
RAPPORT FINAL DE DESCRIPTION DU PIE : COURSE DE VOITURES AUTONOMES

Equipe P12B : Madeleine Becker, Gabriel Hadjerci,
Titouan Lemaitre, Ilyes Er-Rammach, Lucie Perrin,
Joao Pedro Araujo Ferreira Campos, Warren Uzbelger

Mai 2020

Table des matières

1	Contextualisation	4
1.1	Contexte	4
1.2	Livrables	4
1.2.1	Livrables matériels	4
1.2.2	Livrables documentaires	4
1.3	Cahier des charges	4
1.4	Équipe	6
1.4.1	Membres et postes	6
1.4.2	Parties prenantes	7
1.5	Mise en place de la gestion de configuration	8
1.6	Analyses des risques projet et produit	8
1.6.1	Risques projet	9
1.6.2	Risques produit	9
1.7	Matrice SWOT	10
2	Conception	12
2.1	Description fonctionnelle	12
2.2	Budget initial	14
2.3	Planification	15
2.4	Premiers objectifs et nouveaux objectifs	16
2.5	Simulation	17
2.5.1	Algorithme de guidage PID	17
2.5.2	Détermination expérimentale des paramètres de simulation	24
2.6	Structure du code de la voiture	25
2.7	Tests prévus	27
3	Réalisation	28
3.1	Montage de la voiture	28
3.2	LIDAR	28
3.3	Servomoteur de direction	29
3.4	Batteries	29
3.5	Structure interne	29
3.5.1	Structure interne provisoire	29
3.5.2	Modélisation sur Inventor	30
3.6	Site internet	30

Introduction

Pendant la deuxième année à l'ENSTA, les étudiants sont amenés à conduire un Projet d'Ingénieur en Equipe (PIE), de façon à mettre en oeuvre les connaissances théoriques acquises pour le développement et la conception d'un livrable. Ce livrable peut être matériel (un robot, une voiture, un outil) ou non (une solution pour un problème industriel, un programme, un modèle d'une situation réelle). L'objectif est donc de faire travailler les élèves en équipe, dans des groupes hétérogènes du point de vue des formations et des origines de chaque membre, ce qui simule le cadre d'un projet tel qu'on peut le trouver dans l'industrie. Ce rapport vise à documenter l'expérience de l'équipe 12B, responsable du projet de création d'une voiture autonome et participation à une course inter-écoles.

Sont documentés les aspects techniques et d'organisation de l'équipe, les outils utilisés, la démarche ingénierie système, ainsi que les résultats obtenus et les difficultés rencontrées pendant ce projet.

1 Contextualisation

1.1 Contexte

Le PIE a pour but de confronter les élèves à une situation de développement d'un projet qui prend plusieurs mois, ce qui leur permet de se familiariser avec le travail d'ingénieur en entreprise, et de mettre en oeuvre des outils de gestion de projet qui ont été vus en cours théorique.

Dans le cas du projet décrit dans ce rapport, le domaine est la création d'un véhicule autonome à l'échelle 1/10. Le projet consiste alors en le développement d'un software de suivi de trajectoire, montage de la voiture, ainsi qu'une analyse d'ingénierie système pour faire l'intégration entre les différentes parties de hardware et software.

1.2 Livrables

1.2.1 Livrables matériels

Le principal produit de ce projet est la voiture autonome, assemblée et capable de rouler conformément au cahier de charges. C'est le livrable matériel qui est la raison finale de notre projet.

En parallèle, nous devons aussi produire un livrable de support de communication. Il s'agit d'un site internet, dont le but est de documenter et exposer à toute personne intéressée les aspects plus importants du projet, tels que la composition de l'équipe, son avancement, quels ont été les problèmes rencontrés pendant ce temps et d'autres informations qui puissent être utiles dans la compréhension de notre travail.

1.2.2 Livrables documentaires

Pour faciliter la conception des livrables matériels, le projet nécessite aussi plusieurs livrables documentaires, conçus au fur et à mesure que le projet avance, ayant pour but de guider l'équipe présente ainsi que de fournir les bases pour que la prochaine équipe puisse continuer le projet. Il s'agit d'un ensemble de documents et d'outils numériques qui regroupent non seulement les informations générales d'organisation du projet, mais aussi les informations techniques. On y trouvera notamment les codes source des programmes développés, les comptes-rendus des réunions et les documents de gestion de projet suivant les démarches d'ingénierie système. Nous espérons que ces documents seront suffisamment clairs pour que l'équipe de l'année prochaine soit capable de reprendre le projet dans son état actuel, de façon à le continuer et aboutir à l'objectif final qui est de gagner la course de voitures autonomes dans les règles. Chacun de ces livrables documentaires est décrit en détails dans la section [1.5](#).

1.3 Cahier des charges

Notre projet a pour particularité d'être intégré dans un contexte précis puisqu'il se place dans le cadre de la course de voitures autonomes. On peut voir ça comme un

inconvenient car ça ne laisse pas beaucoup de place à notre créativité. Cependant, c'est aussi un avantage. En effet, à l'instar de certains autres groupes, notre objectif est clairement défini et il est donc facile d'extraire du règlement de la course un cahier des charges.

L'objectif est simple : « gagner la course ». Ce qui peut se traduire en « franchir la ligne d'arrivée en premier, tout en respectant les contraintes imposées par le règlement ». On peut donc découper notre cahier des charges en deux types d'exigences, les exigences de respect du règlement et les exigences de performances.

Dans le cas des contraintes relatives au règlement, elles sont déjà formulées clairement et nous n'avons pas à ajouter de modifications.

En ce qui concerne les exigences de performance, on peut, rien qu'en analysant les détails de la course, les définir assez clairement sans même stipuler sur les choix techniques qui vont être utilisés. En effet, certains choix des organisateurs imposent des contraintes implicites que nous sommes obligés de suivre si nous voulons que la voiture ait une chance de gagner. Par exemple, la forme du circuit n'est pas connue à l'avance, il faut donc que la voiture soit capable de détecter le périmètre du circuit. Ce n'est pas obligatoire, mais sans ça il est impossible de gagner. On peut donc dresser une liste de critères de performance, tout en essayant de prendre en compte les meilleures stratégies. Il ne s'agit pas simplement de faire le minimum requis pour finir la course, mais bien évidemment de gagner.

Liste des critères liés au règlement :

- Le châssis utilisé doit être celui du kit Tamiva TT02.
- Le véhicule doit être commandé par un programme autonome.
- Le véhicule doit s'arrêter si elle perd la connexion avec l'équipe pendant plus de deux secondes.
- Le véhicule doit utiliser une batterie NiMH 7,2V 3000 mAh maximum.
- Le véhicule ne doit pas avoir un comportement agressif envers les autres véhicules.

Liste des critères de performance :

- Le véhicule doit avoir une autonomie suffisante pour fonctionner durant toute la course.
- Le véhicule doit détecter les obstacles, les bordures et les véhicules adverses afin de les éviter.
- Le véhicule doit suivre une trajectoire optimale jusqu'à la ligne d'arrivée.
- Le véhicule doit se déplacer le plus vite possible tout en respectant les autres critères.

Nous nous sommes basés sur ce cahier des charges pour toute la conception de la voiture autonome.

1.4 Équipe

1.4.1 Membres et postes

Nous sommes une équipe de huit étudiants, d'origines et de filières différentes, ce qui nous permet de disposer de compétences variées et complémentaires à exploiter pour mener à bien ce projet.

- **Madeleine Becker**, filière STIC : Chef de Projet et Responsable des achats et des relations experts
- **Lucie Perrin**, filière STIC : Secrétaire, Responsable Data et Qualité
- **Joao Pedro Araujo Ferreira Campos**, filière STIC et AST Brésilien : Conseiller Prévention
- **Ziqi Xing**, filière STIC et AST Chinois : Coéquipier responsable achat et gestion du matériel
- **Titouan Lemaitre**, filière mécanique : Responsable Simulation et Montage, coéquipier responsable trajectoire
- **Gabriel Hadjerci**, filière mécanique : Responsable Communication
- **Warren Uzbelger**, filière Maths Appliquées : Responsable trajectoire, coéquipier responsable simulation
- **Ilyes El-Rammach**, filière Maths Appliquées : Coéquipier responsable data

Les rôles ont été choisis selon les compétences et les appétences de chacun pour les différentes tâches. Certaines tâches sont effectuées avec l'aide d'autres membres que ceux désignés d'office, en fonction de la charge de travail de chaque membre à tout instant donné.

Pour assurer le bon déroulement de la réalisation du projet et favoriser la meilleure exploitation des compétences de tous les membres du groupe, nous avons défini des postes avec différentes tâches attribuées, que nous nous sommes ensuite réparties.

- **Chef de Projet :**
 - Assurer le bon fonctionnement de l'équipe et l'avancement du projet
 - Planifier et animer les réunions qui concernent l'équipe au complet
- **Secrétaire :**
 - Maintenir un journal de bord de chaque séance
 - Rédiger le compte rendu des réunions
 - Reporter les problèmes rencontrés
 - Identifier le travail effectif de chaque coéquipier
- **Conseiller Prévention :**
 - Veiller à faire respecter les consignes de sécurité
 - Identifier et analyser les sources potentielles de nuisance à la santé et/ou à la sécurité
- **Relation Expert :**
 - Assurer la correspondance avec l'expert
 - Assister aux réunions avec l'expert
- **Responsable Data :**
 - Gérer l'accès et la maintenance de la plateforme collaborative

- Gérer le processus de validation des données
- **Responsable Achats et Gestion du matériel :**
 - Identifier les besoins en matériel
 - Choisir les fournisseurs en tenant compte des prix et des délais
 - Procéder aux commandes
 - Assurer le suivi des commandes
 - Tenir les comptes
- **Responsable Qualité :**
 - Vérifier la bonne tenue du Wiki
 - Vérifier que les codes sont bien commentés et compréhensibles pour les membres du projet comme des personnes extérieures au projet (notamment pour la passation)
- **Responsable Communication :**
 - Créer et gérer le site web du projet
 - Gérer la promotion de notre projet
- **Responsable Simulation :**
 - Développement de l'algorithme de calcul de la trajectoire optimale
 - Développement d'un simulateur physique permettant de simuler le comportement dynamique de la voiture
 - Développement d'un simulateur physique permettant de simuler les signaux émis par les capteurs en fonction de l'environnement du véhicule
 - Développement d'un logiciel de test des algorithmes de calcul de trajectoire à partir des différents simulateurs physiques
- **Responsable Montage :**
 - Montage de la voiture autonome
 - Vérification de la bonne tenue des différents éléments de la voiture entre eux
- **Responsable Trajectoire :**
 - Calcul de la trajectoire optimale
 - Motricité de la voiture
 - Détermination des caractéristiques dynamiques de la voiture

1.4.2 Parties prenantes

Une partie prenante désigne « *tout acteur (individu, organisation, groupe) concerné par un projet, une décision ou action, c'est-à-dire dont les intérêts sont affectés d'une façon ou d'une autre par sa mise en place* ». On peut diviser les parties prenantes en deux catégories : les parties prenantes internes et externes.

