

Stratégie de transition entre localisation autonome et localisation intérieure de haute précision pour un robot mobile manipulateur

par

Alexandre LAPOINTE

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE LA MAÎTRISE
AVEC MÉMOIRE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M. Sc. A.

MONTREAL, LE 12 AOÛT 2026

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC



Alexandre Lapointe, 2026



Cette licence Creative Commons signifie qu'il est permis de diffuser, d'imprimer ou de sauvegarder sur un autre support une partie ou la totalité de cette oeuvre à condition de mentionner l'auteur, que ces utilisations soient faites à des fins non commerciales et que le contenu de l'oeuvre n'ait pas été modifié.

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE:

M. David St-Onge, directeur de mémoire
Département de génie mécanique, École de technologie supérieure

M. Lionel Birglen, codirecteur
Département de génie mécanique, Polytechnique Montréal

M. Jean-Philippe Roberge, président du jury
Département de génie de la production automatisée, École de technologie supérieure

M. Bruno Belzile, membre du jury
Département de génie mécanique, École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 3 AOÛT 2026

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Je tiens d'abord à remercier mon directeur de recherche, le professeur David St-Onge, pour son encadrement, sa disponibilité et ses conseils tout au long de ce projet de maîtrise. Je remercie également les membres du laboratoire INIT Robots et du laboratoire de métrologie pour leur aide, leurs échanges techniques et leur soutien lors du développement, des essais et de l'analyse des résultats.

Je remercie l'École de technologie supérieure pour son soutien financier, notamment par l'entremise de la bourse d'excellence de l'ÉTS, ainsi que le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) pour le financement accordé dans le cadre de la Bourse de recherche du Canada à la maîtrise. Je remercie aussi Kinova Robotics, partenaire industriel de ce projet de recherche. Ces différents appuis ont contribué directement à la réalisation de mes travaux de recherche et à la rédaction de ce mémoire.

Stratégie de transition entre localisation autonome et localisation intérieure de haute précision pour un robot mobile manipulateur

Alexandre LAPOINTE

RÉSUMÉ

Ce mémoire étudie la transition de localisation d'un robot mobile manipulateur entre une localisation autonome et une localisation de haute précision fournie, dans une zone instrumentée, par un système de localisation intérieure destiné à la métrologie. L'objectif est de conserver une navigation autonome hors de cette zone et d'exploiter la mesure externe lorsque le robot s'y trouve, sans perturber les repères requis pour la navigation et la manipulation. Le travail est mené sur le RanGen, une base mobile à quatre roues directrices portant un bras Kinova équipé pour exploiter cette mesure externe.

Il traite du jumeau numérique, de l'intégration de cette mesure au bras, de la calibration des repères et de la transition entre sources globales de localisation. L'hypothèse de travail est qu'une architecture de transition, fondée sur des interfaces ROS 2 cohérentes et des calibrations géométriques, peut relier navigation autonome, localisation de haute précision et manipulation locale. Les méthodes combinent modélisation numérique, intégration ROS 2, calibrations rigides et essais comparés sur le robot réel.

Les résultats montrent une continuité fonctionnelle entre la simulation et le robot réel. En manipulation, une boucle de correction de la pose du bras fondée sur la mesure externe réduit la RMSE 3D d'environ 7,7 mm à 1,7 mm. En navigation, la localisation de haute précision réduit l'erreur médiane aux poses cibles Nav2 de 78,0 mm à 3,5 mm, et l'erreur de lacet de $2,17^\circ$ à $0,24^\circ$. La démonstration finale valide l'enchaînement entre la navigation autonome, la transition vers cette localisation de haute précision et l'exécution d'une tâche d'assemblage par le bras robotique. Le mémoire conclut que cette transition est fonctionnelle dans le contexte expérimental étudié. Les suites recommandées portent sur la robustesse du dispositif de mesure externe monté sur le bras et sur la définition d'un critère de décision permettant de déterminer quand le robot doit revenir à sa localisation autonome lorsque cette mesure n'est plus disponible ou n'est plus suffisamment fiable.

Mots-clés: robot mobile manipulateur, localisation intérieure de haute précision, transition de localisation, jumeau numérique, métrologie, manipulation robotique

Localization Transition Strategy from Autonomous Localization to High-Precision Indoor Localization for a Mobile Manipulator Robot

Alexandre LAPOINTE

ABSTRACT

This thesis studies the localization transition of a mobile manipulator between autonomous localization and high-precision localization provided, in an instrumented area, by an indoor localization system designed for metrology. The objective is to preserve autonomous navigation outside this area and to use the external measurement when the robot is inside it, without disrupting the reference frames required for navigation and manipulation. The work is conducted on the RanGen, a four-wheel-steering mobile base carrying a Kinova arm equipped to use this external measurement.

It addresses the digital twin, the integration of this measurement with the arm, frame calibration, and the transition between global localization sources. The working hypothesis is that a transition architecture, based on consistent ROS 2 interfaces and geometric calibrations, can connect autonomous navigation, high-precision localization, and local manipulation. The methods combine digital modeling, ROS 2 integration, rigid-body calibrations, and comparative tests on the real robot.

The results show functional continuity between simulation and the real robot. In manipulation, a feedback loop correcting the arm pose based on the external measurement reduces the 3D RMSE from about 7.7 mm to 1.7 mm. In navigation, the high-precision localization reduces the median error at the Nav2 target poses from 78.0 mm to 3.5 mm, and the yaw error from 2.17° to 0.24° . The final demonstration validates the sequence linking autonomous navigation, the transition to this high-precision localization, and the execution of an assembly task by the robotic arm. The thesis concludes that this transition is functional in the experimental context studied. Recommended future work concerns the robustness of the external measurement setup mounted on the arm and the definition of a decision criterion for determining when the robot should return to autonomous localization if this measurement is no longer available or no longer sufficiently reliable.

Keywords: mobile manipulator, high-precision indoor localization, localization transition, digital twin, metrology, robotic manipulation

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 REVUE DE LA LITTÉRATURE	7
1.1 Infrastructures de métrologie pour le suivi de position et de pose	7
1.2 Jumeau numérique en robotique	9
1.3 Calibration et commande du manipulateur à partir d'une mesure externe	11
1.4 Architecture de localisation et transition entre sources globales	12
1.4.1 Séparation entre estimation locale et localisation globale	12
1.4.2 Sources de localisation globale	13
1.4.3 Alignement des repères de localisation	14
1.4.4 Gestion de la transition entre sources globales	15
1.5 Synthèse de la revue	16
CHAPITRE 2 MÉTHODOLOGIE	17
2.1 Objectif, périmètre et logique méthodologique	17
2.2 Plateforme robotique et instrumentation	17
2.3 Architecture logicielle et chaîne d'intégration	20
2.4 Environnement expérimental et contexte de démonstration	21
2.5 Infrastructure iGPS et usages retenus	22
2.6 Stratégie de validation et articulation des protocoles	24
CHAPITRE 3 JUMEAU NUMÉRIQUE	27
3.1 Contexte initial, portée et écart à combler	27
3.2 Démarche de conception et critères de fidélité	28
3.3 Construction du modèle numérique du RanGen	29
3.4 Continuité des interfaces entre simulation et robot réel	29
3.5 Développement du contrôleur de la base mobile	32
3.5.1 Frontière de modélisation et point de comparaison	32
3.5.2 Sélection des modes et conversion de la consigne	32
3.5.3 Contraintes appliquées à la commande intermédiaire	36
3.5.4 Construction de la boîte noire simulée	37
3.5.5 Calcul de l'odométrie du contrôleur	41
3.6 Validation du jumeau numérique	43
3.6.1 Processus automatisé d'évaluation et essais retenus	43
3.6.2 Plan de validation	44
3.6.3 Effet de la garde de braquage	44
3.6.4 Effet de la représentation minimale en déplacement latéral pur	47
3.6.5 Effet du plafonnement selon le secteur de braquage	49
3.6.6 Réglage de l'accélération et de la décélération	51
3.6.7 Synthèse de la validation	55

3.7	Discussion	55
3.8	Conclusion	56
CHAPITRE 4 BRAS ROBOTIQUE		59
4.1	Objectif, portée et chaîne instrumentée	59
4.2	Conception de la couronne	59
4.3	Conception des supports de préhension	62
4.4	Intégration logicielle iGPS dans ROS 2	64
4.5	Repères, notations et organisation expérimentale	65
4.6	Calibration et validation du montage de la couronne	67
4.6.1	Formulation du problème	67
4.6.2	Protocole de calibration et résolution numérique	70
4.6.3	Protocole de validation et résultats	71
4.7	Calibration et validation de la chaîne terminale	72
4.7.1	Formulation géométrique	72
4.7.2	Valeurs retenues et validation	73
4.8	Commande cartésienne en vitesse et paramètres retenus	74
4.9	Évaluation expérimentale	77
4.9.1	Exactitude de positionnement	77
4.9.2	Régularité du suivi de chemin	81
4.10	Synthèse	83
CHAPITRE 5 NAVIGATION INTÉGRÉE AVEC IGPS		85
5.1	Objectif et portée	85
5.2	Chaîne autonome retenue pour la base mobile	86
5.3	Architecture de transition SLAM-iGPS	86
5.4	Calibration de la transformation entre les repères	87
5.5	Bascule vers la localisation iGPS	90
5.5.1	Choix de la stratégie de transition	90
5.5.2	Réexpression de la carte en mode switch	92
5.6	Erreur de pose aux cibles Nav2	92
5.7	Démonstration intégrée	95
CONCLUSION ET RECOMMANDATIONS		101
6.1	Bilan des contributions	101
6.2	Limites	103
6.3	Perspectives	103
ANNEXE I	CONTRÔLE DE COHÉRENCE DE L'ODOMÉTRIE SIMULÉE	105
ANNEXE II	ANALYSES DÉTAILLÉES DU BRAS ROBOTIQUE	107
ANNEXE III	COMPARAISON DES MÉTHODES DE LOCALISATION AUTONOME DE LA BASE MOBILE	113

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES	119
--	-----

LISTE DES TABLEAUX

		Page
Tableau 4.1	Fréquence de publication de la couronne iGPS selon l'interface utilisée, la commande Surveyor et les repères iGPS interrogés. Les commandes <i>allégée</i> et <i>complète</i> correspondent respectivement à <code>GetFrameDataLite</code> et <code>GetFrameData</code> . Les symboles de la première colonne désignent les repères ajoutés cumulativement : couronne (c), sonde <i>i5</i> tenue dans la pince (p), référence de base du bras (b), sonde <i>i5</i> opérateur (o) et sonde <i>i6</i> (i6). Les valeurs sont les moyennes des essais répétés	65
Tableau 4.2	Correspondance entre la nomenclature de la chaîne bras-iGPS utilisée dans la suite du mémoire et les repères ROS associés	66
Tableau 4.3	Comparaison des métriques avant/après sur l'ensemble d'optimisation de la couronne et sur l'ensemble de validation indépendant. Les écarts-types σ_t et σ_r sont calculés sur les normes d'erreur de translation et de rotation, échantillon par échantillon	71
Tableau 4.4	Valeurs calibrées de la chaîne terminale, réestimations et erreur de position après cinq remises en place de la sonde <i>i5</i>	74
Tableau 4.5	Paramètres du contrôleur cartésien retenus pour l'ensemble des essais du chapitre. Les gains sont regroupés en un triplet commun pour les trois translations et un second pour les trois rotations. La régularisation et la limite de vitesse articulaire sont maintenues fixes pendant la campagne de réglage	77
Tableau 4.6	Synthèse des erreurs de positionnement de la sonde <i>i5</i> en régime stabilisé. Les trois essais comprennent 27 échantillons retenus chacun. Pour une série d'essais, les valeurs numériques correspondent à la médiane inter-essais des métriques calculées séparément par essai. L'écart type est calculé, pour chaque essai, sur les normes $\ e_k\ _2$. La répétabilité correspond à la dispersion radiale moyenne autour de chaque groupe de cibles	80
Tableau 5.1	Synthèse des erreurs de pose aux cibles Nav2 pour les deux sources globales. Chaque essai comprend 16 poses cibles retenues. La RMSE, la médiane et l'écart type sont calculés séparément pour chaque essai sur ces 16 poses. Les valeurs numériques correspondent à la médiane inter-essais de ces métriques	93

LISTE DES FIGURES

		Page
Figure 2.1	Plateforme robotique nommée RanGen	18
Figure 2.2	Abstraction logicielle simplifiée du système autour de ROS 2	20
Figure 2.3	Plan simplifié de l'environnement expérimental formé des deux laboratoires et du corridor de liaison. Les éléments représentés à l'intérieur des laboratoires correspondent aux équipements et obstacles statiques de l'environnement. Les marqueurs rouges indiquent la position des six émetteurs iGPS, décrits dans la section suivante	21
Figure 2.4	Sondes et capteurs iGPS utilisés dans les essais : (a) sonde <i>i5</i> complète, (b) capteurs issus d'une sonde <i>i5</i> démontée, (c) sonde <i>i6</i>	23
Figure 2.5	Quatre des six émetteurs du système iGPS installés dans le laboratoire de métrologie	24
Figure 3.1	Vue d'ensemble du monde de simulation réalisé pour le jumeau numérique du RanGen	30
Figure 3.2	Chaîne de construction du jumeau numérique du RanGen	30
Figure 3.3	Chaîne de commande simplifiée entre la simulation et le robot réel. La logique reste commune jusqu'au niveau de commande commun, puis diverge selon les contraintes propres à chaque branche	31
Figure 3.4	Chaîne fonctionnelle de préparation et d'exécution de la commande de la base mobile. La commande <i>Twist</i> sert à déterminer le mode actif et à produire la commande intermédiaire associée, avant l'application des contraintes puis l'exécution par la boîte noire	34
Figure 3.5	Variables géométriques utilisées dans les conversions cinématiques de la base mobile	38
Figure 3.6	Vérification <code>cmd_mode_timeline</code> associée à l'essai de garde de braquage. Elle confirme la comparabilité des entrées et montre le parcours carré rejoué	45

Figure 3.7	Comparaison entre l'odométrie du contrôleur en simulation et celle du robot réel lors de l'essai de garde de braquage. Les deux trajectoires (x, y) sont présentées avec une échelle de vitesse commune, sur un parcours carré. Le cas sans garde est présenté en (a), et le cas avec garde en (b) 46
Figure 3.8	Vérification <code>cmd_mode_timeline</code> associée à l'essai d'allers-retours latéraux selon Y uniquement. Elle confirme la comparabilité des entrées et montre les inversions de sens successives 47
Figure 3.9	Comparaison entre l'odométrie du contrôleur en simulation et celle du robot réel lors de l'essai d'allers-retours latéraux purs. Les trajectoires (x, y) illustrent les inversions successives. Le cas sans contrainte est montré en (a), et le cas avec contrainte en (b) 48
Figure 3.10	Vérification <code>cmd_mode_timeline</code> associée à l'essai de transition entre secteurs de braquage. Elle confirme la comparabilité des entrées et montre le balayage commandé qui fait franchir la frontière entre secteurs 49
Figure 3.11	Schéma des secteurs de braquage utilisés pour le plafonnement de vitesse. La frontière de transition est estimée au voisinage de la limite de braquage Ackermann 50
Figure 3.12	Comparaison de l'effet du secteur de braquage sur le plafond de vitesse lors d'une transition contrôlée. Les évolutions temporelles de l'angle de braquage et de la vitesse linéaire sont présentées en simulation et sur le robot réel 51
Figure 3.13	Vérification <code>cmd_mode_timeline</code> associée à la séquence d'échelons sur les axes. Elle confirme la comparabilité des entrées et montre la montée en v_x , la rotation intermédiaire, puis la montée en v_y 52
Figure 3.14	Comparaison des réponses temporelles obtenues avec la séquence d'échelons sur les axes. La composante v_x est montrée en (a), et la composante v_y en (b) 53
Figure 3.15	Comparaison des temps de montée et de descente avec la séquence d'échelons sur les axes. Le temps de montée 10–90 % est présenté en (a), et le temps de descente 90–10 % en (b) 54

Figure 3.16	Comparaison entre la simulation et le robot réel avec la séquence d'échelons sur les axes. Pour chaque cas, les estimations odométrique et théorique de l'accélération et de la décélération moyennes en v_x et en v_y sont comparées	54
Figure 4.1	Configurations compactes étudiées pour la couronne : les deux capteurs d'une même paire sont orientés différemment afin d'élargir la couverture autour du poignet	60
Figure 4.2	Couronne finale adaptée au poignet du bras Kinova	61
Figure 4.3	Écarts de montage retenus pour la calibration du montage de la couronne : (a) décalage axial δx , (b) roulis $\delta\phi$	62
Figure 4.4	Sonde $i5$ et éléments de préhension utilisés dans la pince Robotiq pendant les calibrations et les essais : (a) sonde $i5$ et interface de fixation, (b) doigts de préhension imprimés, (c) support cylindrique imprimé	63
Figure 4.5	Repères principaux de la chaîne instrumentée. La branche supérieure relie la base du bras, le poignet et la couronne iGPS. La branche inférieure relie le poignet, la pince et la sonde $i5$ saisie comme repère terminal. Les traits pleins représentent les transformations géométriques connues ou calibrées. Les traits pointillés indiquent les poses iGPS disponibles dans map	66
Figure 4.6	Paramètres géométriques calibrés dans le chapitre. Les corrections du poignet ($\delta x, \delta\phi$) sont présentées séparément des composantes de translation locales de la chaîne terminale ($t_{tool}^{i5,x}, t_{tool}^{i5,y}, t_{gripper_base}^{tool,z}$)	67
Figure 4.7	Montages expérimentaux utilisés pour l'évaluation : (a) bras fixé sur table, utilisé pour la comparaison du suivi de chemin, (b) bras embarqué sur la base mobile, utilisé pour le protocole d'exactitude de positionnement et pour cette comparaison	78
Figure 4.8	Lecture par cible de l'erreur externe 3D de la sonde $i5$. Dans le panneau (a), chaque point correspond, pour un groupe de cibles, à la médiane inter-essais de l'erreur médiane. La bande indique l'étendue min-max inter-essais. Le panneau (b) situe les cibles dans le plan XY de arm_base et indique leur coordonnée Z	81

