



Mémoire

Modélisation et contrôle d'un système robotique autonome boîte noire pour la robotique d'essaim : la cas pratique de l'hoverboard.

Auteur :

M. Antoine MORVAN
Robotique Autonome
FISE25

Jury :

Pr. L. HARDOUIN
Pr. L. JAULIN

Version du
25 août 2025

Remerciements

Je tiens à remercier tout particulièrement Luc JAULIN et Fabrice LEBARS qui m'ont accompagné pour toute la durée du stage. Leurs nombreuses explications et conseils m'ont été précieux.

Je remercie également Romain Bornier, un ami et collègue stagiaire qui m'a épaulé pendant toute cette période. Merci également à tout le centre de ressource du laboratoire de robotique de l'ENSTA Bretagne pour leurs expertise sur la conception de robot et les procédés de fabrication.

Je remercie enfin tous les membres du laboratoire avec qui j'ai pu discuté et qui m'ont conseillé aussi bien sur la réalisation de mon stage que pour mon parcours professionnel.

Table des matières

Introduction	1
1 Modélisation	5
1.1 Identification du modèle	5
1.1.1 Expérimentation de l’hoverboard	6
1.2 Modélisation	7
1.2.1 Modèle	7
1.2.2 Contrôleur	9
1.2.3 Observateur	10
1.2.4 Simulation	13
1.3 Discussion	15
2 Conception du système de contrôle	17
2.1 Architecture mécanique du système de contrôle	17
2.1.1 Premier prototype et défis mécaniques	17
2.1.2 Second prototype et procédés de fabrications	18
2.2 Architecture électronique du système de contrôle	22
2.2.1 Schéma électronique	22
2.2.2 Composants	23
2.3 Architecture logicielle du système de contrôle	25
2.4 Mode de fonctionnement	27
2.4.1 Mode téléopération	27
2.4.2 Mode autonome sans boussole	28
2.4.3 Mode autonome avec boussole	31

3	Expérimentation	33
3.1	Preuve de Concept	33
3.2	Essais Kalman sans boussole	33
3.3	Essais autonome Kalman sans boussole	35
3.4	Essais autonome Kalman avec boussole	37
3.5	Conclusion des essais	38
	Conclusion	41

Table des figures

1.1	Le pilote en bleu ne pilote l'hoverboard uniquement via l'angle de ses pieds β .	6
1.2	Représentation de l'expérimentation avec les masses.	7
1.3	Expérimentation avec les masses.	7
1.4	Hoverboard.	8
1.5	Gauche : accélération dans le repère monde ; Droite : accélération dans le repère de la planche.	11
1.6	Illustration du problème d'ambiguïté pour les accéléromètres. (a) et (b) montrent deux situations symétriques où l'accéléromètre mesure les mêmes données ; (c) dans le repère de la planche ; (d) triangle d'accélération. . . .	12
1.7	Le contrôle inertiel de l'hoverboard est supervisé par un contrôleur humain.	14
1.8	Résultats de la simulation.	14
2.1	Prototype 1 vu de côté	18
2.2	Prototype 1 vu de dessus	18
2.3	Servomoteur plié après premiers essais.	19
2.4	Nouveau mécanisme dans sa réalisation finale	20
2.5	CAO du nouveau mécanisme	20
2.6	Pièces résines ayant cassées sous l'effort	20
2.7	pièce résine ayant certains défauts d'impression	20
2.8	Pièce FDM ayant rompu sous l'effort.	21
2.9	Mécanisme en PVC/POM-C.	21
2.10	Prototype final réalisé.	22
2.11	AHRS Sparkfun OpenLog Artemis	24
2.12	GNSS SIRF EM-506	24

2.13	Electronique embarquée.	24
2.14	Electronique embarquée.	25
2.15	Schéma architecture logicielle.	27
2.16	Modèle de voiture de Dubins.	29
3.1	Trajectoire GNSS sur le stade.	34
3.2	Estimation du vecteur vitesse par le Kalman.	34
3.3	Détail de du tracé sur la trajectoire particulière.	34
3.4	Trajectoire confluante, tracé GNSS.	35
3.5	Trajectoire confluante estimée par Kalman.	35
3.6	Cap estimé par Kalman.	36
3.7	Trajectoire GNSS sur le stade.	37
3.8	Estimation du vecteur vitesse par le Kalman.	38
3.9	Détail de du tracé sur la trajectoire particulière.	38

List of acronyms

AHRS	<i>Attitude and Heading Referenc System</i>
GNSS	<i>Global Navigation Satellite System</i>
FDM	<i>Fused Deposition Modelling</i>
ROS2	<i>Robot Operating System 2</i>

Introduction

Dans un monde en constante évolution technologique, la robotique s'impose progressivement comme un pilier incontournable de nos sociétés modernes. Qu'il s'agisse de bras mécaniques dans les chaînes de production industrielle, de robots chirurgicaux assistés en médecine ou de véhicules autonomes en logistique, les robots sont désormais présents dans des domaines variés, contribuant à accroître l'efficacité, la précision et la sécurité des activités humaines.

Parmi les branches émergentes de cette discipline, la robotique d'essaim (ou swarm robotics) occupe une place de plus en plus importante. Inspirée du comportement collectif des animaux ou insectes sociaux tels que les fourmis, les abeilles, les bancs de poissons ou les groupes d'oiseaux, cette approche repose sur l'utilisation coordonnée de nombreux robots simples agissant localement mais produisant collectivement un comportement global complexe. Contrairement à la robotique centralisée, où un robot unique possède une intelligence globale, la robotique d'essaim mise sur la décentralisation, la redondance et la résilience du groupe.

Les applications potentielles sont nombreuses et prometteuses : exploration d'environnements dangereux ou inaccessibles (zones sinistrées, fonds marins, planètes), agriculture de précision, surveillance environnementale distribuée ou encore missions militaires de reconnaissance. Dans tous ces cas, l'utilisation d'un grand nombre de robots autonomes offre une robustesse naturelle face aux défaillances individuelles, tout en permettant une flexibilité opérationnelle.

Cependant, la robotique d'essaim soulève également plusieurs défis technologiques et économiques. L'un des enjeux majeurs est la conception de robots à la fois simples, peu coûteux, fiables et faciles à entretenir. En effet, la viabilité économique d'un essaim repose sur la possibilité de déployer un grand nombre d'unités sans compromettre les performances globales. Il faut donc trouver un équilibre délicat entre minimalisme matériel, robustesse des composants et performance de contrôle.

Ces contraintes techniques appellent une nouvelle approche dans la conception robotique. Aujourd'hui l'émergence de systèmes robotiques conçus pour la grande distribution offre une opportunité unique de répondre à ces défis, en effet, ces systèmes déjà autonomes, fiables car éprouvés et peu onéreux car produits en grande quantité peuvent constituer un

nouvel élément basique dans la conception de robots au même titre qu'un servomoteur par exemple. Il existe déjà certains travaux qui ont mis en lumière ce type de démarche à l'image de *iRobot Create*, cependant ceux-ci passent souvent par une modification lourde du logiciel et/ou de la structure mécanique interne du robot original, or parmi les intérêts de choisir un robot issu de la grande distribution, il y a l'idée de se reposer sur la fiabilité et la robustesse du robot choisis, ce qui suppose de ne pas modifier ni le logiciel ni la structure mécanique interne du robot, d'autant plus que dans l'objectif d'une utilisation au sein d'un essaim on cherche à réduire au plus le travail à réaliser sur chaque robots.

C'est dans cet optique que s'inscrit ce mémoire, qui vise à explorer les possibilités d'utilisations, comme élément de base dans la construction d'un robot autonome, un système robotique de grande distribution. Dans cette perspective de conception de systèmes robotisés à la fois économiques, robustes et facilement accessibles, le choix de la base matérielle joue un rôle crucial. C'est dans cette logique qu'un hoverboard a été retenu comme élément de base du robot mobile. Ce choix repose sur plusieurs atouts majeurs liés à la popularité massive de ces dispositifs auprès du grand public.

Tout d'abord, la large diffusion commerciale des hoverboards ces dernières années a entraîné une baisse significative de leur coût, tant pour les appareils neufs que pour les modèles de seconde main. Cela les rend particulièrement attractifs pour des projets de recherche ou d'expérimentation à budget limité, notamment lorsqu'un grand nombre de plateformes est requis, comme dans le cas de la robotique d'essaim.

Ensuite, leur architecture matérielle intégrée, comprenant deux moteurs à commande indépendante, une batterie rechargeable, une carte de contrôle et un ensemble de capteurs, constitue un ensemble mécatronique complet. Ce niveau d'intégration permet de réduire significativement la complexité de conception d'un robot mobile, tout en bénéficiant d'un matériel robuste, modulaire et éprouvé en conditions réelles d'utilisation.

Enfin, sa capacité à transporter des charges relativement importantes, sa manœuvrabilité (grâce à ses deux roues motorisées), ainsi que sa compacité et son autonomie, en font une base particulièrement intéressantes dans son rapport qualité-prix. Ainsi, loin d'être un simple choix opportuniste, l'utilisation d'un hoverboard comme base de robot s'inscrit dans une démarche pragmatique, visant à réutiliser intelligemment des technologies déjà existantes, accessibles et fonctionnelles, pour construire une plateforme mobile adaptée à la robotique d'essaim low-cost et adaptable.

Cependant, l'utilisation d'un hoverboard, ou autre système robotique similaire, comme base robotique n'est pas exempte de limitations et de défis, en particulier lorsqu'on souhaite l'exploiter tel quel, sans en modifier l'architecture interne. L'un des principaux obstacles réside dans l'absence de documentation accessible sur les algorithmes de contrôle embarqués. En tant que système commercial propriétaire, celui-ci fonctionne comme une véritable boîte noire [1] [2] : son comportement est observable de l'extérieur, mais ses mécanismes internes, tant sur le plan logiciel que matériel, restent largement opaques. Cette opacité

complicque considérablement la compréhension fine des lois de commande, la nature exacte des retours capteurs ou encore la manière dont sont gérées les sécurités internes. Ainsi, le défi n'est pas seulement technique, mais aussi méthodologique : il s'agit d'étudier le système à partir de son comportement externe, d'en déduire un modèle suffisamment exploitable pour le contrôle global, tout en respectant les contraintes de non-intrusion. Cette approche expérimentale, inspirée des méthodes d'analyse de systèmes black-box, suppose un travail rigoureux d'observation, de mesure, de modélisation empirique et de calibration, afin de garantir un pilotage fiable sans connaissance directe des algorithmes embarqués.

Ce rapport s'organise en trois parties, la première consacrée à la problématique de modélisation de l'hoverboard, ensuite nous aborderons en accord avec les observations issus de la modélisation, la conception et réalisation d'un système pour le pilotage et le contrôle de l'hoverboard et enfin nous l'expérimenterons.

Chapitre 1

Modélisation

On cherche donc à réaliser un robot mobile autonome sur la base d'un hoverboard, ne voulant pas modifier le logiciel ou la structure mécanique de celui-ci, nous allons devoir créer un système de pilotage qui s'appuie sur les entrées et sorties de l'hoverboard. Pour cela nous allons devoir modéliser le comportement de l'hoverboard, c'est à dire comprendre comment il réagit aux commandes que nous lui envoyons, ainsi que les retours qu'il nous envoie. Or, comme évoqué plus tôt, la documentation sur ce sujet étant maigre voir inexistante nous nous retrouvons donc face à un système type "boîte noire". La première étape consistera donc à identifier le modèle de l'hoverboard. Pour cela nous allons devoir réaliser des expériences et en déduire un modèle. Ce travail de modélisation repose en grande partie sur l'article *Modeling and control of an hoverboard* [3] par Luc Jaulin.

