

Institut de Recherche en Systèmes Électroniques Embarqués - ESIGELEC

Projet A2RL : Test et développement en simulation d'un contrôleur pour une Super Formula Autonome

Tuteur Industriel : Nassim FOUADH
Stage Effectué du 2 Juin au 25 Juillet 2025


irseeem

ESIGELEC 
ÉCOLE D'INGÉNIEURS-ES GÉNÉRALISTES

CHERON MARIE
INGÉNIERIE DES SYSTÈMES EMBARQUÉS MOBILES AUTONOMES ET CONNECTÉS

Remerciements

Je tiens tout d'abord à remercier mon tuteur de stage, **Nassim FOUADH**, ingénieur de recherche à l'ESIGELEC, qui m'a accompagnée et soutenue tout au long de mon stage. Ses conseils avisés, ainsi que la confiance qu'il m'a accordée lors de mes missions, ont fortement contribué à la réussite de ce stage et à mon épanouissement sur ce projet.

Je tiens également à remercier particulièrement **Edgard PETIT**, ingénieur de recherche, qui fut mon tuteur lors de mon projet S8 sur le challenge A2RL, et qui a continué à m'accompagner durant ce stage.

Je souhaite aussi adresser mes remerciements à **Pascal FALLA**, professeur à l'ESIGELEC, qui m'a aidée à obtenir ce stage après avoir pris connaissance de ma passion pour l'automobile et de mon projet de mobilité en Angleterre dans une dominante liée à ce domaine.

Leur encadrement, leurs conseils, et leur aide précieuse tout au long de ce projet m'ont permis de m'épanouir pleinement et d'apprendre énormément, tant sur le plan technique que sur les aspects humains et personnels.

Je remercie aussi **l'ensemble des membres de l'équipe du Laboratoire de Navigation Autonome (LNA)**, qui m'ont encouragée et conseillée pendant la durée de mon stage. Ces personnes ont contribué, par leur disponibilité, leur accueil et leur expertise à rendre cette expérience enrichissante et motivante.

Enfin, je souhaiterais remercier **l'ESIGELEC**, qui m'a donné l'opportunité d'effectuer un stage sur un projet à la fois formateur et en parfaite adéquation avec mes objectifs professionnels.

TABLE DES FIGURES

Figure 0 : Logo A2RL	5
Figure 1 : Image du bâtiment de l'IRSEEM	6
Figure 2 : Organisation du Laboratoire de l'IRSEEM	7
Figure 3 : Logo de ALADIN	8
Figure 4 : Logo de l'équipe FR4IAV	8
Figure 5 : Résultat de la prédiction avec la méthode des Moindres Carrés, peu fiable car les scores R^2 sont faibles	10
Figure 6 : Résultat de la prédiction avec le modèle Random Forest, très fiable car les scores R^2 sont élevés	10
Figure 7 : Résultat de la prédiction avec le modèle XGBoost, moyennement fiable car risque d'overfitting	11
Figure 8 : Affichage des données du topic, avec les température ("tiretemp") égales à 0.0 pour les quatres pneus	12
Figure 9 : Affichage des températures sur le simulateur Autonomia au même moment, impossible à récupérer sur les topics	12
Figure 10 : Notre pseudo code sur le tableau pour réfléchir au ralentissement de la voiture	13
Figure 11 : Apparition d'un opposant dans le simulateur pour tester notre code	13
Figure 12 : Live Twitch de a2rl_simsprint lors de la deuxième qualification pour la première course sprint. L'équipe FR4IAV est représentée par la voiture nommée "Juliette", en bas à droite, à l'arrêt afin d'éviter une collision avec la voiture "tgm", de l'équipe TGM.	14
Figure 13 : Passage du niveau de difficulté de 0 à 1	15
Figure 14 : Séparation en portion du circuit de Yas Marina pour le Curriculum Learning	15
Figure 15 : Courbe d'apprentissage en Curriculum Learning, ici, on observe le nombre de rewards obtenus en fonction du nombre d'environnement généré.	16
Figure 16 : Schéma méthodologique de l'approche LMPC et intégration des contraintes CBF	18
Figure 17 : Logique de fonctionnement alterné entre PID, MPC et LMPC	19
Figure 18 : Terminal Linux qui montre le temps de calcul du LMPC	20

TABLE DES MATIÈRES

Remerciements	2
Introduction	5
I. Présentation du Contexte	6
1. ESIGELEC et IRSEEM	7
2. Laboratoire de Navigation Autonome (LNA)	7
3. Équipe FR4IAV et Projet A2RL	8
II. Travail Effectué	9
1. Continuation du Projet S8	9
2. Récupération du Pure Pursuit et Dépassement des Adversaires	13
3. Recherches Effectuées et Tentatives d'Implémentation	15
a. Curriculum Learning	15
b. PID de Trajectoire et Vitesse	17
c. Learning Model Predictive Controller et Control Barrier Function	18
III. Bilan	21
1. Humain	21
2. Personnel	21
3. Technique	22
4. Charte Ethique de l'Ingénieur	23
Conclusion	24
Bibliographie / Sitographie	25
Glossaire	26

Introduction

Dans le cadre de ma formation d'ingénieur à l'ESIGELEC, j'ai réalisé un stage de deux mois à l'Institut de Recherche en Systèmes Électroniques Embarqués (IRSEEM), du 2 juin au 25 juillet 2025. Ma mission était celle de Technicienne Robotique Systèmes Embarqués, au sein du Laboratoire de Navigation Autonome (LNA), et portait sur le thème : « Test et développement sur les environnements de simulation dans le cadre du projet A2RL ».

Ce projet, l'Abu Dhabi Autonomous Racing League (A2RL), est une compétition internationale où des monoplaces de Super Formula autonomes s'affrontent sur le circuit de Yas Marina. L'équipe française FR4IAV, dont l'ESIGELEC fait partie, représente la France pour la première fois dans cette compétition. Mon rôle était d'apporter une contribution technique au développement du contrôleur autonome en simulation, en explorant différentes méthodes allant du renforcement learning (sur Unity) aux approches PID, Pure Pursuit, MPC et LMPC dans le simulateur Autonomia.

Dans ce rapport, je présenterai d'abord le contexte de mon stage et du projet A2RL, avant de détailler les missions que j'ai menées, les difficultés rencontrées et les solutions proposées. Enfin, j'exposerai le bilan de cette expérience, en prenant en compte les aspects humains, personnels et techniques, ainsi que les valeurs éthiques qui ont guidé ma démarche.



Figure 0 : Logo A2RL

I. Présentation du Contexte

1. L'ESIGELEC et l'IRSEEM

Fondée en 1901, l'ESIGELEC est une grande école d'ingénieurs française située sur le campus du Technopôle du Madrillet, à Saint-Étienne-du-Rouvray, près de Rouen. Elle forme des ingénieurs généralistes dans des domaines variés tels que les systèmes embarqués, l'électronique, l'intelligence artificielle, les réseaux, l'énergie, la mobilité ou encore la transition numérique. L'ESIGELEC se distingue par son ouverture internationale, grâce à ses nombreux partenariats, doubles diplômes et possibilités d'études à l'étranger. Elle s'appuie sur des laboratoires de recherche modernes et, en lien étroit avec l'industrie, accompagne la réalisation de projets innovants en partenariat avec de grandes entreprises et institutions.