Dans le cadre de ce projet, les parties prenantes internes sont :

- les dirigeants de ce projet : l'ENSTA,
- les personnes le réalisant : notre équipe,
- nos collaborateurs, qui sont ici les experts dont on reçoit de l'aide :
 - Gabriel Ballet qui nous encadre dans la réalisation,
 - Eric Fenaux qui nous donne des conseils sur le véhicule,
 - Severine Bournaux qui nous guide pour la gestion de projet.

Les parties prenantes externes sont les clients, ici les organisateurs de la course (Ecole Normale Supérieure de Paris-Saclay), et les fournisseurs de chacun de nos composants : Conrad en ce qui concerne le kit de montage de la voiture, la Raspberry Pi et son boîtier adapté, et Robotshop pour le LIDAR.

1.5 Mise en place de la gestion de configuration

Dès le début de notre projet, nous avons mis en place la gestion de configuration grâce à la plateforme Gitlab. Sur cette plateforme nous stockons :

- Les documents de **définition de la mission**, et notamment le règlement de la course et le cahier des charges tels que nous l'avons défini, en se fondant sur le règlement.
- Les **compte-rendus** de toutes les réunions que nous faisons. Dans ces compte-rendus sont notamment détaillées notre démarche, et les décisions que nous prenons au cours du projet. À chaque séance, nous détaillons les objectifs du jour, ainsi que les avancées notables et les raisons qui justifient nos décisions. Il y a également un compte-rendu du cours de dynamique du véhicule donné par M. Éric Fenaux.
- Les documents de **gestion de projet** et de **gestion de configuration**. On trouve notamment tous les diagrammes créés au cours du projet : FAST, GANTT, matrice SWOT, WBS, gestion des risques.
- La documentation et les **spécifications** que nous trouvons sur les éléments que nous utilisons, notamment le châssis, le LIDAR etc.

Sur le Gitlab, nous pouvons également rassembler tout le code que nous écrivons au cours du projet.

Pour faciliter la prise en main de notre plateforme, nous avons rédigé un document d'aide (Readme) qui détaille comment est organisé le Wiki, comment sont agencés les codes et à quoi ils servent.

Lorsque le diagramme FAST a été terminé, nous avons une meilleure idée de la voiture que nous voulions. Nous avons donc pu séparer les tâches en différents pôles, et détailler ces tâches. Nous avons alors accompagné notre Gitlab d'un compte Trello, où nous détaillons les tâches à réaliser, qui les réalise et combien de temps nous estimons que leur réalisation va durer. Cependant, la pratique a montré que cette nouvelle plateforme dispersait les mêmes informations à trop d'endroits différents. Nous n'avons donc pas vraiment utilisé le Trello. À la place, la gestion de l'avancement s'est faite à l'aide des compte-rendus de l'activité appuyés par le GANTT, qui sont tous sur le Gitlab. Cette solution nous convient parce que l'ampleur du projet est raisonnable : nous pouvons facilement réunir toute l'équipe et suivre l'avancement de chacun. La hiérarchie des responsabilités est assez horizontale, et le nombre restreint de membres de l'équipe permet la motivation et l'implication de tous.

1.6 Analyses des risques projet et produit

Les dangers que nous pouvons rencontrer à travers ce projet peuvent être séparés en deux types : les risques projet, qui pèsent sur la réalisation en elle-même de notre

projet, et les risques produit, qui sont les risques encourus lors de la manipulation du véhicule.

Tous les risques sont cependant catégorisés avec la même échelle : la multiplication de la gravité G (entre 1 et 100), la fréquence d'exposition F (entre 1 et 14) et le niveau de maîtrise M (entre 0.2 et 1) donne un nombre qui détermine le niveau de risque selon la hiérarchisation détaillée ci-après.

Hiérarchisation

Résultat	Niveau de risque	Capacité d'action
+ 600	Risque très important	Action immédiate
350 à 600	Risque important	Action à planifier
150 à 350	Risque modéré	Action possible
- 150	Risque faible	Maintenir la vigilance

FIGURE 1 – Hiérarchisation des risques

1.6.1 Risques projet

Les risques qui pèsent sur la réalisation du projet sont :

- **Beaucoup d'apprentissage** $G = 30$, $F = 6$, $M = 1$: Un grand nombre d'aspects du projet nous sont inconnus. Ils nécessitent donc un apprentissage, et rendent la planification du projet difficile. Malheureusement, nous n'avons pas de moyen d'action contre ce risque.
- **Charge mentale** $G = 50$, $F = 9$, $M = 1$: La charge de travail demandée par l'ENSTA est variable et assez peu prévisible, ce qui peut engendrer des difficultés à tenir tous les projets. Pour cela, une planification précise, régulière et en amont est nécessaire.
- **Blessures, coupures, projections de débris** $G = 50$, $F = 6$, $M = 0.5$: les différentes manipulations d'outils lors de la réalisation du projet peuvent entraîner différentes blessures. Pour pallier à ce risque, le strict respect des consignes de sécurité est obligatoire.

1.6.2 Risques produit

Les risques produits sont eux de natures différentes :

- **Risques électriques** $G = 30$, $F = 1$, $M = 0.5$: L'utilisation de matériel électronique et la puissance des batteries font que des intensités non négligeables peuvent circuler. De plus, il faut prendre garde aux courts-circuits accidentels qui peuvent provoquer une surchauffe. Pour prévenir ce risque, une manipulation attentive du matériel est primordiale.
- **Risques de surchauffe** $G = 30$, $F = 1$, $M = 0.5$: les batteries utilisées pour l'alimentation nécessite une manipulation attentive parce qu'elles peuvent être sujettes aux surchauffes. Le respect attentif des consignes de bonne utilisation est donc essentiel.

- **Risque de panne de composant électronique, de batterie ou de mécanique** $G = 30$, $F = 4$, $M = 0.8$: l'ensemble du véhicule est constitué de matériel délicat qui peut être sujet à des pannes. Cependant, notre budget ne nous permet pas d'avoir des pièces de rechange, on ne peut donc pas vraiment prémunir ce risque.
- **Risque de dommages par collision avec un obstacle fixe ou mouvant** $G = 50$, $F = 4$, $M = 0.2$: une erreur de trajectoire de notre voiture ou d'un adversaire peut causer une collision, ce qui peut entraîner des dommages graves sur notre véhicule. La prévention de ces risques est faite par une conception méticuleuse de l'algorithme de guidage de la voiture.
- **Risque de demi-tour** $G = 10$, $F = 4$, $M = 0.8$: la conception actuelle du programme fait que la voiture peut accidentellement faire demi-tour sur la piste. Elle fait alors la course à l'envers, sans s'en rendre compte, augmentant drastiquement les chances de collision avec un adversaire. La détection du fait qu'on roule à l'envers est extrêmement compliquée et donc coûteuse à ajouter. Cependant, le règlement de la course autorise les participants à replacer leur véhicule dans ce cas là, donc nous avons choisi de ne pas adresser ce risque.

1.7 Matrice SWOT

Pour accompagner la définition des pôles et des postes, nous avons réalisé une matrice SWOT. Elle nous permet de voir quels sont les éléments pouvant représenter un frein ou une menace à notre avancement ou à la réussite de notre projet, ou bien à l'inverse les éléments sur lesquels nous pouvons nous appuyer pour mener à bien notre travail.



FIGURE 2 – Matrice SWOT

En prenant en compte en particulier les faiblesses et les menaces auxquelles nous devons faire face, nous avons tenté de mettre en place des actions pour y pallier. Tout

d'abord, nos faiblesses s'estompent au fur et à mesure de l'avancement du projet, car la course n'aura finalement de toute façon pas lieu en raison de la crise actuelle, et nos connaissances dans le sujet se développent de plus en plus. Concernant les menaces, le financement, bien que tardif, a bien été effectué pour les premières pièces. Enfin, les derniers éléments de la catégorie MENACES n'ont plus réellement de sens en raison de l'épidémie que nous traversons actuellement.

2 Conception

2.1 Description fonctionnelle

La première fonction principale que doit assurer la voiture est de gagner la course. Elle doit donc dans un premier temps être en mesure de calculer et de suivre une trajectoire la plus efficace possible sans toucher les murs tout en esquivant les concurrents et les éventuels obstacles. La voiture sera donc équipée d'algorithmes permettant de calculer la meilleure trajectoire à suivre en fonction de ses paramètres (position, vitesse et niveau d'énergie restant) et en fonction de son entourage qu'elle doit donc pouvoir détecter. Une fois la trajectoire calculée, elle doit être en mesure de la suivre en un temps minime, ce qui sera assuré par des actionneurs soigneusement étudiés et directement soumis aux algorithmes. Dans un deuxième temps, toujours en vue de favoriser la victoire de la course, la voiture doit être capable de gérer l'énergie dont elle dispose. Pour ça, la capacité disponible est donnée par une batterie imposée pour la course, et l'affichage et la prise en compte du niveau d'énergie restant seront assurés par des algorithmes implémentés à cet effet.