1.1 Identification du modèle

Lorsqu'un système est considéré comme une boîte noire, c'est-à-dire que sa structure interne et ses algorithmes de commande ne sont pas accessibles, plusieurs approches méthodologiques peuvent être envisagées pour en extraire un modèle exploitable. L'interpolation permet par exemple d'approximer le comportement du système à partir de données connues, tandis que l'identification de systèmes cherche à ajuster des équations dynamiques aux observations disponibles. De leur côté, les algorithmes d'intelligence artificielle, tels que les réseaux de neurones, offrent la possibilité de capturer des dynamiques complexes sans connaissance préalable des équations physiques. Dans le cadre de ce travail, c'est toutefois une approche plus empirique qui a été retenue. Elle repose sur la réalisation d'essais expérimentaux directement sur la plateforme hoverboard, afin d'observer son comportement en conditions réelles. Les résultats issus de ces expérimentations permettront non seulement de dégager les grandes lignes de son fonctionnement, mais également de guider le dimensionnement du sursystème de contrôle qui viendra par la suite orienter et stabiliser

le dispositif.

1.1.1 Expérimentation de l'hoverboard

L'hoverboard est un véhicule constitué de deux roues reliées par un axe unique. Sa stabilité est assurée par une électronique embarquée, mais si l'on souhaite en analyser le comportement, il est pertinent de le modéliser de manière simplifiée. Dans cette étape de modélisation, on suppose que le moteur n'est pas commandé et que le comportement en boucle ouverte de la planche se rapproche de celui d'un monocycle plan. Dans ce modèle, u représente le moment exercé entre la planche et la roue. L'angle talocrural β , défini comme l'angle entre le corps du pilote et la planche, est supposé constant et constitue la seule variable effectivement contrôlée par le pilote. Autrement dit, le pilote n'agit pas directement sur son tangage θ (il ne décide pas de se pencher en avant ou en arrière), mais uniquement sur β via l'inclinaison de ses pieds par rapport à son corps. Ce mécanisme de contrôle, illustré dans les travaux de Jaulin, implique que le pilote agit indirectement sur la vitesse du véhicule en imposant un décalage entre son corps et la planche.

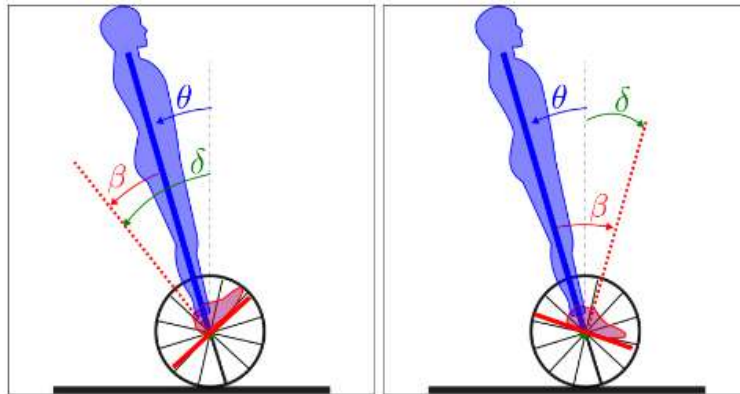


FIGURE 1.1 – Le pilote en bleu ne pilote l'hoverboard uniquement via l'angle de ses pieds β .

Ce principe a été confirmé par une série d'expérimentations. L'essai direct de l'hoverboard a montré que l'on pouvait effectivement le piloter uniquement par l'inclinaison des plateformes, ce qui établit un lien clair entre l'angle appliqué par les pieds et la consigne en vitesse ou en accélération transmise aux moteurs. Une autre expérience a consisté à fixer deux masses de 1 kg à l'avant de chaque plateforme : l'hoverboard est alors entré dans une phase d'accélération continue jusqu'à sa chute vers l'avant, confirmant que l'angle d'inclinaison $\delta = \theta + \beta$ constitue bien une entrée déterminante correspondant à une consigne d'accélération.

Ainsi, le déplacement résulte d'un principe simple : lorsque l'inclinaison globale de l'ensemble planche-pilote est positive, les roues tournent vers l'avant ; lorsqu'elle est négative,

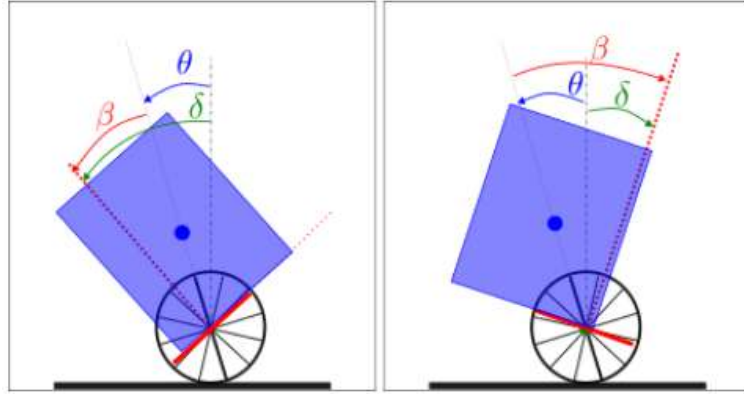


FIGURE 1.2 – Représentation de l'expérimentation avec les masses.



FIGURE 1.3 – Expérimentation avec les masses.

elles tournent vers l'arrière. Un principe qui semble se confirmer dans le manuel de l'hoverboard et dans d'autres vidéos explicatives (lien). Le problème central devient alors la mesure et l'estimation de cet angle δ à partir d'une unité inertielle, qui fournit uniquement des accélérations et des vitesses de rotation.

1.2 Modélisation

1.2.1 Modèle

La modélisation d'un hoverboard peut être réalisée en utilisant une approche lagrangienne. Dans ce travail, nous choisissons plutôt une approche de type *Newton-Euler*, similaire à celle utilisée pour le Segway[11].

Notons b le centre de gravité du pilote et a le centre de la roue. Prenons un point c sur la roue et définissons s comme l'angle entre le vecteur $\vec{ac}$ et l'axe horizontal, tandis que θ représente l'angle de tangage. Le système possède deux degrés de liberté, s et θ . L'état du système est alors donné par le vecteur :

$$x = (s, \theta, v, \omega)^T,$$

où $v = \dot{s}$ et $\omega = \dot{\theta}$.

Les paramètres physiques sont les suivants :

- pour la roue : sa masse M , son rayon ρ , et son moment d'inertie $J_M = \frac{1}{2}M\rho^2$,
- pour le pilote : sa masse m , et son moment d'inertie supposé $J_p = m\ell^2$, où ℓ désigne la distance entre son centre de gravité et le centre de la roue.

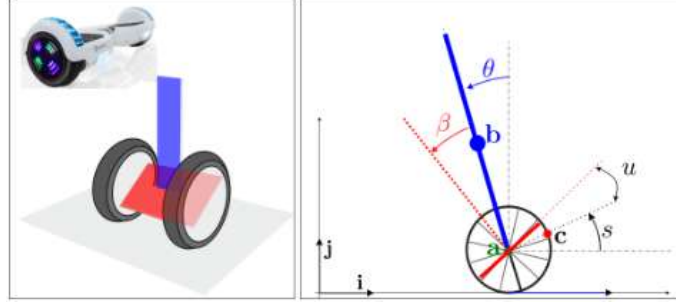


FIGURE 1.4 – Hoverboard.

En appliquant le principe fondamental de la dynamique à chaque sous-système (la roue et le corps), on obtient :

$$\begin{cases} -r_x + f_x = -M\rho\ddot{s} & \text{(roue en translation)} \\ f_x\rho + u = \frac{1}{2}M\rho^2\ddot{\theta} & \text{(roue en rotation)} \\ r_x\mathbf{i} + r_y\mathbf{j} - mg_0\mathbf{j} = m\ddot{\mathbf{b}} & \text{(corps en translation)} \\ r_x\ell\cos\theta + r_y\ell\sin\theta - u = m\ell^2\ddot{\theta} & \text{(corps en rotation)} \end{cases}$$

où f_x, r_x, r_y sont les forces de réaction et $g_0 = 9.81 \text{ m.s}^{-2}$ est la gravité.

La position du centre de gravité $\mathbf{b}$ est donnée par :

$$\mathbf{b} = (-\rho s - \ell\sin\theta)\mathbf{i} + (\ell\cos\theta + \rho)\mathbf{j},$$

et sa dérivée seconde s'écrit :

$$\ddot{\mathbf{b}} = (-\rho\ddot{s} - \ell\ddot{\theta}\cos\theta + \ell\dot{\theta}^2\sin\theta)\mathbf{i} - (\ell\ddot{\theta}\sin\theta + \ell\dot{\theta}^2\cos\theta)\mathbf{j}.$$

En éliminant les forces internes, on obtient finalement le système d'équations :

$$\begin{cases} \ddot{s} = \frac{\mu_3(\mu_2\omega^2 - \mu_g\cos\theta)\sin\theta + (\mu_2 + \mu_3\cos\theta)u}{\mu_4 + \mu_3^2\sin^2\theta} \\ \ddot{\theta} = \frac{(\mu_1\mu_g - \mu_3^2\omega^2\cos\theta)\sin\theta - (\mu_1 + \mu_3\cos\theta)u}{\mu_4 + \mu_3^2\sin^2\theta} \end{cases}$$

où les paramètres positifs sont définis par :

$$\mu_1 = \frac{3}{2}M\rho^2 + m\rho^2, \quad \mu_2 = 2m\ell^2, \quad \mu_3 = \rho m\ell, \quad \mu_4 = \rho^2 m\ell^2(3M + m), \quad \mu_g = g_0\ell m.$$

Ainsi, les équations d'état du système complet s'écrivent :

$$\begin{pmatrix} \dot{s} \\ \dot{\theta} \\ \dot{v} \\ \dot{\omega} \end{pmatrix} = \begin{pmatrix} v \\ \omega \\ \frac{\mu_3(\mu_2\omega^2 - \mu_g \cos \theta) \sin \theta + (\mu_2 + \mu_3 \cos \theta)u}{\mu_4 + \mu_3^2 \sin^2 \theta} \\ \frac{(\mu_1\mu_g - \mu_3^2\omega^2 \cos \theta) \sin \theta - (\mu_1 + \mu_3 \cos \theta)u}{\mu_4 + \mu_3^2 \sin^2 \theta} \end{pmatrix}.$$

Ces équations traduisent la dynamique non linéaire de l'hoverboard et constituent la base pour l'analyse de stabilité et la conception du contrôleur.

1.2.2 Contrôleur

Tous les capteurs embarqués sur l'hoverboard sont de nature inertielle : deux accéléromètres et un gyromètre placés au centre de la planche, c'est-à-dire au niveau de l'axe des roues. La grandeur essentielle à considérer est l'inclinaison ou *tilt* de la planche, notée $\delta = \theta + \beta$, où θ correspond au tangage et β à l'angle talocrural imposé par le pilote.

Puisqu'aucun capteur externe (comme un GPS, une caméra ou un capteur d'angle absolu) n'est disponible, il est nécessaire de concevoir un contrôleur qui n'utilise que les mesures issues de l'unité inertielle. En supposant que cette dernière fournisse une estimation de δ et de sa dérivée angulaire $\omega = \dot{\theta}$, Jaulin propose un contrôleur proportionnel-dérivé (PD) de la forme :

$$u = k_1\delta + k_2\omega,$$

où u désigne le couple moteur appliqué entre la roue et la planche. Ce contrôleur interprète l'inclinaison δ comme une erreur à corriger et cherche donc à maintenir la planche horizontale.