Figure 1 : Image du bâtiment de l'IRSEEM

L'IRSEEM (Institut de Recherche en Systèmes Électroniques Embarqués) est l'unité de recherche (UR 4353) placée sous la double tutelle de l'ESIGELEC et l'Université de Rouen Normandie. Créé en 2001, il rassemble des chercheurs, enseignants-chercheurs, doctorants et experts dont les travaux couvrent notamment la conception, la modélisation et la caractérisation de composants électroniques, les systèmes mobiles autonomes et connectés, la sécurité et la fiabilité des systèmes embarqués. L'IRSEEM répond aussi à des enjeux applicatifs dans des secteurs comme l'automobile, l'aéronautique, l'énergie, les télécommunications ou la santé et s'inscrit dans des partenariats académiques et industriels pour répondre à des enjeux technologiques d'avenir.

I. Présentation du Contexte

2. Laboratoire de Navigation Autonome (LNA)

Au sein de l'IRSEEM, la plateforme de Navigation Autonome est consacrée à la recherche et au développement de solutions pour la mobilité intelligente. Ses travaux vont de la robotisation de véhicules à la validation de systèmes avancés d'aide à la conduite (ADAS), en passant par le développement de technologies innovantes comme la recharge automatisée des véhicules électriques autonomes.

Le laboratoire dispose de moyens expérimentaux de pointe : bancs à rouleaux pour véhicules hybrides et électriques, salle de navigation autonome, chambres anéchoïques pour tester la compatibilité électromagnétique, ainsi qu'un large ensemble d'équipements (véhicules robotisés, capteurs lidar, systèmes de positionnement haute précision, robots mobiles, etc).

Au cours de mon stage, j'ai travaillé au sein du Centre d'Innovation et de Transfert Technologique (CI2T), une équipe pluridisciplinaire spécialisée en recherche et développement, directement impliquée dans les projets du LNA. Cette expérience m'a permis de contribuer à des travaux concrets en lien avec l'innovation et la mobilité autonome.

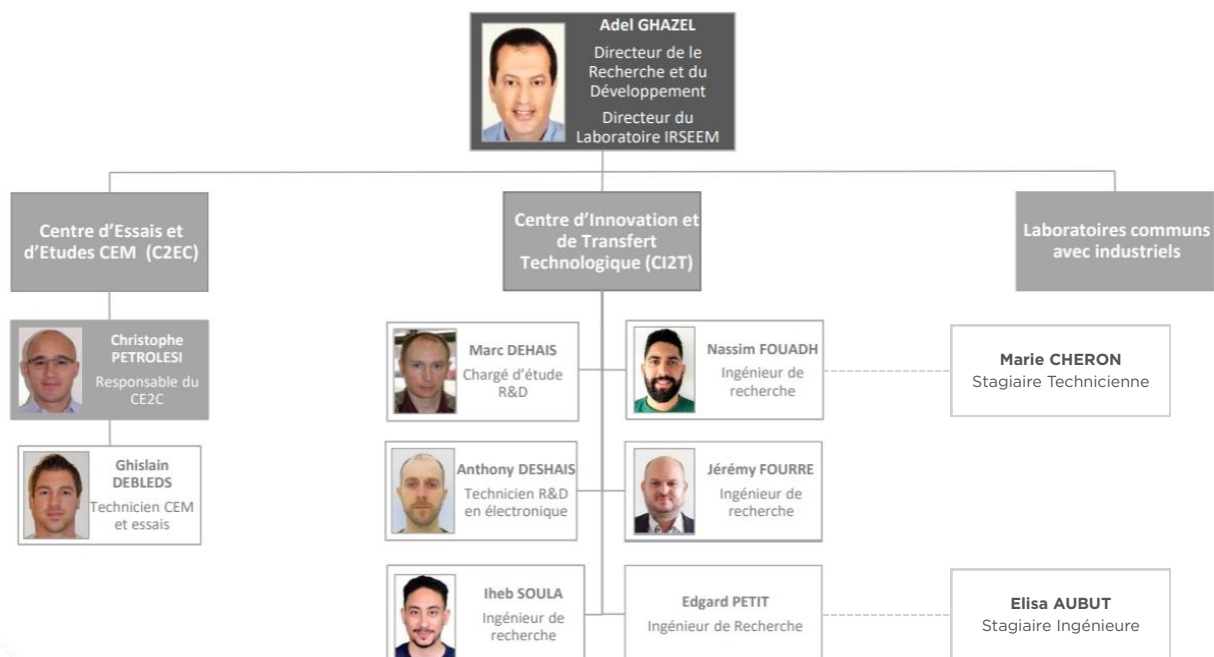


Figure 2 : Organisation du Laboratoire de l'IRSEEM

I. Présentation du Contexte

3. Équipe FR4IAV et Projet A2RL

Le projet Abu Dhabi Autonomous Racing League (A2RL) est une compétition internationale organisée par ASPIRE, où des monoplaces de Super Formula autonomes s'affrontent sur le circuit de Yas Marina, afin de faire progresser la recherche en mobilité intelligente, en intelligence artificielle et en robotique.

L'équipe française FR4IAV (French Team for Artificial Intelligence Autonomous Vehicle), propulsée par le cluster de mobilité ALADIN (Advanced Landcraft Association for Driving INnovation), représentera la France pour la première fois à l'A2RL en 2025. Elle rassemble des étudiant·es-ingénieur·es de l'ESIGELEC, de l'INSA Rouen Normandie, de l'ESIEE Paris, de l'ECE Paris et de Sorbonne Université, avec le soutien de laboratoires de recherche, d'institutions régionales et d'acteurs industriels.



Figure 3 : Logo de ALADIN



Figure 4 : Logo de l'équipe FR4IAV

Le pilotage scientifique est assuré par Rodolphe Debeauvais, ingénieur technique chez ALADIN, qui coordonne les activités techniques et assure le lien avec les organisateurs internationaux.

Chaque école constitue une sous-équipe avec une mission spécifique : l'ESIGELEC développe le contrôleur autonome et l'IA (modélisation, entraînement, optimisation de l'agent et communication avec le véhicule via ROS2) ; l'INSA Rouen travaille sur la perception et les capteurs (simulation embarquée, gestion des capteurs virtuels, prétraitement des données) ; et l'ESIEE Paris élabore et valide les architectures logicielles, notamment via des machines d'état pour la prise de décision. L'ECE Paris et Sorbonne Université viennent compléter l'ensemble par leur expertise transversale.

C'est au sein de la sous-équipe contrôleur autonome et IA que j'ai réalisé mon stage technicien, en contribuant directement aux travaux de modélisation et d'implémentation du système.

II. Travail Effectué

1. Continuité du Projet S8

J'ai réalisé le semestre 8 sur ce même projet, ce qui constitue le point de départ direct de mon stage. Lors du S8, notre groupe avait travaillé exclusivement sur le simulateur Unity, développé par l'équipe du PING, afin d'améliorer les performances de la voiture en reinforcement learning.

Le stage s'inscrit donc dans la continuité de ce travail. Alors qu'au S8 j'étais sur un ordinateur personnel, j'ai utilisé pour le stage un ordinateur du LNA, plus puissant et rapide, ce qui m'a permis d'installer le simulateur Autonoma (développé directement par ASPIRE) et l'ensemble des logiciels nécessaires. J'ai ainsi pu poursuivre mes travaux de reinforcement learning, mais cette fois en utilisant directement les données issues du simulateur, que je n'avais pas pu installer sur mon ordinateur personnel.