Figure 4.9	DSP de Welch de la norme du résidu cartésien pour les montages fixe et mobile. Chaque courbe correspond à la moyenne inter-essais des DSP pour un plan YZ, et les bandes colorées indiquent l'écart type inter-essais. La zone grisée indique la bande f_c-10 Hz intégrée dans l'équation (4.23) 83
Figure 5.1	Repères de la carte SLAM $\mathcal{F}_S$ et de l'iGPS $\mathcal{F}_I$. Les deux ensembles de points correspondent aux mêmes poses collectées, mais exprimées dans chacun des deux repères 88
Figure 5.2	Comparaison des deux utilisations de la transformation entre les repères SLAM et iGPS. Le contour en pointillés indique l'opération qui applique la transformation calibrée entre les repères. 91
Figure 5.3	Erreur XY aux poses cibles Nav2 pour les sources globales SLAM et iGPS. Le panneau (a) compare, pour chaque pose cible, la médiane sur les cinq essais. Les zones ombrées indiquent l'intervalle interquartile. La pose évaluée est reconstruite en combinant la source globale sélectionnée, SLAM ou iGPS, avec la même odométrie locale, selon la méthode décrite à la section 5.3. Le panneau (b) situe ces poses sur la carte. Les flèches indiquent l'orientation de la base et de la couronne 94
Figure 5.4	Interface RViz utilisée pour lancer et superviser la démonstration intégrée depuis le poste opérateur. Le panneau de gauche regroupe la carte, les zones du scénario et la commande de départ du robot. La scène 3D centrale affiche la plateforme mobile, le bras et les repères utilisés pendant la calibration SLAM-iGPS. Le panneau de droite sert au suivi des séquences du bras, notamment pour le dépôt de l'outil et la manipulation des vis 96
Figure 5.5	Récupération de l'outil : (a) départ du poste opérateur et navigation autonome vers l'emplacement de prise de l'outil, (b) arrivée de la base devant cet emplacement 97
Figure 5.6	Retour vers le poste opérateur avec changement de source globale : (a) pose de transition pour la calibration, (b) retour au poste opérateur après bascule vers l'iGPS 98
Figure 5.7	Manipulation locale au poste opérateur : (a) désignation d'une cible de prise avec la sonde <i>i5</i> , (b) saisie d'une vis avec le bras 99

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

AMCL	Adaptive Monte Carlo Localization
API	Application Programming Interface, interface de programmation d'application
BLE	Bluetooth Low Energy
CAO	Conception assistée par ordinateur
CAN	Controller Area Network
DSP	Densité spectrale de puissance
EKF	Extended Kalman Filter, filtre de Kalman étendu
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
ICR	Instantaneous Center of Rotation, centre instantané de rotation
iGPS	indoor Global Positioning System
IMU	Inertial Measurement Unit, centrale inertielle
INS	Inertial Navigation System
JSON	JavaScript Object Notation
LiDAR	Light Detection and Ranging
LIO-SAM	LiDAR Inertial Odometry via Smoothing and Mapping
Nav2	Navigation 2
NIST	National Institute of Standards and Technology
PID	Proportionnel intégral dérivé
REP-105	ROS Enhancement Proposal 105
RMS	Root Mean Square, moyenne quadratique
RMSE	Root Mean Square Error, erreur quadratique moyenne
ROS 2	Robot Operating System 2
RTLS	Real-Time Locating Systems, systèmes de localisation en temps réel
RViz	ROS Visualization

SLAM	Simultaneous Localization and Mapping
STL	Standard Tessellation Language
SVG	Scalable Vector Graphics
SVD	Singular Value Decomposition, décomposition en valeurs singulières
TCP	Transmission Control Protocol
TF	Système de transformations entre repères de ROS
URDF	Unified Robot Description Format
UWB	Ultra-Wideband, ultra-large bande

INTRODUCTION

0.1 Contexte industriel et scénario visé

Dans un contexte manufacturier marqué par un manque de main-d'œuvre et par une demande croissante de produits personnalisables, les entreprises recherchent des systèmes robotisés flexibles, faciles à reconfigurer et capables de s'adapter à des tâches variables (Ghodsian, Benfriha, Olabi, Gopinath & Arnou, 2023). Le robot mobile manipulateur constitue une architecture pertinente pour répondre à ces besoins. En associant une base mobile à un bras robotique, il peut rejoindre différents postes de travail, transporter du matériel et réaliser des tâches locales comme l'assemblage ou l'inspection.

Le projet européen ColRobot illustre ce type d'application industrielle du robot mobile manipulateur. Coordonné par Arts et Métiers, il réunissait notamment les groupes industriels Thales Alenia Space et Renault autour d'applications d'assemblage dans les secteurs aérospatial et automobile (Ghodsian *et al.*, 2023; Arts et Métiers, 2018; CORDIS, 2019). Dans le cas de Thales Alenia Space, le robot avait notamment pour rôle de préparer et d'acheminer le matériel nécessaire à l'assemblage, ainsi que d'assister l'opérateur pendant certaines opérations d'assemblage (Thales Alenia Space, 2019).

Le scénario retenu dans ce mémoire s'inspire du cas d'usage industriel de Thales Alenia Space. Un opérateur travaillant à un poste d'assemblage demande l'assistance d'un robot mobile manipulateur. Le robot quitte le poste, rejoint une autre zone de l'atelier pour y récupérer un outil, puis retourne auprès de l'opérateur afin de réaliser une manipulation locale. La mission demeure supervisée par l'opérateur, qui déclenche ou valide les étapes clés.

Ce scénario est adapté à une mission combinant localisation autonome et localisation externe de précision. Le poste d'assemblage se trouve dans une zone instrumentée par un système de mesure externe, alors que le reste du trajet ne bénéficie pas de cette infrastructure. Hors du

volume instrumenté, le robot s'appuie sur ses capteurs embarqués pour naviguer. Dans le volume instrumenté, il peut tirer profit d'une mesure de pose externe plus précise afin de soutenir la navigation finale et la manipulation locale.

0.2 Problématique

Cette configuration met en évidence une difficulté d'intégration : la transition de localisation dépend de plusieurs composantes qui doivent rester cohérentes pendant la mission. Le déplacement et l'assemblage n'imposent pas les mêmes exigences techniques à la stratégie de localisation. Un déplacement dans un corridor peut tolérer une incertitude plus grande qu'une opération locale où le robot doit aligner un outil, déposer une pièce ou saisir un élément précis. La manipulation ajoute aussi plusieurs sources d'incertitude indépendantes : localisation de la base mobile, modèle géométrique du bras, défauts de montage et localisation de la pièce.

Les capteurs embarqués permettent au robot de conserver une estimation de sa pose pendant ses déplacements dans l'atelier, mais cette estimation ne garantit pas nécessairement une précision compatible avec des opérations locales d'assemblage. À l'inverse, un système de mesure externe peut fournir une mesure de pose plus précise, mais seulement dans le volume instrumenté. Le problème n'est donc pas de choisir une seule source de localisation. Il consiste plutôt à organiser la transition entre ces sources selon leur disponibilité et le niveau de précision requis par la mission.

Cette transition doit être réalisée sans perturber la carte d'occupation du robot, les transformations de repères et les interfaces nécessaires à la navigation et à la manipulation. L'exploitation de la mesure externe nécessite également une chaîne de transformations reliant le repère du système de mesure aux repères de la base mobile, du bras et de l'outil par une calibration géométrique. L'architecture doit ensuite maintenir cette cohérence lorsque la source de localisation globale change au cours de la mission.

La question centrale du mémoire peut donc être formulée ainsi : comment concevoir et valider une architecture de transition capable de conserver une localisation autonome hors de la zone instrumentée, d’assurer le passage vers une localisation de haute précision lorsque la mesure externe devient disponible, puis de maintenir des repères cohérents pour la navigation finale et la manipulation locale ?

0.3 Objectif et contributions

L’objectif général de ce mémoire est de développer et d’évaluer, sur une plateforme robotique expérimentale, une architecture de transition de localisation qui relie la navigation autonome, une localisation de haute précision fondée sur une mesure externe et la manipulation locale dans une mission supervisée. Le travail vise une transition fonctionnelle et cohérente dans les conditions expérimentales déterminées pour le projet.

Pour atteindre cet objectif, le mémoire apporte quatre contributions. La première est le développement d’un jumeau numérique comme environnement d’intégration et de validation. Celui-ci met à disposition des interfaces logicielles cohérentes avec celles du robot réel afin de faciliter le transfert des fonctions développées entre la simulation et la plateforme expérimentale.

La deuxième contribution est la définition et la calibration d’une chaîne de transformations géométriques permettant d’exploiter les poses fournies par le système de mesure externe dans les repères du robot. Cette chaîne relie notamment le repère associé à la mesure externe aux repères de la base mobile, du bras et de l’outil. Elle permet d’exprimer la mesure externe dans les repères nécessaires à la localisation, à la navigation et à la manipulation.

La troisième contribution constitue l’axe principal du mémoire. Elle porte sur la conception d’une architecture assurant la transition entre la localisation autonome et la localisation fondée sur la mesure externe. Cette architecture permet de sélectionner la source de localisation globale active tout en présentant une interface stable aux fonctions de navigation. Elle vise également

à préserver la cohérence entre la carte et les transformations utilisées par la navigation et la manipulation.

La quatrième contribution consiste en une validation fonctionnelle intégrée dans le scénario manufacturier défini précédemment. Elle prend la forme d'une démonstration qui couvre successivement la navigation hors de la zone instrumentée, la transition vers la localisation de haute précision, le positionnement de la base mobile au poste de travail et l'exécution d'une tâche d'assemblage par le bras.

0.4 Portée et limites

Le travail porte sur l'intégration et l'évaluation d'une architecture de transition de localisation dans un environnement intérieur contrôlé. L'environnement expérimental comprend une zone couverte par un système de mesure externe et une zone située hors de sa couverture. Cette configuration permet d'étudier, dans un cadre maîtrisé, la transition entre deux sources de localisation.

Dans l'architecture étudiée, la mesure externe constitue une source alternative de localisation globale disponible uniquement dans la zone instrumentée. Elle n'est pas considérée comme une solution couvrant l'ensemble des déplacements du robot. Le travail porte principalement sur l'intégration fonctionnelle des sources, la calibration des repères et la cohérence des transformations lors de la transition.

Les performances du système de mesure externe peuvent notamment être influencées par les occlusions, la géométrie de l'environnement et la stabilité mécanique de l'instrumentation. Ces facteurs sont pris en compte dans l'interprétation des résultats, tandis que leur caractérisation détaillée dépasse le périmètre du mémoire.

La démonstration finale constitue ainsi une validation fonctionnelle de l'architecture de transition dans une mission supervisée. Elle ne vise pas à démontrer une autonomie complète en conditions industrielles.

0.5 Structure du mémoire

Le premier chapitre présente la revue de la littérature. Il situe le travail par rapport aux infrastructures de mesure externe, au jumeau numérique en robotique, à la calibration géométrique, à l'utilisation d'une mesure externe pour la manipulation, puis aux architectures permettant de gérer plusieurs sources de localisation en navigation mobile.

Le deuxième chapitre définit la méthodologie commune aux travaux. Il présente la plateforme robotique, son architecture logicielle, l'environnement expérimental, l'infrastructure de mesure externe ainsi que la stratégie générale de validation.

Le troisième chapitre traite du jumeau numérique de la plateforme robotique. Il décrit la construction du modèle, la continuité des interfaces avec le robot réel et le développement d'un contrôleur de base mobile visant à rapprocher le comportement simulé de celui observé expérimentalement.

Le quatrième chapitre établit la chaîne métrologique nécessaire à la manipulation locale. Il présente la conception du dispositif de mesure externe monté sur le bras, son intégration à l'architecture logicielle du système, les calibrations géométriques et l'évaluation d'une manipulation corrigée par mesure de pose externe.

Le cinquième chapitre étend cette chaîne métrologique à la localisation de la base mobile. Il décrit la localisation autonome retenue, la relation entre le repère du système de mesure externe et le repère global de navigation, l'architecture de transition entre les sources de localisation ainsi que la validation intégrée du scénario supervisé.

La conclusion résume les contributions du mémoire, précise les limites de l'approche et présente les principales pistes de poursuite des travaux.

CHAPITRE 1

REVUE DE LA LITTÉRATURE

Cette revue situe les travaux nécessaires à l’objectif du mémoire : concevoir et valider une architecture de transition capable de conserver une localisation autonome hors de la zone instrumentée, d’assurer le passage vers une localisation de haute précision lorsque la mesure externe devient disponible, puis de maintenir des repères cohérents pour la navigation finale et la manipulation locale. Elle présente d’abord les infrastructures de métrologie externe, puis le rôle du jumeau numérique comme support d’intégration entre la simulation et le robot réel. Elle traite ensuite de la calibration et de la commande du manipulateur à partir d’une mesure externe, puis des contraintes de pose, de carte et de transformations entre repères associées à la transition entre sources de localisation.

1.1 Infrastructures de métrologie pour le suivi de position et de pose

Les systèmes de métrologie externe fournissent une mesure de position ou de pose dans un repère défini par une infrastructure extérieure au robot. Selon l’architecture, cette observation peut servir à caractériser l’erreur après acquisition des données, à comparer deux sources de pose ou à alimenter une boucle de correction pendant l’exécution. Des travaux en suivi optique montrent que cette mesure externe peut être exploitée pour améliorer l’exactitude cartésienne lorsque l’erreur provient notamment d’un biais géométrique, d’un défaut de montage ou d’une mauvaise localisation du point d’outil (Liu, Li, Zhuang & Song, 2020).

Parmi ces infrastructures, un système iGPS (*indoor Global Positioning System*) est une infrastructure de métrologie distribuée conçue pour le suivi précis de positions et de poses dans un volume instrumenté. Il repose sur des émetteurs laser installés dans le volume de mesure et sur des capteurs placés sur les objets suivis. Les émetteurs définissent le volume mesurable et balayent l’espace avec des faisceaux optiques. À partir des signaux détectés par les capteurs, le système estime les angles entre les émetteurs et les capteurs, puis reconstruit la position de chaque capteur dans un repère global commun par triangulation. Une pose complète est obtenue

lorsque plusieurs capteurs, dont la géométrie relative est connue, sont combinés dans un même repère. La mesure dépend donc directement de la ligne de visée, du nombre d'émetteurs observés, de leur disposition et de l'orientation des capteurs (Norman, Schönberg, Gorlach & Schmitt, 2012).

Cette architecture est adaptée aux grands volumes de mesure, mais sa performance dépend de la configuration expérimentale. Des résultats expérimentaux montrent qu'un système iGPS peut atteindre une mesure tridimensionnelle à l'échelle millimétrique, voire submillimétrique lorsque la configuration est optimisée (Muelaner, Wang, Martin, Jamshidi & Maropoulos, 2012). Dans un contexte de robots coopératifs, des récepteurs iGPS fixés aux effecteurs ont également été validés comme source externe de pose à six degrés de liberté pour la mesure et la correction de mouvements (Norman *et al.*, 2012). Ces résultats appuient son utilisation comme source cartésienne précise, tout en rappelant que la performance reste liée à la configuration de l'infrastructure et au montage des capteurs.

L'iGPS a également été étudié dans le contexte de cellules d'assemblage reconfigurables, où les ressources de production et le produit peuvent être déplacés et reconfigurés en fonction de l'opération d'assemblage. Dans un scénario simulé, un récepteur iGPS est placé au point d'outil d'un robot et la position de cinq émetteurs est optimisée afin de réduire l'incertitude de mesure sous contraintes de portée et de visibilité (Nicksch, Hüttner & Schmitt, 2022a). Ce travail met en évidence l'intérêt de l'iGPS comme infrastructure de métrologie pour des systèmes d'assemblage utilisant des équipements mobiles, mais il reste centré sur l'optimisation du positionnement des émetteurs et sur l'incertitude à l'intérieur du volume instrumenté. Un travail complémentaire propose un modèle d'iGPS virtuel pour estimer a priori l'incertitude de mesure dans des environnements reconfigurables. L'application à un cas d'assemblage aile-fuselage montre que la configuration des émetteurs influence fortement la distribution spatiale de l'incertitude et la capacité du système à respecter les tolérances d'assemblage (Nicksch, Sabzehi & Schmitt, 2022b).

Ces travaux situent l'iGPS parmi les infrastructures externes adaptées au suivi précis de systèmes robotiques en assemblage. Ils montrent qu'une mesure externe peut être utilisée comme référence de position ou de pose lorsque le robot évolue dans un volume instrumenté. En revanche, ils traitent surtout la qualité de la mesure dans ce volume, notamment la visibilité, la configuration des émetteurs, l'incertitude et le montage des capteurs. Ils ne traitent pas directement de la continuité de localisation lors des transitions d'entrée et de sortie de ce volume.

1.2 Jumeau numérique en robotique

La transition entre navigation autonome, mesure externe et manipulation locale nécessite d'intégrer plusieurs modules logiciels avant les essais sur le robot réel. Une représentation simulée du système permet de vérifier ces interfaces et de comparer les comportements attendus dans un environnement contrôlé.

La littérature sur les jumeaux numériques ne propose pas une définition unique. Cette diversité entretient une ambiguïté entre les domaines et complique la comparaison des approches (National Institute of Standards and Technology, 2026). Une distinction utile peut être faite selon le degré d'intégration des données entre modèle numérique, ombre numérique et jumeau numérique complet (Kritzinger, Karner, Traar, Henjes & Sihn, 2018). Dans cette classification, le modèle numérique représente le système sans synchronisation automatique avec l'objet physique. L'ombre numérique ajoute un transfert automatisé unidirectionnel de données de l'objet physique vers l'objet numérique, tandis que le jumeau numérique complet repose sur des transferts automatisés dans les deux directions.

Dans ce mémoire, le terme *jumeau numérique* ne désigne donc pas un jumeau numérique complet au sens strict de cette classification. Il est employé dans un sens opérationnel restreint pour désigner une représentation simulée de la plateforme robotique expérimentale étudiée. Cette représentation sert au développement, à l'intégration et à la comparaison des fonctions de commande et de navigation du projet dans un environnement contrôlé. La fidélité recherchée n'est donc pas exhaustive : elle est principalement fonctionnelle. Elle vise à conserver les conventions

logicielles et les grandeurs observables nécessaires pour comparer les comportements obtenus en simulation et sur le robot réel.

