driving simulators used for the validation of advanced systems and the autonomous vehicle. In order to address the issues of performance and motion perception, we have presented different approaches to improve the Motion Cueing Algorithms (MCA).
All our studies show that the model predictive control (MPC) strategy is the best choice to MCA on high-performance driving simulators. Indeed, compared to other strategies, it allows to better take advantage of the workspace without endangering the simulator and/or the driver. However, in this MCA, the real-time optimization and the perception model must be guaranteed in order to improve the driver's immersion in the virtual environment. Therefore, we compared different techniques to solve constrained optimization problems. We proposed a based optimization technique, which provides an intuitive and fast solution to the MPC constrained optimization problem. Finally, we established recommendations for MCA parameterization according to the self-declared driving behavior allowing a better perception of motion in a driving simulator, in interactive driving and in autonomous mode.

Keywords : motion cueing algorithms, model predictive control, autonomous driving, dynamic driving simulators, human motion perception.