La deuxième fonction principale est d'assurer la sécurité des concurrents en accord avec les règles de la course. Il faut alors d'une part s'assurer que la voiture garde un comportement acceptable et soit sécurisée dans le sens où les court-circuits et les erreurs de montage doivent absolument être évités. On intègre alors des cas d'erreurs dans nos algorithmes pour détecter tout comportement abusif, et on procède au montage dans le plus grand des soins et avec l'aide des experts.

Voici les diagrammes FAST apportant plus de précisions à notre analyse fonctionnelle (c.f. partie [cachier des charges](#)) :

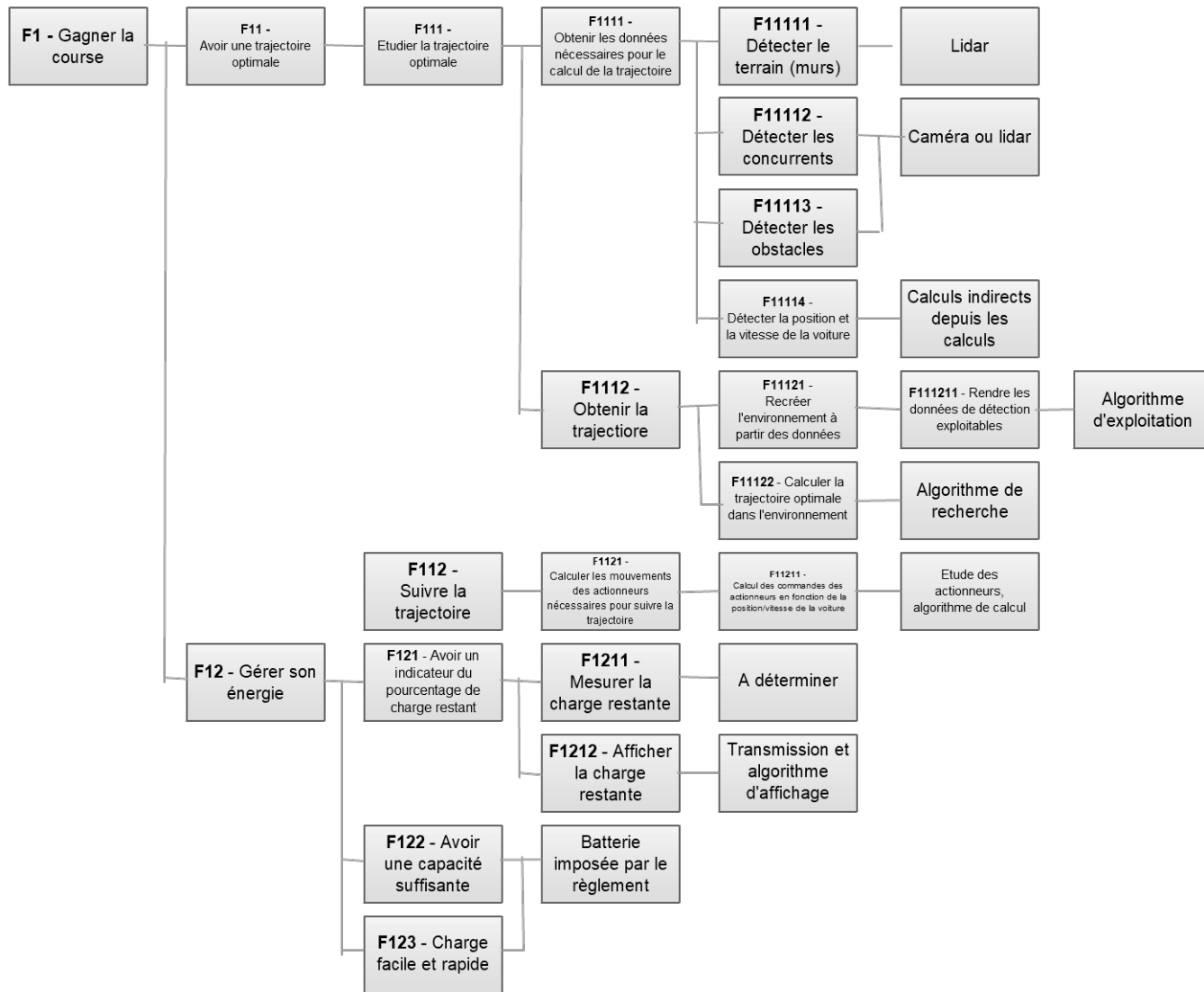


FIGURE 3 – Diagramme FAST (première fonction principale)

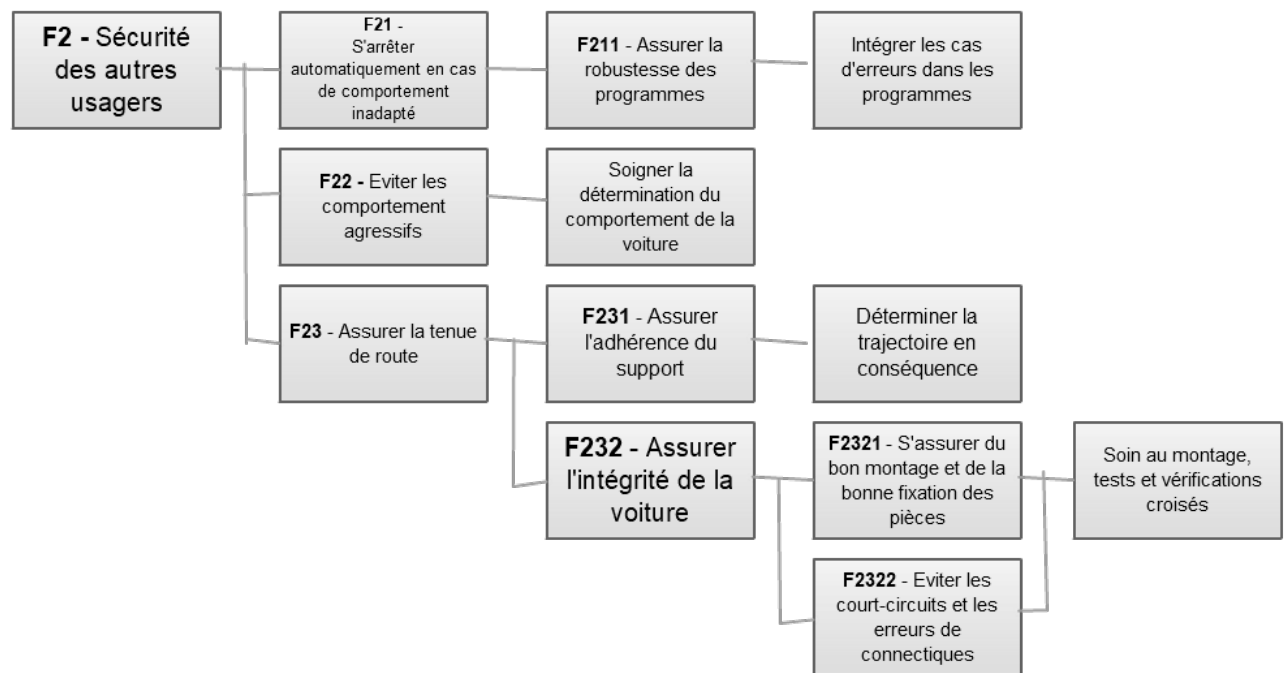


FIGURE 4 – Diagramme FAST (deuxième fonction principale)

2.2 Budget initial

Le budget initial estimé était d'environ 1000 €. Les éléments que nous étions certains de devoir acheter étaient :

- Le châssis de notre voiture, qui est imposé par le règlement de la course. Son prix unitaire est de 169 €.
- Un LIDAR est également imposé, pour détecter les bord du circuit ainsi que les adversaire et/ou obstacles. Son prix unitaire est de 334 €.
- La batterie, non fournie, pour alimenter les moteurs. Son prix unitaire est d'environ 40 €.
- Un servomoteur, non fourni non plus, pour pouvoir contrôler la direction de notre voiture. Son prix est d'environ 10 €.
- Une carte de programmation. Note choix s'est porté sur une RaspberryPi 4B pour ses performances, sa facilité d'utilisation et sa large communauté en ligne, auprès de laquelle nous pouvons trouver de l'aide en cas de problème. Son prix unitaire est de 74,99 €.
- Une boîte pour protéger la carte, dont le prix est de 4,33 €.
- Un câble d'alimentation pour alimenter la carte, dont le prix est d'environ 10 €.

À ces éléments déjà prévus s'est rajoutée une carte d'interface de puissance entre les différents composants, d'une valeur de 155,53 €. Cela monte le total de nos achats à 797,85 €, ce qui est toujours dans le budget que nous avons estimé.

2.3 Planification

Fin décembre nous avons terminé les premières parties de conception générale de la voiture, et nous avons pu nous lancer dans la conception des composantes principales. En janvier nous avons reçu notre première commande de matériel, donc nous avons réalisé un nouveau GANTT. En effet, il nous était impossible auparavant d'avoir de la visibilité sur la disponibilité du matériel, donc la majorité des tâches n'étaient pas planifiables.

La dernière grosse mise à jour date du 28 février, soit très peu de séances avant le confinement.