Dans cette perspective, la fidélité fonctionnelle du jumeau numérique repose d’abord sur la continuité des interfaces entre la simulation et le système réel. Les modules de navigation, de commande et de supervision doivent pouvoir échanger les mêmes types de données, selon les mêmes conventions, que ces données proviennent du simulateur ou du robot réel. À l’échelle du système, ROS 2 (*Robot Operating System 2*) fournit ce cadre de communication distribué, dans lequel les nœuds échangent des données au moyen d’interfaces typées telles que les *topics*, les services et les actions (Macenski, Foote, Gerkey, Lalancette & Woodall, 2022). Pour la chaîne d’actionnement, *ros2_control* prolonge cette abstraction en séparant les contrôleurs des composants matériels au moyen d’un gestionnaire de contrôleurs et d’interfaces d’état et de commande (*ros2_control Development Team*, 2026b). Cette séparation permet de conserver une architecture de commande commune entre la simulation et le robot réel, en limitant les adaptations à l’interface avec le matériel réel ou simulé.

Le choix du simulateur dépend alors du niveau de fidélité recherché et de sa capacité à préserver cette continuité d’interfaces entre la simulation et le système réel. Gazebo est largement utilisé dans l’écosystème ROS 2, notamment avec le paquet *gz_ros2_control*, qui permet d’instancier l’architecture *ros2_control* autour d’un modèle simulé (Gazebo Project, 2026a,b; *ros2_control Development Team*, 2026a). Isaac Sim est particulièrement orienté vers la richesse des scènes simulées, la simulation de capteurs et les scénarios de perception complexes fondés sur des données synthétiques (NVIDIA Corporation, 2026a,b,c). Des intégrations avec *ros2_control* existent également pour ce simulateur (*ros2_control Development Team*, 2026c; PickNik Robotics, 2025).

Pour le périmètre retenu, le simulateur est donc avant tout un outil d’intégration. Sa fonction principale n’est pas de reproduire tous les effets physiques du robot réel, mais de conserver les interfaces de commande, d’état et de navigation nécessaires pour préparer le transfert vers la plateforme expérimentale.

1.3 Calibration et commande du manipulateur à partir d'une mesure externe

Lorsqu'un manipulateur mobile exploite une mesure externe portée par l'effecteur, cette mesure doit être rattachée à la chaîne de repères du robot et au repère de travail. Ce rattachement fait intervenir le repère du capteur, le repère de l'outil, le repère de la base mobile et le repère de l'infrastructure de mesure externe. Il combine des transformations calibrées et des transformations variables liées à l'état du système, afin d'exprimer l'observation externe dans les repères requis pour la manipulation ou la localisation de la plateforme.

Les méthodes de calibration *hand-eye* constituent un cadre classique pour ce type de problème. Un capteur fixé à l'effecteur est observé pendant plusieurs mouvements afin d'estimer la transformation fixe qui le relie à la chaîne cinématique du robot (Tsai & Lenz, 1989). Des formulations plus générales permettent aussi d'estimer les liens entre le monde, la base du robot et un capteur externe lorsque plusieurs transformations fixes sont inconnues (Shah, 2013).

Cette chaîne de repères s'étend lorsque le manipulateur est porté par une base mobile. Dans un système à base fixe, le positionnement du point d'outil peut être analysé principalement à partir de la géométrie du manipulateur, de sa calibration et du repère de la pièce. L'ajout d'une base mobile fait intervenir la localisation de la plateforme dans l'environnement. Les travaux du NIST (*National Institute of Standards and Technology*) sur la caractérisation de robots mobiles manipulateurs destinés à l'assemblage illustrent ce lien. Ils utilisent notamment un système de suivi optique pour mesurer la pose de la base et l'incertitude de positionnement du point d'outil (Bostelman, Li-Baboud, Yoon, Shah & Aboul-Enein, 2020). Une référence métrologique externe sert ainsi à évaluer la performance du système lorsque la précision du point d'outil dépend à la fois de la base mobile et du manipulateur.

Une fois la chaîne de repères établie, la mesure externe peut aussi être utilisée dans une boucle de commande. Elle peut alors fermer la boucle sur la pose mesurée du repère outil et alimenter une commande cinématique en vitesse, où l'erreur cartésienne est convertie en vitesses articulaires par la jacobienne du robot. Dans ce cadre, une pseudo-inverse amortie de la jacobienne limite l'amplification des vitesses articulaires près des singularités ou des

configurations mal conditionnées, avec un compromis sur le suivi instantané (Whitney, 1969; Nakamura & Hanafusa, 1986).

1.4 Architecture de localisation et transition entre sources globales

Une transition entre sources de localisation ne consiste pas simplement à remplacer une mesure par une autre. Elle modifie la source responsable du rattachement global du robot, tandis que l'estimation de son mouvement local doit demeurer continue. Elle doit également préserver la cohérence entre la carte et les transformations de repères utilisées par la pile de navigation. La présente section organise l'analyse autour de l'estimation locale, des sources de localisation globale, de l'alignement de leurs repères et de la gestion du basculement entre ces sources.

1.4.1 Séparation entre estimation locale et localisation globale

Nav2 (*Navigation 2*) est une pile de navigation ROS 2 qui regroupe notamment la planification globale, le suivi local, les cartes de coût et les comportements de récupération (Macenski, Martín, White & Ginés Clavero, 2020; Open Navigation LLC, 2026c). Son fonctionnement repose sur des poses exprimées dans un ensemble cohérent de repères. Dans ROS 2, cette cohérence est assurée par TF, qui représente et maintient les transformations entre les repères nommés du système (Foote, 2013).

Conformément aux conventions de REP-105 (*ROS Enhancement Proposal 105*) (Meeussen, 2010) et à la configuration TF attendue par Nav2 (Open Navigation LLC, 2026b), l'arbre TF d'un robot mobile est structuré autour des transformations principales suivantes :

$$\text{map} \rightarrow \text{odom} \rightarrow \text{base_link}.$$

Le repère `base_link` est attaché à la base du robot. Le repère `odom` fournit une référence locale continue, mais susceptible de dériver avec le temps. Le repère `map` constitue une référence

globale liée à la carte ou à l’environnement de travail. La transformation entre `map` et `odom` relie ainsi la continuité de l’estimation locale au rattachement global du robot.

L’estimation locale peut être obtenue par fusion de mesures proprioceptives, comme l’odométrie des roues et une centrale inertielle (IMU, *Inertial Measurement Unit*). Les filtres de Kalman étendus sont couramment utilisés pour produire une estimation continue en temps réel à partir de ces mesures (Moore & Stouch, 2014). Dans ROS 2, cette fonction est notamment fournie par le paquet `robot_localization` (Charles River Analytics, 2026).

Cette organisation permet de traiter séparément l’estimation locale et son rattachement global. L’estimation locale continue de fournir la transformation `odom` vers `base_link`. Le changement de source globale se fait donc sur la transformation `map` vers `odom`, qui relie cette estimation locale au repère global utilisé par Nav2.

1.4.2 Sources de localisation globale

La localisation globale autonome sur une carte peut être réalisée par plusieurs approches. AMCL (*Adaptive Monte Carlo Localization*) constitue une méthode classique fondée sur la localisation de Monte-Carlo, dans laquelle la distribution de la pose du robot est représentée par un ensemble de particules pondérées (Dellaert, Fox, Burgard & Thrun, 1999). Dans ROS 2, cette méthode est disponible dans Nav2 avec le paquet `nav2_amcl` (Open Navigation LLC, 2026a).

SLAM Toolbox (*Simultaneous Localization and Mapping*) offre une autre solution de localisation fondée sur les capteurs embarqués. Il combine notamment le recalage de scans et l’optimisation d’un graphe de poses. Selon sa configuration, il peut être utilisé pour construire ou mettre à jour une carte, ou dans un mode consacré à la localisation sur une carte existante (Macenski & Jambrecic, 2021; Macenski, 2026). AMCL et SLAM Toolbox constituent ainsi des sources globales autonomes : la pose est estimée à partir des capteurs du robot, sans recours à une mesure externe de pose.

L'iGPS fournit une source de localisation globale d'une nature différente. Sa mesure provient d'une infrastructure de métrologie fixe et est exprimée dans le repère propre à cette infrastructure. Elle n'est disponible que dans le volume instrumenté et lorsque les conditions nécessaires à l'observation des capteurs sont réunies. Avant d'être utilisée par la navigation, cette mesure doit être exprimée dans le repère de la carte et convertie en pose de la base du robot.

Une source autonome sur carte et une source métrologique externe ne fournissent donc pas directement une pose exploitable de la même manière. Leur intégration impose d'abord d'exprimer les poses dans un même cadre de référence. Elle impose ensuite de choisir, à chaque instant, la source qui fournit le rattachement global actif.

1.4.3 Alignement des repères de localisation

Une transition entre deux sources globales suppose l'existence d'un lien géométrique entre leurs repères. Une localisation autonome sur carte exprime la pose du robot dans le repère `map`, tandis qu'une infrastructure de métrologie exprime ses observations dans son propre repère. Ces repères peuvent différer par leur origine et leur orientation.

Leur relation peut être estimée à partir de positions ou de poses observées simultanément dans les deux repères. Les méthodes d'alignement rigide par moindres carrés permettent d'estimer la transformation qui minimise les écarts entre deux ensembles de points correspondants (Umeyama, 1991). Pour une base mobile dont le mouvement est supposé planaire, cette transformation peut être limitée au groupe $SE(2)$, avec deux composantes de translation dans le plan (x, y) et une rotation de lacet autour de l'axe z du repère de la carte.

La qualité de l'estimation dépend de la configuration des poses utilisées pour l'alignement et de la cohérence des correspondances entre les deux repères. Les résidus après alignement permettent ensuite d'évaluer la cohérence entre les deux sources de localisation.

Une fois cette transformation connue, les mesures de l'infrastructure externe peuvent être exprimées dans le repère de la carte et utilisées pour déterminer le rattachement global du robot.

L'alignement des repères constitue ainsi une condition préalable à une transition cohérente entre une localisation autonome et une localisation métrologique.

1.4.4 Gestion de la transition entre sources globales

Une fois les repères alignés, il reste à gérer le basculement entre sources. La transition doit préciser quelle source fournit le rattachement global actif et à quelles conditions ce rattachement peut changer. Elle doit notamment tenir compte de la disponibilité de la source, de la qualité de la pose fournie et de la continuité de la pose transmise à la navigation. Une discontinuité importante peut perturber le contrôleur local, déplacer la représentation des obstacles dans les cartes de coût ou modifier la position apparente des objectifs de navigation.

Les travaux sur le *handover* de localisation abordent des problèmes comparables, notamment lors du passage entre des environnements intérieurs et extérieurs. Ils mettent en évidence l'importance de la continuité de pose, de la sélection de la source utilisée et de l'évaluation de la qualité des mesures (Cao, Usai, Gu & Rizzo, 2025).

Le passage entre sources peut être traité par fusion ou par commutation. Dans une stratégie de fusion, plusieurs mesures compatibles sont combinées après avoir été exprimées dans un repère commun. Dans une stratégie de commutation, une seule source est retenue pour définir le rattachement global actif. Le basculement peut alors produire un saut apparent de la pose si la source sélectionnée n'est pas cohérente avec la précédente. Dans les deux cas, l'architecture doit préciser le rôle attribué à chaque source, les conditions d'acceptation des mesures et le comportement adopté lorsqu'une source devient indisponible.

Une première approche traite le passage entre une localisation extérieure et une localisation intérieure. Le basculement vers le mode intérieur est déclenché à l'entrée d'un garage, avant que la source extérieure ne soit complètement perdue. Après ce changement, l'estimation intérieure est exprimée dans un repère relatif initialisé au point de transition. Ce repère est ensuite rattaché à la trajectoire extérieure afin de maintenir une trajectoire continue entre les deux environnements (Li *et al.*, 2020).

Une autre architecture utilise une commutation explicite entre deux sources globales de localisation. Contrairement à l'approche précédente, la localisation intérieure n'est pas traitée comme une trajectoire relative initialisée à partir de la dernière pose extérieure. Le rattachement global est alors basculé vers l'une des deux sources lorsque le robot atteint un point de passage situé dans une zone où les deux sources peuvent être disponibles (Ostuni, 2022).

Ces travaux montrent que la transition de localisation est généralement traitée comme un problème de continuité, de qualité de mesure et de sélection de source.

1.5 Synthèse de la revue

La littérature met en évidence quatre besoins liés à l'intégration d'une métrologie externe sur un robot mobile manipulateur : disposer d'une mesure fiable dans le volume instrumenté, représenter le système dans un environnement simulé, relier la mesure externe aux repères du robot, puis maintenir une chaîne de transformations cohérente lors du changement de source globale de localisation.

Les travaux recensés traitent surtout ces besoins séparément. Le problème abordé dans ce mémoire est de les réunir dans une même architecture, capable d'exploiter la mesure externe dans le volume instrumenté tout en conservant une localisation autonome en dehors de ce volume. Cette architecture doit aussi aligner les repères des deux sources et fournir à la navigation comme à la manipulation une chaîne de transformations cohérente. Cette intégration guide les choix techniques présentés dans les chapitres suivants.

CHAPITRE 2

MÉTHODOLOGIE

2.1 Objectif, périmètre et logique méthodologique

Ce mémoire s'appuie sur une démarche en trois volets : la conception du système, son intégration et sa validation. Ce chapitre définit la base commune du mémoire. Il fixe les choix techniques et les critères de validation utilisés dans les chapitres suivants.

Le projet porte sur un robot mobile manipulateur dont la localisation repose sur deux modes complémentaires : une estimation autonome fondée sur les capteurs embarqués et une localisation assistée par iGPS dans la zone instrumentée. L'enjeu méthodologique est de préparer la transition entre ces deux modes tout en conservant des repères et des interfaces cohérents pour la navigation de la base et la manipulation. Dans cette logique, la mesure iGPS sert à la fois de source de localisation globale dans la zone instrumentée et de mesure cartésienne externe pour le suivi de l'effecteur. Le chapitre se limite aux éléments transversaux nécessaires à la compréhension du système et de la transition visée : la plateforme robotique, l'architecture d'intégration, le contexte expérimental, les usages de l'iGPS et la logique générale de validation.

Les chapitres 3 à 5 traitent respectivement du jumeau numérique, de l'intégration de l'iGPS au niveau du bras et de la transition entre la localisation autonome de la base et la localisation assistée par iGPS dans une démonstration intégrée.

2.2 Plateforme robotique et instrumentation

Le système étudié repose sur un robot mobile manipulateur composé d'une base mobile AgileX Ranger Mini 2.0, d'un bras robotique Kinova Gen3 et d'une pince adaptative Robotiq 2F-85 (Weston Robot, 2026; Kinova Robotics, 2024; Robotiq, 2019). Cet ensemble constitue la plateforme robotique *RanGen*, présentée à la figure 2.1.

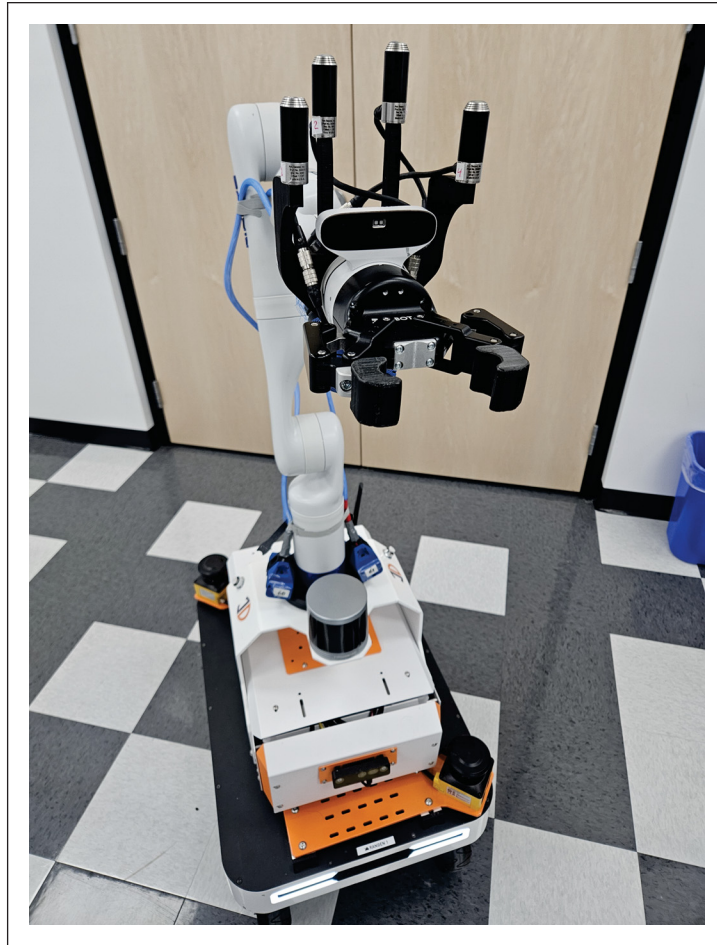


Figure 2.1 Plateforme robotique nommée RanGen

La base mobile constitue le support principal de la plateforme et assure sa mobilité dans l'environnement. Son architecture à quatre roues directrices lui donne un comportement holonome, c'est-à-dire la capacité de commander la vitesse longitudinale, la vitesse latérale et la vitesse angulaire selon la tâche. Elle exploite ainsi plusieurs modes de déplacement, notamment le pivotement sur place, la translation latérale et le déplacement de type double Ackermann. Cette polyvalence impose une contrainte pour le mémoire : le contrôleur développé au chapitre 3 doit reproduire un comportement cohérent en simulation et sur le robot réel.

Le bras Kinova Gen3 assure les tâches de manipulation du système. Il porte aussi une instrumentation de localisation de haute précision. Cette instrumentation repose sur un dispositif fixé au

poignet, qui fournit une mesure de pose par iGPS lorsque le signal est disponible. La conception mécanique de ce dispositif, son intégration physique sur le bras et sa calibration sont réservées au chapitre 4.

La plateforme comporte aussi plusieurs capteurs embarqués. Elle comprend des encodeurs de roues pour l'odométrie et une centrale inertielle pour l'estimation du mouvement. Elle intègre également des capteurs LiDAR 2D pour la perception plan et la sécurité, un capteur LiDAR 3D pour la perception volumique, ainsi que des caméras utilisées pour la détection de repères visuels et le retour vidéo destiné à la supervision par l'opérateur. L'iGPS s'ajoute à ces capteurs comme source de localisation de haute précision lorsqu'une couverture suffisante est disponible.

La plateforme utilise deux ordinateurs embarqués : un Jetson Orin et un Intel NUC. La base mobile et les capteurs sont reliés au Jetson pour exploiter sa capacité de calcul, tandis que le bras est géré par le NUC pour simplifier l'intégration et le développement.