À la demande de Rodolphe, je me suis ensuite concentrée sur l'étude des glissements de la voiture afin de les prédire le mieux possible en simulation. Pour cela, j'ai analysé les différents topics ROS2 renvoyés par le simulateur afin de calculer le glissement selon la formule :

$$\text{Glissement} = \frac{\text{vitesse}_{\text{roues}} - \text{vitesse}_{\text{vehicule}}}{\max(\text{vitesse}_{\text{roues}}, \text{vitesse}_{\text{vehicule}})}$$

J'ai testé trois méthodes de prédiction pour modéliser ce phénomène :

- La méthode des moindres carrés
- Un modèle Random Forest
- Un modèle XGBoost

Pour chacune de ces méthodes, j'ai calculé le score R^2 afin d'évaluer leur précision et de déterminer laquelle était la plus efficace pour prédire le glissement. Le score R^2 (aussi appelé coefficient de détermination, ou R^2 sur les images) est un indicateur statistique qui mesure la qualité de prédiction d'un modèle par rapport aux données réelles. Plus il est proche de 1, plus le modèle est performant.

$$R^2 = 1 - \frac{\sum (y_{\text{réel}} - y_{\text{prédit}})^2}{\sum (y_{\text{réel}} - \bar{y}_{\text{réel}})^2}$$

$y_{\text{réel}}$: valeurs réelles observées
 $y_{\text{prédit}}$: valeurs prédites par le modèle
 $\bar{y}_{\text{réel}}$: moyenne des valeurs réelles

II. Travail Effectué

1. Continuité du Projet S8

La méthode des moindres carrés, même en complexifiant l'équation, ne donnait pas de résultats satisfaisants car elle manquait de précision. La méthode Random Forest, en revanche, produisait des équations plus simples mais efficaces pour prédire le glissement.

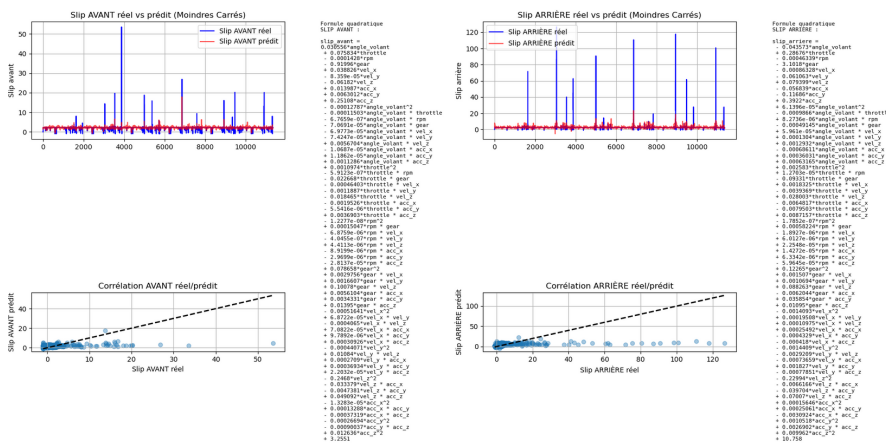


Figure 5 : Résultat de la prédiction avec la méthode des Moindres Carrés, peu fiable car les scores R^2 sont faibles

Score R^2 SLIP AVANT : 0.1941

Score R^2 SLIP ARRIÈRE : 0.1128

La courbe du haut montre la comparaison entre le slip réel et prédit, et celle du bas illustre leur corrélation. Les écarts importants indiquent une faible précision du modèle, confirmée par les scores R^2 très bas (0.1941 à l'avant et 0.1128 à l'arrière).

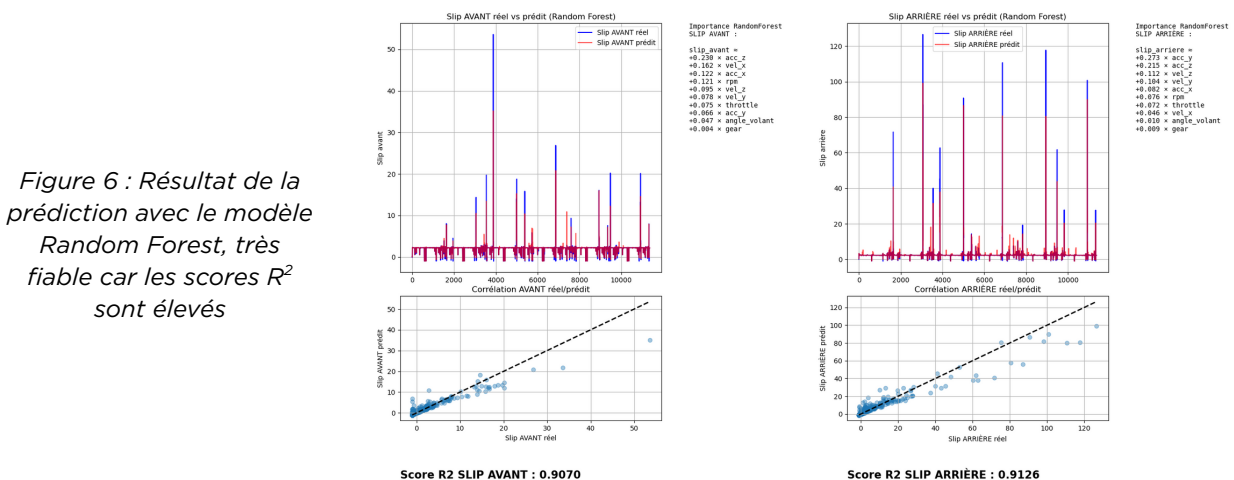


Figure 6 : Résultat de la prédiction avec le modèle Random Forest, très fiable car les scores R^2 sont élevés

Score R^2 SLIP AVANT : 0.9070

Score R^2 SLIP ARRIÈRE : 0.9126

Les courbes réelle et prédite se superposent beaucoup mieux, et la corrélation en bas se rapproche de la diagonale. Les scores R^2 élevés (0.9070 et 0.9126) montrent que le modèle capte bien la dynamique du glissement.

II. Travail Effectué

1. Continuité du Projet S8

Enfin, la méthode XGBoost était extrêmement précise, mais cette précision était probablement due à un phénomène de surapprentissage (overfitting) qui faussait les résultats. Le modèle mémorise trop fidèlement les données d'entraînement, au lieu d'apprendre à en tirer une tendance générale. Autrement dit, les résultats paraissent excellents pendant la phase d'apprentissage, mais ne sont pas représentatifs dans un contexte réel ou sur des situations légèrement différentes.

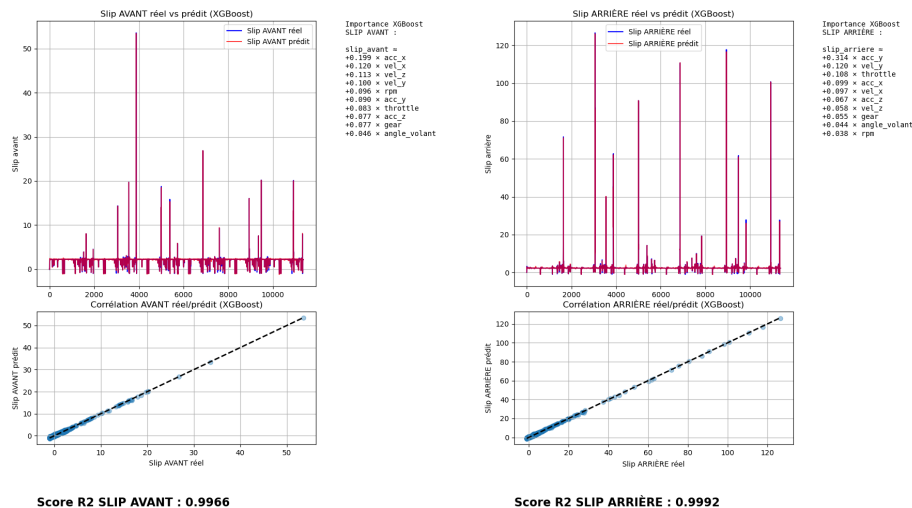


Figure 7 : Résultat de la prédiction avec le modèle XGBoost, moyennement fiable car risque d'overfitting

Les courbes du haut se confondent presque totalement, et les points du graphique de corrélation suivent parfaitement la diagonale. Les scores R^2 proches de 1 (0.9966 et 0.9992) traduisent une précision quasi parfaite, probablement due à un surapprentissage.