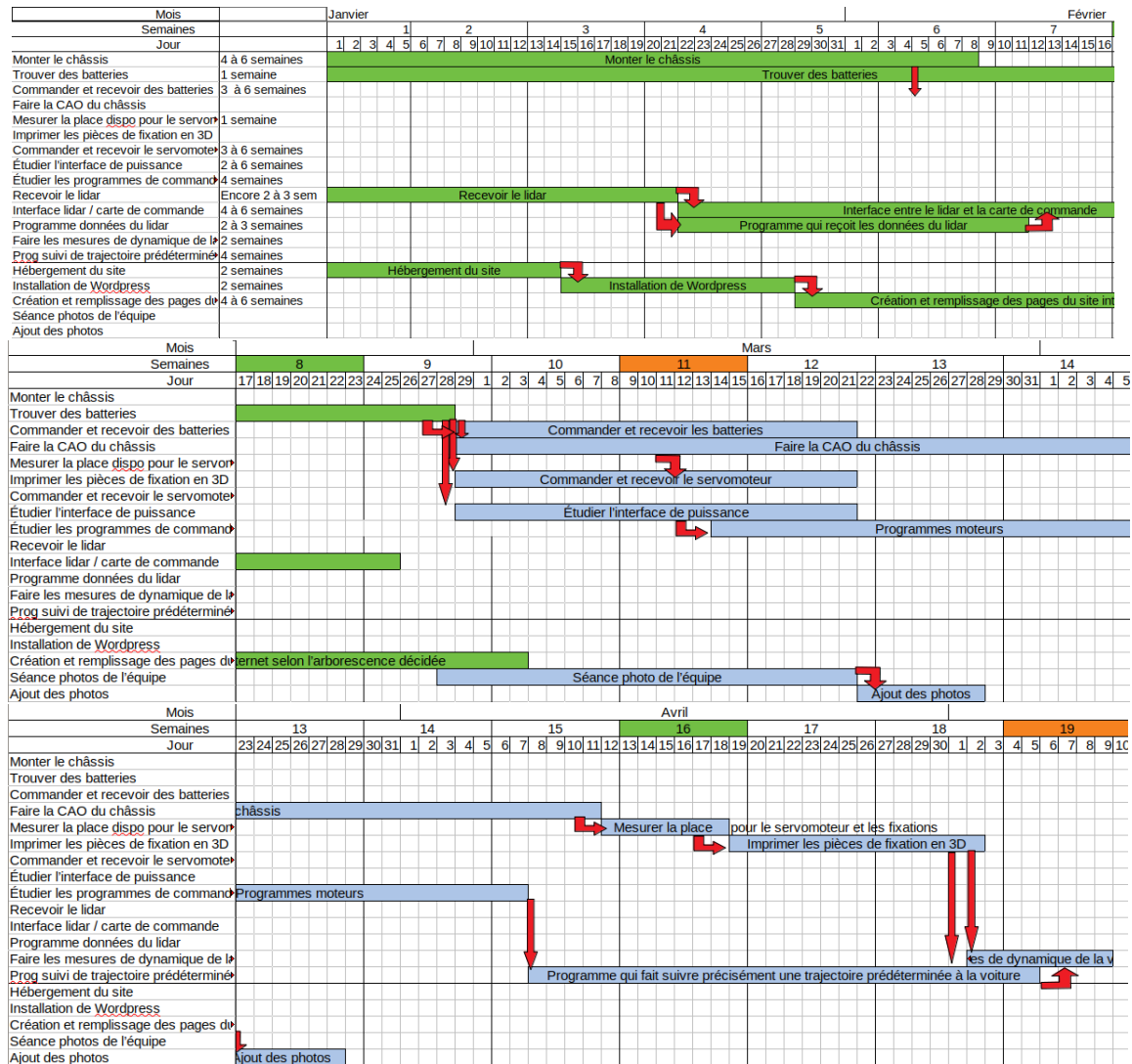


FIGURE 5 – Diagramme de GANTT au 28 février 2020

La plupart des tâches qui étaient prévues avant cette date ont mis plus de temps que prévu à être réalisées, ce qui nous a amenés à revoir la durée estimée pour les tâches suivantes. Lors de la mise à jour de ce GANTT nous nous sommes rendus compte que nous étions beaucoup plus contraints par le temps que ce que nous avions estimé au préalable. En effet, une des tâches majeures nécessaire à plein de

tâches qui en découlaient était la création du support à l'intérieur de la voiture. En effet, pour pouvoir monter la voiture et donc faire des tests par exemple de guidage moteur ou de direction, il faut que la voiture soit presque entièrement montée. Pour ce faire, une structure à l'intérieur de la voiture est nécessaire, non seulement pour permettre à tous les éléments de tenir dans la voiture mais aussi pour éviter des problèmes mécaniques graves, par exemple que des fils d'alimentation ou de connexion s'emmêlent dans l'arbre moteur. Or, pour cette structure interne nous avons opté pour l'impression 3D pour pouvoir l'adapter le plus possible à nos besoins. Mais pour cela, il nous faut la modéliser et modéliser le châssis sur ordinateur, ce qui est long et compliqué. Pour pallier à ce problème et pouvoir paralléliser les tâches, nous avons décidé d'opter pour une structure provisoire, pas idéale mais beaucoup moins longue à fabriquer. Cette structure est détaillée dans la partie 3.5.1. Malheureusement, le confinement est intervenu avant que nous ne puissions l'essayer.

2.4 Premiers objectifs et nouveaux objectifs

Au cours du projet, nous avons été amenés à revoir nos objectifs, et ce pour deux raisons principales. D'une part, le projet est assez long et est prévu pour être fait sur plusieurs années. Nous avons donc voulu nous concentrer sur les aspects basiques qui permettent à la voiture de rouler et de se déplacer dans un circuit, avant de s'intéresser aux aspects de performance qui pourront être étudiés les années suivantes. D'autre part, durant l'année nous avons subi plusieurs contretemps plus ou moins dérangeants suivant la situation. Peut-être que nous avons été trop ambitieux au départ.

Ceci étant dit, nous avons recentré nos objectifs sur des points plus abordables. Au départ nous avions prévu de réaliser la voiture au complet, avec en bonus des systèmes optionnels comme une caméra pour améliorer la détection des voitures adverses. Nous avons donc revu nos objectifs à la baisse une première fois, par manque de temps, mais surtout car les pièces que nous attendions comme le châssis ou le LIDAR sont arrivés extrêmement en retard. Pour éviter de perdre du temps, nous nous sommes concentrés sur la simulation et avons axé notre travail sur ce qui pouvait être fait sans matériel.

La situation sanitaire de fin d'année était difficilement prévisible et même les objectifs revus à la baisse une première fois sont devenus trop compliqués à réaliser. La principale raison est que le matériel s'est retrouvé éparpillé voire inaccessible à cause du confinement. Les aspects mécaniques que nous avions déjà repoussés une première fois à cause du délai de livraison, se sont retrouvés une nouvelle fois infaisables. Comme nous étions déjà bien avancés dans la partie programmation, nous nous sommes concentrés sur l'amélioration de ce que nous avions déjà produit. La simulation par exemple a été améliorée, et l'algorithme de détermination de trajectoire est désormais plus performant. Nous avons aussi pu finir le site internet. Les objectifs et les livrables que nous avions prévu de rendre en fin d'année ne sont donc plus les mêmes pour des raisons évidentes.

2.5 Simulation

Afin de mieux comprendre la dynamique de la voiture, nous avons conçu un programme (en langage Python) qui simule le suivi de trajectoire. Son fonctionnement s'appuie sur la modélisation de l'environnement de la voiture à partir de ce qui est détecté par le LIDAR, puis sur un choix de trajectoire dans cet environnement modélisé.

2.5.1 Algorithme de guidage PID

Récréer l'environnement

Dans un premier temps, il nous faut recréer un circuit virtuel 2D dans lequel on veut que notre voiture évolue, on s'inspire pour dessiner ce circuit des circuits utilisées pour les compétitions des années précédentes. Il faut ensuite simuler ce que "voit" la voiture grâce au LIDAR en fonction de sa position sur le circuit. Enfin il faut définir la zone où la voiture "peut se trouver" à partir des données simulées par le LIDAR. C'est-à-dire la zone dont le mur le plus proche est à une distance supérieure au rayon du cercle circonscrit au rectangle qui représente notre voiture.

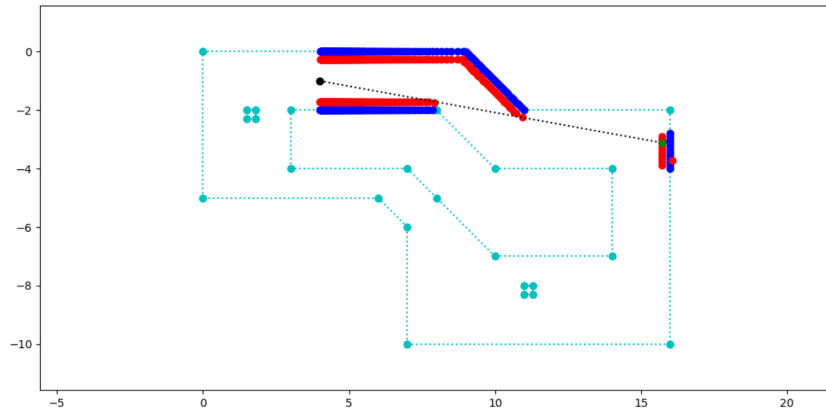


FIGURE 6 – Illustration de la simulation de l'environnement de la voiture par le code.

Légende :

- Les traits pointillés bleu clair représentent les limites du circuit.
- Le point noir représente la position actuelle de la voiture.
- Les points bleu foncé sont les points relevés par le LIDAR, c'est ce que "voit" la voiture.
- Les points rouges délimitent la zone où la voiture peut aller d'après les données recueillies par le LIDAR.
- Le point vert est le point visé parmi les points rouges, comme détaillé ci-après.

Choisir la direction à viser (fonction cible)

L'environnement de la voitures est recréé, et la voiture est capable de percevoir cet environnement et où elle peut aller. Il faut désormais programmer la voiture pour

qu'elle puisse déterminer de manière autonome où elle doit aller. On définit ce point à viser comme le point le plus lointain pouvant être atteint par la voiture en ligne droite sans sortir de la zone où elle peut aller (points rouges) et se trouvant dans le demi-plan devant la voiture (lui-même illustré en figure 7). Ce point « cible » est illustré en vert sur la figure 6.