Cette répartition répond à des contraintes logicielles et de déploiement. Le Jetson fonctionne sous Ubuntu 22.04 avec ROS 2 Humble, alors que le projet cible ROS 2 Jazzy sous Ubuntu 24.04. Les services liés à la base et aux capteurs sont donc isolés dans des conteneurs orchestrés par *Docker Compose*, avec un service par sous-système. Cette isolation permet d'exécuter ces modules dans un environnement ROS 2 Jazzy, indépendamment de l'environnement natif du Jetson. Les conteneurs sont lancés avec le mode réseau de l'hôte. Les données ROS 2 publiées par chaque service sont ainsi disponibles sur ce même réseau et peuvent être utilisées par les autres modules du système. Cette organisation facilite l'exploitation, mais alourdit le cycle de développement lors des reconstructions d'images. À l'inverse, le NUC fonctionne sous Ubuntu 24.04 avec ROS 2 Jazzy et sert surtout au développement des fonctions propres au projet. Les fonctions de SLAM et de navigation avec Nav2 restent toutefois sur le Jetson, car ces composantes exigeantes en calcul tirent mieux parti du matériel NVIDIA.

2.3 Architecture logicielle et chaîne d'intégration

ROS 2 structure l'intégration logicielle du système. Il fournit une infrastructure de communication distribuée pour les données des capteurs, les commandes des actionneurs, les estimations d'état et les informations nécessaires à la supervision du robot. Cette infrastructure maintient des interfaces communes entre les fonctions étudiées dans les chapitres suivants. ROS 2 a été retenu pour sa modularité, sa large adoption en robotique et la richesse de son écosystème logiciel. Cette architecture découple les fonctions logicielles du matériel sous-jacent. Dans ROS 2, cette séparation repose principalement sur trois types d'interfaces : les *topics*, les *services* et les *actions*. Ces interfaces s'appuient sur des types de messages qui permettent aux fonctions du projet de rester indépendantes de la source exacte des données.

Comme l'illustre la figure 2.2, les mêmes interfaces peuvent être exploitées en simulation comme sur le robot réel. Cette continuité *sim-to-real* limite les ajustements nécessaires au transfert entre les deux environnements.

Gazebo est retenu comme simulateur principal pour la mise en œuvre du jumeau numérique. Ce choix tient à son intégration directe avec ROS 2 et avec l'architecture `ros2_control`. La chaîne de commande reste ainsi cohérente entre la simulation et le robot réel. Le chapitre 3 détaille la mise en place de cet environnement de simulation.

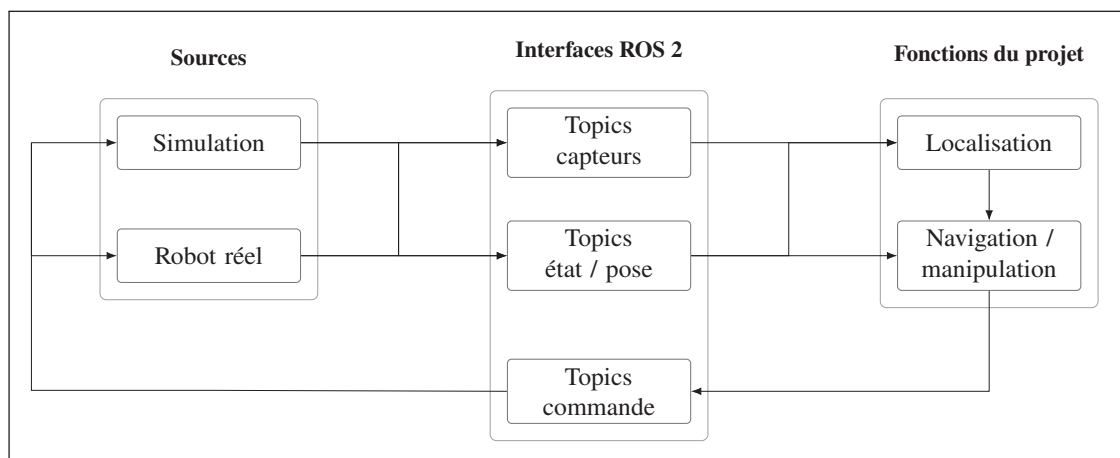


Figure 2.2 Abstraction logicielle simplifiée du système autour de ROS 2

Cette structure sert de base au développement, aux essais et aux comparaisons des composantes présentées dans les chapitres suivants, en simulation comme sur le robot réel.

2.4 Environnement expérimental et contexte de démonstration

La démonstration finale et les essais associés sont réalisés dans un environnement formé par le laboratoire de métrologie, le laboratoire InitRobots et le corridor qui les relie, comme l'illustre la figure 2.3.

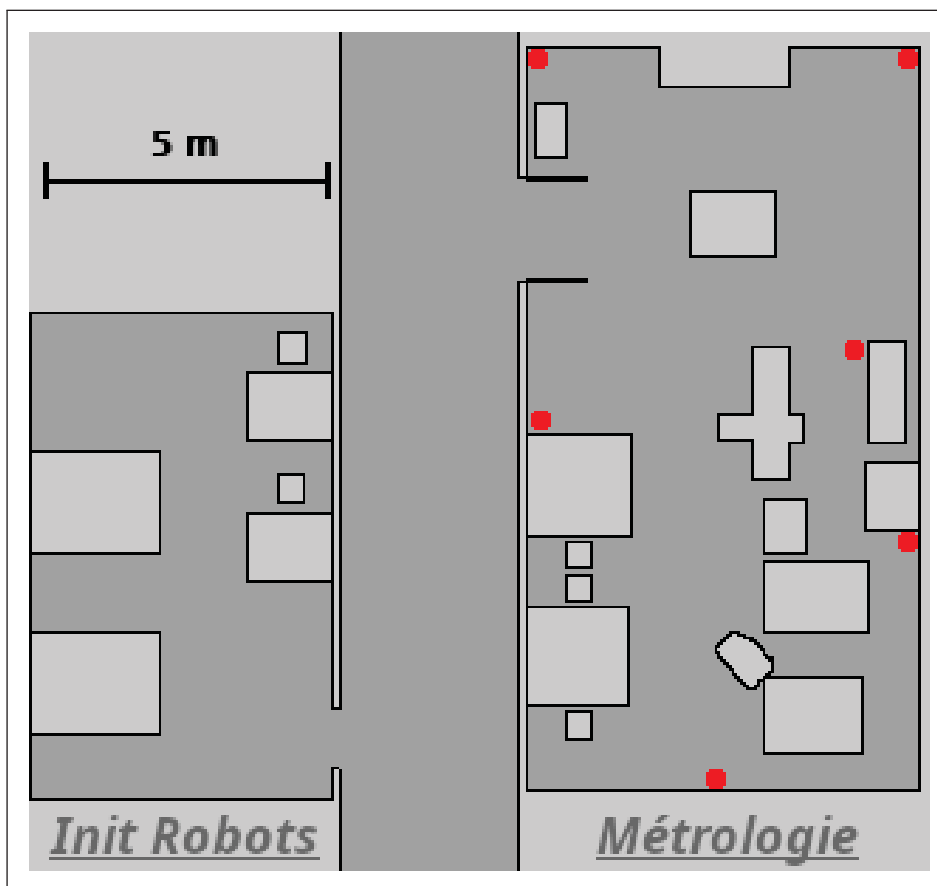


Figure 2.3 Plan simplifié de l'environnement expérimental formé des deux laboratoires et du corridor de liaison. Les éléments représentés à l'intérieur des laboratoires correspondent aux équipements et obstacles statiques de l'environnement. Les marqueurs rouges indiquent la position des six émetteurs iGPS, décrits dans la section suivante

Le choix d'un environnement de laboratoire répond à des contraintes méthodologiques. Il donne accès à l'infrastructure iGPS et permet des essais maîtrisés et répétables. La principale limite est la couverture partielle de l'iGPS, dont la zone de mesure exploitable se restreint au laboratoire de métrologie. L'usage de l'iGPS se limite donc aux phases du scénario réalisées dans cette zone. Cette contrainte structure les protocoles expérimentaux : elle impose la coexistence de plusieurs sources de localisation, motive les mécanismes de transition entre repères et justifie la place accordée à la calibration. L'environnement expérimental permet ainsi d'évaluer l'intégration du système et les transitions entre modes de localisation.

2.5 Infrastructure iGPS et usages retenus

Le principe de fonctionnement des systèmes iGPS et les facteurs qui influencent leur mesure sont présentés à la section 1.1. La présente section décrit l'infrastructure particulière utilisée dans ce projet et les usages expérimentaux retenus.

Le système iGPS utilisé dans ce projet est fourni par 7D Kinematic Metrology (7D Kinematic Metrology, 2026d). Il fournit une mesure de pose externe de haute précision, utilisée à la fois comme source de localisation et comme référence pour l'évaluation expérimentale. Cette précision correspond ici aux ordres de grandeur annoncés par le constructeur : une incertitude de point 3D inférieure à $150\ \mu\text{m}$ et une incertitude dynamique inférieure à $0,3\ \text{mm}$ pour le suivi en mouvement (7D Kinematic Metrology, 2026d,c). Ces valeurs ne constituent pas des performances vérifiées expérimentalement dans ce mémoire. Son rôle n'est pas de remplacer les capteurs embarqués du robot, mais de les compléter dans la zone couverte.

Les sondes et capteurs iGPS utilisés dans le mémoire sont présentés à la figure 2.4. La sonde *i5* sert de cible mobile mesurée par l'iGPS (7D Kinematic Metrology, 2026a). Dans les usages retenus, seule la composante de position de sa mesure à cinq degrés de liberté est exploitée. Elle sert aux campagnes de calibration et de validation du bras, aux essais d'exactitude et de régularité, ainsi qu'à la désignation de cibles dans la démonstration intégrée.

Des capteurs issus d'une sonde *i5* démontée sont aussi employés. Cette configuration permet de placer les capteurs dans un montage mécanique propre au système, au lieu d'utiliser la sonde dans son boîtier d'origine. Ces capteurs sont utilisés pour l'instrumentation iGPS du bras.

La sonde *i6* fournit une pose complète à six degrés de liberté (7D Kinematic Metrology, 2026b). Elle est utilisée pour la mise en place et la calibration de l'installation iGPS. Elle sert aussi de référence ponctuelle dans certains essais, par exemple pour reconstruire un repère virtuel du bras ou pour mesurer la position de la base mobile pendant des essais de navigation.

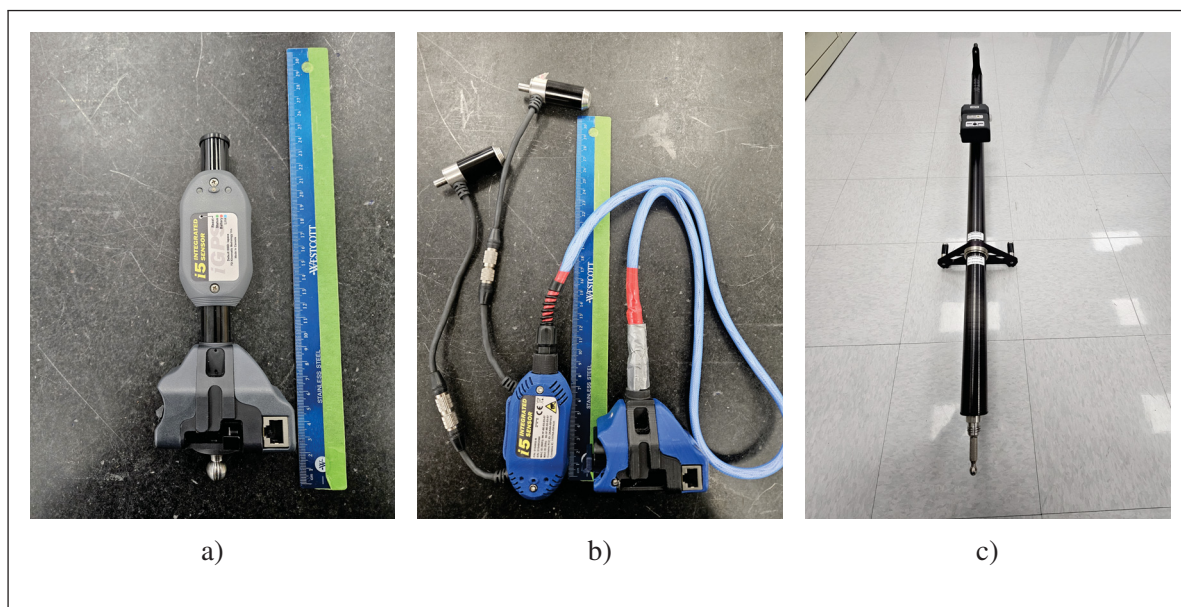


Figure 2.4 Sondes et capteurs iGPS utilisés dans les essais : (a) sonde *i5* complète, (b) capteurs issus d'une sonde *i5* démontée, (c) sonde *i6*

La figure 2.5 situe l'installation iGPS dans le laboratoire de métrologie. Dans la configuration retenue, six émetteurs installés dans la salle définissent le volume de mesure et produisent le signal détecté par les sondes. La couverture n'est toutefois pas complète, car elle demeure limitée par le champ de vision des émetteurs et par les occlusions possibles causées par les équipements présents dans la salle. Cette infrastructure fournit la référence métrologique utilisée dans les protocoles expérimentaux des chapitres 4 et 5.



Figure 2.5 Quatre des six émetteurs du système iGPS installés dans le laboratoire de métrologie

2.6 Stratégie de validation et articulation des protocoles

La validation est organisée autour de trois volets qui convergent vers l'évaluation de la transition entre localisation autonome et localisation assistée par iGPS. Le premier porte sur le jumeau numérique et évalue la cohérence du comportement entre la simulation et le robot réel. Le deuxième porte sur l'intégration de l'iGPS au niveau du bras. Il comprend la calibration du dispositif fixé au poignet, l'intégration logicielle de la mesure et l'évaluation de l'exactitude de positionnement et de la régularité du suivi de chemin.

Le troisième volet concerne la localisation de la base mobile et la transition vers l'iGPS. Il s'appuie sur une configuration de localisation autonome fondée sur les capteurs embarqués et définit l'architecture de transition vers une localisation assistée par iGPS dans la zone instrumentée. La démonstration intégrée sert de validation fonctionnelle de cette transition. Elle

inclut l'utilisation de l'iGPS comme source de pose globale pour la navigation finale et comme source de pose cartésienne pour la manipulation.

La validation ne repose donc pas sur une seule métrique. Selon le cas, elle s'appuie sur la cohérence *sim-to-real*, l'exactitude de pose, la régularité du suivi, la continuité d'une transition de localisation ou la réussite fonctionnelle d'un scénario. Les chapitres 3 à 5 présentent les protocoles détaillés, les résultats associés et leur interprétation.

CHAPITRE 3

JUMEAU NUMÉRIQUE

3.1 Contexte initial, portée et écart à combler

Ce chapitre présente la représentation numérique du RanGen utilisée dans le mémoire. Elle sert de base commune pour développer, tester et comparer les fonctions qui préparent l'intégration de la navigation, de l'iGPS et de la manipulation sur le robot réel. Elle regroupe le modèle de la plateforme, l'environnement Gazebo et la chaîne ROS 2 qui exécute des commandes comparables des deux côtés.

Dans l'état initial du projet, la plateforme réelle était déjà largement intégrée. La base mobile, le bras Kinova, la pince et les capteurs étaient accessibles dans ROS 2 par l'intermédiaire des ordinateurs embarqués. La base pouvait déjà être utilisée pour des essais simples de déplacement, et la télécommande fournie par AgileX permettait un pilotage direct par la carte mère. Le point de départ du chapitre n'est donc pas un robot à construire, mais une plateforme fonctionnelle dont la contrepartie en simulation restait incomplète pour les besoins du mémoire.

L'objectif n'est pas de reproduire l'ensemble des comportements du système réel, mais de construire une représentation suffisamment complète et cohérente pour l'intégration, le développement et la validation des fonctions nécessaires à la transition de localisation. La contribution principale du chapitre porte sur la base mobile. Sur le robot, un nœud de commande existait déjà et encapsulait correctement le bas niveau, mais sa logique devait être reprise pour mieux s'intégrer au reste de la chaîne. En simulation, des paquets étaient disponibles en ROS 1 et en ROS 2 (AgileX Robotics, 2026), sans offrir un contrôleur exploitable pour les besoins de ce travail. Le contrôleur disponible ne prenait pas en charge la translation latérale pure ni le pivot. Un nouveau contrôleur de simulation a été développé. Le nœud du robot réel a aussi été adapté afin de conserver une logique de commande cohérente. Les essais du chapitre valident le comportement obtenu.

La situation du bras était plus simple, puisque l'essentiel de la chaîne existait déjà en simulation comme sur le robot réel. Le travail a surtout porté sur l'harmonisation des interfaces de commande entre les deux branches. Cette continuité entre la simulation et le robot réel est brièvement présentée dans la section 3.4.

3.2 Démarche de conception et critères de fidélité

La démarche retenue pour la base mobile ne cherche pas à reconstruire le fonctionnement interne complet du contrôleur, car celui-ci n'est pas entièrement accessible. Elle vise plutôt à reproduire en simulation le comportement observable de la base à partir des mêmes commandes de haut niveau. La comparaison s'appuie donc sur l'observation du robot réel et sur le réglage du contrôleur simulé jusqu'à une réponse comparable.

Le jumeau numérique vise une cohérence opérationnelle entre la simulation et le robot réel. La consigne de haut niveau et les conventions d'interface ROS 2 restent les mêmes des deux côtés. Pour la base mobile, la chaîne commune repose sur une consigne *Twist*, c'est-à-dire une vitesse du châssis exprimée par (v_x, v_y, ω) , et sur une commande intermédiaire associée au mode de déplacement actif. Dans ce chapitre, cette commande intermédiaire est définie par le mode actif et par le triplet (v_{lin}, ϕ, ω) , où v_{lin} représente la vitesse linéaire, ϕ l'angle de braquage et ω la vitesse angulaire.

Cette commande intermédiaire constitue le dernier niveau à la fois commun, observable et explicitement maîtrisé dans le logiciel du projet, avant la divergence vers les deux chaînes d'exécution. Le critère principal de fidélité est donc comportemental. Pour une même intention de commande et pour une même commande intermédiaire, la simulation doit produire une réponse proche de celle du robot réel. Les étapes internes propres à chaque boîte noire ne sont pas comparées directement.

3.3 Construction du modèle numérique du RanGen

Les modèles du bras Kinova et de la pince Robotiq existaient déjà. La base mobile ne disposait pas d'un modèle exploitable pour la simulation. Elle a donc été reconstruite à partir de sa géométrie CAO, puis intégrée dans une description URDF complète avec les liaisons et interfaces nécessaires à la simulation et au reste de la chaîne logicielle.