Analyse de stabilité

Le système en boucle fermée s'écrit :

$$\begin{pmatrix} \dot{\theta} \\ \dot{\omega} \end{pmatrix} = \begin{pmatrix} \omega \\ \frac{(\mu_1\mu_g - \mu_3^2\omega^2 \cos \theta) \sin \theta - (\mu_1 + \mu_3 \cos \theta)(k_1(\theta + \beta) + k_2\omega)}{\mu_4 + \mu_3^2 \sin^2 \theta} \end{pmatrix}.$$

Au voisinage de $(\theta, \omega) = (0, 0)$, le système peut être linéarisé sous la forme matricielle :

$$\begin{pmatrix} \dot{\theta} \\ \dot{\omega} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{\mu_1\mu_g - k_1(\mu_1 + \mu_3)}{\mu_4} & -\frac{k_2(\mu_1 + \mu_3)}{\mu_4} \end{pmatrix} \begin{pmatrix} \theta \\ \omega \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{k_1(\mu_1 + \mu_3)}{\mu_4} \end{pmatrix} \beta.$$

Le polynôme caractéristique associé est :

$$P(s) = s^2 + \frac{k_2(\mu_1 + \mu_3)}{\mu_4}s + \frac{-\mu_1\mu_g + k_1(\mu_1 + \mu_3)}{\mu_4}.$$

Pour que le système soit stable au sens de Hurwitz, les conditions suivantes doivent être satisfaites :

$$k_1 > \frac{\mu_1\mu_g}{\mu_1 + \mu_3}, \quad k_2 > 0.$$

Ces inégalités garantissent que le contrôleur proportionnel-dérivé stabilise le tangage θ et maintient la planche en équilibre.

Ainsi, le contrôleur conçu est entièrement *model-free* dans la mesure où il ne nécessite pas l'utilisation explicite des équations dynamiques de l'hoverboard. Il se base uniquement sur des mesures inertielle simples (tilt δ et vitesse angulaire ω), ce qui en fait une solution robuste, économique et adaptée à un système dont les paramètres internes (masse du pilote, position de son centre de gravité, etc.) ne sont pas connus avec précision.

1.2.3 Observateur

La mise en œuvre du contrôleur présenté précédemment nécessite de disposer de deux grandeurs : l'inclinaison ou *tilt* de la planche, notée δ , et la vitesse angulaire $\omega = \dot{\theta}$. La seconde est directement mesurée par le gyromètre de l'unité inertielle. En revanche, la première doit être reconstruite à partir des données des accéléromètres, ce qui rend nécessaire la conception d'un observateur.

Principe de l'observation

L'unité inertielle fournit le vecteur d'accélération mesuré dans le repère de la planche, noté $\mathbf{a}_m$. Ce vecteur combine deux contributions : l'accélération réelle de la planche et la gravité. Il s'exprime comme :

$$\mathbf{a}_m = \begin{pmatrix} \cos \delta & \sin \delta \\ -\sin \delta & \cos \delta \end{pmatrix} \begin{pmatrix} \dot{v} \\ g_0 \end{pmatrix},$$

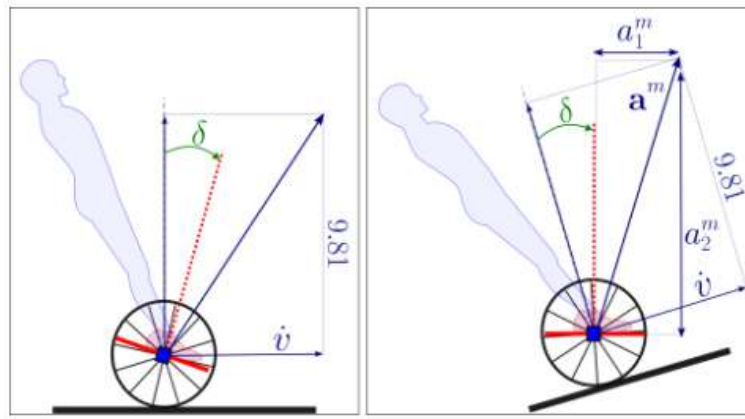


FIGURE 1.5 – Gauche : accélération dans le repère monde ; Droite : accélération dans le repère de la planche.

où $\dot{v}$ désigne l'accélération longitudinale de la planche et $g_0 = 9.81 \text{ m.s}^{-2}$ est l'accélération de la pesanteur.

On obtient alors :

$$\|\mathbf{a}_m\|^2 = \dot{v}^2 + g_0^2.$$

Cette relation montre que les accéléromètres ne permettent pas de distinguer directement une accélération due à un mouvement horizontal d'une inclinaison par rapport à la gravité.

Lien avec l'ascenseur d'Einstein

Cette ambiguïté est précisément ce qu'Einstein a mis en évidence dans son expérience de pensée dite de *l'ascenseur d'Einstein*. Un observateur enfermé dans un ascenseur ne peut pas, à partir des seules mesures locales, distinguer si la force qu'il ressent provient d'une accélération uniforme de l'ascenseur ou de la gravité terrestre (voir également Kalman, 1960 [5] ; Thrun et al., 2005 [7], pour une formalisation probabiliste de ce type de problème dans le contexte robotique).. De manière analogue, un accéléromètre fixé sur l'hoverboard ne peut pas savoir si la composante mesurée résulte d'une inclinaison ou bien d'une accélération réelle du système.

Cette différence rend la reconstruction de l'angle δ plus complexe que dans le cas d'un pendule inversé classique[4], où la gravité fournit directement une référence absolue.

Utilisation du gyromètre pour lever l'ambiguïté

Pour surmonter cette difficulté, on exploite la mesure fournie par le gyromètre. En notant ω_m la vitesse angulaire mesurée et en supposant que δ est continu en t , une prédiction

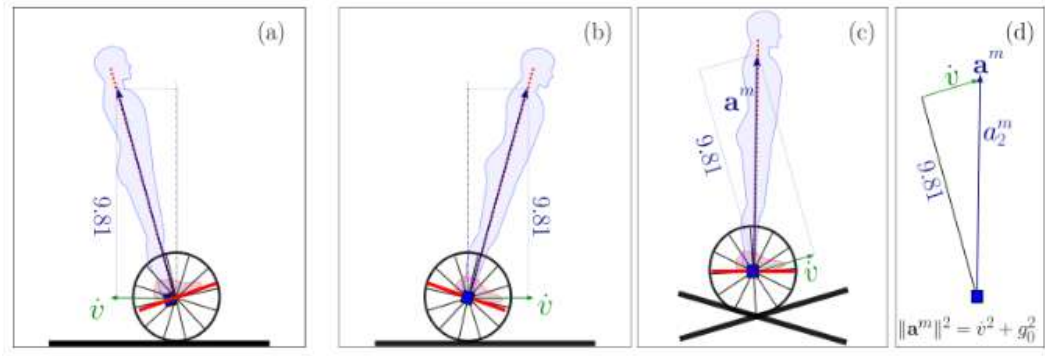


FIGURE 1.6 – Illustration du problème d'ambiguïté pour les accéléromètres. (a) et (b) montrent deux situations symétriques où l'accéléromètre mesure les mêmes données ; (c) dans le repère de la planche ; (d) triangle d'accélération.

de l'angle δ peut être obtenue par intégration :

$$\delta_p(t) = \delta_m(t - \Delta t) + \Delta t \cdot \omega_m(t),$$

où δ_m correspond à l'estimation précédente de l'angle et Δt est le pas d'échantillonnage.

À partir de cette prédiction δ_p , on déduit une estimation de l'accélération horizontale :

$$\dot{v} \simeq \cos \delta_p a_{m1} - \sin \delta_p a_{m2},$$

où a_{m1} et a_{m2} désignent respectivement les composantes horizontale et verticale du vecteur $\mathbf{a}_m$.

Le signe de cette estimation permet de lever l'ambiguïté et de choisir la bonne orientation pour δ .

Estimation finale du tilt

En combinant les données des accéléromètres et du gyromètre, l'estimation finale de l'angle d'inclinaison δ est donnée par :

$$\hat{\delta} = \text{angle} \left(\mathbf{a}_m, \begin{pmatrix} \text{sign}(\cos \delta_p a_{m1} - \sin \delta_p a_{m2}) \cdot \sqrt{\|\mathbf{a}_m\|^2 - g_0^2} \\ g_0 \end{pmatrix} \right).$$

Cette combinaison permet de reconstruire en temps réel l'inclinaison δ à partir de mesures bruitées et incomplètes.

Intégration dans la commande

L'estimation $\hat{\delta}$ est ensuite utilisée dans la loi de commande :

$$u = k_1 \hat{\delta} + k_2 \omega_m.$$

Ainsi, l'observateur inertiel permet de résoudre une ambiguïté fondamentale (accélération vs gravité) et de fournir les informations nécessaires au contrôleur sans recourir à des capteurs externes. Cette approche, bien qu'empirique, est robuste et illustre comment une problématique classique de la relativité générale peut se retrouver dans un contexte pratique de robotique mobile.

1.2.4 Simulation

Afin d'illustrer le comportement du contrôleur, il est nécessaire de modéliser également l'action du pilote sur l'hoverboard. Plutôt que d'introduire un modèle anthropomorphique complexe, cette action est représentée par un contrôleur secondaire, appelé *contrôleur humain*.

Ce contrôleur reçoit en entrée la vitesse mesurée v_m de l'hoverboard et la vitesse désirée $\bar{v}$, et génère une consigne sous la forme d'un angle talocrural β . La relation choisie est une loi proportionnelle simple :

$$\beta = k_3(\bar{v} - v_m),$$

où k_3 est un gain positif.

Ce modèle traduit l'idée que le pilote ajuste inconsciemment son inclinaison β pour compenser l'écart entre la vitesse actuelle et la vitesse souhaitée. L'angle β devient alors une variable intermédiaire qui agit sur l'inclinaison totale $\delta = \theta + \beta$, laquelle est régulée par le contrôleur inertiel.

Résultats numériques

Les simulations numériques ont été réalisées avec les paramètres suivants :

$$m = 10 \text{ kg}, \quad M = 1 \text{ kg}, \quad \ell = 1 \text{ m}, \quad \rho = 1 \text{ m}, \quad g_0 = 9.81 \text{ m.s}^{-2},$$

$$k_1 = 200, \quad k_2 = 100, \quad k_3 = -0.2.$$

La vitesse désirée a été fixée à $\bar{v} = 1 \text{ m.s}^{-1}$. Les résultats montrent que la vitesse réelle $v(t)$ converge vers cette valeur de consigne après une phase transitoire de quelques secondes.

En parallèle, le tilt estimé $\hat{\delta}$ reste proche de zéro, ce qui confirme que le contrôleur inertiel parvient à stabiliser l'inclinaison malgré les perturbations induites par l'action du pilote. Le couple moteur u généré par la loi de commande s'avère régulier et limité, ce qui laisse envisager une implémentation réaliste sur une plateforme physique.

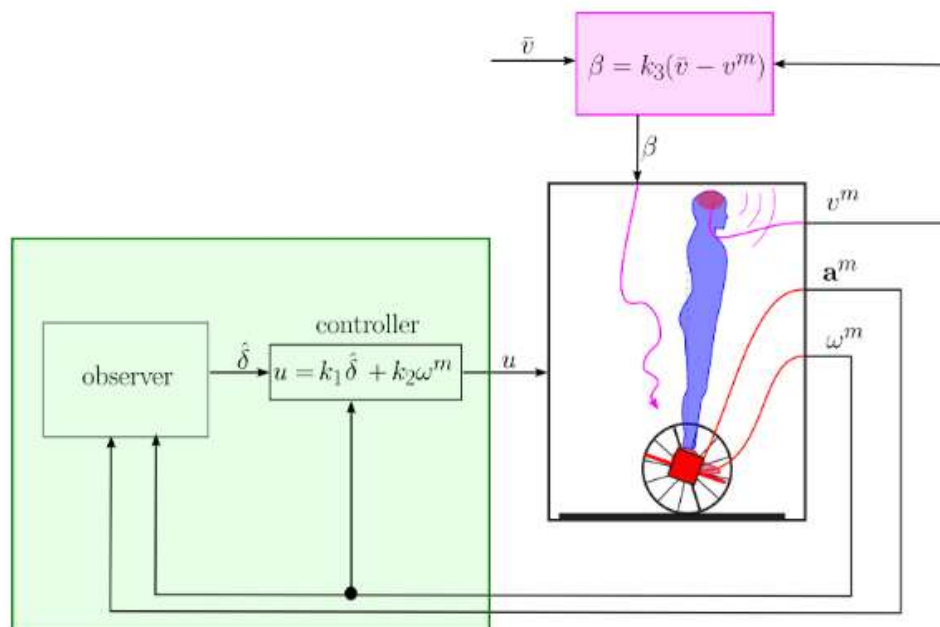


FIGURE 1.7 – Le contrôle inertiel de l’hoverboard est supervisé par un contrôleur humain.