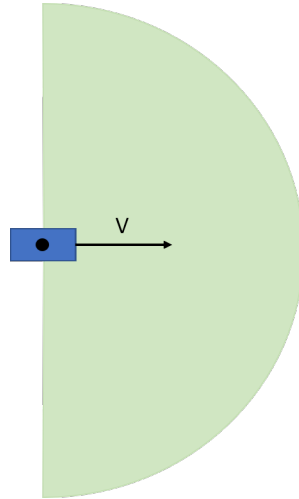


FIGURE 7 – Demi plan devant la voiture

Calculer les commandes pour se ramener à la trajectoire

Le vecteur vitesse de la voiture n'est généralement pas colinéaire avec le vecteur qui détermine la direction à viser. De plus la voiture ne peut se déplacer que selon des arcs de cercle dont le rayon minimal est déterminé par l'accélération latérale maximale que peuvent supporter les pneumatiques et la vitesse actuelle de déplacement. Il nous faut donc déterminer une série de commandes de directions et d'accélérateur/frein qui nous permettent de ramener la voiture dans la direction souhaitée

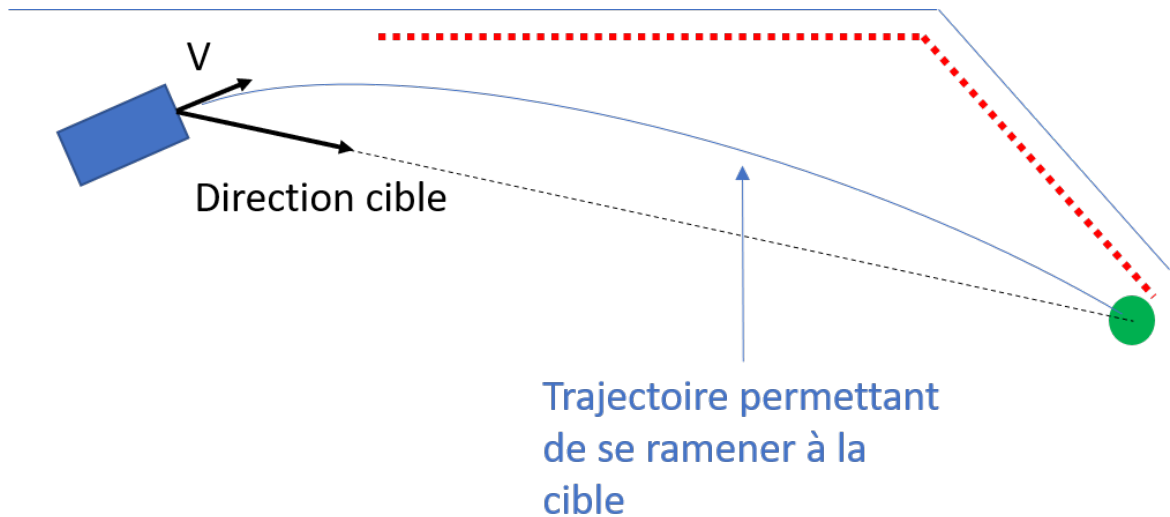


FIGURE 8 – Illustration de la non colinéarité de V et de la direction cible

Première approche : Commande PID

- On calcule l'angle θ entre le vecteur direction de cible et le vecteur vitesse
- On en déduit l'angle au volant par un asservissement avec un terme proportionnel à θ et un terme proportionnel à la dérivée de θ (on calcule $\theta(t + dt)$ en considérant que la voiture s'est déplacée en ligne droite pendant dt selon son vecteur vitesse actuel)
- On calcule le rayon de courbure nécessaire pour rejoindre cible, on calcule la vitesse maximale à laquelle ce rayon de courbure peut être pris en fonction de l'accélération latérale maximale de la voiture on en déduit l'accélération (variation de vitesse) positive ou négative nécessaire pour se ramener à la vitesse souhaitée et on en déduit la commande d'accélération

Cette première approche assez simple fonctionne de manière satisfaisante dans une grande majorité des situations rencontrées lorsque la voiture est seule sur la piste, à condition que les coefficients devant le terme proportionnel et dérivatif soient bien réglés.

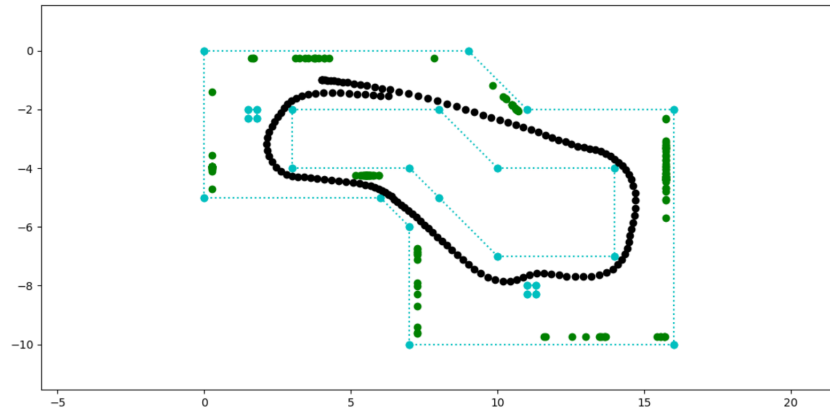


FIGURE 9 – Trajectoire obtenue pour la voiture seule avec un algorithme de guidage PID

Cependant dans certaines situations extrêmes comme des virages très serrés, des mouvements rapides d'une voiture concurrente devant notre voiture ou la présence d'un obstacle qui réduit la largeur de la piste, cet algorithme s'avère peu fiable : les sorties de pistes sont fréquentes.

Simulateur physique et trajectoire

Notre voiture est à présent capable de reconnaître son environnement, de choisir la cible qu'elle doit viser et de fournir les commandes permettant de se ramener à cette cible en fonction de sa vitesse et de son environnement.

À partir des commandes fournies, le programme calcule la position et la vitesse effectivement atteintes par la voiture au pas de temps suivant (dans notre cas $T = 0.1s$). Puis relance une acquisition par le LIDAR de l'environnement en fonction de la nouvelle position et orientation spatiale. Le programme génère alors un nouveau point cible et les commandes volant et accélérateur pour tenter de s'y ramener.

En répétant ce procédé P fois dans une boucle de calcul, on produit la trajectoire de la voiture guidée par notre algorithme.

Pour simuler la course de manière plus réaliste, on peut simuler deux voitures tournant sur la même piste et ainsi observer comment elles réagissent aux comportements imprévisibles de concurrents.

Affichage de la simulation comme animation vidéo

Pour mieux évaluer la qualité de notre programme de guidage, il est avantageux de mettre en place un système de rendu vidéo qui transforme la trajectoire calculée en animation vidéo. Ceci permet en effet de remarquer facilement des comportements anormaux (qu'un conducteur humain supposé optimal n'aurait pas) ainsi que de repérer les événements qui conduisent à des sorties de pistes.

Cet outil précieux permet de faire progresser l'algorithme de guidage, que ce soit le choix de cible où le calcul de commandes volant/accélérateur pour essayer de s'y ramener.

Paramétrage

Notre simulation dépend évidemment de nombreux paramètres physiques, qui sont souvent mal connus ou susceptibles d'évoluer. Par exemple l'accélération latérale maximale tolérée par les pneumatiques dépend fortement du sol sur lequel la course se déroule. Pour pouvoir tester notre programme de guidage dans une grande variété de conditions et assurer la cohérence des paramètres dans l'ensemble du code, ces paramètres sont définis dans une fonction à part appelée *paramètres* et peuvent y être modifiés facilement et à tout moment.

Deuxième approche : Commande prédictive

Bien que le programme de guidage PID s'avère suffisant dans certaines situations, le fait qu'il ne tienne pas compte de la trajectoire obtenue en fonction des commandes de volant et d'accélérateur pour calculer ces mêmes commandes volant et accélérateur conduit à des commandes inappropriées à la situation, qui entraînent des sorties de route.

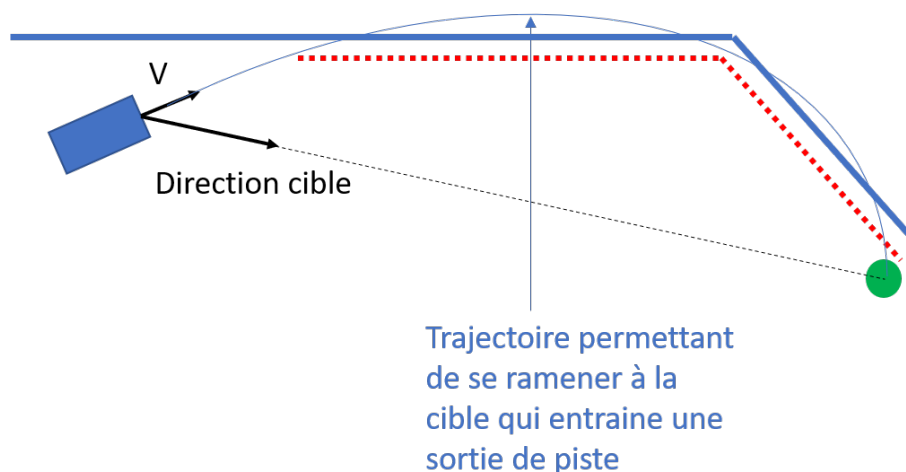


FIGURE 10 – Exemple de sortie de route engendrée par l'algorithme de guidage PID

Pour essayer de remédier à cela, on implémente l'algorithme suivant :

- De la même manière que dans le programme précédent, on choisit le point « cible » comme le point le plus lointain pouvant être atteint par la voiture en ligne droite sans sortir de la zone où elle peut aller (points rouges) et se trouvant dans le demi plan devant la voiture.
- On simule la trajectoire que produisent les commandes « tourner à fond vers la cible et freiner à fond ».
- Si cette trajectoire intersecte le segment reliant la voiture au point cible, avant d'intersecter l'un des segments qui constituent la limite de la zone où la voiture peut se trouver, alors la cible est déclarée atteignable et le programme continue. Sinon, la cible est déclarée inatteignable et on recommence les étapes précédentes avec la cible suivante : le 2ème point le plus lointain pouvant être atteint par la voiture en ligne droite sans sortir de la zone où elle

- peut aller (points rouges) et se trouvant dans le demi plan devant la voiture.
- On répète l'étape précédente jusqu'à obtenir une cible atteignable.