Le principe de ces méthodes était de récupérer les données du simulateur, de rechercher la courbe qui correspond le mieux, puis d'obtenir une fonction utilisable comme reward pour la voiture. Pour ce faire, j'ai collecté un grand nombre de variables telles que le temps, l'appui sur la pédale d'accélération et de frein, l'angle du volant, les vitesses (x, y, z), les accélérations (x, y, z), le rapport de vitesse engagé et les valeurs de glissement calculées grâce à la formule précédente.

II. Travail Effectué

1. Continuité du Projet S8

Cependant, il me manquait une donnée essentielle : la température des pneus. Celle-ci joue un rôle majeur dans le calcul et la précision du glissement. En raison d'un problème interne dans le développement du simulateur Autonomia, cette information restait bloquée à zéro. Ce défaut ne pouvant pas être corrigé de notre côté, aucune des méthodes testées ne pouvait être utilisée de manière réellement fiable.

```
wheelspeedfr: 163.5  
wheelspeedfl: 163.05  
wheelspeedrr: 164.22500000000002  
wheelspeedrl: 163.72500000000002  
rpm: 4573.6  
gear: 6  
tiretempfr: 0.0  
tiretempfl: 0.0  
tiretemprr: 0.0  
tiretemprrl: 0.0
```

Figure 8 : Affichage des données du topic, avec les température ("tiretemp") égales à 0.0 pour les quatre pneus



Figure 9 : Affichage des températures sur le simulateur Autonomia au même moment, impossible à récupérer sur les topics

Initialement, Rodolphe souhaitait appliquer cette approche dans la simulation Unity, mais celle-ci ne donnait accès qu'à la vitesse globale et à l'angle du volant, sans les vitesses des roues individuelles. J'ai donc poursuivi mes recherches pour intégrer cette logique directement dans du code de reinforcement learning sous Autonomia, avant que Rodolphe me demande finalement de l'implémenter dans le contrôleur PID de Victor DUCRAY, étudiant de l'ESIGELEC en stage chez ALADIN, qui travaillait alors sur un Pure Pursuit associé à un PID vitesse. Cela devait permettre d'améliorer la gestion des freinages et des accélérations afin de limiter les glissements et d'optimiser la performance globale du véhicule.

II. Travail Effectué

2. Récupération du Pure Pursuit et Dépassement des Adversaires

Après avoir travaillé sur des méthodes pour éviter le glissement de la voiture, nous avons travaillé sur la récupération du code de Pure Pursuit couplé à un PID vitesse développé par Victor Ducray. Le principe du Pure Pursuit est de suivre une trajectoire définie par une suite de coordonnées (x et y), auxquels est associée une vitesse cible. Le PID vitesse permet alors d'adapter l'accélération et le freinage pour que la voiture atteigne cette vitesse à chaque point. L'objectif était d'y intégrer un système de ralentissement et d'esquive des adversaires afin de pouvoir éviter une voiture lente ou arrêtée sur la piste.

Avec Elisa Aubut (stagiaire ingénieure), nous avons dû implémenter cette fonctionnalité dans un délai très court, nous n'avons eu accès au code que deux jours avant la qualification avec des opposants sur la piste. La principale difficulté était que ce code avait été modifié depuis les dernières versions partagées et que ni Elisa ni moi n'y étions vraiment familières. De plus, nous travaillions jusque-là sur d'autres aspects du projet, pour ma part, j'avais surtout développé sur Unity en C# avec du reinforcement learning, tandis qu'ici il s'agissait de Python avec Pure Pursuit et PID sur le simulateur Autonomo.

Le premier jour, nous avons surtout cherché à comprendre la logique du code, testant quelques idées avec l'aide de nos tuteurs. Mais faute de temps, rien de concluant n'en est ressorti. Le lendemain, nous avons pris le temps de poser nos idées clairement sur un tableau, en partant d'une note que j'avais griffonnée la nuit précédente (une sorte de pseudo-code assez brouillon mais qui a servi de base). Après avoir clarifié la logique de notre algorithme, je me suis concentrée sur l'implémentation, pendant qu'Elisa poursuivait son rapport et m'aidait sur certains points de doute.

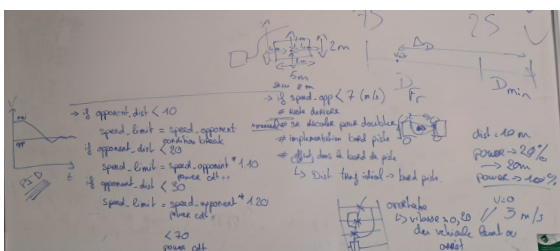


Figure 10 : Notre pseudo code sur le tableau pour réfléchir au ralentissement de la voiture



Figure 11 : Apparition d'un opposant dans le simulateur pour tester notre code

II. Travail Effectué

2. Récupération du Pure Pursuit et Dépassement des Adversaires

Après de nombreux essais, nous avons réussi à implémenter un système de ralentissement et d'arrêt. Concrètement, si une voiture devant nous était très lente ou arrêtée en travers de la piste (crash ou autre), notre véhicule pouvait ralentir pour s'adapter ou s'arrêter juste avant l'impact. En revanche, nous n'avons pas réussi à mettre en place le dépassement. En Pure Pursuit, cela s'avère extrêmement complexe, il ne suffit pas de quitter la trajectoire, il faut aussi prévoir le retour progressif sur la trajectoire initiale, gérer l'angle de braquage, la réaccélération, les changements de rapport, et éviter de faire déraeper la voiture. Ces contraintes rendaient la manœuvre très difficile à stabiliser en si peu de temps.

Lors de la deuxième partie des qualifications, où nous devions rouler avec deux autres voitures sur le circuit de Yas Marina, notre système a toutefois fonctionné, nous avons pu nous arrêter proprement devant une voiture accidentée et bloquée en travers de la piste, démontrant l'efficacité du ralentissement et de l'arrêt.



Figure 12 : Live Twitch de a2rl_simsprint lors de la deuxième qualification pour la première course sprint.

L'équipe FR4IAV est représentée par la voiture nommée "Juliette", en bas à droite, à l'arrêt afin d'éviter une collision avec la voiture "tgm", de l'équipe TGM

Cet échec sur la gestion des dépassements nous a conduit à une conclusion importante, le Pure Pursuit est une méthode trop rigide pour implémenter des comportements complexes comme l'évitement et les dépassements. Cela m'a amenée à commencer à explorer de nouvelles pistes, d'une part en continuant avec d'autres méthode pour le reinforcement learning, d'autre part avec des contrôleurs plus complets de type PID trajectoire, couplé avec un PID vitesse, voire des approches alternatives que j'ai découvertes dans des projets similaires plus simples sur internet.