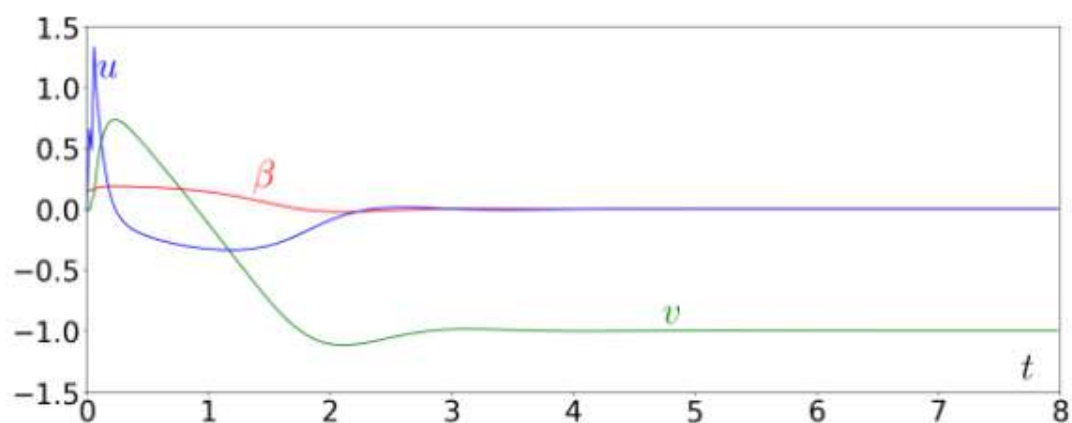


FIGURE 1.8 – Résultats de la simulation.

1.3 Discussion

Ces résultats mettent en évidence deux points essentiels : d'une part, le rôle central de l'utilisateur qui, par le biais de l'angle β , assure indirectement la régulation de la vitesse ; d'autre part, la robustesse du contrôleur inertiel qui stabilise le système sans connaissance fine de ses paramètres internes.

L'approche proposée combine ainsi un modèle réduit du comportement humain et une stratégie de contrôle locale simple, ce qui rend la méthode à la fois efficace, peu coûteuse et adaptée à des systèmes de type « boîte noire » dont les équations exactes sont difficiles à identifier.

Ainsi, de part nos observations et expérimentations, nous avons pu établir qu'elles étaient les entrées et sorties du système, desquelles nous avons pu développer une modélisation de l'hoverboard cohérente avec la réalité. De plus, en simulant ce modèle nous avons également pu déterminer un contrôleur stable pour l'hoverboard montrant qu'il est théoriquement possible de l'asservir en vitesse de manière indirecte en agissant uniquement sur l'angle de la plateforme. Il nous reste donc à implémenter ce contrôleur sur l'hoverboard et à vérifier que celui-ci fonctionne correctement.

Chapitre 2

Conception du système de contrôle

Maintenant que nous avons modélisé et simulé l’hoverboard, nous connaissons les entrées et sorties du système et savons qu’il est possible le piloter de manière stable. De plus notre approche empirique nous a déjà donné quelques pistes d’approche et une idées du dimensionnement des composants. Nous pouvons donc désormais concevoir le système de contrôle.

2.1 Architecture mécanique du système de contrôle

Lors de nos essais sur l’hoverboard, nous avons pu observer que placer une masse d’un kilo sur une extrémité de la plateforme était largement suffisant pour l’incliner et ainsi induire une accélération. Notre système de contrôle s’appuie donc sur ce principe en utilisant des actionneurs capables de reproduire cet effet d’inclinaison en bougeant une masse le long de la plateforme.

2.1.1 Premier prototype et défis mécanique

La première idée de prototype consistait à utiliser un pendule inversé, constitué d’une masse fixée à l’extrémité d’une tige verticale, qui pouvait s’incliner vers l’avant ou vers l’arrière afin de provoquer l’oscillation de la plateforme. Si ce principe est intuitif et facile à comprendre, il présente toutefois des limites mécaniques importantes. En effet, dans une telle configuration, l’ensemble des efforts est reporté directement sur l’axe du servomoteur, notamment ceux liés au poids de la masse. Or cette force s’exerce perpendiculairement à l’axe de rotation, ce qui sollicite le servomoteur dans une direction pour laquelle il n’a pas été conçu, risquant ainsi d’endommager prématurément le mécanisme.

Pour pallier ces contraintes mécaniques, il a été envisagé de déplacer la masse non plus

verticalement, mais horizontalement, en la plaçant sur une petite plateforme supportée par une bille. Cette configuration permet de limiter les efforts liés au couple généré par la distance entre la masse et l'axe du servomoteur, tout en transmettant directement le mouvement via l'axe de ce dernier. Une autre alternative consistait à installer la masse sur un rail, offrant ainsi la possibilité d'un positionnement précis et d'une réduction significative des contraintes appliquées aux actionneurs. Ce concept présentait de nombreux avantages en termes de stabilité et de contrôle, mais sa mise en œuvre aurait été plus complexe et chronophage. Pour cette raison, cette solution a finalement été écartée.

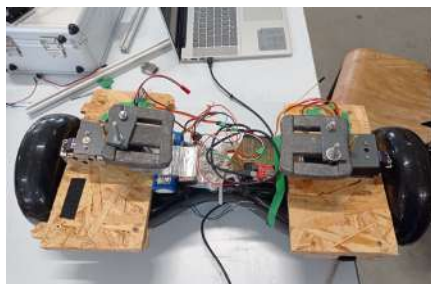


FIGURE 2.1 – Prototype 1 vu de côté

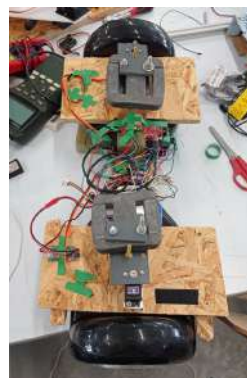


FIGURE 2.2 – Prototype 1 vu de dessus

Une fois le prototype mécanique assemblé, les premiers essais ont confirmé la validité du principe : l'inclinaison de la plateforme pouvait effectivement être obtenue par le déplacement d'une masse. Cependant, les tests ont également mis en évidence une fragilité importante face aux chocs, en particulier lors des arrêts brusques. Il est apparu que la pièce assurant la liaison entre la plateforme et l'axe du servomoteur ne bloquait pas complètement certains degrés de liberté. De ce fait, une partie des efforts était transmise directement aux servomoteurs, qui n'étaient pas conçus pour les supporter. En conséquence, à chaque impact important, au moins un servomoteur devait être remplacé, leurs axes se déformant sous la contrainte.

La conception d'un prototype doit donc débuter par une étude approfondie de la structure responsable du déplacement de la masse, afin de garantir une robustesse mécanique suffisante. Cette étape est indispensable pour réaliser les tests préliminaires nécessaires à l'implémentation du contrôleur.

2.1.2 Second prototype et procédés de fabrications

L'objectif de ce second prototype est d'obtenir un système plus robuste, capable de résister aux chocs inévitables lors du développement du contrôleur. Comme mentionné



FIGURE 2.3 – Servomoteur plié après premiers essais.

précédemment, la principale faiblesse du premier prototype provenait de la liaison pivot entre l'axe du servomoteur et la plateforme supportant la masse, notamment de son incapacité à contraindre certains degrés de liberté.

Pour y remédier, l'idée est de concevoir une structure capable d'encaisser les efforts autres que ceux associés à l'axe de rotation du servomoteur. Cette structure a été pensée en deux parties :

- une première, solidaire du bâti, servant d'ancrage et supportant l'axe de rotation,
- une seconde, mobile, portant à la fois le servomoteur et la masse fixée à ses extrémités.

Comme illustré à la Figure 2.4, les deux parties sont reliées par une vis alignée sur l'axe du servomoteur, jouant le rôle d'axe de rotation. Cette vis traverse deux roulements à billes espacés d'environ 2 cm, et un roulement plat est ajouté entre la partie mobile et le bâti. L'ensemble forme ainsi une fourche qui, par serrage, bloque tous les degrés de liberté à l'exception de celui du pivot, assurant ainsi la robustesse mécanique recherchée.

Une fois la structure modélisée en CAO, il a fallu procéder à sa fabrication. Plusieurs procédés étaient envisageables, mais rappelons que l'objectif du projet est la conception d'un robot autonome à partir d'un système « boîte noire » issu de la grande distribution, dans une perspective de robotique d'essaim. L'un des impératifs est donc de concevoir un système à la fois simple et reproductible, afin de permettre la fabrication d'un grand nombre d'unités en un minimum de temps.

Dans cette optique, la première solution retenue a été l'impression par résine. Ce procédé présente plusieurs avantages : il impose peu de contraintes de conception, permet la réalisation simultanée de plusieurs pièces complexes, offre une qualité de détail supérieure à l'impression 3D classique, et garantit ainsi une utilisation aisée ainsi qu'un gain de temps considérable par rapport à l'impression FDM ou au fraisage.

Cependant, les essais ont révélé certaines limites majeures rendant la résine inadaptée

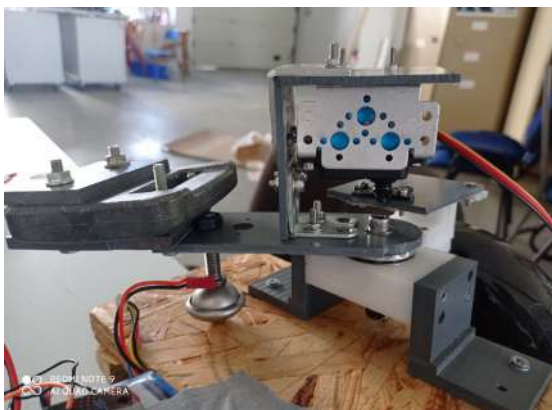


FIGURE 2.4 – Nouveau mécanisme dans sa réalisation finale

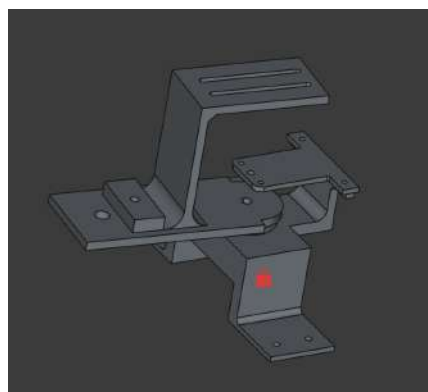


FIGURE 2.5 – CAO du nouveau mécanisme

à notre système. D'une part, elle ne permettait pas de respecter les cotes issues de la CAO, nécessitant un retravail systématique des pièces après chaque impression. D'autre part, la résine, matériau cassant, réagissait mal aux contraintes mécaniques : les pièces se brisaient fréquemment, rendant leur usinage ou ajustement pratiquement impossible.



FIGURE 2.6 – Pièces résines ayant cassées sous l'effort



FIGURE 2.7 – pièce résine ayant certains défauts d'impression

Il a donc été décidé de se tourner vers l'impression 3D FDM, qui offre une meilleure résistance mécanique tout en conservant la simplicité d'utilisation et la faible contrainte de conception.

Cependant, les essais ont également révélé certaines limites : certaines pièces, réalisables en impression résine et théoriquement compatibles avec la technologie FDM, échouaient fréquemment lors de l'impression. Cela a conduit à une révision de la conception de plusieurs éléments.

Bien que le résultat obtenu ait montré une robustesse supérieure à celui de l'impression résine, il s'est néanmoins avéré insuffisant pour nos besoins, les pièces finissant systématiquement par se briser.



FIGURE 2.8 – Pièce FDM ayant rompu sous l'effort.