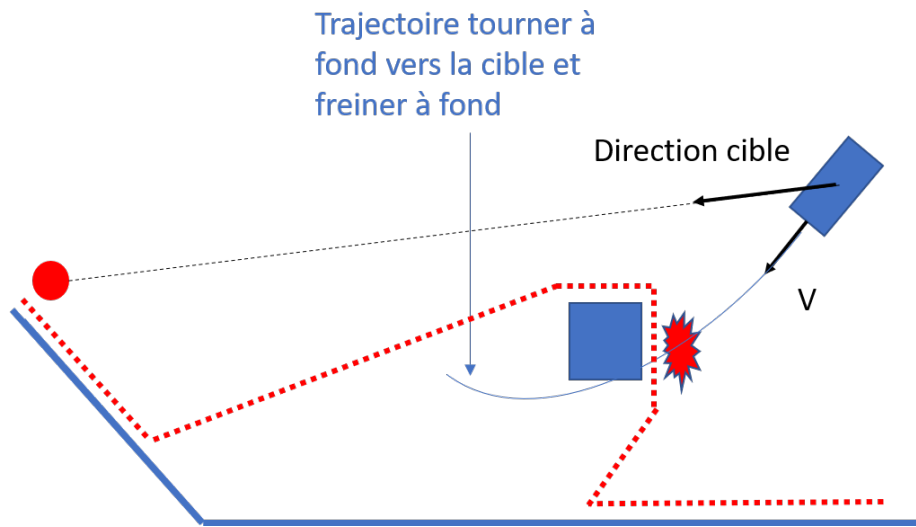


FIGURE 11 – Exemple de cible inatteignable

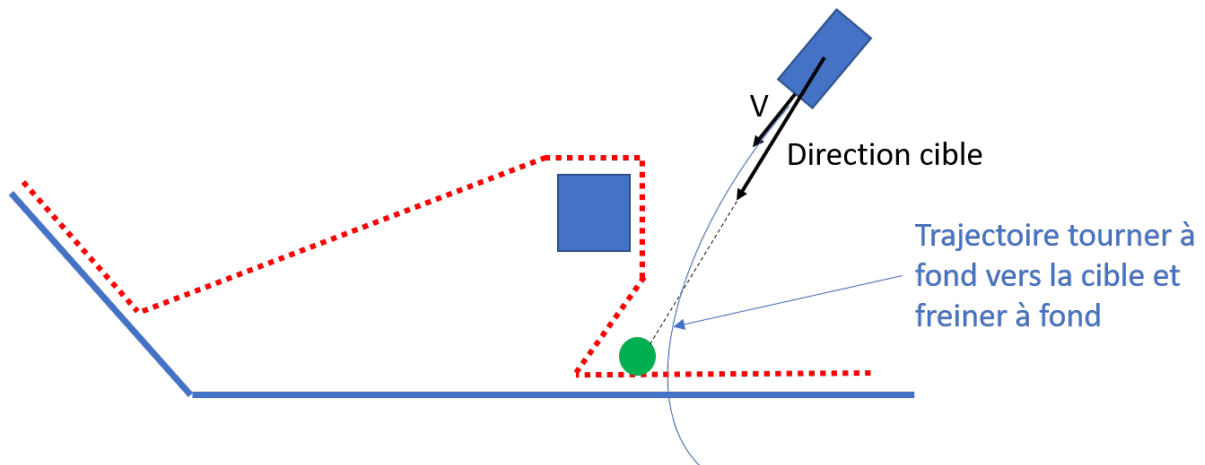


FIGURE 12 – Exemple de cible atteignable

Comme illustré ci dessus, cette stratégie de sélection de cible permet au programme de prendre en compte les capacités motrices de la voiture. Dans l'exemple de la figure 11, la voiture va trop vite pour pouvoir passer au dessus de l'obstacle. La meilleure trajectoire d'évitement de l'obstacle par le dessus est la trajectoire résultant des commandes « braquer à fond vers la cible et freiner à fond ». Elle conduit cependant à un choc avec l'obstacle. Toutes les cibles « au dessus » de l'obstacle sont donc inatteignables. Cela est pris en compte par le programme qui sélectionne une cible en dessous de l'obstacle qui lui permettra de l'éviter. On sait que la cible choisie dans la figure 12 est atteignable, il faut maintenant

déterminer les commandes les plus appropriées pour rejoindre cette cible. Pour ce faire on réalise la succession d'étapes suivantes :

- À partir de la vitesse de la voiture, de l'accélération latérale maximale tolérée par les pneus, de l'angle de braquage maximal des roues et de la longueur de la voiture, on calcule le rayon de courbure minimal que peut suivre la voiture à sa vitesse actuelle, considérée constante sur toute la période d'échantillonnage $T = 0.1s$.
- Si le rayon de courbure qui permet de rejoindre la cible est plus petit que ce rayon minimal, on choisit les commandes d'évitement : braquer à fond vers la cible et freiner à fond. Bien que loin d'être optimale, cette approximation « prudente » a été faite pour simplifier le code.
- Sinon, le programme choisit les commandes volant qui permettent à la voiture de décrire un cercle de rayon $R_{commande}$ à la vitesse actuelle V où $R_{commande}$ est défini comme le plus grand rayon compris entre R_{min} et $R_{cible}/3$ (choisi expérimentalement) tel que le cercle de rayon $R_{commande}$ intersecte le segment voiture cible avant d'intersecter une limite du domaine ou peut se trouver la voiture (R_{cible} est le rayon de courbure qui permet d'atteindre la cible et R_{min} est le rayon de courbure minimal à vitesse actuelle V). Ce choix a été fait car la cible n'est jamais un objectif à atteindre en soi mais plutôt une indication de la direction dans laquelle aller pour rester sur le circuit. Cette tendance à surbraquer permet aussi de compenser les faibles variations de vitesse dues au commandes d'accélération sur le temps d'échantillonnage.
- On calcule ensuite la vitesse V_{maxR} , la vitesse maximale à laquelle peut se déplacer la voiture sur un arc de cercle de rayon $R_{commande}$. La vitesse V_{maxAnt} est la vitesse maximale qui permet de s'arrêter avant la cible en freinant à fond. On définit la vitesse à laquelle la voiture peut se déplacer comme $V_{commande} = \min(V_{maxAnt}, V_{maxR})$. On déduit de $V_{commande}$ et V la valeur de la commande d'accélération qui est ensuite bornée si nécessaire : à nouveau les pneus et la puissance des freins et du moteur limitent la vitesse d'accélération et de décélération.

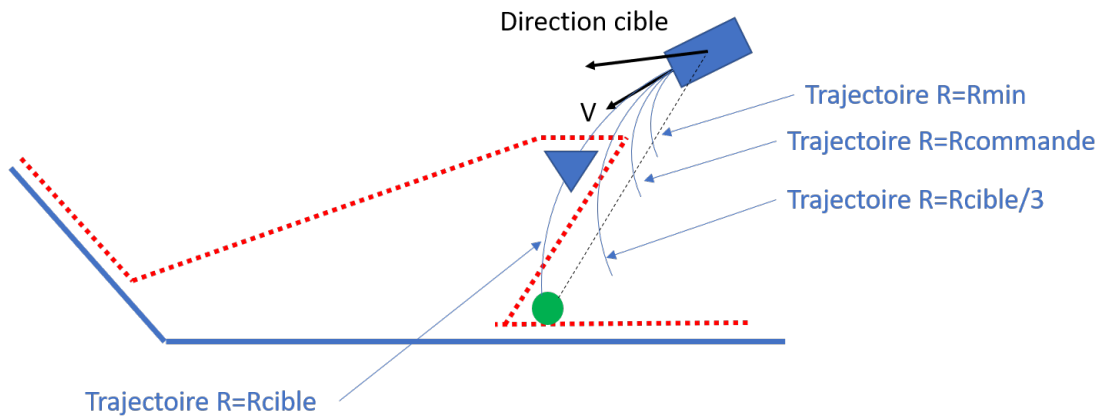


FIGURE 13 – Illustration graphique du calcul de $R_{commande}$ dans le cas où $R_{commande} < R_{cible}/3$

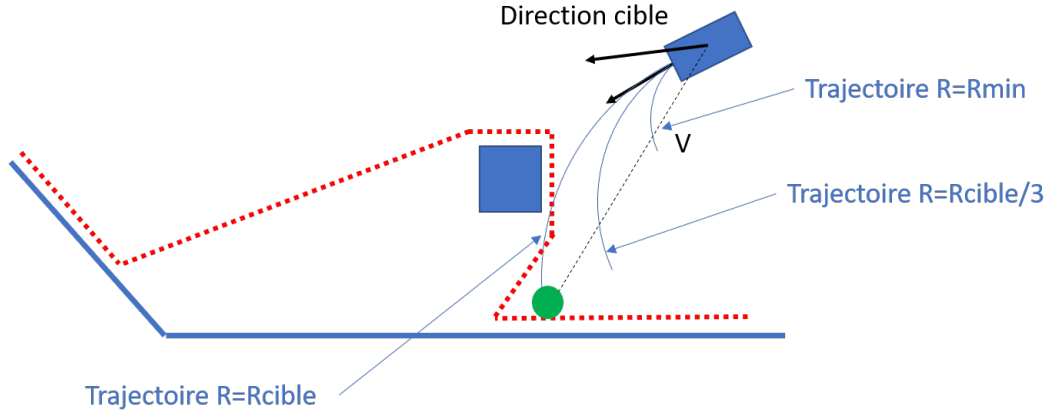


FIGURE 14 – Illustration graphique du calcul de $R_{commande}$ dans le cas où $R_{commande} = R_{cible}/3$

Ce nouvel algorithme permet donc de générer des cibles qui sont atteignables avec certitude ainsi que des commandes qui permettent de les atteindre de façon certaine. Les performances de ce second algorithme se sont avérées bien supérieures à celles du premier algorithme notamment lors de la simulation d'une course entre deux voitures équipées respectivement d'un des deux algorithmes.