Gazebo sert d'environnement principal pour le jumeau numérique. Le modèle du RanGen, la chaîne `ros2_control` de la base mobile et le monde de simulation y sont réunis dans un environnement de développement, d'observation et d'essai.

Le monde de simulation a été modélisé à partir du plan vectoriel des deux laboratoires et du corridor utilisés dans les essais réels. Le niveau de détail retenu privilégie la cohérence spatiale des essais. Comme la perception exploitée dans ce chapitre repose surtout sur un LiDAR 2D, une représentation essentiellement plane de l'environnement est suffisante pour le mémoire. La figure 3.1 présente une vue d'ensemble de ce monde de simulation.

Comme l'illustre la figure 3.2, le jumeau numérique repose sur deux éléments complémentaires : un modèle intégré du RanGen et un monde de simulation cohérent avec le contexte expérimental. Ensemble, ces éléments fournissent la base de travail nécessaire pour développer la commande, observer le comportement du système et comparer la simulation au robot réel.

3.4 Continuité des interfaces entre simulation et robot réel

ROS 2 fournit le cadre commun où les commandes, les états et les observations transitent par les mêmes interfaces entre la simulation et le robot réel. La continuité recherchée porte sur la commande intermédiaire introduite plus haut. La comparaison est ancrée à ce niveau, avant la divergence des deux exécutions dans leurs boîtes noires respectives.

Cette continuité se retrouve aussi dans la chaîne de commande du bras. Elle repose sur un PID cartésien qui produit une vitesse cartésienne, ensuite convertie en vitesses articulaires. En simulation, un nœud d'adaptation convertit ces vitesses en trajectoires très courtes pour

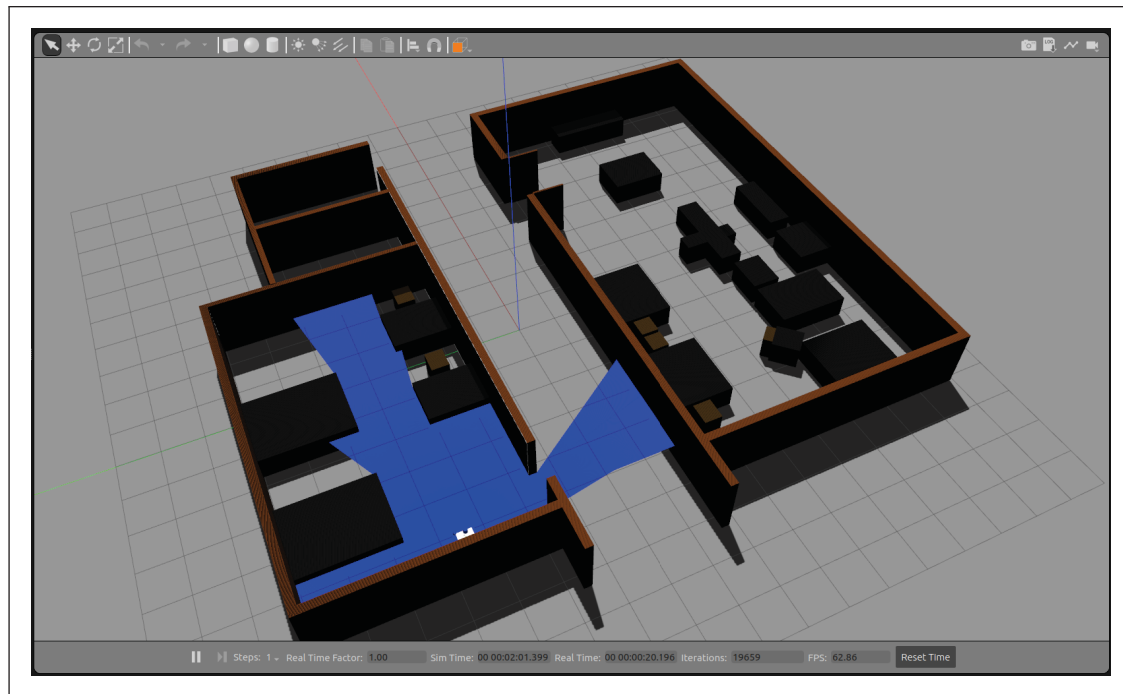


Figure 3.1 Vue d'ensemble du monde de simulation réalisé pour le jumeau numérique du RanGen

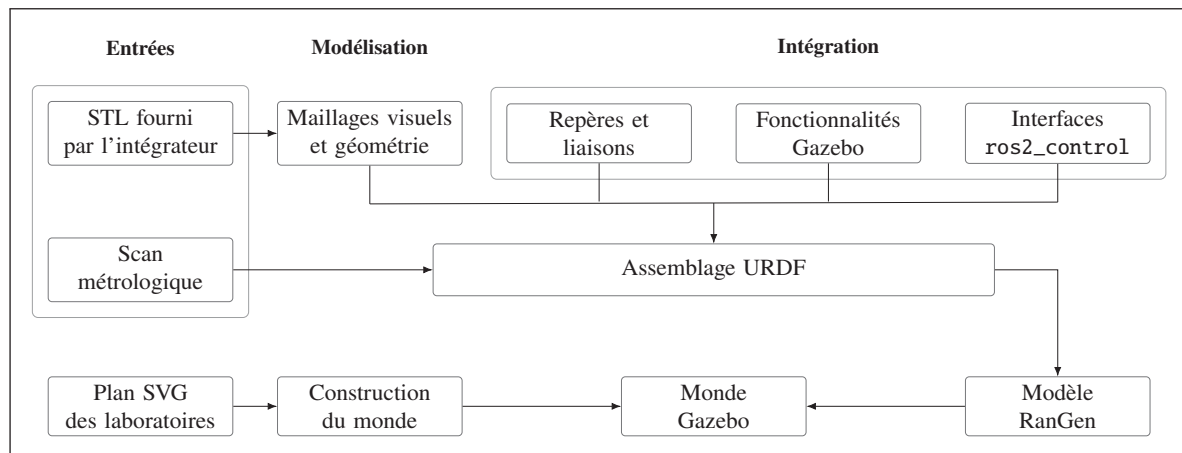


Figure 3.2 Chaîne de construction du jumeau numérique du RanGen

le `joint_trajectory_controller`. Sur le robot réel, un *wrapper* ROS 2 fondé sur l'API KINOVA KORTEK™ (Kinova Robotics, 2022) conserve la même interface d'entrée avec une exécution plus stable que le simple réemploi de la chaîne de simulation. Le détail de cette

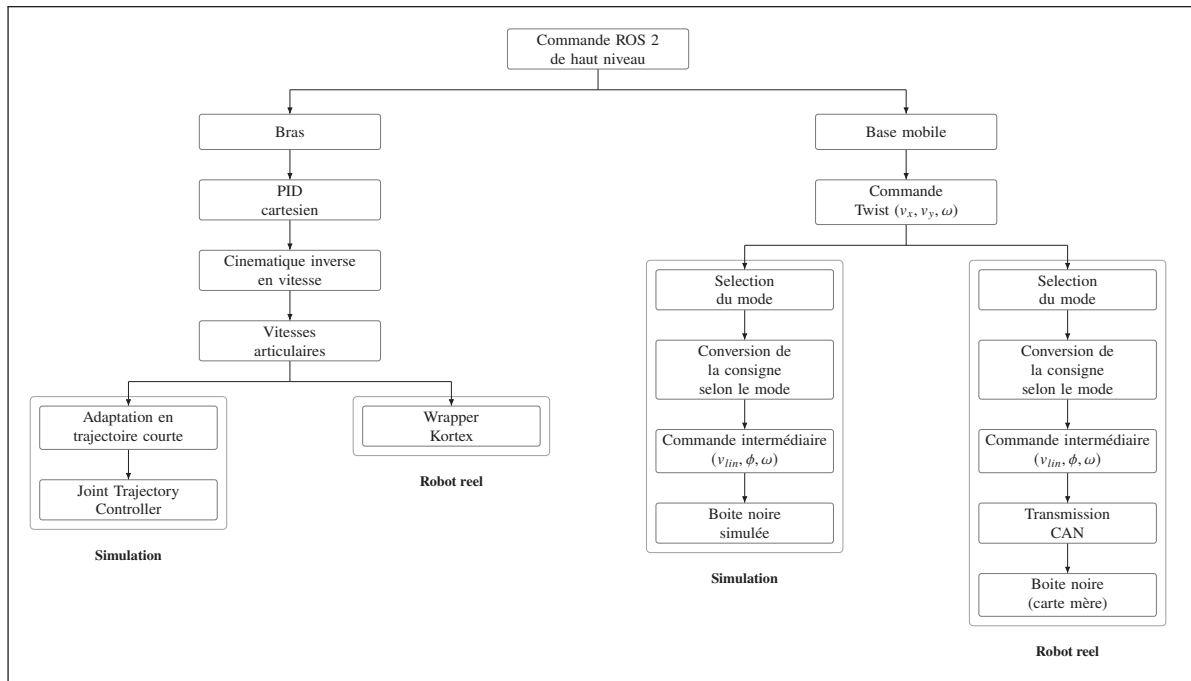


Figure 3.3 Chaîne de commande simplifiée entre la simulation et le robot réel. La logique reste commune jusqu’au niveau de commande commun, puis diverge selon les contraintes propres à chaque branche

branche n’est pas l’objet du chapitre. Le point retenu est de conserver le même nœud amont des deux côtés.

Pour la branche de la base mobile, la figure 3.3 représente la continuité logique de la chaîne de commande, et non la répartition exacte de ses étapes dans les composants logiciels. Elle met en évidence que le dernier point d’entrée explicitement commun aux deux branches est une commande en vitesse cartésienne de type *Twist* (v_x, v_y, ω) .

Les deux branches reconstruisent toutefois la même commande intermédiaire. Celle-ci constitue le dernier niveau de commande commun avant l’exécution propre à chaque branche. La comparaison entre la logique de commande en simulation et celle du robot réel s’ancre à ce niveau.

Cette commande intermédiaire n’apparaît pas comme une interface commune unique sur la figure, car elle n’est pas produite par la même chaîne logique des deux côtés. En simulation,

elle est construite dans la chaîne `ros2_control`, tandis que, sur le robot réel, la même logique amont est reprise dans le nœud `ranger_messenger` jusqu'à ce niveau avant l'envoi vers la carte mère. Une continuité plus directe n'était pas possible sur le robot réel, car la carte mère ne donnait pas accès à une interface matérielle `ros2_control` capable d'envoyer directement des consignes de vitesse aux roues motrices et de position aux roues directrices.

Ce choix conserve une architecture de simulation structurée autour de `ros2_control`. Le contrôleur devient ainsi plus facile à maintenir, à réutiliser et à adapter à d'autres environnements de simulation. Il évite aussi de multiplier les nœuds spécifiques.

La continuité d'interface est donc plus directe pour le bras que pour la base mobile. Pour la base mobile, elle impose des choix d'architecture et de commande qui structurent directement la suite du chapitre.

3.5 Développement du contrôleur de la base mobile

3.5.1 Frontière de modélisation et point de comparaison

Sur le robot réel, la base mobile n'est accessible qu'à un haut niveau de commande. Le mode de mouvement, l'état des actionneurs et l'odométrie restent observables, alors que la conversion vers le bas niveau et la reconstruction interne de l'odométrie ne le sont pas. Cette opacité fixe la frontière de modélisation retenue. Le contrôleur simulé porte donc sur la logique commune située en amont de cette boîte noire. Cette logique couvre la sélection du mode, la conversion de la consigne selon le mode et la construction de la commande intermédiaire.

3.5.2 Sélection des modes et conversion de la consigne

La plateforme mobile exploite trois modes de déplacement : le mode double Ackermann, le mode parallèle et le pivot sur place. Le contrôleur sélectionne le mode à partir des composantes (v_x, v_y, ω) de la consigne. Une composante latérale active le mode parallèle. Une rotation sans

translation mène au pivot sur place. Une translation longitudinale combinée à une rotation active le mode double Ackermann.

Le changement de mode intègre une zone morte autour des seuils de transition : tant que la commande reste sous ces seuils, le contrôleur conserve le mode précédent au lieu de revenir à un mode par défaut. Un délai de commutation limite ensuite les basculements répétés. Un nouveau mode n'est appliqué qu'une fois ce délai écoulé.

Pour chaque mode actif, le contrôleur convertit la consigne *Twist* vers une commande intermédiaire de forme commune. En double Ackermann, (v_x, ω) donne une vitesse linéaire et un angle de braquage. En mode parallèle, (v_x, v_y) donne une vitesse linéaire signée et une direction de translation. En pivot, seule la composante angulaire est conservée. La commande intermédiaire reste ainsi décrite par le mode actif et par le triplet (v_{lin}, ϕ, ω) .

La figure 3.4 synthétise cette chaîne fonctionnelle pour la base mobile.

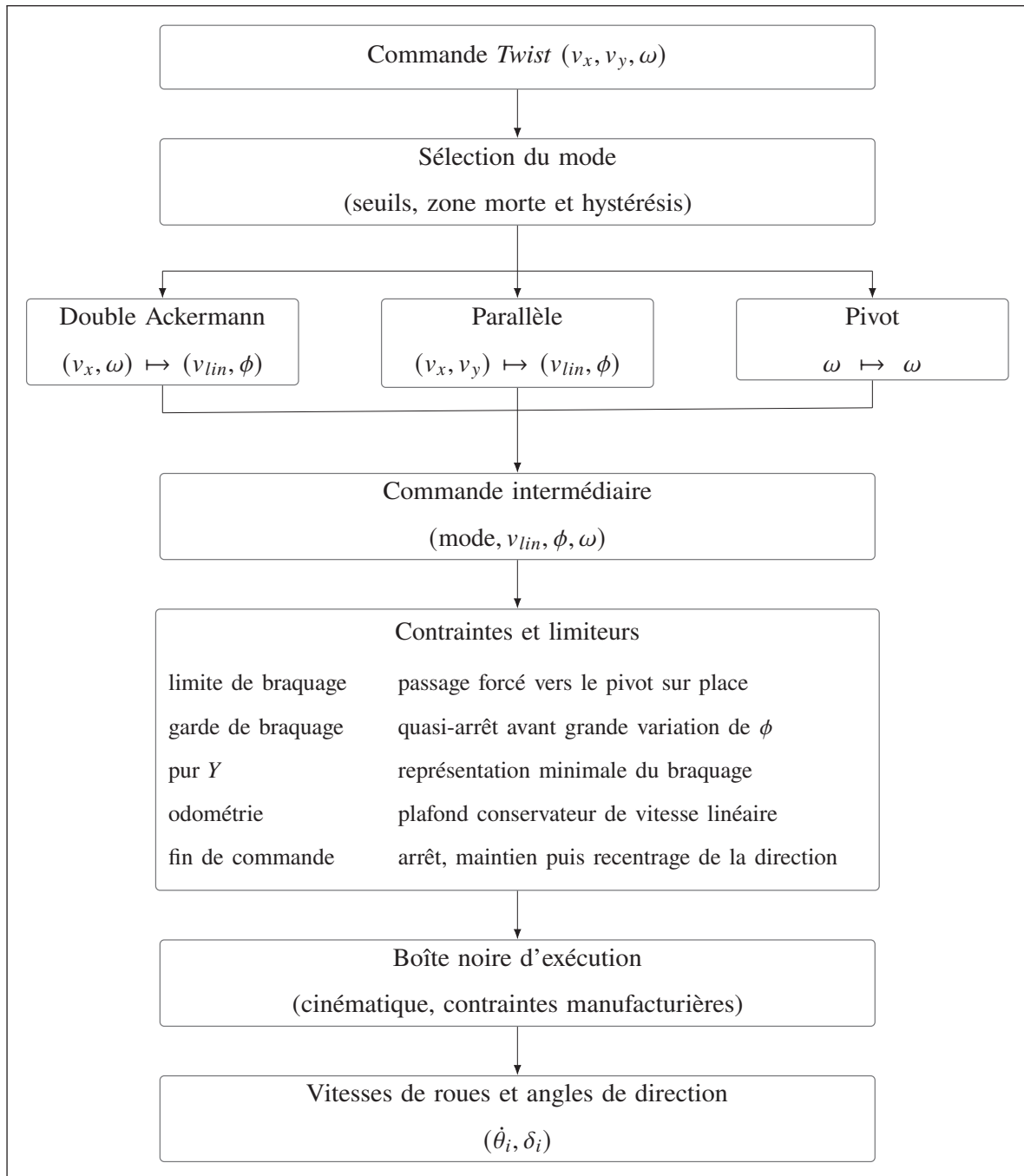


Figure 3.4 Chaîne fonctionnelle de préparation et d'exécution de la commande de la base mobile. La commande *Twist* sert à déterminer le mode actif et à produire la commande intermédiaire associée, avant l'application des contraintes puis l'exécution par la boîte noire

En mode double Ackermann, la consigne est convertie à partir de (v_x, ω) pour construire le couple

$$u_{\text{Ackermann}} = (v_{lin}, \phi), \quad (3.1)$$

où v_{lin} reprend la vitesse longitudinale demandée et où ϕ correspond à l'angle de braquage interne transmis à la boîte noire. La demande de rotation sert donc à calculer le braquage, puis n'agit plus comme variable indépendante dans l'exécution.

En mode parallèle, la consigne est convertie à partir de (v_x, v_y) pour construire

$$u_{\text{parallel}} = (v_{lin}, \phi). \quad (3.2)$$

La norme de la commande de translation donne $|v_{lin}|$, tandis que sa direction donne ϕ . La direction demandée est ramenée à une représentation compatible avec la plage de braquage admissible. Lorsque ce rabattement impose un décalage de π , le signe de v_{lin} est inversé de manière à conserver le même déplacement du châssis. Le cas des déplacements latéraux purs est précisé dans la sous-section 3.6.4.

En mode pivot, la conversion de la consigne conserve uniquement

$$u_{\text{pivot}} = \omega. \quad (3.3)$$

La vitesse linéaire et le braquage restent nuls, car la géométrie de pivot est imposée par la chaîne d'exécution elle-même.

Dans les trois cas, le point d'ancrage de la comparaison reste le même : la commande intermédiaire conserve sa structure, même si ses composantes utiles dépendent du mode actif. Cette conversion prépare l'interface commune. La conversion cinématique vers les vitesses de roues et les angles de direction reste prise en charge par la boîte noire d'exécution.