Pour pallier les limites rencontrées avec l'impression 3D, il a été décidé de recourir au fraisage. Ce procédé permet la fabrication de pièces dans des matériaux tels que le POM-C ou le PVC, offrant une robustesse nettement supérieure. Toutefois, il présente plusieurs inconvénients. Contrairement aux techniques additives précédemment utilisées, le fraisage repose sur le retrait de matière, ce qui impose de repenser entièrement la conception des pièces afin d'optimiser leur usinage.

De plus, ce procédé exige davantage de temps et de savoir-faire. L'utilisation d'une fraiseuse requiert une formation spécifique, certaines opérations étant particulièrement complexes et nécessitant une réflexion approfondie en amont de la conception. Enfin, les fraiseuses elles-mêmes sont des machines coûteuses, fragiles et difficiles à réparer, ce qui ajoute une contrainte supplémentaire à leur emploi.

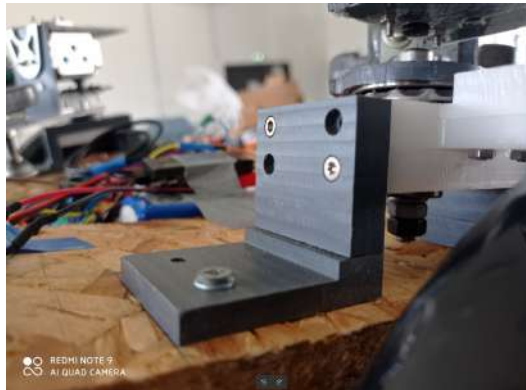


FIGURE 2.9 – Mécanisme en PVC/POM-C.

Bien que la conception des pièces se soit révélée plus complexe et chronophage, le résultat obtenu présente une structure robuste, offrant une excellente qualité de finition et pouvant être facilement retravaillée si nécessaire. C'est donc ce procédé qui a été retenu pour la fabrication du second prototype.



FIGURE 2.10 – Prototype final réalisé.

Comparaison des procédés

Trois procédés de fabrication ont été envisagés pour la réalisation des pièces : l’impression par résine, l’impression 3D FDM et le fraisage. Chacun présente des avantages et des inconvénients spécifiques, résumés dans le tableau 2.1.

En définitive, malgré une conception plus complexe et chronophage, le fraisage a été retenu pour la réalisation du second prototype. Il garantit une robustesse mécanique suffisante, une qualité de finition élevée et une possibilité de retravailler facilement les pièces si nécessaire.

2.2 Architecture électronique du système de contrôle

2.2.1 Schéma électronique

L’architecture électronique du système a été conçue dans la perspective de réaliser un robot entièrement autonome. Pour atteindre cet objectif, le système de contrôle doit être capable d’estimer en continu la position, la vitesse et le cap du robot afin de permettre une navigation fiable. Le design a également été pensé en vue d’une utilisation future dans des applications de robotique d’essaim, ce qui impose des contraintes fortes en termes de simplicité, de fiabilité et de coût réduit.

Dans cette optique, le choix a été fait d’utiliser un module GNSS *SIRF EM-506* pour l’estimation de la position, ainsi qu’une centrale inertielle *AHRS SparkFun OpenLog Artemis*, capable de mesurer les accélérations, les vitesses de rotation et le cap grâce à ses magnétomètres intégrés. L’ajout d’odomètres, permettant d’estimer la vitesse à partir de la rotation des roues, avait initialement été envisagé, mais cette solution a été abandonnée pour des raisons de coût et de complexité de mise en œuvre. À la place, l’estimation de la

TABLE 2.1 – Comparaison des procédés de fabrication

Procédé	Avantages	Inconvénients
Impression résine	- Grande liberté de conception - Possibilité de produire plusieurs pièces complexes simultanément - Très bonne qualité de détail	- Faible respect des cotes CAO - Nécessité de retravailler les pièces - Matériau fragile et cassant
Impression 3D FDM	- Simplicité d'utilisation - Peu de contraintes de conception - Meilleure résistance mécanique que la résine	- Échecs fréquents d'impression sur certaines pièces - Robustesse encore insuffisante - Pièces finissant par se briser
Fraisage	- Utilisation de matériaux robustes (POM-C, PVC, etc.) - Pièces solides et durables - Bonne qualité de finition - Pièces facilement retravaillables	- Conception plus complexe (procédé soustractif) - Temps de fabrication plus long - Nécessite savoir-faire et formation - Machines coûteuses et fragiles

vitesse sera obtenue via un observateur de Kalman, dont la conception et les performances seront détaillées ultérieurement.

L'ensemble des calculs et de la fusion de données est assuré par un microcontrôleur *Raspberry Pi 3*, qui communique avec le module GNSS par liaison UART et avec l'AHRS via USB. Pour les phases de test et de mise au point, une fonctionnalité de téléopération a été ajoutée : la Raspberry Pi est donc également connectée à un récepteur radio. Ce même microcontrôleur assure enfin la commande des actionneurs, en générant directement les signaux PWM nécessaires à leur pilotage.

L'alimentation du système repose sur deux batteries distinctes. Une première batterie LiPo 3S de 12V lié à un convertisseur de tension (12V vers 5V) est dédiée à l'alimentation de la Raspberry Pi 3 ainsi que de l'ensemble des capteurs, garantissant une autonomie électrique adaptée aux contraintes du système embarqué. Une seconde batterie LiPo 2S de 7.4V alimente les servomoteurs.

2.2.2 Composants

- **Capteur GNSS :**
- **SIRF EM-506**

Rôle : Estimation de la position en coordonnées géographiques.

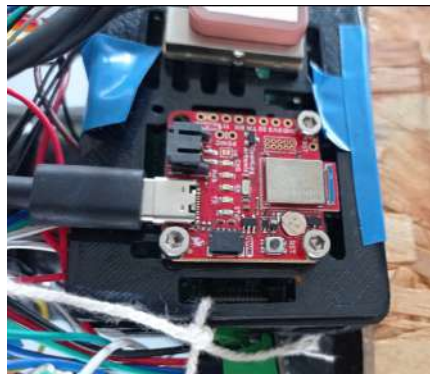


FIGURE 2.11 – AHRS Sparkfun OpenLog Artemis



FIGURE 2.12 – GNSS SIRF EM-506

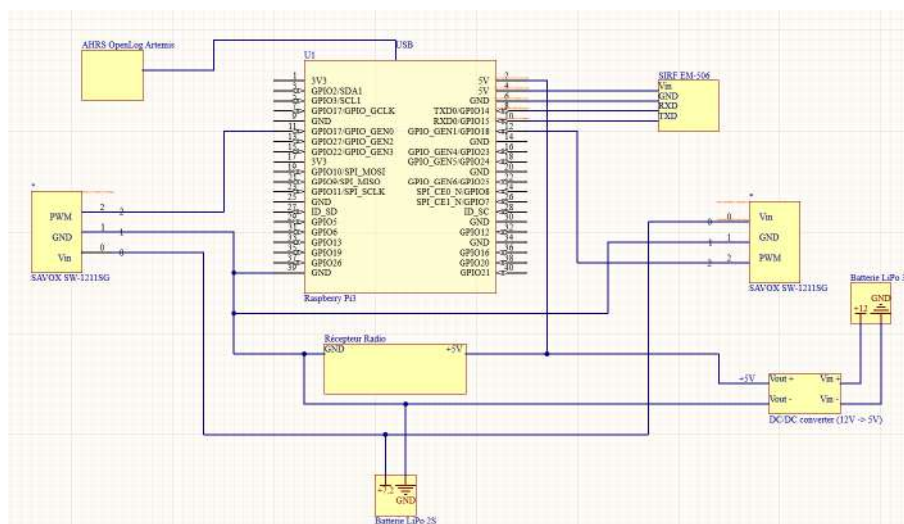


FIGURE 2.13 – Electronique embarquée.

Fréquence : 1Hz.

Précision : 2,50 m.

Alimentation : 3.3V depuis la Raspberry Pi3.

Interface : UART.

— Capteur AHRS :

— RAZOR SparkFun OpenLog Artemis [10]

Rôle : Fournit les données d'accélération, de vitesses de rotations et le cap (via la magnétométrie).

Fréquence : 50Hz.

Précision : $\pm 0.03g$ pour les accéléromètres, $\pm 8^\circ/s$ pour les gyroscopes, $\pm 2^\circ$ pour les magnétomètres.

Alimentation : 5V depuis la Raspberry Pi3.

— **Microcontrôleur :**

— **Raspberry Pi3**

Rôle : Traite les signaux, traduit la réception et commande les actionneurs.

Alimentation : 5V depuis la batterie 12V.

— **Actionneurs :**

— **Servomoteurs DiGit 4096**

Rôle : Déplacent les masses pour induire une accélération.

Alimentation : 7.4V depuis la batterie 2S.

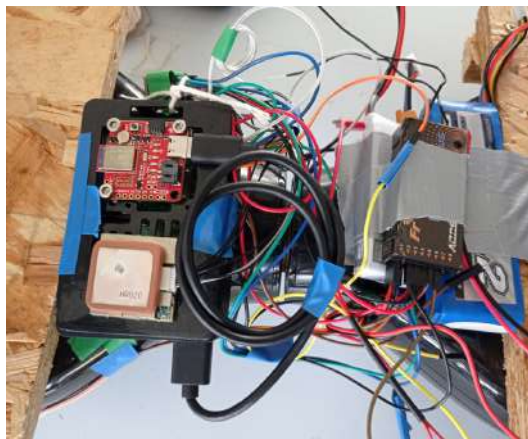


FIGURE 2.14 – Electronique embarquée.

2.3 Architecture logicielle du système de contrôle

L'architecture logicielle du système de contrôle repose sur une organisation en nœuds ROS 2, assurant à la fois l'acquisition des données capteurs, leur fusion et le calcul des consignes de commande. Les informations de position sont fournies par le module GNSS via le nœud *GPSNode*, qui publie sur le topic `/gps/fix`. Ces positions, exprimées initialement en coordonnées géographiques (latitude, longitude), sont projetées dans un repère cartésien en mètres dont l'origine correspond à la première mesure enregistrée par l'algorithme. C'est également dans ce même repère que sont définis les *waypoints* utilisés pour la navigation. Les mesures d'accélérations, de vitesses de rotation et de cap magnétique sont, quant à elles, transmises par l'AHRS au travers de son nœud *IMUNode*, sur les topics `/imu/data_raw` pour les données brutes et `/imu/yaw_deg` pour l'orientation.

Pour garantir l'autonomie du robot, trois grandeurs doivent être estimées en continu : la position, la vitesse et le cap. Si la position et le cap sont directement accessibles via les capteurs, la vitesse doit en revanche être estimée. De plus, la fréquence d'émission du GNSS, limitée à 1 Hz, s'avère insuffisante pour assurer une boucle de contrôle fluide. Afin de pallier ces limitations, un observateur de Kalman a été mis en place. Celui-ci reconstruit en temps réel la position, la vitesse et le cap à chaque appel des fonctions de *callback* liées aux capteurs, portant la fréquence effective des estimations à environ 50 Hz.

Cet estimateur est intégré au nœud **ControllerNode**, qui combine ainsi la fonction d'observation et la logique de commande. Le contrôleur reçoit en entrée un *waypoint* servant d'objectif de navigation. De là un guidage par champs de potentiel est mis en place, celui-ci est constitué d'une ligne attractive passant par le dernier waypoint atteint et le waypoint courant et d'un point attraitif situé sur cette ligne à 10m derrière le waypoint courant. À partir de ce champ, il déduit son cap de objectif qu'il compare au cap estimé par Kalman afin de calculer l'erreur d'orientation. Selon la valeur de cette erreur, le contrôleur choisit une consigne en vitesse de rotation : tourner à gauche (vitesse angulaire positive), tourner à droite (vitesse angulaire négative), ou poursuivre tout droit (vitesse nulle) si l'erreur reste dans un cône de tolérance prédéfini.

Une fois cette consigne angulaire définie ($-\omega_0$, ω_0 ou 0), un régulateur proportionnel-dérivé (PD) est appliqué sur l'erreur de vitesse de rotation entre la consigne et la mesure gyroscopique issue de l'AHRF. Cet asservissement fournit une correction dynamique robuste de la trajectoire.