2.5.2 Détermination expérimentale des paramètres de simulation

Les paramètres de simulation à déterminer expérimentalement sont :

- a_{maxlat} , l'accélération latérale maximale supportée par les pneumatiques
- ϵ_{max} , l'angle de braquage maximal des roues
- $a_{max}(v)$, la valeur maximale d'accélération en fonction de la vitesse.
- $a_{min}(v)$, la valeur minimale d'accélération (i.e. de freinage)
- τ_{sb} , le taux de surbraquage qui est supposé constant pour les faibles vitesses.

Mesure de a_{maxlat}

Lorsque la voiture suit une trajectoire circulaire de rayon r , l'accélération latérale est $|a_{lat}| = v^2/r$.

Pour mesurer a_{maxlat} il suffit de faire décrire une trajectoire de rayon r fixe à la voiture puis d'augmenter progressivement la vitesse de la voiture en la maintenant sur la trajectoire circulaire jusqu'à observer un décrochage (= dérapage/perte de contrôle).

Mesure de $a_{max}(v)$

En filmant le déplacement de la voiture depuis l'arrêt jusqu'à ce qu'elle atteigne sa vitesse maximale (devant une référence de dimension connue) avec la commande d'accélérateur au maximum, on peut mesurer par pointage son accélération en fonction de sa vitesse, et ainsi tracer expérimentalement $a_{max}(v)$.

Mesure de $a_{min}(v)$

En filmant le déplacement de la voiture depuis sa vitesse maximale jusqu'à l'arrêt (devant une référence de dimension connue) avec la commande d'accélérateur au

minimum et le freinage maximal, on peut mesurer par pointage son accélération en fonction de sa vitesse, et ainsi tracer expérimentalement $a_{min}(v)$.

Mesure de $\tau_{sb}(v)$

Lorsque le taux de glissement des pneumatiques est faible, (cadre dans lequel on compte se placer pour ne pas prendre de risques lors de la course)

$$\epsilon = \frac{\tau_{sb} \cdot v^2}{R} + \frac{l}{R}$$

où l est la longueur de la voiture, v est sa vitesse, R le rayon de courbure local de la trajectoire qu'elle suit et ϵ l'angle de braquage en radians.

En fixant la commande de volant et d'accélérateur à une position arbitraire et en mesurant l'angle de braquage ainsi que le rayon et la vitesse de la trajectoire circulaire uniforme obtenue, on peut déterminer le taux de surbraquage.

2.6 Structure du code de la voiture

Avant de se jeter directement dans le codage de l'algorithme qui dirige le comportement de la voiture, nous nous sommes intéressés à la structure globale que nous souhaitons, et notamment aux interactions entre les différents morceaux de code. Cette structure est schématisée en figure 15.

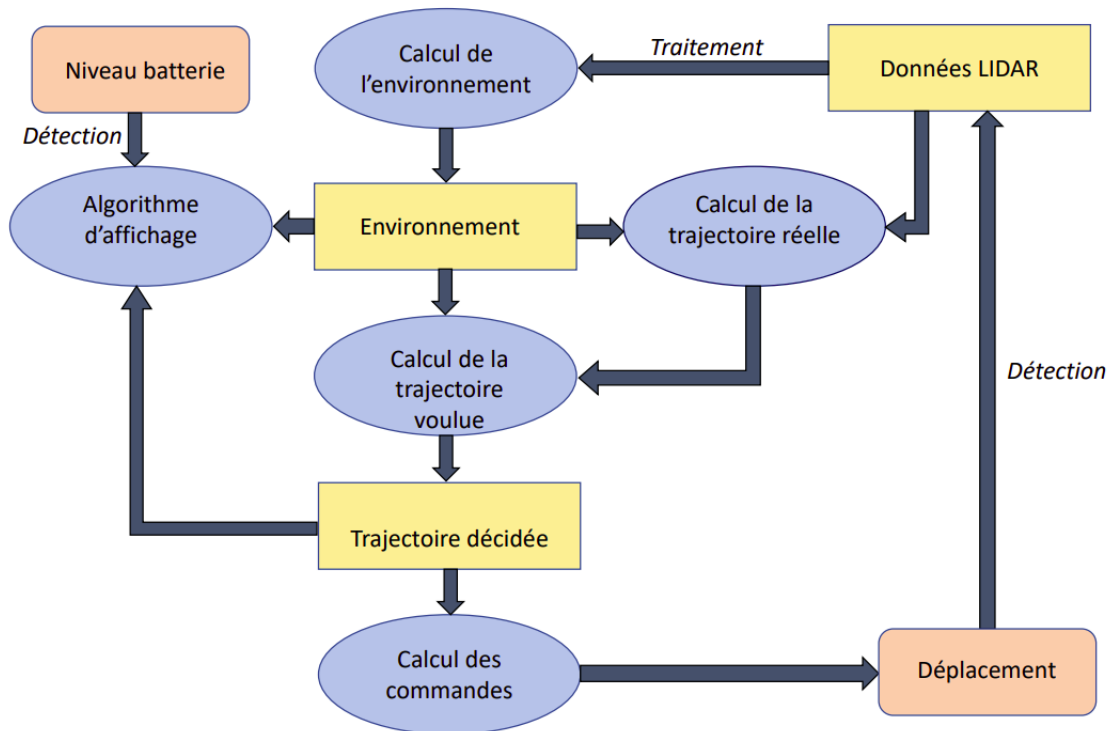


FIGURE 15 – Squelette du programme de direction de la voiture

Sur ce schéma, on peut voir les différentes parties de l'algorithme dans des ellipses bleues, et les données qu'ils s'échangent dans des rectangles jaunes.

En plus de nous intéresser au comportement de la voiture lors de la course, nous sommes penchés sur ses interactions avec l'utilisateur avant, pendant et après la course. Ces réflexions nous ont amenés à construire le diagramme de séquence, visible ci-dessous en figure 16.

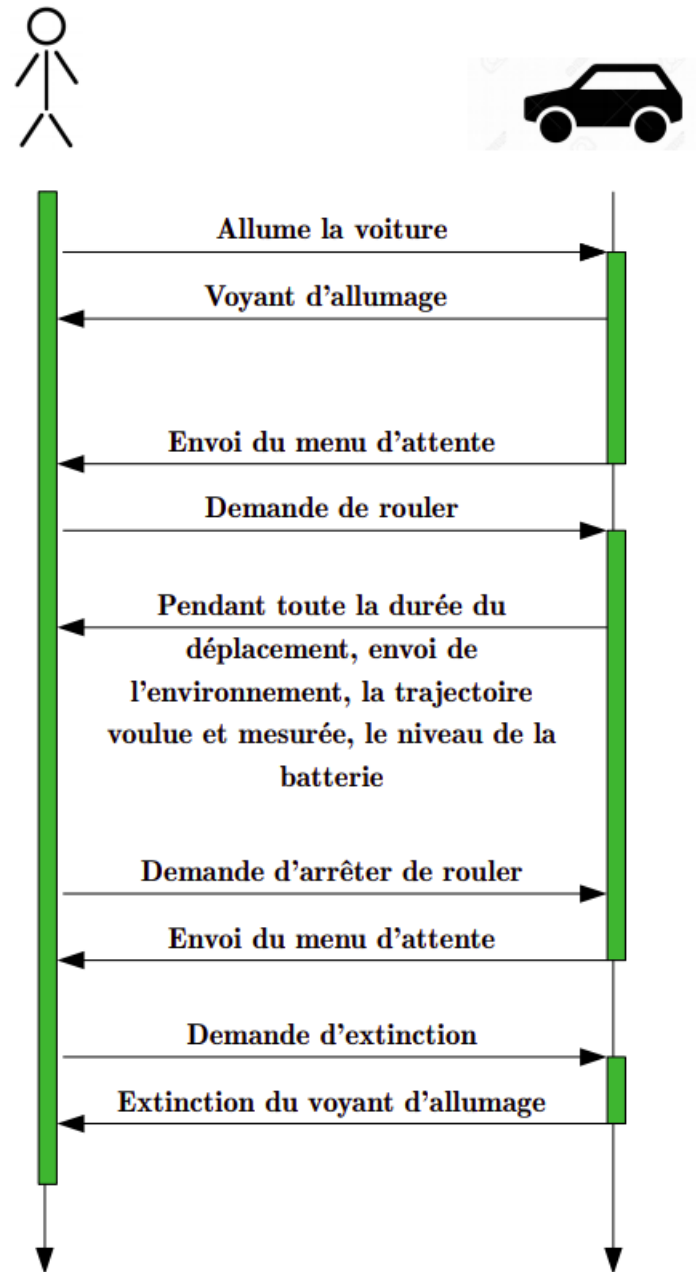


FIGURE 16 – Diagramme de séquence

2.7 Tests prévus

L'état de l'avancement de la réalisation de la voiture ne nous a pas permis de faire de tests de vérification du bon fonctionnement de la voiture. Cependant, les différents tests à effectuer sont :

- **Électronique** : il est important de vérifier la bonne communication entre tous les composants électroniques de la voiture (capteurs, carte de commande, carte de puissance, alimentation, moteurs, etc).
- **Communication avec l'utilisateur** : dans la façon dont nous avons réfléchi notre véhicule, nous avons prévu une communication entre l'utilisateur et la voiture, certainement à travers un ordinateur. Il nous faut donc nous assurer de la fiabilité de cette communication. De plus, le mode dégradé de la communication est une exigence du cahier des charges : la voiture doit s'arrêter si elle perd la communication avec l'utilisateur pendant plus de deux secondes. (c.f. partie [cahier des charges](#))
- **Mécanique** : en plus de l'électronique, le bon fonctionnement de notre voiture s'appuie également sur la mécanique du châssis et des transmissions. Il est donc important de mettre à l'épreuve tous les éléments mécaniques de la voiture.
- **Fonctionnement nominal** : un ensemble de tests et de mises à l'épreuve de notre voiture doivent s'assurer que le fonctionnement nominal de notre véhicule est bien celui prévu.
- **Fonctionnement dégradé** : dans la mesure du possible, il faut prévoir un comportement de la voiture en cas de problème, et les mettre à l'épreuve dans différentes configurations que l'on peut rencontrer
- **Dynamique de la voiture** : un ensemble de tests doivent être réalisés pour vérifier que les paramètres de dynamique du véhicule conviennent à son fonctionnement nominal. L'ensemble des valeurs mesurées et des protocoles de tests est détaillé dans la partie [2.5.2](#) du présent rapport.