3.5.3 Contraintes appliquées à la commande intermédiaire

La commande intermédiaire peut aussi recevoir des contraintes optionnelles. Celles-ci peuvent être activées ou désactivées selon le besoin d'essai. Elles stabilisent la commande et rendent le comportement produit par la boîte noire plus lisible. Cet apport est nécessaire, car le contrôleur initial revenait toujours en Ackermann. Certains effets dépendants du mode, comme le maintien de la direction, devenaient alors plus difficiles à observer.

La première contrainte concerne la limite géométrique du braquage en mode Ackermann. Lorsqu'une commande impose un virage trop serré pour l'angle de braquage maximal de la plateforme, le contrôleur force un passage vers le pivot sur place. Une hystérésis conserve aussi temporairement ce mode lorsque la base pivote déjà et que la commande reste proche de cette limite. Cette règle évite des bascules inutiles entre pivot et Ackermann pendant les manœuvres serrées.

Une deuxième contrainte porte sur les changements importants de braquage. Lorsqu'une transition impose une forte variation de l'angle de direction alors que la base roule encore, le contrôleur impose un quasi-arrêt avant le changement de braquage des roues directrices. Cette règle, appelée *garde de braquage en virage* (`steering_turn_gate`), vise à limiter les coins arrondis et les courts départs dans la mauvaise direction observés lors de certaines transitions. Elle encadre ces changements de direction et rend leur forme plus homogène d'un virage à l'autre.

Une troisième contrainte cible les déplacements latéraux purs. Dans ce cas, une même commande ($v_x = 0, v_y \neq 0$) peut conduire à deux représentations cinématiques valides, associées à un changement de signe de la vitesse linéaire. Le contrôleur retient alors celle qui demande le plus petit changement de braquage. Dans le texte, cette règle est désignée comme la contrainte de représentation minimale pour les déplacements selon Y uniquement. Elle limite les reconfigurations inutiles de la direction pour une commande latérale simple. Cet aspect est particulièrement utile en téléopération, en calibration et lors des phases de positionnement précis.

Une quatrième contrainte, plus conservatrice, peut aussi agir sur la commande intermédiaire avant l'entrée dans la boîte noire. Lorsqu'elle est activée, l'option de sécurité odométrique `common_cap_linear_vel_for_odom` applique une limite commune de vitesse linéaire, égale au minimum des plafonds de secteur. Son rôle est d'éviter des discontinuités odométriques lors des changements de plafond de vitesse entre secteurs. La motivation de cette mesure est détaillée dans la sous-section 3.5.5.

La chaîne de commande a aussi été renforcée par des mécanismes liés à la direction et à la fin de commande. Après une commande non nulle, l'orientation des roues n'est pas recentrée immédiatement. Lorsque la commande dépasse un premier délai d'expiration, la composante de vitesse est ramenée à zéro, tandis que la direction reste maintenue à sa dernière valeur. Cette phase impose une décélération dans le sens du mouvement en cours. Elle réduit les chocs mécaniques et améliore la fluidité en téléopération comme en contrôle autonome. Le recentrage de la direction n'est engagé qu'après un second délai réglable, dédié au maintien temporaire de la direction.

Cette préparation de commande n'est pas limitée à la simulation. Elle est aussi reportée dans le nœud de contrôle utilisé sur le robot réel, afin que le même traitement soit appliqué des deux côtés avant la divergence finale vers la base réelle ou son équivalent simulé. Cette harmonisation réduit les écarts introduits par le logiciel de commande et rend la comparaison *sim-to-real* plus directement interprétable.

3.5.4 Construction de la boîte noire simulée

La boîte noire simulée exécute la commande dans un modèle capable de reproduire le comportement observé du robot. La commande intermédiaire ne peut pas être appliquée directement dans la simulation. Son interprétation dépend du mode actif, et sa conversion en consignes de roues et de direction relève de cette boîte noire. Celle-ci s'appuie sur les comportements relevés en séquences de mouvement et en téléopération pour reconstruire une réponse observable proche de celle du robot.

Un premier mécanisme gère les états d'arrêt et les phases transitoires. Lorsque la commande demande un arrêt avec une direction déjà établie, une phase de maintien conserve les roues directrices en place pendant que la vitesse des roues motrices décroît vers zéro. Lorsqu'un arrêt complet est demandé, le contrôleur arrête les roues et applique une stratégie de direction dépendante du mode. En pivot, la direction est maintenue selon la géométrie imposée par le mode, comme décrit en sous-section 3.5.2.

La conversion cinématique dépend du mode actif. La figure 3.5 synthétise les variables géométriques utilisées dans les trois cas cinématiques et sert de repère pour les relations présentées ci-dessous.

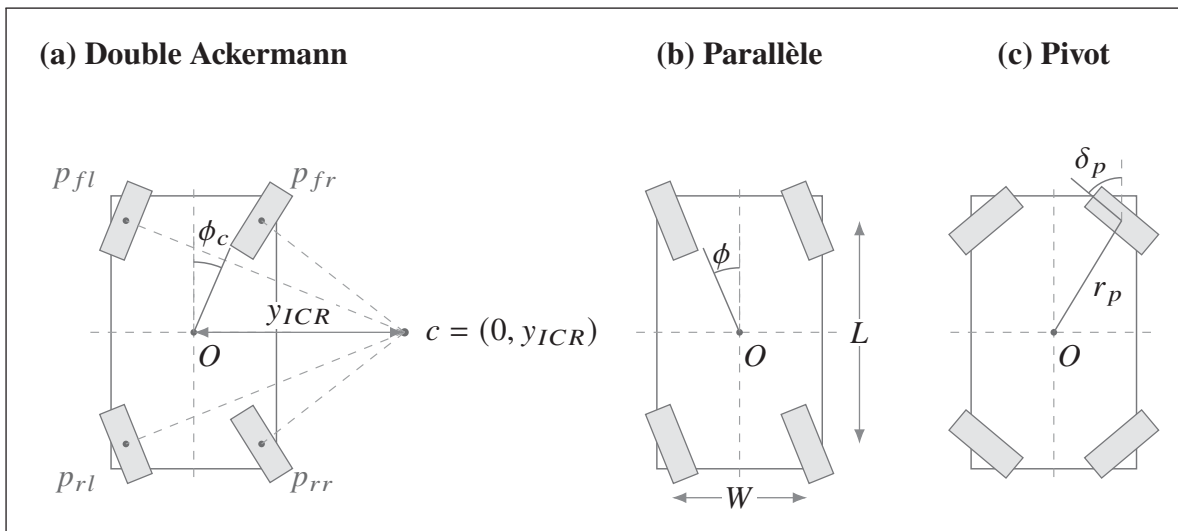


Figure 3.5 Variables géométriques utilisées dans les conversions cinématiques de la base mobile

Dans les relations qui suivent, L désigne l'empattement, W la largeur utile entre les roues directrices et r le rayon des roues. Les angles ϕ_c , ϕ et δ_p correspondent respectivement au braquage central en double Ackermann, à la direction retenue en mode parallèle et à l'angle de pivot imposé aux roues en rotation sur place.

En double Ackermann, la boîte noire reconstruit une vitesse de rotation du châssis à partir de la commande intermédiaire, dont elle déduit les vitesses de roues et les angles de direction.

Le contrôleur convertit l'angle de braquage interne en angle central et reconstruit la vitesse angulaire du châssis

$$\omega = \frac{2v_{lin} \sin(\phi_c)}{L}. \quad (3.4)$$

Le centre instantané de rotation est alors défini par

$$y_{ICR} = \frac{|v_{lin}|}{\omega}, \quad c = (0, y_{ICR}). \quad (3.5)$$

En notant les positions des quatre roues dans le repère du châssis

$$p_{fl} = \left(\frac{L}{2}, \frac{W}{2}\right), \quad p_{fr} = \left(\frac{L}{2}, -\frac{W}{2}\right), \quad p_{rl} = \left(-\frac{L}{2}, \frac{W}{2}\right), \quad p_{rr} = \left(-\frac{L}{2}, -\frac{W}{2}\right), \quad (3.6)$$

la vitesse linéaire de chaque roue devient

$$v_i = |\omega| \|p_i - c\|, \quad \omega_i = \text{sign}(v_{lin}) \frac{v_i}{r}. \quad (3.7)$$

Les angles de direction correspondants sont :

$$\delta_{fl} = \arctan\left(\frac{\omega L}{2|v_{lin}| - \omega W}\right), \quad \delta_{fr} = \arctan\left(\frac{\omega L}{2|v_{lin}| + \omega W}\right), \quad (3.8)$$

$$\delta_{rl} = -\delta_{fl}, \quad \delta_{rr} = -\delta_{fr}. \quad (3.9)$$

En mode parallèle, la conversion est plus directe :

$$\delta_{fl} = \delta_{fr} = \delta_{rl} = \delta_{rr} = \phi, \quad \omega_{fl} = \omega_{fr} = \omega_{rl} = \omega_{rr} = \frac{v_{lin}}{r}. \quad (3.10)$$

En pivot, la géométrie imposée devient

$$\delta_p = \arctan\left(\frac{L}{W}\right), \quad (\delta_{fl}, \delta_{fr}, \delta_{rl}, \delta_{rr}) = (-\delta_p, \delta_p, \delta_p, -\delta_p), \quad (3.11)$$

et, en notant

$$r_p = \sqrt{\left(\frac{L}{2}\right)^2 + \left(\frac{W}{2}\right)^2}, \quad v_p = |\omega| r_p, \quad (3.12)$$

les vitesses de roue deviennent

$$(\omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr}) = \left(-\text{sign}(\omega) \frac{v_p}{r}, \text{sign}(\omega) \frac{v_p}{r}, -\text{sign}(\omega) \frac{v_p}{r}, \text{sign}(\omega) \frac{v_p}{r} \right). \quad (3.13)$$

Compte tenu des entrées retenues dans chacun des trois modes, les relations cinématiques des équations (3.4) à (3.13) constituent un point de départ cohérent pour reconstruire en simulation les consignes de roues et de direction.

Les consignes de direction et de vitesse sont toutes deux conditionnées avant l'application finale des commandes. Une limitation du taux de variation de la direction borne la variation des angles d'un cycle à l'autre, ce qui évite des sauts trop brusques et réduit les oscillations autour de 0° . Un seuil d'alignement compare les consignes de direction au retour de direction disponible. Tant que la direction n'est pas suffisamment alignée, la consigne de vitesse des roues reste nulle. Ce seuil est plus strict en mode pivot ainsi que lors d'une translation latérale pure. Dans les autres cas de translation, une tolérance plus large est utilisée.

La boîte noire détermine aussi le secteur de braquage à partir d'un angle estimé par le retour de direction. Elle choisit entre deux limiteurs de vitesse distincts, associés à un secteur peu braqué et à un secteur fortement braqué. La frontière entre ces deux secteurs a été estimée au voisinage de la limite de braquage retenue en mode Ackermann. La sélection fixe le plafond de vitesse par secteur à utiliser pour la commande en cours. Cette logique est illustrée par la figure 3.11 et par la figure 3.12.

Le plafond de vitesse ainsi défini est ensuite appliqué par le limiteur dynamique standard. Celui-ci borne la vitesse, l'accélération et le jerk selon les paramètres du contrôleur. Le réglage expérimental de ses paramètres d'accélération et de décélération est présenté dans la sous-section 3.6.6.

3.5.5 Calcul de l'odométrie du contrôleur

Sur cette base mobile à quatre roues directrices, les roues peuvent prendre des orientations différentes de l'axe du châssis. L'odométrie doit donc exploiter directement les vitesses de roue et les angles de direction mesurés.

Une première odométrie reconstruisait le mouvement séparément pour chaque mode actif, à partir des relations cinématiques correspondantes. Cette approche restait suffisante tant que les roues accéléraient et s'arrêtaient de façon quasi instantanée. L'ajout du limiteur dynamique standard sur les roues a rendu les transitions plus réalistes et plus difficiles à suivre avec cette approche par mode. Le calcul a donc été reformulé comme une estimation par moindres carrés de la vitesse du châssis (v_x, v_y, ω) à partir des contraintes apportées par les quatre roues.

En reprenant les notations de la figure 3.5, $p_i = (x_i, y_i)$ désigne la position fixe de la roue i dans le repère du châssis et δ_i son angle de direction instantané. La vitesse mesurée à cette roue est d'abord convertie en vitesse linéaire par $v_i = r \omega_i$. La vitesse du point p_i s'écrit alors :

$$v(p_i) = \begin{pmatrix} v_x \\ v_y \end{pmatrix} + \omega \begin{pmatrix} -y_i \\ x_i \end{pmatrix}. \quad (3.14)$$

En notant

$$u_i = \begin{pmatrix} \cos(\delta_i) \\ \sin(\delta_i) \end{pmatrix}, \quad (3.15)$$

le vecteur unitaire porté par la direction de roulement de la roue, le modèle cinématique impose que la composante de $v(p_i)$ dans cette direction soit égale à la vitesse mesurée, soit $u_i^\top v(p_i) = v_i$. On obtient ainsi :

$$\cos(\delta_i) v_x + \sin(\delta_i) v_y + (-y_i \cos(\delta_i) + x_i \sin(\delta_i)) \omega = v_i. \quad (3.16)$$

Chaque roue fournit ainsi une contrainte scalaire. En empilant les quatre roues $i \in \{fl, fr, rl, rr\}$, on obtient le système surdéterminé suivant :

$$A \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix} = b, \quad b = \begin{pmatrix} v_{fl} \\ v_{fr} \\ v_{rl} \\ v_{rr} \end{pmatrix}, \quad (3.17)$$

avec

$$A = \begin{pmatrix} \cos \delta_{fl} & \sin \delta_{fl} & -y_{fl} \cos \delta_{fl} + x_{fl} \sin \delta_{fl} \\ \cos \delta_{fr} & \sin \delta_{fr} & -y_{fr} \cos \delta_{fr} + x_{fr} \sin \delta_{fr} \\ \cos \delta_{rl} & \sin \delta_{rl} & -y_{rl} \cos \delta_{rl} + x_{rl} \sin \delta_{rl} \\ \cos \delta_{rr} & \sin \delta_{rr} & -y_{rr} \cos \delta_{rr} + x_{rr} \sin \delta_{rr} \end{pmatrix}. \quad (3.18)$$

Les quatre roues apportent quatre contraintes pour seulement trois inconnues (v_x, v_y, ω) . Le contrôleur estime alors la vitesse du châssis par moindres carrés :

$$A^\top A \begin{pmatrix} \hat{v}_x \\ \hat{v}_y \\ \hat{\omega} \end{pmatrix} = A^\top b. \quad (3.19)$$

Les composantes estimées $(\hat{v}_x, \hat{v}_y)$ sont exprimées dans le repère du châssis. Pour une période Δt , on calcule d'abord l'incrément de pose dans le repère du châssis :

$$\Delta x_b = \hat{v}_x \Delta t, \quad \Delta y_b = \hat{v}_y \Delta t, \quad \Delta \theta = \hat{\omega} \Delta t, \quad (3.20)$$

puis on le projette dans le repère monde à l'aide de l'orientation courante θ_k :

$$\left\{ \begin{array}{l} x_{k+1} = x_k + \Delta x_b \cos \theta_k - \Delta y_b \sin \theta_k, \\ y_{k+1} = y_k + \Delta x_b \sin \theta_k + \Delta y_b \cos \theta_k, \\ \theta_{k+1} = \theta_k + \Delta \theta. \end{array} \right\}. \quad (3.21)$$

Le calcul d'odométrie, défini par les équations (3.16) à (3.21), est vérifié par un essai dédié présenté en annexe I. Cette vérification compare l'odométrie publiée par le contrôleur à la pose de référence fournie par Gazebo. Elle montre que l'odométrie simulée suit correctement le déplacement du modèle et peut donc être utilisée dans la comparaison avec le robot réel.

Un point plus spécifique concerne la transition entre les secteurs de vitesse admissible. Sur le robot réel, le logiciel d'odométrie embarqué peut réagir avec un délai ou une hystérésis lors du passage d'un secteur de braquage à l'autre. L'erreur d'odométrie peut alors croître rapidement. Le contrôleur propose donc une option de sécurité odométrique, qui borne la vitesse linéaire au minimum des plafonds des secteurs peu et fortement braqués. La vitesse maximale devient ainsi identique dans les deux secteurs, ce qui évite l'accumulation d'erreur odométrique lors des transitions. Cette option est destinée aux cas où la qualité de l'odométrie doit être privilégiée, au prix d'une vitesse de déplacement plus faible.

3.6 Validation du jumeau numérique

3.6.1 Processus automatisé d'évaluation et essais retenus

Les essais retenus dans ce chapitre s'appuient sur un processus automatisé d'évaluation. Il comprend la génération d'une séquence de mouvement déterministe, l'exécution de la campagne avec enregistrement des *topics* ROS 2 utiles, l'extraction et la vérification des données, puis l'analyse qui produit les figures et les métriques du mémoire. Ce processus rend les comparaisons entre la simulation et le robot réel reproductibles sans redéfinir les étapes de traitement pour chaque essai.

Le chapitre ne retient toutefois que les essais qui évaluent les mécanismes de commande retenus. Le parcours carré (square) est conservé pour évaluer l'effet de la garde de braquage. L'essai d'allers-retours latéraux selon Y uniquement (*pure_y_roundtrip*) évalue la représentation minimale. L'essai de transition entre secteurs de braquage (*steer_sector_transition*) isole

l'effet du secteur de braquage sur le plafond de vitesse. La séquence d'échelons sur les axes (`axis_step`) sert au réglage de l'accélération et de la décélération.

Les résultats qui suivent ne visent pas à comparer les erreurs des odométries du robot et de la simulation. Les trajectoires servent plutôt à comparer la forme des déplacements et à vérifier si la simulation retrouve les tendances observées sur le robot réel.

3.6.2 Plan de validation

La validation repose sur deux volets complémentaires. Le premier concerne les entrées imposées aux deux branches. Les campagnes utilisées pour la comparaison sont exécutées sur le robot, et les commandes enregistrées y sont rejouées en simulation avec le même profil temporel. La vérification `cmd_mode_timeline` est consultée de façon systématique. Elle confirme que les deux chaînes ont reçu des entrées comparables et donne une lecture compacte de la séquence rejouée. Les composantes de commande et les changements de mode y sont visibles sur la même ligne temporelle.

Le deuxième volet porte sur les comportements directement observables : forme des coins, inversions lors des déplacements selon Y uniquement, variation du plafond de vitesse selon le secteur de braquage, et dynamique d'accélération et de décélération. La conversion interne de la carte mère AgileX et la reconstruction interne de son odométrie ne sont pas évaluées.