En parallèle, le contrôleur ajuste également la vitesse linéaire. Il compare la consigne de vitesse (en m/s) à la vitesse estimée par Kalman et en déduit un *offset* angulaire appliqué aux actionneurs. Si la vitesse est trop faible, cet offset incline les masses vers l'avant afin de générer une tendance à l'accélération ; inversement, si la vitesse est trop élevée, un offset négatif induit une décélération. Si l'erreur reste dans la tolérance, l'offset est nul.

La commande finale envoyée aux servomoteurs résulte de la combinaison de cet offset et du signal d'asservissement (noté u) pour obtenir une consigne angulaire totale réinterprétée en PWM par le microcontrôleur :

$$\begin{bmatrix} \text{Consigne droite} \\ \text{Consigne gauche} \end{bmatrix} = \begin{bmatrix} 45 \\ 45 \end{bmatrix} + \text{offset} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + u \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad (2.1)$$

où 45 correspond à la position centrale neutre des servomoteurs.

Ainsi, les masses se déplacent et induisent une consigne en accélération sur les roues de l'hoverboard. Ce mouvement modifie à son tour les mesures des capteurs, qui sont intégrées par l'observateur de Kalman, fermant ainsi la boucle de contrôle complète.

Une démonstration vidéo du système d'asservissement en vitesse rotation est disponible à l'adresse suivante : <https://youtube.com/shorts/kuQPnSBtMAO>

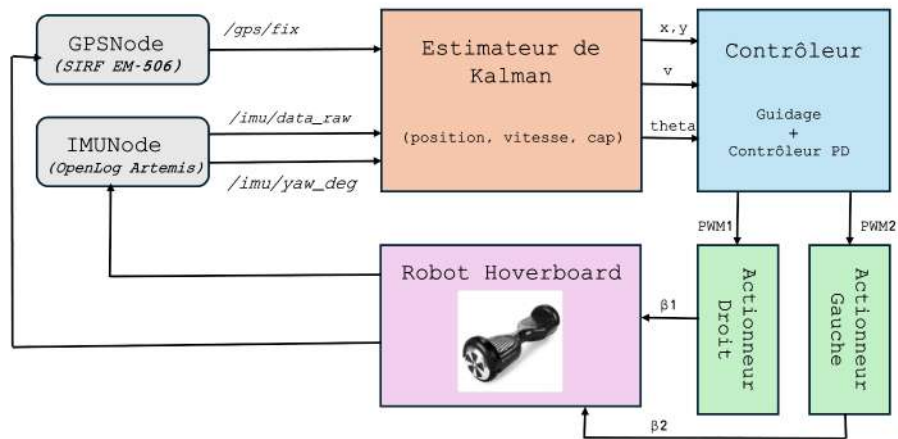


FIGURE 2.15 – Schéma architecture logicielle.

2.4 Mode de fonctionnement

Au cours du développement du prototype, trois modes de fonctionnement ont été implémentés. Le premier, déjà évoqué, est un mode de téléopération. Il a été conçu principalement pour faciliter les phases de mise au point et de débogage, tout en offrant une preuve de concept du système. Les deux autres modes correspondent à un fonctionnement entièrement autonome et se distinguent par la nature de leur observateur de Kalman. Dans le premier cas, l'observateur estime séparément les composantes de la position et de la vitesse selon les axes x et y , à partir des mesures du GNSS et du gyroscope sur l'axe z . Dans le second cas, l'observateur reconstruit la position ainsi que la norme du vecteur vitesse, en intégrant cette fois les données du GNSS, du gyroscope selon z et du cap magnétique fourni par l'AHRS via ses magnétomètres.

2.4.1 Mode téléopération

Principe

Le mode de téléopération permet à un opérateur humain de contrôler directement le robot à distance. Il est particulièrement utile pour vérifier le bon fonctionnement du système et l'ajustement des paramètres du contrôleur. En effet, la téléopération utilise le même contrôleur que pour le mode autonome présenté plus tôt à la différence près que ici c'est l'opérateur qui constitue la boucle d'asservissement et non pas l'observateur de Kalman et que l'objectif est déterminé non pas par des waypoints mais par l'utilisateur. De ce fait l'asservissement du cap et de la vitesse linéaire se fait au travers des observations du pilote qui ne communique que deux signaux PWM via la manette. Le premier pour communiquer

la volonté d'avancer et le second la volonté de tourner. Le signal qui communique la volonté d'avancer est réinterprété par le microcontrôleur comme l'offset angulaire à imposer aux actionneurs tandis que le signal communiquant la volonté de tourner est réinterprété comme la consigne de vitesse angulaire selon z . De là, ces consignes sont communiquées au contrôleur qui se charge de créer le PWM correspondant en utilisant la formule 2.1.

Une vidéo de démonstration du mode de téléopération est disponible à l'adresse suivante : <https://youtu.be/zYmKNu2Ntm8>

2.4.2 Mode autonome sans boussole

Principe

Comme mentionné précédemment (2.3), l'autonomie du robot repose sur sa capacité à estimer en continu trois grandeurs essentielles : la position, la vitesse et le cap. Pour cela, nous disposons des mesures de position fournies par le GNSS ainsi que des données issues de l'AHRs, notamment le gyroscope sur l'axe z et les magnétomètres. Dans un premier temps, le choix a été fait de ne pas exploiter ces derniers pour deux raisons principales. D'une part, l'architecture électronique interne de l'hoverboard, inconnue, pourrait générer des interférences susceptibles d'altérer la fiabilité de la boussole. D'autre part, il est souhaitable de pouvoir évoluer dans des environnements où le champ magnétique terrestre peut être perturbé, comme en milieu urbain dense. Ce choix présente également un intérêt académique, puisqu'il permet d'évaluer dans quelle mesure une navigation fiable peut être obtenue en s'appuyant uniquement sur un GNSS peu coûteux.

L'estimation simultanée de la position, de la vitesse et du cap soulève néanmoins deux difficultés majeures. Tout d'abord, ni la vitesse ni le cap ne sont directement mesurables : la seule observation disponible est la position. Ensuite, cette mesure est à la fois bruitée (avec une précision de l'ordre de $\pm 2.50\text{m}$) et lente, la fréquence d'échantillonnage du GNSS étant limitée à 1 Hz. Ces contraintes rendent impossible un contrôle satisfaisant en temps réel. Pour pallier ces limitations, le recours à un observateur de Kalman s'impose [5].

Observateur de Kalman

Hypothèse 1

L'hoverboard est assimilable à une voiture de Dubins

On souhaite estimer la position (x, y) , la vitesse $\vec{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$ et le cap θ du robot. Or, dans le cas d'un modèle de Dubins :

$$\theta = \arctan\left(\frac{v_y}{v_x}\right)$$

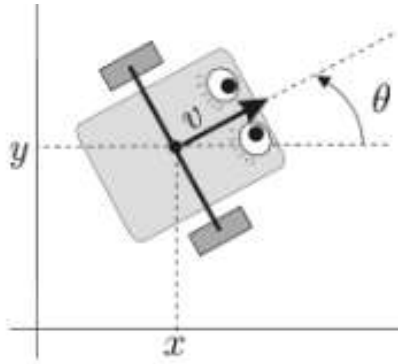


FIGURE 2.16 – Modèle de voiture de Dubins.

Il est donc possible de réduire le problème en estimant uniquement les composantes de la position et de la vitesse selon les axes x et y .

Notre vecteur d'état devient donc :

$$X = \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x \\ y \\ v * \cos(\theta) \\ v * \sin(\theta) \end{bmatrix}$$

On a alors :

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v}_x \\ \dot{v}_y \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ \dot{v} * \cos(\theta) - v * \sin(\theta) \dot{\theta} \\ \dot{v} * \sin(\theta) + v * \cos(\theta) \dot{\theta} \end{bmatrix}$$

Pour simplifier le problème et conserver un observateur de Kalman linéaire, on fait une seconde hypothèse.

Hypothèse 2

L'accélération de l'hoverboard est nulle, soit : $\dot{v} = 0$.

On a donc :

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v}_x \\ \dot{v}_y \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ -v * \sin(\theta) \dot{\theta} \\ v * \cos(\theta) \dot{\theta} \end{bmatrix}$$

soit,

$$\dot{X} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_3 \\ x_4 \\ -\omega * x_4 \\ \omega * x_3 \end{bmatrix} \text{ avec } \omega = \dot{\theta}$$

donc,

$$\dot{X} = AX, \text{ avec } A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\omega \\ 0 & 0 & \omega & 0 \end{bmatrix}$$

Il est important de noter que ω correspond à la mesure du gyroscope selon z , et que cette matrice sera donc mise à jour à chaque mesure.

Pour ce Kalman, notre unique observation est la position (x, y) , soit :

$$Y = CX, \text{ avec } C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \text{ et } Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Discussion

L'hypothèse 2 constitue certes une simplification notable du modèle réel. Néanmoins, elle reste justifiée par le fait que, sur de courtes périodes, les variations de vitesse demeurent généralement limitées et peuvent être prises en compte par la matrice de covariance associée aux incertitudes du modèle, notamment sur les composantes v_x et v_y . De plus, cette approche permet de conserver la linéarité de l'observateur de Kalman, simplifiant ainsi son implémentation et son calcul en temps réel. Dans notre algorithme, l'observateur de Kalman est appelé en mode prédiction à chaque mesure du gyroscope et en mode correction à chaque mesure du GNSS. Cela nous fournit une estimation de la position et de la vitesse à une fréquence d'environ 10Hz, bien supérieure à celle du GNSS seul.

Une autre critique pourrait être formulée à l'égard de la seconde observation prise en compte à savoir celle du gyroscope selon z . Celle-ci est directement intégrée à la matrice plutôt que de passer par le vecteur d'observation, cette approche est nécessaire si l'on veut garder un Kalman classique mais elle peut également introduire des erreurs. En effet, le gyroscope n'est pas exempt de bruit et d'erreurs de mesure, or ici nous le considérons comme une mesure parfaite. Pour autant, les gyroscopes étant des capteurs relativement stables et faibles, cette hypothèse reste raisonnable dans le cadre de notre application.

2.4.3 Mode autonome avec boussole

Principe

Ce second mode de fonctionnement repose sur le même principe que le précédent, à savoir l'utilisation d'un observateur de Kalman pour estimer l'état du système. Cependant, à la différence du premier, celui-ci intègre les données issues des magnétomètres de l'AHRs pour estimer le cap θ du robot. Ayant désormais une mesure du cap θ intégrée, notre observateur de Kalman doit être modifié. Il nous permettra de comparer l'approche avec et sans boussole.

Observateur de Kalman

Hypothèse 1

L'hoverboard est assimilable à une voiture de Dubins

On souhaite désormais uniquement estimer la position (x, y) , la norme de la vitesse v . Notre vecteur d'état devient donc :

$$X = \begin{bmatrix} x \\ y \\ v \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

On a alors,

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v * \cos(\theta) \\ v * \sin(\theta) \\ \dot{v} \end{bmatrix}$$

Pour conserver notre Kalman linéaire, nous devons également faire l'hypothèse que $\dot{v} = 0$.

Hypothèse 2

L'accélération de l'hoverboard est nulle, soit : $\dot{v} = 0$.

On a donc,

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v * \cos(\theta) \\ v * \sin(\theta) \\ 0 \end{bmatrix}$$

soit,

$$\dot{X} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} x_3 * \cos(\theta) \\ x_3 * \sin(\theta) \\ 0 \end{bmatrix}$$

donc,

$$\dot{X} = AX, \text{ avec } A = \begin{bmatrix} 0 & 0 & \cos(\theta) \\ 0 & 0 & \sin(\theta) \\ 0 & 0 & 0 \end{bmatrix}$$

Tout comme le premier observateur de Kalman l'observation du cap θ est ici directement intégrée dans la matrice A , elle est donc mise à jour à chaque mesure des magnétomètres.