II. Travail Effectué

3. Recherches Effectuées et Tentatives d'Implémentation

a. Curriculum Learning

En reinforcement learning, le comportement de l'agent est guidé par un système de récompenses (rewards) et de pénalités. Plus la voiture adopte un comportement souhaité (par exemple, avancer rapidement dans la bonne direction), plus elle accumule de rewards, ce qui l'encourage à répéter ce comportement. À l'inverse, des actions indésirables (comme heurter un mur ou mal s'orienter) entraînent des pénalités, ce qui décourage la voiture de reproduire ces erreurs. L'agent apprend donc progressivement à maximiser ses rewards et à éviter les comportements pénalisés.

Pour tenter d'accélérer cet apprentissage, j'ai expérimenté le curriculum learning sur le simulateur Unity. L'idée était de découper le circuit de Yas Marina en portions de difficulté croissante afin que la voiture apprenne d'abord les bases sur des situations simples, avant de passer à des configurations plus complexes :

- Niveau 0 : Ligne droite
- Niveau 1 : Virage à droite
- Niveau 2 : Virage à gauche
- Niveau 3 : Virage en S
- Niveau 4 : Virage en S serré
- Niveau 5 : Secteur 1 (circuit de F1)
- Niveau 6 : Secteur 2 (circuit de F1)
- Niveau 7 : Secteur 3 (circuit de F1)
- Niveau 8 : Circuit complet

[10:39:02] [Curriculum] Difficulty set to 1
UnityEngine.Debug.Log (object)

Figure 13 : Passage du niveau de difficulté de 0 à 1

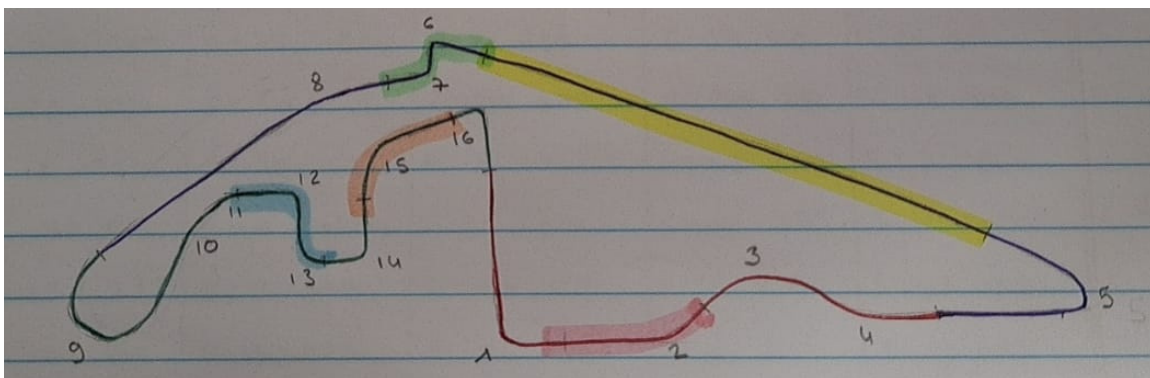


Figure 14 : Séparation en portion du circuit de Yas Marina pour le Curriculum Learning

II. Travail Effectué

3. Recherches Effectuées et Tentatives d'Implémentation

a. Curriculum Learning

Les rewards et pénalités reposaient sur plusieurs critères : la vitesse, la progression sur le circuit, l'orientation du véhicule (alignement de la voiture par rapport à la courbure de la piste), les collisions avec le mur et le temps au tour.

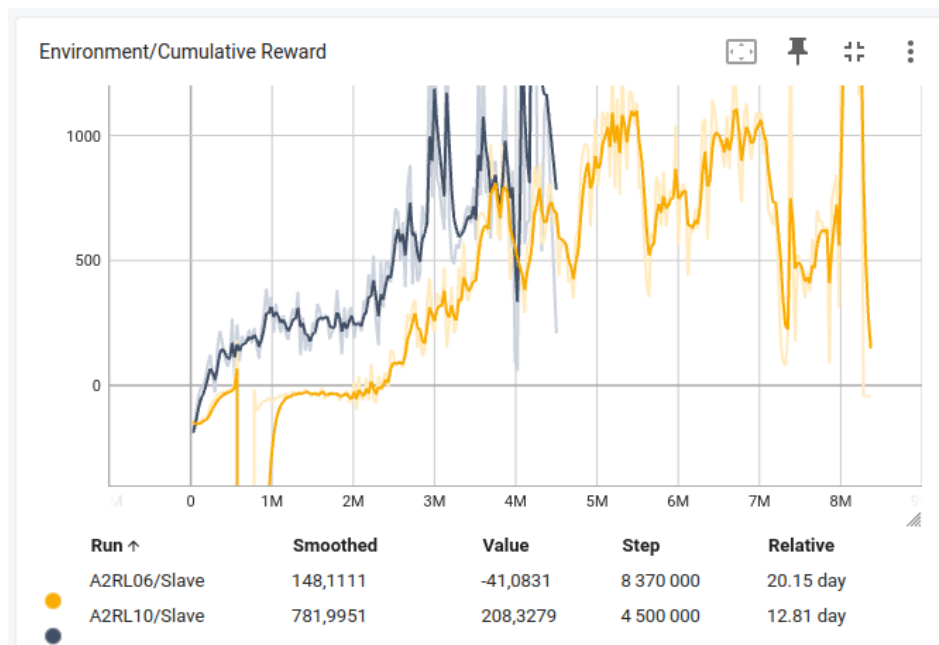


Figure 15 : Courbe d'apprentissage en Curriculum Learning, ici, on observe le nombre de rewards obtenus en fonction du nombre d'environnement généré.

Au final, nous avons constaté que l'apprentissage devenait de plus en plus lent à mesure que la difficulté augmentait. De plus, ce découpage entraînait des comportements incohérents : après avoir appris à aller vite en ligne droite, la voiture n'anticipait pas le ralentissement nécessaire pour un virage. Puis, une fois entraînée à prendre un virage à droite, elle ne comprenait pas pourquoi elle recevait des pénalités lors du niveau suivant avec un virage à gauche, et ainsi de suite. En pratique, le curriculum learning n'a donc pas apporté les améliorations espérées.

II. Travail Effectué

3. Recherches Effectuées et Tentatives d'Implémentation

b. PID de Trajectoire et Vitesse

Afin de dépasser les limites du Pure Pursuit, j'ai exploré l'utilisation d'un PID vitesse couplé à un PID trajectoire. Un PID vitesse permet de contrôler l'accélération et le freinage afin que la voiture atteigne une vitesse cible donnée. En y ajoutant une rampe d'accélération et de décélération, il devient possible d'arriver plus progressivement à la consigne, ce qui évite les comportements brusques. C'était particulièrement intéressant pour corriger les problèmes observés avec le Pure Pursuit lors de nos essais de dépassement, où la voiture avait tendance à déraiper à cause d'accélération ou décélération trop brutales.

L'idée était de coupler ce PID vitesse avec un PID trajectoire. Contrairement au Pure Pursuit, qui suit passivement une série de points, un PID trajectoire ajuste dynamiquement la direction de la voiture en fonction de l'erreur de position latérale et d'orientation par rapport à la trajectoire cible. En combinant les deux, il devenait théoriquement possible d'intégrer la présence des adversaires dans le calcul et de gérer des dépassements, en s'écartant temporairement de la trajectoire idéale pour ensuite y revenir de manière contrôlée.

J'ai donc entrepris des recherches approfondies et commencé à coder un premier PID trajectoire cohérent pour tester cette approche. Cependant, le réglage des nombreux paramètres (gains P, I, D pour la vitesse et pour la trajectoire, gestion des rampes, stabilité en virage, etc.) s'est avéré long et fastidieux. De plus, comme je ne connaissais toujours pas bien le code de Victor, repartir dessus aurait été trop complexe et chronophage. Il me paraissait donc plus simple de repartir de zéro avec un PID neuf plutôt que de modifier en profondeur celui existant, conçu initialement pour un Pure Pursuit.