3.6.3 Effet de la garde de braquage

Le parcours carré est retenu ici, car ses virages rendent le problème visé directement visible. La garde de braquage en virage a été ajoutée pour encadrer les transitions où les roues changent encore d'orientation pendant que la base avance. La lecture porte sur la forme des quatre virages et sur la manière dont la trajectoire entre et sort de chacun d'eux. L'effet de la garde devient le plus lisible à ces transitions. La figure 3.6 vérifie que ce carré est bien rejoué des deux côtés.

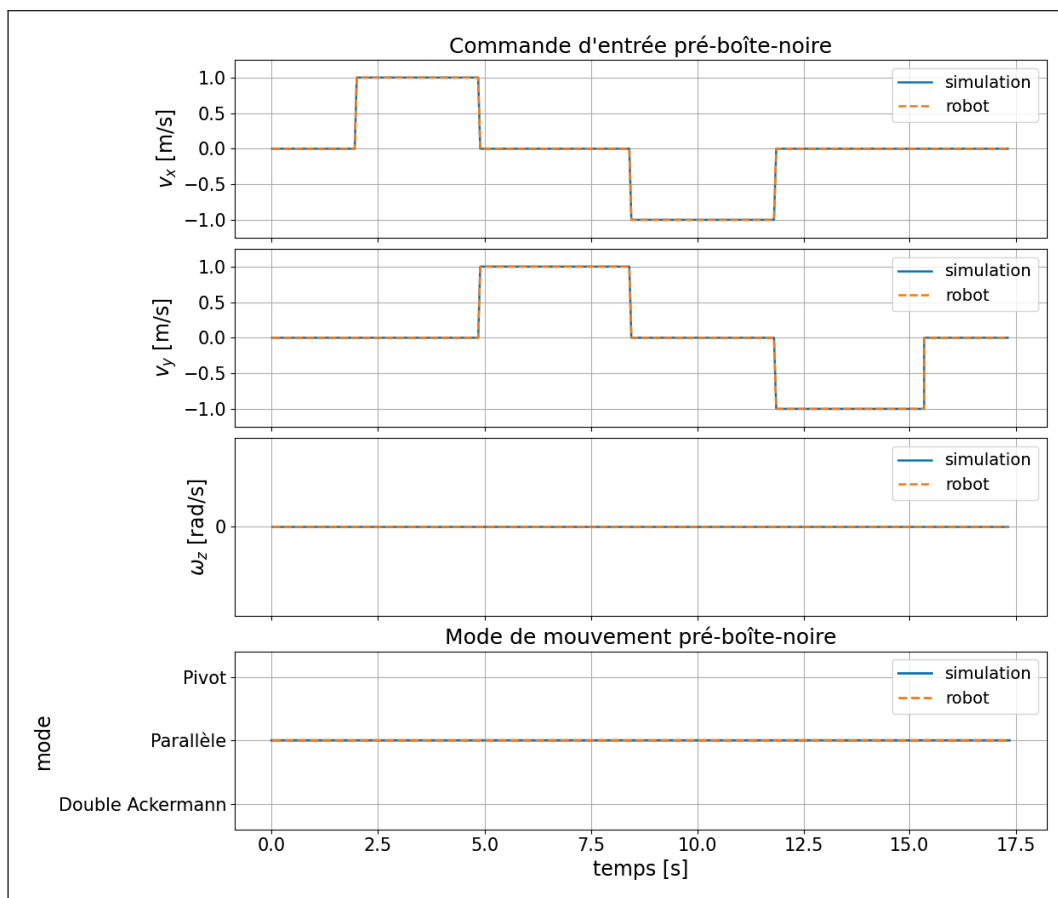


Figure 3.6 Vérification `cmd_mode_timeline` associée à l'essai de garde de braquage. Elle confirme la comparabilité des entrées et montre le parcours carré rejoué

La figure 3.7 montre que, dans la vue (a) sans garde, les quatre virages ne présentent pas une forme homogène. Certains restent arrondis, tandis que d'autres font apparaître un léger crochet au moment de la transition, en simulation comme sur le robot. Dans la vue (b), avec la garde active, les virages deviennent plus réguliers et perturbent moins les segments droits. Cet essai confirme l'intérêt de la garde pour régulariser les transitions de braquage dans les virages serrés.

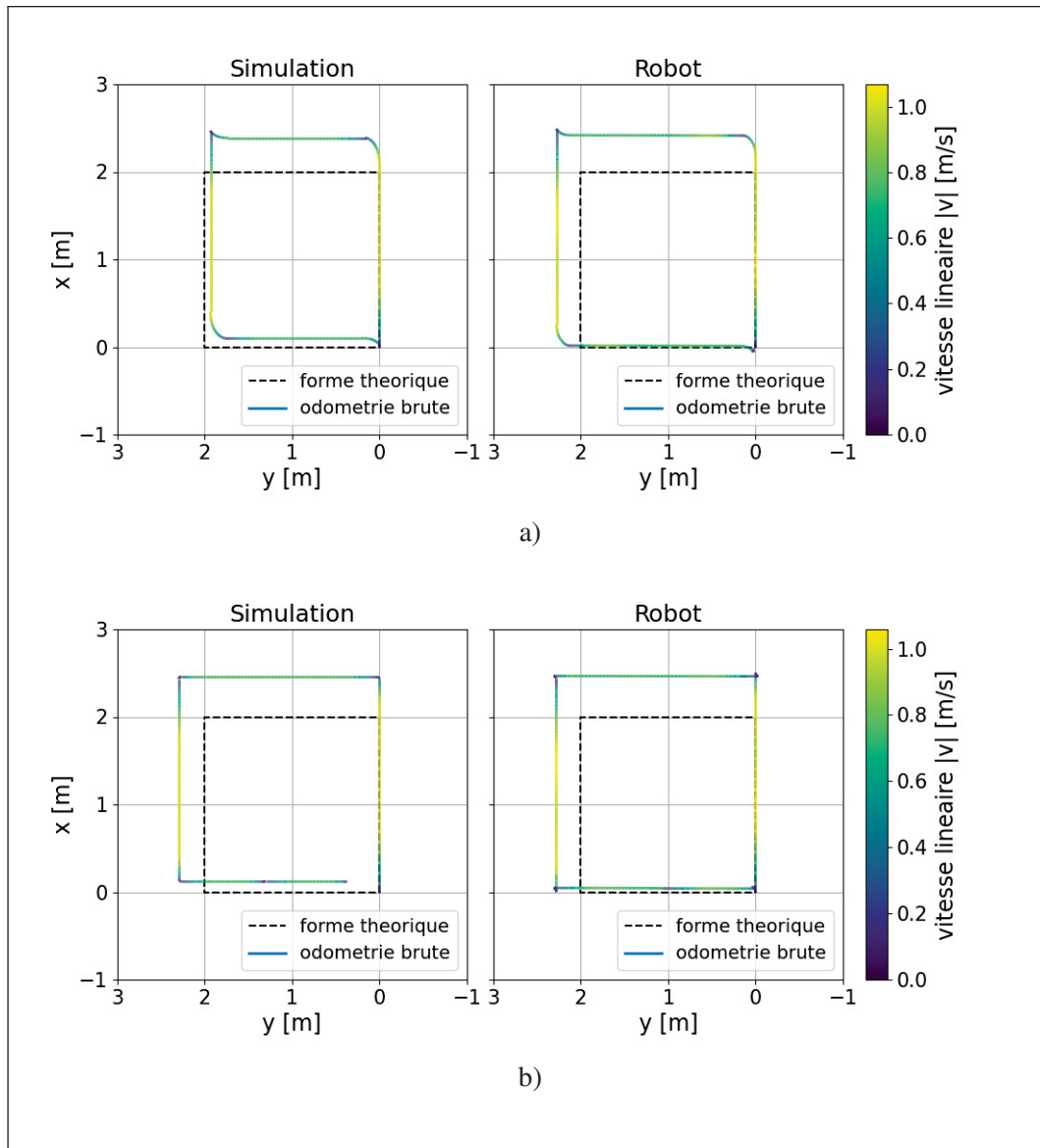


Figure 3.7 Comparaison entre l'odométrie du contrôleur en simulation et celle du robot réel lors de l'essai de garde de braquage. Les deux trajectoires (x, y) sont présentées avec une échelle de vitesse commune, sur un parcours carré. Le cas sans garde est présenté en (a), et le cas avec garde en (b)

3.6.4 Effet de la représentation minimale en déplacement latéral pur

L'essai d'allers-retours latéraux selon Y uniquement caractérise le comportement du contrôleur au moment des inversions de sens. La figure 3.8 vérifie que ces allers-retours restent comparables entre robot et simulation.

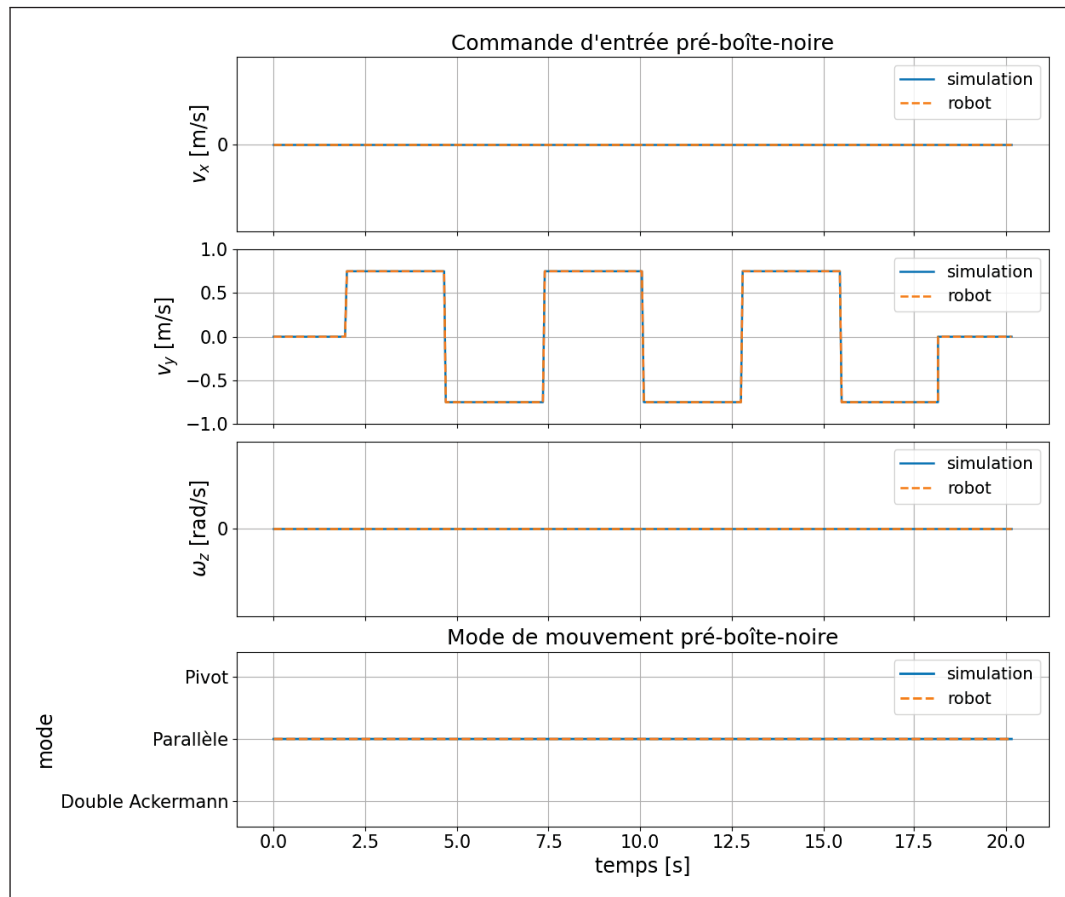


Figure 3.8 Vérification `cmd_mode_timeline` associée à l'essai d'allers-retours latéraux selon Y uniquement. Elle confirme la comparabilité des entrées et montre les inversions de sens successives

La figure 3.8 montre que les phases de commande latérale gardent la même durée d'un cycle à l'autre, y compris au premier aller-retour. La figure 3.9 se lit à partir de ce constat. Dans la vue (a), sans contrainte, les roues doivent être réorientées lors des inversions de sens. Le temps de déplacement latéral effectif est donc réduit, ce qui raccourcit les segments suivants. Dans la

vue (b), avec la représentation minimale, l'inversion repose surtout sur un changement de signe de la vitesse linéaire, et les segments gardent ainsi des longueurs plus homogènes.

La même figure montre aussi que, dans la vue (a), la réorientation des roues lors des inversions provoque une dérive progressive de l'odométrie selon x au fil des allers-retours selon Y . Dans la vue (b), la trajectoire reste plus regroupée et plus proche d'un déplacement selon Y uniquement. La représentation minimale réduit ainsi la dérive hors axe et raccourcit les phases d'inversion.

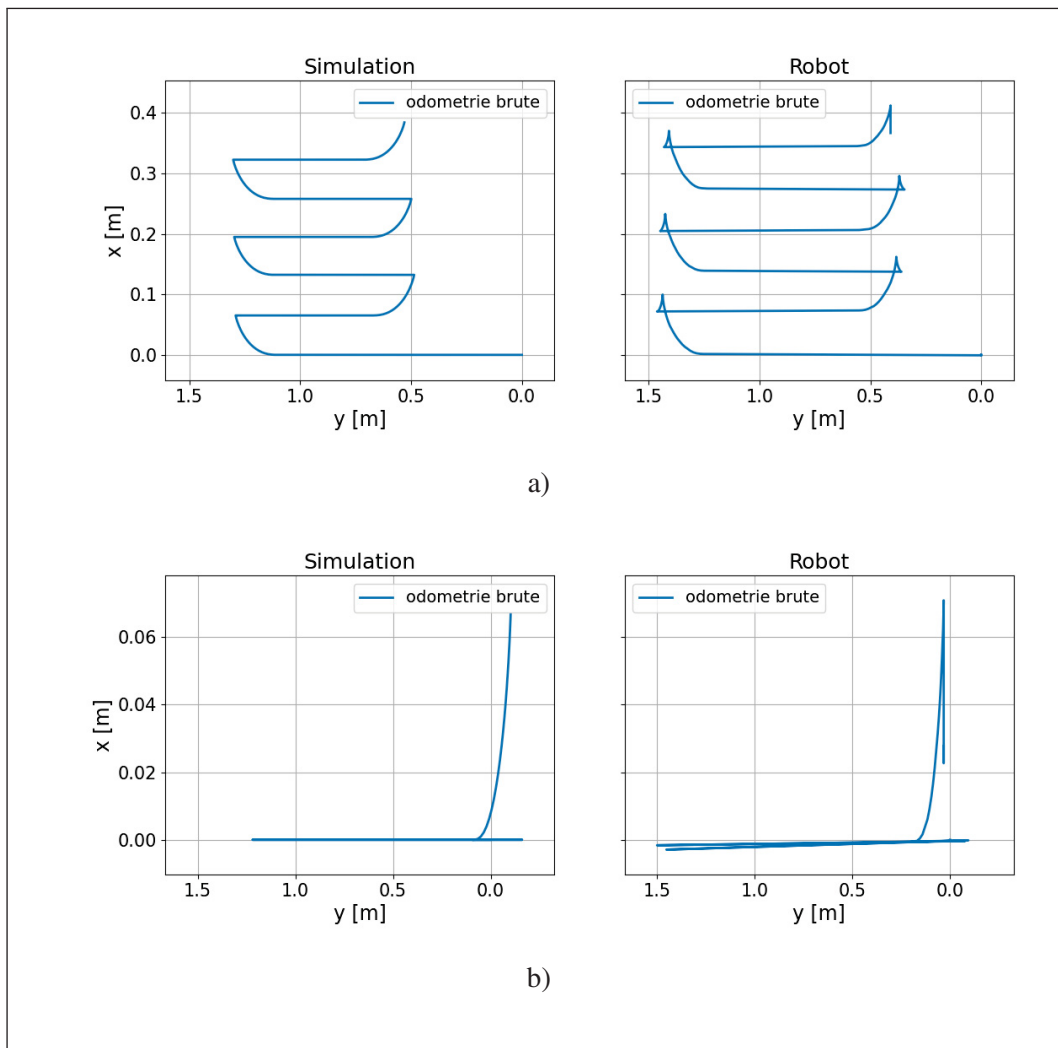


Figure 3.9 Comparaison entre l'odométrie du contrôleur en simulation et celle du robot réel lors de l'essai d'allers-retours latéraux purs. Les trajectoires (x, y) illustrent les inversions successives. Le cas sans contrainte est montré en (a), et le cas avec contrainte en (b)

3.6.5 Effet du plafonnement selon le secteur de braquage

L'essai de transition entre secteurs de braquage balaie la zone de transition entre secteurs et rend visible son effet sur la vitesse admissible. La figure 3.10 vérifie la comparabilité des entrées. Elle montre aussi le balayage commandé qui provoque le franchissement de la frontière entre secteurs.