Pour ce Kalman également l'unique observation considérée comme telle est celle de la position (x, y) , ce qui nous donne notre fonction d'observation :

$$Y = CX, \text{ avec } C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \text{ et } Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Discussion

Les hypothèses faites pour ce Kalman sont similaires à celles du précédent, avec une simplification notable du modèle réel qui se justifie par le comportement de l'hoverboard et la matrice de covariance G_α .

Chapitre 3

Expérimentation

3.1 Preuve de Concept

Afin de valider la modélisation proposée, nous avons conçu un dispositif expérimental permettant de simuler l'imposition d'un angle talocrural. Ce système consistait en une planche d'aggloméré fixée sur l'une des plateformes de l'hoverboard, sur laquelle était monté un servomoteur actionnant une masse. Cette masse était installée à l'extrémité d'un tube en PVC, lui-même soutenu par une liaison sphère-plan, de manière à reproduire le mouvement souhaité. Le servomoteur était piloté à distance grâce à un récepteur radio connecté à une télécommande. La vidéo de l'essai est disponible au lien suivant :

<https://youtu.be/PdPQJRcWlWA>

Au cours de cet essai, il a été démontré qu'il était possible de provoquer et contrôler l'accélération du robot en modifiant simplement l'angle d'inclinaison. Cette étape confirme l'hypothèse fondamentale du modèle physique choisi et pose les bases du contrôle indirect par action mécanique.

3.2 Essais Kalman sans boussole

Dans le but d'évaluer la performance de notre observateur de Kalman et de vérifier la validité des hypothèses fortes formulées sur le modèle de l'hoverboard, nous avons réalisé, avant tout essai autonome sur ce dernier, une première expérimentation pratique. Pour cela, un chariot a été utilisé comme plateforme de test. Notre électronique embarquée, comprenant le microcontrôleur, le module GNSS et l'AHRS, a été installée au-dessus de l'essieu arrière. L'expérience a consisté à effectuer un tour complet de piste d'athlétisme, puis à tenter, à l'aide de l'observateur de Kalman et des données enregistrées, de reconstruire avec précision la trajectoire parcourue.

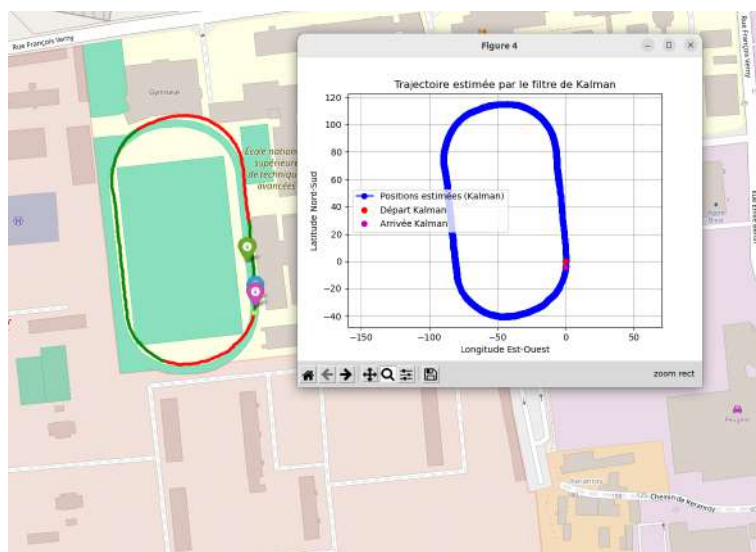


FIGURE 3.1 – Trajectoire GNSS sur le stade.

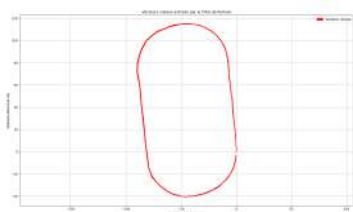


FIGURE 3.2 – Estimation du vecteur vitesse par le Kalman.

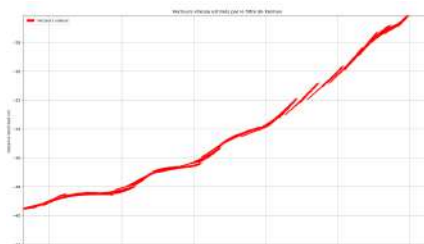


FIGURE 3.3 – Détail de du tracé sur la trajectoire particulière.

Comme l'illustre la figure 3.1, l'observateur de Kalman permet d'obtenir une estimation globale de la trajectoire relativement précise. Toutefois, on observe que certaines portions rectilignes ne sont pas parfaitement droites, en raison des imprécisions inhérentes aux mesures GNSS. Malgré cela, l'estimation du vecteur vitesse produite par l'observateur apparaît particulièrement satisfaisante.

Afin d'évaluer plus finement la qualité de la reconstruction, une seconde expérience a été menée sur une portion de la piste, consistant à parcourir une trajectoire volontairement sinueuse. Les résultats présentés en figure 3.3 montrent que l'observateur restitue correctement une trajectoire lisse et sinueuse, fidèle au mouvement réellement effectué. Cette expérience a permis de valider la pertinence du modèle et de l'observateur de Kalman, tout en optimisant ses paramètres.

Il est toutefois important de noter que, durant ces essais avec le chariot, la vitesse est restée relativement constante, du fait qu'il était tracté manuellement. Il sera donc particulièrement intéressant, dans la suite du travail, d'analyser dans quelle mesure la

précision de l'observateur est affectée lors d'expérimentations réelles à bord de l'hoverboard, où la vitesse est naturellement plus variable.

3.3 Essais autonome Kalman sans boussole

Une fois l'observateur de Kalman validé, nous avons pu engager la phase de tests en conditions réelles sur l'hoverboard. L'objectif fixé consistait à atteindre un point GPS prédéfini. La figure 3.4 illustre un essai concluant au cours duquel l'hoverboard est parvenu à atteindre sa cible avec succès.

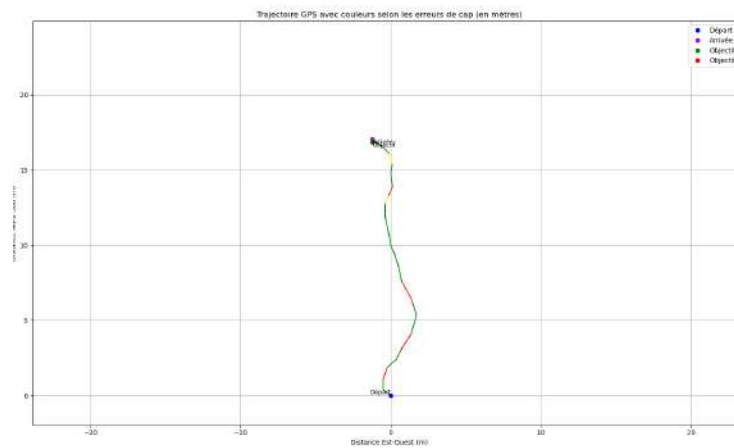


FIGURE 3.4 – Trajectoire concluante, tracé GNSS.

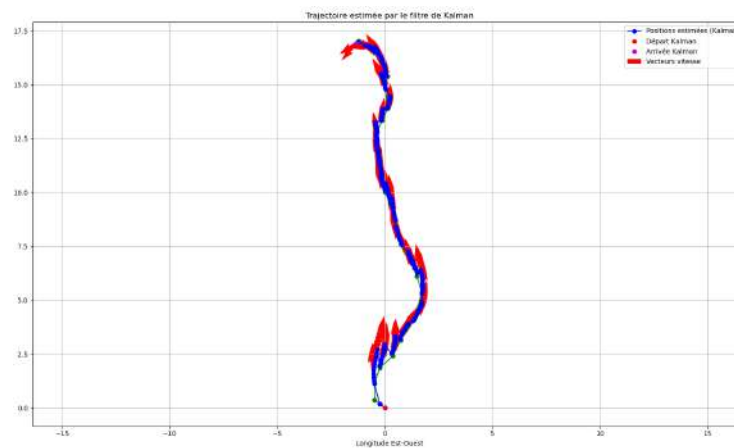


FIGURE 3.5 – Trajectoire conluante estimée par Kalman.

Comme le montrent également les figures 3.5 et 3.6, le système a démontré une bonne

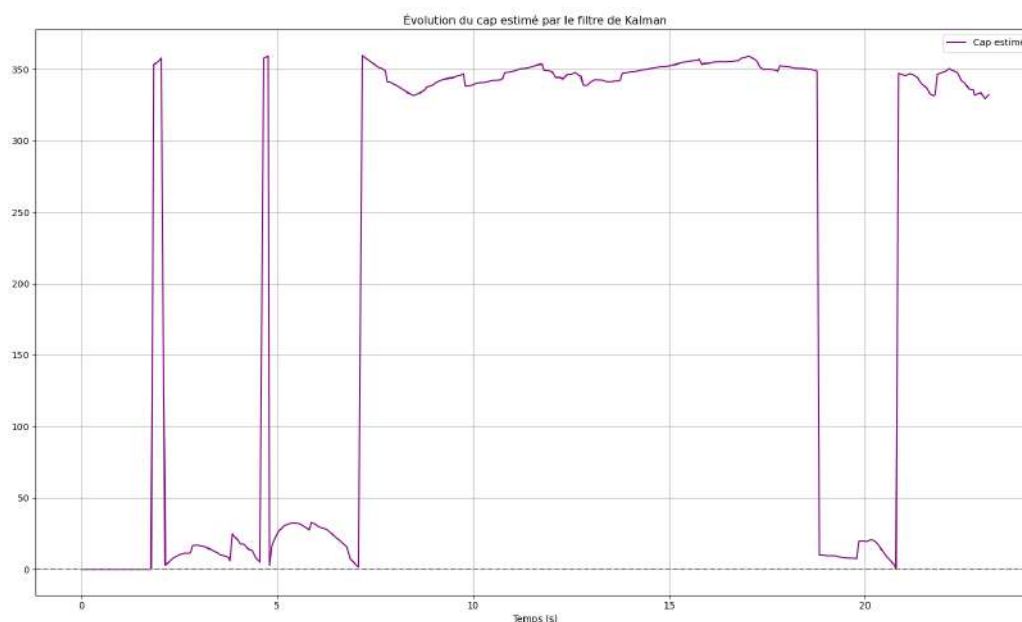


FIGURE 3.6 – Cap estimé par Kalman.

précision dans l'atteinte de l'objectif, accompagnée d'une estimation du cap stable et cohérente.

Cependant, si plusieurs essais se sont révélés concluants, d'autres ont mis en lumière certaines limitations de notre approche, en particulier dans la gestion des variations de vitesse et des perturbations extérieures. Lors de nombreux tests, il a été observé que l'hoverboard pouvait finir par s'arrêter sous l'effet du déplacement des masses, sans que l'algorithme ne le détecte. Dans ce cas, le système considérait à tort que la vitesse restait suffisante et n'envoyait donc aucune consigne d'accélération, conduisant l'hoverboard à un état statique qui dégradait progressivement l'estimation du Kalman.

Un second problème est apparu au moment du lancement de l'hoverboard. En raison de l'imprécision du GNSS [9] [8], si la vitesse initiale n'était pas assez élevée, les mesures consécutives pouvaient ne pas refléter correctement la dynamique du mouvement, perturbant ainsi fortement le filtre de Kalman. Enfin, l'instabilité du GNSS s'est révélée être une source de difficulté supplémentaire : certaines sessions offraient des mesures relativement fiables, tandis qu'à d'autres moments les données étaient fortement bruitées, voire entachées de sauts brutaux, rendant l'estimation impossible et condamnant l'essai à l'échec.

Une autre contrainte rencontrée concernait la piste d'essai elle-même. Avec une largeur d'environ 5m, et compte tenu de l'imprécision du GNSS estimée à $\pm 2.50\text{m}$, l'hoverboard devait évoluer avec une grande précision pour éviter de sortir de la piste, ce qui limitait considérablement la marge de manœuvre.