Après en avoir discuté avec Rodolphe, il m'a finalement demandé de ne pas poursuivre ce travail, car il ne trouvait pas cela pertinent, Victor disposait déjà d'un PID vitesse et le Pure Pursuit fonctionnait correctement tant qu'il n'y avait pas de dépassement. J'ai donc stoppé mes essais à ce stade, bien que cette piste restait intéressante pour la gestion des adversaires.

II. Travail Effectué

3. Recherches Effectuées et Tentatives d'Implémentation

c. Learning Model Predictive Controller et Control Barrier Function

Le Learning Model Predictive Control (LMPC), ou Commande Prédictive Apprenante, est une extension du Model Predictive Control (MPC). Le principe du MPC est de calculer à chaque instant la commande optimale (accélération, freinage, braquage) sur un horizon de temps donné, en tenant compte du modèle dynamique de la voiture et des contraintes. Le LMPC ajoute une dimension apprentissage : il s'appuie sur les trajectoires et actions des tours précédents pour améliorer progressivement la trajectoire et réduire le temps au tour.

La Control Barrier Function (CBF), ou fonction barrière de contrôle, est une méthode permettant d'intégrer des contraintes de sécurité en temps réel dans l'optimisation. Concrètement, elle empêche le véhicule de sortir de la piste ou de s'approcher trop près des adversaires en ajoutant des conditions strictes sur les distances de sécurité.

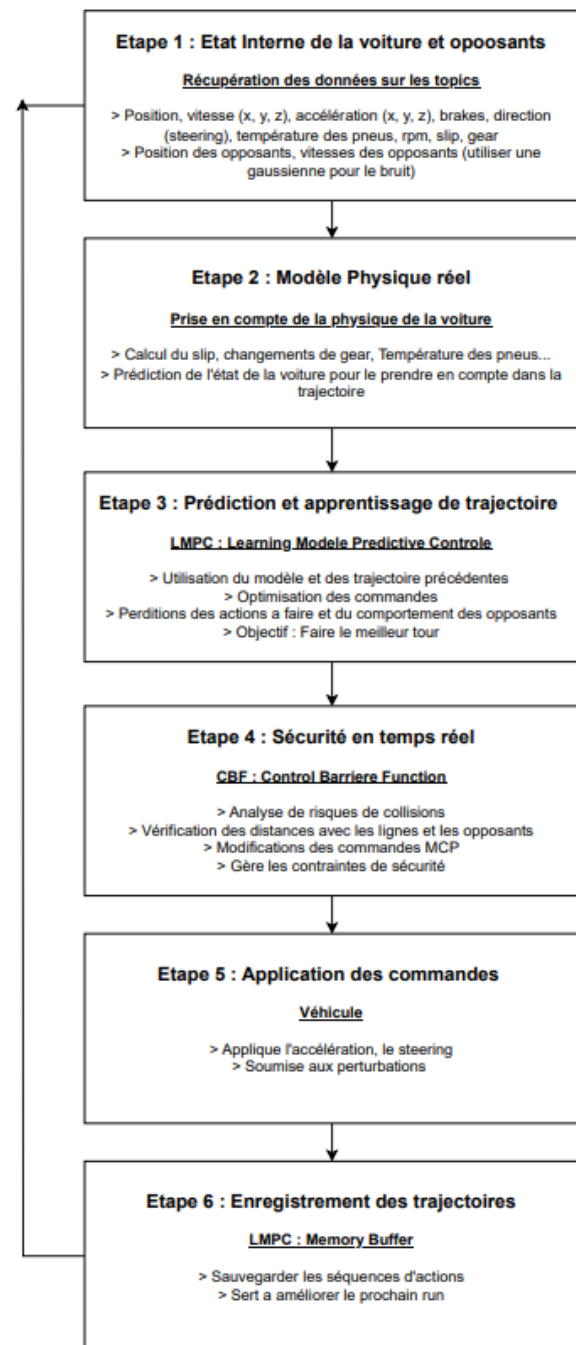


Figure 16 : Schéma méthodologique de l'approche LMPC et intégration des contraintes CBF

II. Travail Effectué

3. Recherches Effectuées et Tentatives d'Implémentation

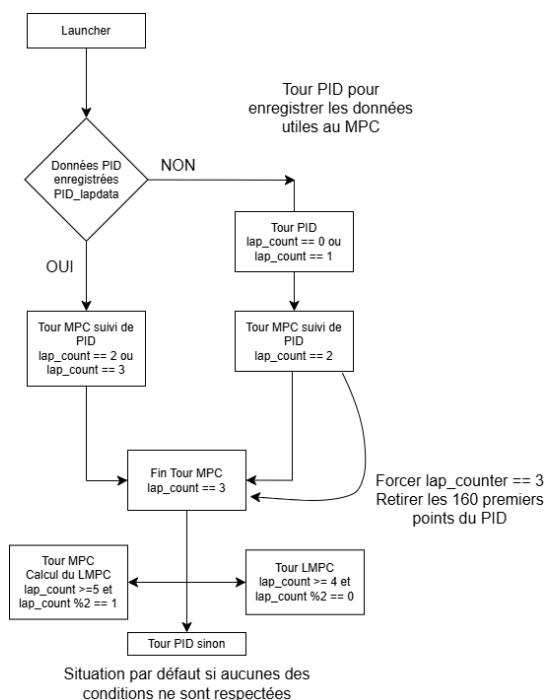
c. Learning Model Predictive Controller et Control Barrier Function

Pour mettre en œuvre cette logique, j'ai procédé ainsi :

- Le premier tour était effectué en PID, à vitesse réduite, afin de collecter un maximum de données : coordonnées, vitesses en x et y, appuis sur les pédales de frein et d'accélération.
- Un solveur MPC était utilisé pour le deuxième tour, servant de base au calcul des trajectoires optimisées.
- Dès le quatrième tour, la voiture devait rouler avec un LMPC, utilisant les informations enregistrées pour affiner sa conduite, apprendre des trajectoires passées et améliorer progressivement son temps

Le CBF intervenait en parallèle pour maintenir la voiture dans les limites de la piste et gérer les distances de sécurité avec les adversaires.

Afin d'alterner calcul et apprentissage, j'avais mis en place une logique de tours pairs et impairs : les tours impairs étaient réalisés en MPC pour stabiliser la trajectoire, tandis que les tours pairs étaient effectués en LMPC afin d'intégrer l'apprentissage à partir des données collectées.



II. Travail Effectué

3. Recherches Effectuées et Tentatives d'Implémentation

c. Learning Model Predictive Controller et Control Barrier Function

Malheureusement, la principale limite rencontrée a été le temps de calcul du solveur. Même en commençant les calculs pendant le tour MPC, l'ordinateur n'arrivait pas à fournir des solutions assez rapidement pour une voiture allant à 300 km/h comme une Super Formula. Les deux dernières semaines de stage ont donc été consacrées à tenter d'optimiser le solveur, mais celui-ci restait trop lent pour être utilisable en conditions réelles.

```
[lmpc_node-9] [LMPC TIMER] total: 3834.4 ms
[lmpc_node-9] - SafeSet ref      : 1712.4 ms
[lmpc_node-9] - NLP init + Q     : 2081.4 ms
[lmpc_node-9] - Interp. bordures : 7.8 ms
[lmpc_node-9] - Solveur CasADi   : 32.8 ms
```

Figure 18 : Terminal Linux qui montre le temps de calcul du LMPC

En conclusion, les solveurs MPC et LMPC étaient fonctionnels, mais leur vitesse de calcul n'était pas suffisante pour le niveau de complexité et de performance attendu dans le cadre du projet A2RL.