3 Réalisation

3.1 Montage de la voiture

Le montage devait se faire en plusieurs parties.

La première consistait à monter une voiture miniature téléguidée sans la batterie (qui n'était pas fournie) qu'on a commandée sur Internet, mais sans tenir compte du téléguidage, puisque ce qui nous intéresse est une voiture autonome. Le modèle de la voiture est imposé par le cahier des charges (c.f. partie [cahier des charges](#)). Le montage de cette dernière s'est faite avec une seule difficulté : une des pièces de la voiture n'était pas de la bonne dimension, il s'agissait d'une roue dentée. Celle-ci devait assurer la liaison directe entre le moteur et la transmission de la voiture. Or, la pièce fournie ne faisait pas le bon diamètre (elle est trop petite), donc elle n'arrivait pas à toucher les dents de la roue dentée du moteur une fois montée, ce qui faisait que le moteur tournait dans le vide. Il a donc fallu commander la bonne pièce afin de pouvoir continuer le montage et passer à la phase de tests. Après avoir remplacé la mauvaise pièce par celle-ci, tout s'est arrangé et on pouvait enfin faire tourner les roues grâce au moteur. Hormis ce problème, cette partie montage s'est bien passée, et les instructions fournies étaient parfaitement claires.

La deuxième partie du montage consistait à mettre en place le côté automatique de notre voiture. Pour ça, on avait en tête d'installer une batterie aux bonnes caractéristiques en adaptant les connectiques par des soudures (c.f. partie [batterie](#)), puis la Raspberry Pi et le LIDAR (c.f. partie [budget initial](#)).

3.2 LIDAR

Pour se repérer dans l'espace, notre voiture utilise un LIDAR comme conseillé par les organisateurs de la course. Le LIDAR pour « light detection and ranging » est l'équivalent d'un radar utilisant des rayons lumineux pour sonder l'espace plutôt que des ondes radios. Il est placé sur le toit de la voiture et permet la détection des murs, des obstacles et des voitures adverses.

Le LIDAR que nous utilisons est le modèle RPLIDAR A2 produit par Slamtec. Malheureusement, il fait partie de la liste de matériels qui ont mis du temps à être livrés, nous avons donc commencé à étudier son fonctionnement très tard. Cependant nous avons quand même réussi à mettre au point un programme en langage Python qui permet d'obtenir en continu les tableaux des coordonnées des obstacles autour du LIDAR à une fréquence d'environ 10Hz. Ces tableaux pourront être ensuite utilisés par le programme de calcul de trajectoire. Ils se décomposent en deux sous-listes, la première contenant les angles de mesures du LIDAR, la seconde contenant la distance LIDAR-obstacle pour ces angles.

Un enjeu est de s'assurer que le LIDAR renvoie un maximum de points pour qu'il n'y ait pas de « trou » qui fausse le calcul de trajectoire. Pour cela, il faut choisir la bonne vitesse de rotation et vérifier que le LIDAR soit bien placé sur le châssis pour pouvoir détecter efficacement les murs et ne pas manquer d'informations. Ce sont des tests que nous n'avons pas pu mettre en place, et qui devront être faits les

années suivantes.

La taille des tableaux est susceptible de varier au cours des mesures, l'algorithme de calcul de trajectoire devra donc adapter son calcul en prenant en compte cette particularité.

Des bibliothèques Python ou C++ en libre accès peuvent être facilement trouvées sur internet et utilisées pour compléter ce qui a déjà été fait.

3.3 Servomoteur de direction

Le servomoteur contrôlant la direction de la voiture n'est pas fourni dans le kit du châssis, il faut donc l'acheter séparément. Pour cela, il nous fallait déterminer les caractéristiques dont nous avons besoin (couple, amplitude etc). Après discussion avec Éric Fenaux, il s'est avéré que la géométrie particulière de la transmission de la voiture a pour conséquence qu'un très faible couple est nécessaire pour manipuler la direction de la voiture. Cependant, avec l'arrivée du confinement, nous n'avons pas eu le temps de passer commande du matériel.

3.4 Batteries

Les batteries d'alimentation de la voiture n'étaient pas non plus fournies dans le kit du châssis. Après des recherches personnelles et des discussions avec notre encadrant, on en a conclu que le laboratoire de robotique pouvait nous prêter des batteries qui convenaient et le matériel pour les adapter. En effet, le port de branchement des batteries et celui du circuit d'alimentation n'étaient pas compatibles. Il a donc fallu souder de nouvelles connectiques, ce qui a été fait. À présent, on peut alimenter notre voiture. Malheureusement, cela fait partie des dernières choses que l'on a faites avant le confinement, nous n'avons donc pas pu tester la chose.

3.5 Structure interne

3.5.1 Structure interne provisoire

Afin de pouvoir procéder aux tests de fonctionnement de notre voiture il fallait construire la structure interne, c'est-à-dire définir l'emplacement de chaque élément au sein de la « carrosserie ». Cette étape est importante pour éviter par exemple que des fils s'emmêlent dans l'arbre du moteur, ce qui abîmerait notre matériel en plus d'entraîner le non fonctionnement de la voiture. Ainsi, pour pouvoir commencer les tests sans attendre d'avoir la structure définitive, nous avons réalisé une structure provisoire en plexiglass.

Nous sommes partis d'une plaque de plexiglass, d'épaisseur 5mm environ, que nous avons découpée aux dimensions de la voiture à l'aide d'une scie à ruban dans le laboratoire de l'U2IS de l'ENSTA. L'objectif était de créer un "étage" dans notre voiture pour séparer les différents éléments, comme sur le schéma en figure 17.

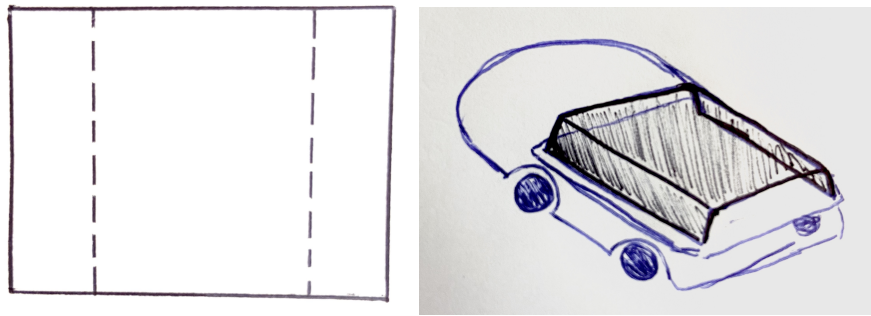


FIGURE 17 – Schéma de la structure provisoire

Une fois la plaque de plexiglass découpée dans les dimensions voulues, nous l'avons pliée le long des pointillés à l'aide d'une plieuse chauffante du laboratoire pour qu'elle prenne la forme voulue. Cela consiste à chauffer le plexiglass à l'aide d'un fil incandescent jusqu'à ce que la plaque devienne pliable. Une fois le plexiglass rendu souple le long du fil, la plieuse permet de le plier selon l'angle voulu, ici environ 60 degrés.

3.5.2 Modélisation sur Inventor

Une modélisation en utilisant un logiciel de CAO (pour nous Inventor, utilisé au laboratoire) était prévue afin d'étudier en détail les mouvements de la voiture sur ordinateur et pour faire des prévisions sur la pertinence de placer du matériel en plus à certains endroits de la voiture. Cela permet par exemple de répondre aux questions « Ce matériel peut-il être rajouté ? Va-t-il tenir ? Va-t-il gêner le mouvement de la voiture ? ». Mais en raison de la crise sanitaire, la modélisation n'a pas pu aboutir car nous n'avions plus accès à la voiture pour pouvoir la modéliser. On a seulement eu le temps de tracer les contours du châssis.

3.6 Site internet

Afin de faciliter au mieux la communication sur notre projet, nous avons mis en place un site internet regroupant les grands points de notre projet, de nos objectifs à la présentation de notre équipe. Celui-ci, bien qu'ayant été avorté en raison de la crise sanitaire, permet tout de même d'avoir un aperçu de nos ambitions pour ce travail, ainsi que de notre investissement. Pour y accéder, nous vous mettons à disposition le lien : <https://pievoitureautonome.wordpress.com/> .

Conclusion

Ce projet a été pour nous une aventure humaine et technique. Nous avons au départ beaucoup d'idées et d'envies, qui se sont confrontées à la réalité de l'organisation et du temps disponible. Cependant, l'amitié au sein de l'équipe nous a permis de continuer ensemble, et de ne jamais céder au découragement. L'état actuel du projet n'est pas ce à quoi nous nous attendions, toutefois l'expérience de conduire un projet et d'avoir une idée de quels sont les problèmes qui peuvent arriver à une équipe, et comment y réagir, est une expérience importante pour un futur ingénieur.

Parmi les idées que nous avons dû abandonner, on peut citer : l'ajout d'une caméra pour compléter la détection du LIDAR, notamment pour des obstacles qui seraient sous le « plan de détection » du LIDAR ; la personnalisation de la carrosserie transparente de notre voiture ; faire rouler la voiture, pour que l'on puisse estimer les paramètres de simulation ; la participation à la première course du projet.

Cependant, ajuster ses envies à la réalité fait partie du travail nécessaire. Nous souhaitons à l'équipe qui reprendra notre projet d'atteindre ses objectifs.