Ces expérimentations ont donc démontré la faisabilité de transformer l'hoverboard en un robot autonome, et ont confirmé la possibilité de réaliser une navigation basée sur un observateur de Kalman n'utilisant que les données d'un capteur GNSS et d'un gyroscope. Toutefois, elles ont également montré que cette approche, en l'état, ne permettait pas d'obtenir une fiabilité suffisante. Si l'on souhaite faire de l'hoverboard une véritable plateforme robotique robuste, il devient indispensable d'améliorer le système de contrôle. La section suivante se concentre ainsi sur l'intégration d'un capteur de boussole magnétique afin de renforcer l'estimation du cap et la stabilité globale du système.

3.4 Essais autonome Kalman avec boussole

Ayant montré, lors des essais précédents, la nécessité d'une nouvelle approche, une seconde campagne expérimentale a été menée en intégrant cette fois-ci les mesures issues de la boussole. La première étape a consisté à vérifier que l'hoverboard ne générât pas de perturbations magnétiques significatives susceptibles de dégrader les mesures du magnétomètre, ce qui a pu être confirmé grâce aux observations réalisées lors des tests initiaux. Il a ensuite été nécessaire de mettre à jour le logiciel de l'AHRs, qui ne proposait pas par défaut de calibration magnétique des capteurs [10]. Cette mise à jour a non seulement permis d'activer cette fonctionnalité, mais a également modifié la structure des données reçues tout en augmentant la fréquence d'émission, passant de 10Hz à 50Hz.

Une fois la boussole correctement intégrée, une nouvelle série d'essais a été lancée, avec un objectif plus ambitieux que dans la campagne précédente : au lieu de viser un simple *waypoint*, le robot devait cette fois parcourir un tour complet de piste en suivant une liste de points GPS prédéfinis. Contrairement aux premiers essais, cette démarche s'est révélée beaucoup plus concluante.

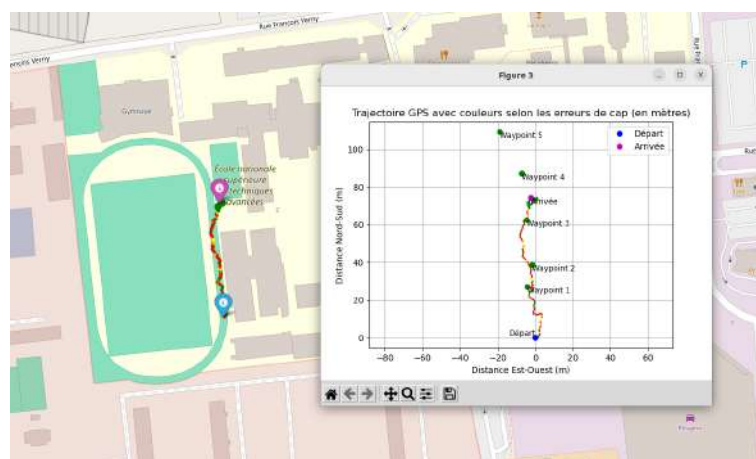


FIGURE 3.7 – Trajectoire GNSS sur le stade.

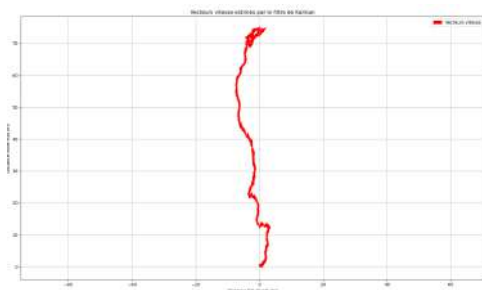


FIGURE 3.8 – Estimation du vecteur vitesse par le Kalman.

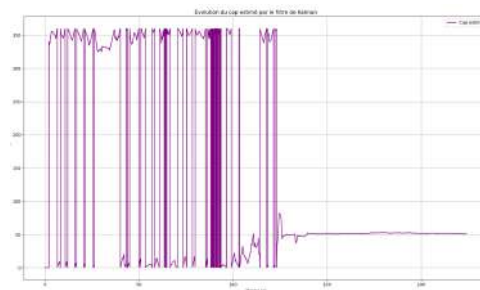


FIGURE 3.9 – Détail de du tracé sur la trajectoire particulière.

La figure 3.7 illustre l'un de ces essais, où l'hoverboard a réussi à parcourir environ une centaine de mètres en franchissant trois *waypoints*, le tout avec une trajectoire globalement stable. Une légère déviation hors de la piste a néanmoins été observée en fin de parcours, probablement imputable à une imprécision du GNSS. Ces résultats suggèrent que l'intégration de la boussole améliore nettement la robustesse de l'algorithme, bien que la précision globale demeure fortement dépendante de la qualité des mesures GNSS.

Faute de temps, le nombre d'essais avec cette configuration est resté limité. Il aurait pourtant été particulièrement intéressant de multiplier les tests afin de quantifier plus précisément le gain de robustesse apporté par la boussole et d'évaluer, le cas échéant, l'intérêt de recourir à un module GNSS plus performant. De plus un défaut important sur la structure mécanique est apparu lors de ces derniers essais, il semblerait que la pièce faisant lien entre le servomoteur et la partie fixe au niveau de l'axe ne soit pas assez résistante. En effet, après chaque quinzaine d'essai la pièce doit être remplacée sous peine de ne plus pouvoir contrôler le robot. il apparait donc nécessaire de revoir la conception mécanique afin de la rendre plus robuste.

Une vidéo de l'essai est disponible à l'adresse suivante :

<https://youtu.be/6aMcUMFg6Go>

3.5 Conclusion des essais

Les différentes campagnes d'essais menées ont permis de valider la faisabilité d'une navigation autonome de l'hoverboard tout en mettant en évidence les limites et les perspectives d'amélioration de notre approche. Les premiers tests, réalisés sans boussole, ont démontré qu'il était possible de transformer l'hoverboard en une plateforme robotique capable d'atteindre un objectif GPS prédéfini à l'aide d'un observateur de Kalman ne s'appuyant que sur les mesures GNSS et gyroscopiques. Toutefois, ces expérimentations ont aussi révélé plusieurs limitations importantes : sensibilité aux variations de vitesse, arrêts imprévus

non détectés, dépendance forte à la qualité du GNSS et instabilité des estimations dans certaines conditions.

Pour pallier ces faiblesses, une deuxième campagne a intégré les mesures issues de la boussole. Cette évolution s'est traduite par une amélioration notable de la robustesse et de la stabilité de la trajectoire, permettant à l'hoverboard de suivre une série de waypoints sur plusieurs dizaines de mètres. Si ces résultats sont encourageants, ils mettent également en lumière la dépendance persistante vis-à-vis de la précision du GNSS, dont les imprécisions et instabilités continuent de limiter la fiabilité du système.

Ainsi, ces campagnes d'essais confirment à la fois la pertinence de l'approche retenue et la nécessité de poursuivre son amélioration. L'intégration d'un capteur magnétique constitue une étape clé vers une navigation plus robuste, mais des efforts supplémentaires devront être consacrés à la gestion des imprécisions GNSS et à la prise en compte des variations dynamiques de vitesse, afin de doter l'hoverboard d'une architecture de contrôle réellement fiable et adaptée à une utilisation en tant que vecteur robotique autonome.

Conclusion et perspectives

Ce travail démontre qu'il est possible de concevoir une plateforme robotique autonome robuste, économique et reproductible à partir d'un hoverboard issu de la grande distribution, en s'appuyant sur une approche de modélisation empirique adaptée aux systèmes de type boîte noire. La démarche conduit à la réalisation d'un système de contrôle intégralement dédié à la robotique d'essaim, intégrant à la fois une architecture mécanique optimisée, une électronique embarquée modulaire et une logique de fusion de données permettant la navigation autonome. Les expérimentations menées, depuis la preuve de concept jusqu'aux tests avancés avec intégration du capteur magnétique, ont permis de valider les choix méthodologiques tout en mettant en lumière les défis inhérents aux capteurs low-cost, notamment la dépendance à la précision GNSS et la gestion des perturbations dynamiques. Les résultats montrent l'apport significatif de la fusion multi-capteurs pour améliorer la stabilité et la robustesse du guidage autonome. En perspective, il apparaît intéressant de continuer les essais avec la configuration intégrant la boussole afin de mesurer l'impact réel des performances du GNSS. L'intégration d'une caméra Picam, capteur simple d'utilisation peu onéreux et compatible avec notre microcontrôleur pourrait constituer une piste d'amélioration intéressante, en remplacement de l'AHRs par exemple, offrant ainsi une solution alternative permettant de ne pas alourdir la charge capteurs et qui nous fournirait plus d'information sur la vitesse notamment. Par ailleurs, une révision de l'architecture mécanique semble nécessaire, sa conception actuelle s'avérant coûteuse en temps et insuffisamment fiable. Il pourrait être également intéressant d'appliquer le résultats de ces travaux sur d'autres plateformes similaires telles que la solowheel ou les roues autonomes qui pourraient facilement être fabriquées à partir de roues de vélo électrique. Ces travaux constituent ainsi une première étape vers la mise à l'échelle de l'hoverboard pour des applications en robotique d'essaim, ouvrant la voie à de nouvelles perspectives d'expérimentation collaborative et de recherche appliquée.

Bibliographie

- [1] E. Şahin, “Swarm Robotics : From Sources of Inspiration to Domains of Application,” in *Swarm Robotics*, LNCS 3342, Springer, 2005.
- [2] M. Dorigo and E. Şahin, “Swarm robotics—special issue editorial,” *Autonomous Robots*, vol. 17, no. 2–3, pp. 111–113, 2004.
- [3] L. Jaulin, “Modeling and control of an hoverboard,” *International Journal of Control, Automation and Systems*, 2021.
- [4] K. Pathak, J. Franch, and S. Agrawal, “Analysis and Control of a Robotic Vehicle with an Inverted Pendulum,” *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 726–736, 2005.
- [5] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [6] B. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. Wiley, 2001.
- [7] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.
- [8] R. Siegwart, I. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd ed. MIT Press, 2011.
- [9] P. Misra and P. Enge, *Global Positioning System : Signals, Measurements and Performance*, 2nd ed. Ganga-Jamuna, 2012.
- [10] SparkFun Electronics, *OpenLog Artemis Product Documentation*, 2020. [Online]. Available : <https://www.sparkfun.com>
- [11] L. Jaulin, “Modeling and control of a Segway robot,” *International Journal of Control, Automation and Systems*, vol. 18, no. 12, pp. 3148–3157, 2020.

Résumé — Ce travail présente la conception et la mise en œuvre d’une plateforme robotique autonome basée sur un hoverboard issu de la grande distribution. L’approche adoptée repose sur une modélisation empirique d’un système de type boîte noire, permettant de contourner l’absence d’accès à l’architecture électronique interne et aux algorithmes de contrôle d’origine. La plateforme développée associe une architecture mécanique dédiée, une électronique embarquée modulaire et une logique logicielle organisée autour de ROS 2, intégrant un observateur de Kalman pour la fusion de données multi-capteurs.

L’hoverboard a ainsi été transformé en un vecteur autonome capable d’atteindre des objectifs GPS prédéfinis. Les premières expérimentations, basées uniquement sur le GNSS et le gyroscope, ont mis en évidence les limites liées à l’imprécision et à la faible fréquence des capteurs low-cost. L’intégration d’un magnétomètre a ensuite permis d’améliorer significativement la robustesse du guidage et la stabilité des estimations de cap. Toutefois, la dépendance au GNSS demeure une contrainte majeure et la conception mécanique actuelle reste perfectible.

Les résultats obtenus montrent qu’il est possible de concevoir une plateforme peu coûteuse, reproductible et adaptée à des applications futures en robotique d’essaim. Ce travail ouvre ainsi la voie à des expérimentations collaboratives sur des systèmes multi-robots et met en lumière les défis associés à l’utilisation de capteurs grand public dans des environnements dynamiques.

Mots clés : Hoverboard, robotique autonome, modélisation empirique, fusion de données, ROS 2, swarm robotics.

Polytech Angers
62, avenue Notre Dame du Lac
49000 Angers