III. Bilan

1. Humain

Ce stage m'a offert une expérience humaine particulièrement enrichissante. Même si toutes les personnes de l'équipe n'étaient pas directement impliquées dans le projet A2RL, j'ai eu la chance de côtoyer des ingénieurs, des techniciens et des chargés d'étude qui m'ont apporté des conseils précieux. Leurs retours et leur expérience m'ont permis de progresser et d'élargir ma vision du travail en ingénierie.

Le projet s'est également déroulé dans un cadre collaboratif impliquant plusieurs écoles, ce qui a nécessité de travailler avec d'autres personnes à distance. Cette dimension a renforcé mes capacités de communication, d'organisation et d'adaptation dans un contexte où les échanges et la coordination étaient essentiels.

La collaboration avec les autres stagiaires et les encadrants techniques m'a également appris à mieux communiquer, à poser les bonnes questions et à m'intégrer dans une équipe pluridisciplinaire où chacun apporte sa contribution.

2. Personnel

Sur le plan personnel, ce stage a confirmé mon intérêt pour les systèmes embarqués appliqués au sport automobile. Le passage d'Unity (utilisé au semestre 8) au simulateur Autonoma, beaucoup plus réaliste et complexe, a été un vrai défi mais aussi une source de motivation.

J'ai appris à sortir de ma zone de confort en travaillant dans de nouveaux environnements, notamment sous Linux, et en approfondissant l'utilisation de nouveaux langages (C# pour Unity et Python/ROS2 pour Autonoma).

Ce stage m'a aussi permis d'améliorer ma capacité à faire des recherches en amont avant de me lancer dans un projet ou un code, en adoptant une démarche plus rigoureuse et méthodique.

Enfin, l'autonomie laissée sur certaines tâches m'a permis de développer mon sens de l'initiative et ma persévérance, notamment face à des implémentations complexes ou à des résultats qui tardaient à arriver

III. Bilan

3. Technique

D'un point de vue technique, ce stage a constitué une immersion intense dans le domaine du contrôle de véhicules autonomes en sport automobile.

J'ai pu manipuler différents outils et approches, du reinforcement learning sous Unity au contrôle PID, Pure Pursuit, MPC et LMPC dans le simulateur Autonomia. La complexité des tâches m'a permis d'approfondir mes compétences en programmation (C#, Python, ROS2), en simulation avancée, ainsi qu'en optimisation et en modélisation de systèmes dynamiques.

J'ai également dû tenir compte de contraintes réelles telles que les fonctions barrière de contrôle (CBF), les limites de piste ou les interactions avec les adversaires, ce qui m'a plongé dans des problématiques concrètes de sécurité et de performance.

Ces expériences techniques sont en parfaite adéquation avec mon projet professionnel, celui de travailler sur les systèmes embarqués dans le sport automobile, et s'alignent parfaitement dans le cadre de ma mobilité en Angleterre pour mon double diplôme à Cranfield en Automotive Mechatronics, une dominante qui aborde précisément les thématiques de contrôle avancé (MPC, CBF, PID) et d'intelligence artificielle appliquée à l'automobile.

III. Bilan

4. Charte Éthique de l'Ingénieur

Définition (Charte éthique de l'ingénieur, IESF) : « En tant que personnes individuelles, l'ingénieur et le scientifique participent à faire la promotion de leurs métiers et de leurs domaines d'activité pour susciter des vocations. »

En tant qu'individuelle, j'ai choisi l'élément Promotion. Être une femme dans une école majoritairement masculine m'a fait prendre conscience de l'importance de promouvoir le rôle des ingénieures afin d'encourager plus de diversité dans ces métiers. Souhaitant travailler dans l'informatique et l'automobile, deux domaines encore très masculins, je trouve essentiel de montrer que notre place est légitime et que notre implication peut inspirer d'autres femmes à s'engager dans cette voie.

Définition (Charte éthique de l'ingénieur, IESF) : « En tant que membres d'un collectif de travail, l'ingénieur et le scientifique travaillent méthodiquement afin d'assurer la traçabilité des résultats et des données qui fondent les décisions de leur collectif de travail. »

Dans le collectif de travail, j'ai retenu la valeur de Transparence. Tout au long de mon stage, j'ai gardé une trace des essais et méthodes testées, en rédigeant avec Elisa un document récapitulatif pour les futures équipes afin d'éviter les mêmes impasses. Déjà lors du semestre 8, j'avais mis en place, avec les membres de mon équipe, des tutoriels d'installation et encouragé l'équipe à documenter ses travaux. Cette rigueur dans la traçabilité reflète ma volonté de travailler de manière claire et utile pour le collectif, en facilitant la continuité du projet.

Définition (Charte éthique de l'ingénieur, IESF) : « En tant que membres de la société toute entière, l'ingénieur et le scientifique osent (ré-)inventer pour l'intérêt collectif. »

Enfin, au sein de la société, j'ai choisi Créativité. Pour moi, elle est essentielle en ingénierie car elle permet de dépasser les méthodes classiques et d'oser inventer de nouvelles solutions. J'ai pu l'illustrer en explorant une approche totalement différente avec le LMPC et les CBF, une piste encore jamais testée sur le projet. Même si elle n'a pas abouti pleinement, cette démarche montre l'importance de penser différemment et d'oser proposer des alternatives innovantes pour faire progresser les systèmes embarqués et la mobilité autonome.

Conclusion

En conclusion de ce stage, je tiens à remercier l'IRSEEM et plus particulièrement le Laboratoire de Navigation Autonome, pour m'avoir donné l'opportunité de travailler sur un projet aussi ambitieux que l'A2RL. Cette expérience m'a permis de plonger au cœur de la recherche et du développement autour de la conduite autonome, en explorant des approches variées allant du reinforcement learning aux contrôleurs prédictifs.

Dans ce rapport, j'ai présenté le contexte du projet et le rôle de l'ESIGELEC au sein de l'équipe FR4IAV, avant de détailler mes missions principales : la prédiction du glissement, la reprise du Pure Pursuit couplé au PID vitesse, ainsi que mes recherches sur le Curriculum Learning, le PID Trajectoire/Vitesse et enfin le LMPC associé aux CBF. J'ai également dressé un bilan humain, personnel et technique de cette expérience, en lien avec la charte éthique de l'ingénieur.

En résumé, ce stage a été une expérience extrêmement formatrice. Il m'a permis de développer mes compétences techniques en programmation (C#, Python, ROS2), en simulation (Unity, Autonomia) et en optimisation de contrôleurs, tout en renforçant mes capacités d'adaptation, de travail collaboratif et de recherche. Il a aussi conforté mon projet professionnel : travailler sur les systèmes embarqués dans le sport automobile, domaine que je poursuivrai lors de ma mobilité en Angleterre dans la spécialité Automotive Mechatronics.

Sitographie & Bibliographie

MPC et LMPC

- Rosolia, U., & Borrelli, F. (2020). Learning Model Predictive Control for Autonomous Racing. IEEE Transactions on Control Systems Technology. [En ligne]. Disponible sur : <<https://ieeexplore.ieee.org/document/9811969>>
- MathWorks. (2020). Vehicle Path Tracking using Model Predictive Control (MPC). MATLAB File Exchange. Disponible sur : <<https://fr.mathworks.com/matlabcentral/fileexchange/84320-vehicle-path-tracking-using-model-predictive-control-mpc>>
- Urosolia, V. (2018). PredictiveModel.py, Racing LMPC Repository. GitHub. Disponible sur : <<https://github.com/urosolia/RacingLMPC>>
- Hybrid Robotics. (2020). Car Racing Control (MPC implementation). GitHub. Disponible sur : <https://github.com/HybridRobotics/car-racing/tree/master/car_racing/control>

PID

- Luo, J. (2018). Self-Driving-Car-PID-Controller. GitHub. Disponible sur : <<https://github.com/jluo-bgl/Self-Driving-Car-PID-Controller/tree/master>>
- Satya Mallick. (2021). PID Controller in ROS2 (CARLA). LearnOpenCV. Disponible sur : <<https://learnopencv.com/pid-controller-ros-2-carla/>>
- Millington, I. (2014). Racing Vehicle Control Systems using PID Controllers. In Game AI Pro: Collected Wisdom of Game AI Professionals. CRC Press. Disponible sur : <https://www.gameai.pro/GameAIPro/GameAIPro_Chapter40_Racing_Vehicle_Control_Systems_using_PID_Controllers.pdf>

Divers

- Klotz, J., et al. (2023). ROS 2: A Next-Generation Framework for Robotic Applications. arXiv. Disponible sur : <<https://arxiv.org/pdf/2311.14276>>
- López-Campos, R., Rojas Pérez, L. O. Y., & Martínez-Carranza, J. (2020). Following and Overtaking: A Policy for Autonomous Car Driving. Computación y Sistemas, 24(3). DOI : 10.13053/cys-24-3-3475. Disponible sur : <https://www.researchgate.net/publication/346219109_Following_and_Overtaking_A_Policy_for_Autonomous_Car_Driving>

Glossaire

Agent

Dans le cadre du reinforcement learning, l'agent est le programme ou modèle d'intelligence artificielle qui prend des décisions. Il interagit avec l'environnement, reçoit des récompenses ou des pénalités, et apprend à adopter les comportements qui maximisent ses performances au fil du temps.

Control Barrier Function (CBF)

Méthode mathématique utilisée pour garantir la sécurité dans un système de contrôle. Elle impose des contraintes, par exemple rester dans les limites de la piste ou éviter les collisions, en empêchant le contrôleur de franchir certaines zones dites « interdites ».

Learning Model Predictive Control (LMPC)

Extension du MPC qui intègre l'apprentissage à partir des trajectoires précédentes. Il permet d'améliorer la performance du véhicule au fil des tours, notamment en optimisant la trajectoire et en réduisant le temps au tour.

Model Predictive Control (MPC)

Méthode de contrôle prédictif où le système anticipe son comportement futur sur un horizon donné. Le contrôleur calcule à chaque instant la meilleure commande possible pour atteindre un objectif tout en respectant des contraintes physiques (vitesse, limites de piste, sécurité, etc.).

PID (Proportionnel - Intégral - Dérivé)

Le PID est un régulateur utilisé en automatique pour stabiliser et contrôler un système. Il agit en comparant une valeur mesurée (comme la vitesse du véhicule) à une consigne cible, puis en calculant un écart appelé erreur.

La partie proportionnelle réagit à l'erreur instantanée, la partie intégrale corrige les écarts cumulés dans le temps, et la partie dérivée anticipe les variations futures. En combinant ces trois effets, le PID permet une régulation stable, précise et réactive, largement utilisée dans les systèmes embarqués automobiles.

Glossaire

Pure Pursuit

Algorithme de suivi de trajectoire couramment utilisé en robotique mobile et en conduite autonome. Il calcule un point cible situé à une certaine distance devant le véhicule et oriente la direction du volant pour le rejoindre. Simple et efficace sur des trajectoires prévisibles, il devient toutefois limité pour les manœuvres complexes comme les dépassements.

Reinforcement Learning (Apprentissage par renforcement)

Méthode d'intelligence artificielle dans laquelle un agent apprend à agir en interagissant avec un environnement. À chaque action, il reçoit une récompense (reward) positive ou négative selon son comportement. L'objectif est de maximiser la somme des récompenses cumulées pour adopter une stratégie optimale.

Reward (Récompense)

Valeur numérique utilisée dans le reinforcement learning pour guider l'apprentissage. Elle renforce les comportements positifs (comme suivre la trajectoire ou augmenter la vitesse) et pénalise les comportements indésirables (comme heurter un mur ou sortir de la piste).

ROS2 (Robot Operating System 2)

Environnement logiciel open source pour la robotique, facilitant la communication entre les différentes parties d'un robot ou d'un véhicule autonome (capteurs, contrôleurs, algorithmes, actionneurs). Il a été utilisé dans le projet pour la gestion des échanges entre le contrôleur et le simulateur.

Simulation Unity

Environnement virtuel créé sous Unity reproduisant le circuit et la voiture du projet A2RL. Cet outil a été utilisé exclusivement pour les phases d'apprentissage en reinforcement learning.

Unity

Moteur de jeu 3D multiplateforme développé par Unity Technologies. Il permet de concevoir des environnements interactifs, des jeux vidéo, des simulations et des expériences immersives.

Ce stage de deux mois à l'IRSEEM (Institut de Recherche en Systèmes Électroniques Embarqués) s'est inscrit dans le cadre du projet A2RL (Abu Dhabi Autonomous Racing League), une compétition internationale de voitures autonomes de type Super Formula. Ma mission, en tant que Technicienne Robotique Systèmes Embarqués, a consisté à contribuer au développement et au test de contrôleurs autonomes en simulation. J'ai travaillé sur différentes approches de contrôle : le reinforcement learning (Unity), les méthodes classiques comme PID et Pure Pursuit, ainsi que des techniques avancées telles que le Model Predictive Control (MPC) et son extension Learning MPC (LMPC) intégrant des Control Barrier Functions (CBF).

Ce stage m'a permis de développer des compétences en programmation (C#, Python, ROS2), en optimisation de trajectoires et en simulation robotique, tout en approfondissant ma compréhension des défis liés à la conduite autonome en sport automobile. L'expérience m'a également appris à documenter méthodiquement mes essais, à collaborer dans une équipe pluridisciplinaire et à innover en explorant des solutions originales. Enfin, elle a conforté mon projet professionnel, qui vise à travailler dans le domaine des systèmes embarqués pour l'automobile.

Mots-clés : Conduite autonome, Simulation, Innovation, Intelligence artificielle, Sport automobile

This two-month internship at IRSEEM (Research Institute in Embedded Electronic Systems) was carried out within the A2RL (Abu Dhabi Autonomous Racing League) project, an international competition involving autonomous Super Formula cars. My mission as an Embedded Robotics Technician was to contribute to the development and testing of autonomous controllers in simulation. I explored several control strategies: reinforcement learning (Unity), classical approaches such as PID and Pure Pursuit, and advanced methods like Model Predictive Control (MPC) and its extension Learning MPC (LMPC) with integrated Control Barrier Functions (CBF).

This internship enabled me to strengthen my skills in programming (C#, Python, ROS2), trajectory optimization, and robotic simulation, while gaining deeper insight into the challenges of autonomous driving in motorsport. It also taught me to systematically document experiments, to collaborate effectively within a multidisciplinary team, and to foster innovation by investigating unconventional approaches. Ultimately, this experience confirmed my professional goal to specialize in embedded systems for the automotive industry.

Keywords: Autonomous driving, Simulation, Innovation, Artificial intelligence, Motorsport