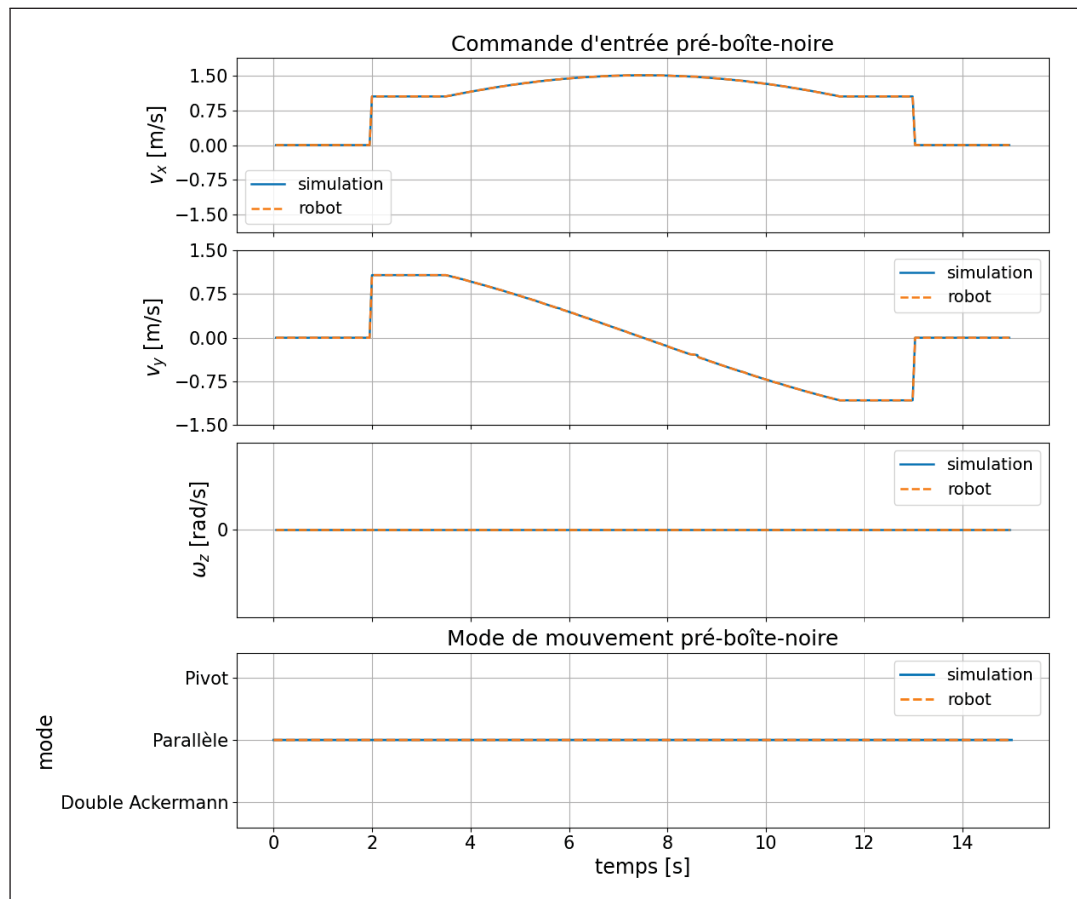


Figure 3.10 Vérification `cmd_mode_timeline` associée à l'essai de transition entre secteurs de braquage. Elle confirme la comparabilité des entrées et montre le balayage commandé qui fait franchir la frontière entre secteurs

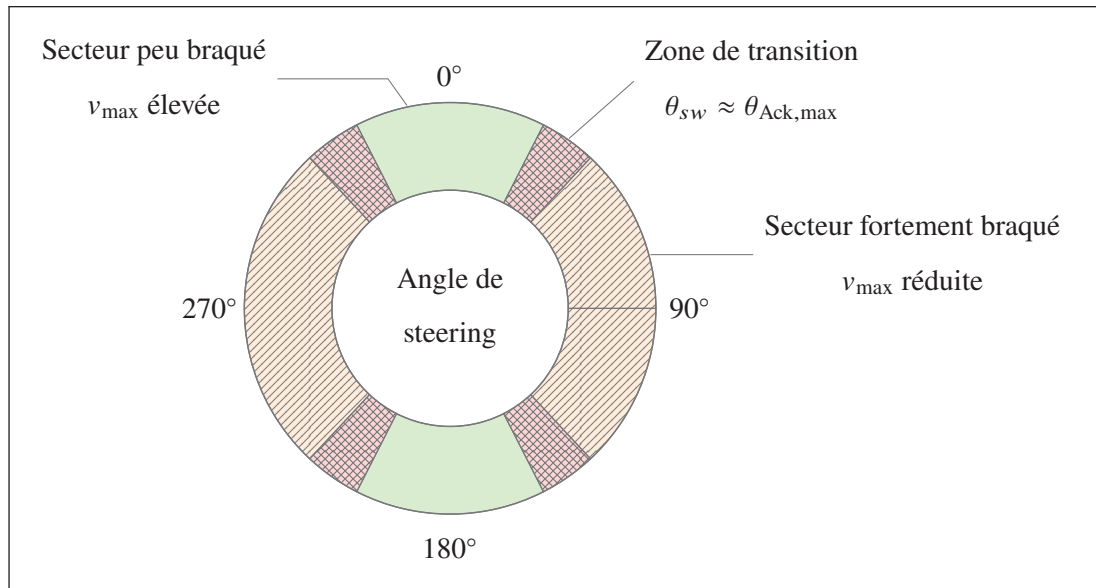


Figure 3.11 Schéma des secteurs de braquage utilisés pour le plafonnement de vitesse. La frontière de transition est estimée au voisinage de la limite de braquage Ackermann

La figure 3.11 présente la carte des secteurs angulaires. Elle distingue les secteurs peu braqués, où le plafond reste plus élevé, des secteurs fortement braqués, où il devient plus contraignant. La zone de transition marque la frontière de bascule entre ces deux régimes. Cette frontière reste ici une estimation pratique, placée au voisinage de la limite de braquage Ackermann, sans identification plus précise dans ce mémoire. Elle met aussi en évidence la zone la plus sensible pour l'odométrie.

La figure temporelle 3.12 se lit par colonnes : la simulation à gauche et le robot à droite. En haut, l'angle de direction est comparé au seuil de secteur, repéré par deux lignes horizontales pointillées. En bas, la vitesse linéaire est comparée à la commande intermédiaire. Cette commande sert de repère pour situer le début et la fin de la séquence. Les lignes verticales indiquent les instants attendus d'entrée et de sortie du secteur peu braqué. Sur le robot, la première transition de vitesse apparaît retardée par rapport à ce franchissement. La seconde reste plus proche du comportement attendu. Ce résultat confirme que le plafond de vitesse dépend du secteur de braquage. La frontière utilisée dans le modèle reste toutefois une estimation du seuil de bascule

interne de la carte mère AgileX. L'objectif est donc de reproduire la dépendance au secteur, sans identifier exactement l'angle de transition réel.

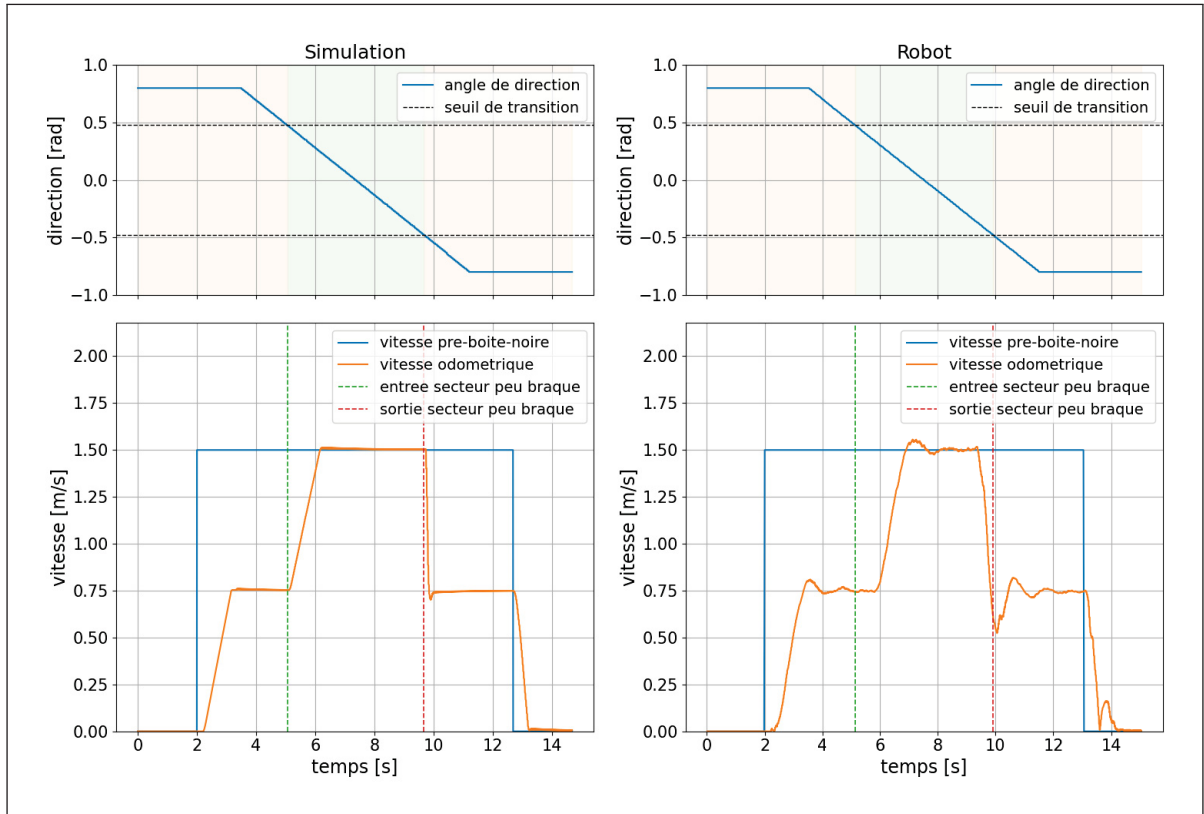


Figure 3.12 Comparaison de l'effet du secteur de braquage sur le plafond de vitesse lors d'une transition contrôlée. Les évolutions temporelles de l'angle de braquage et de la vitesse linéaire sont présentées en simulation et sur le robot réel

3.6.6 Réglage de l'accélération et de la décélération

La séquence d'échelons sur les axes applique d'abord un échelon en v_x , revient à l'arrêt, effectue une rotation sur place d'environ 90° , puis applique un échelon en v_y . Les phases d'arrêt isolent les transitoires et permettent de lire séparément les réponses sur les deux axes de translation. Le réglage des paramètres d'accélération et de décélération a été mené avec cette séquence. La figure 3.13 présente le `cmd_mode_timeline` associé. Elle vérifie la comparabilité des entrées et rappelle cette succession de phases commandées.

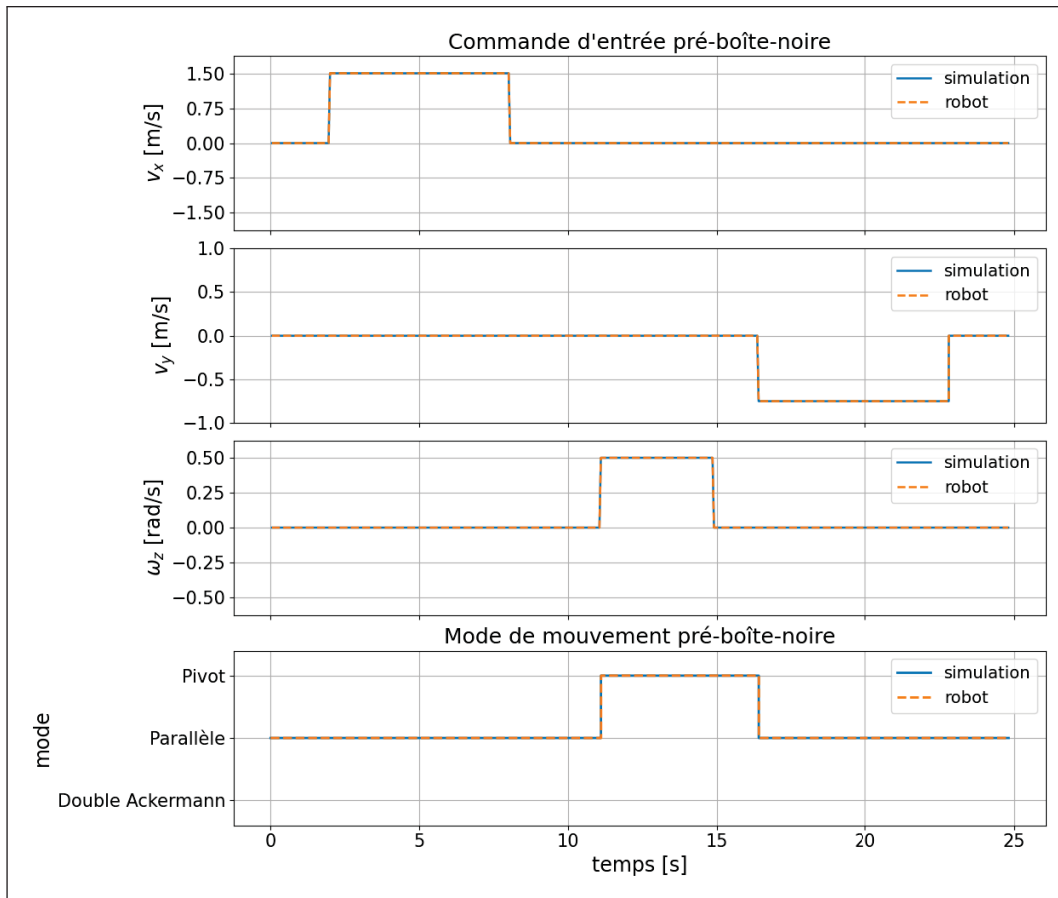


Figure 3.13 Vérification `cmd_mode_timeline` associée à la séquence d'échelons sur les axes. Elle confirme la comparabilité des entrées et montre la montée en v_x , la rotation intermédiaire, puis la montée en v_y

La figure 3.14 regroupe les réponses temporelles sur v_x , puis sur v_y . Dans chaque graphique, la consigne est comparée à l'odométrie. La lecture porte surtout sur les phases transitoires, où les écarts de dynamique entre la simulation et le robot apparaissent le plus nettement.

La figure 3.15 constitue la base principale du réglage. Elle présente la moyenne de quatre essais. La vue (a) sert à ajuster l'accélération à partir des temps de montée, tandis que la vue (b) sert à ajuster la décélération à partir des temps de descente.

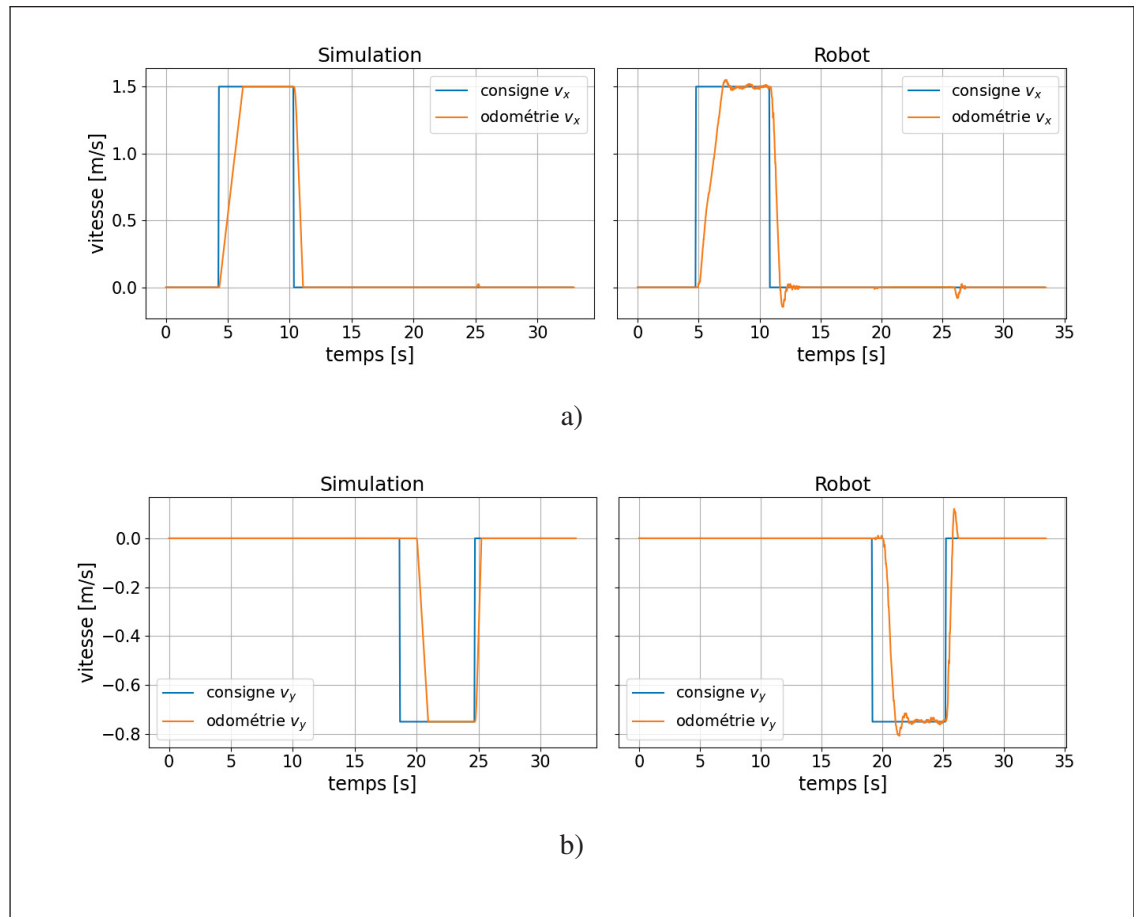


Figure 3.14 Comparaison des réponses temporelles obtenues avec la séquence d'échelons sur les axes. La composante v_x est montrée en (a), et la composante v_y en (b)

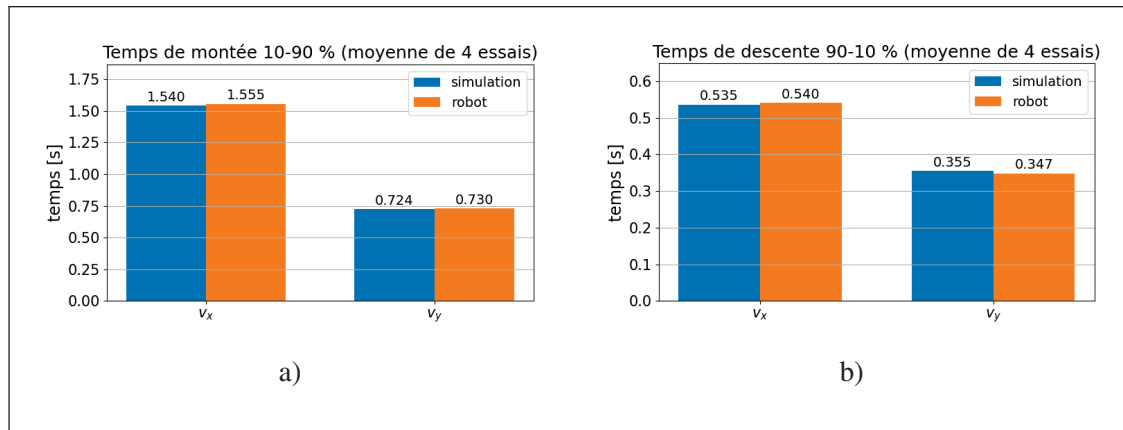


Figure 3.15 Comparaison des temps de montée et de descente avec la séquence d'échelons sur les axes. Le temps de montée 10–90 % est présenté en (a), et le temps de descente 90–10 % en (b)

La figure 3.16 sert de vérification secondaire. Elle compare, pour la simulation et pour le robot, la lecture odométrique et une estimation théorique des accélérations et décélérations moyennes.

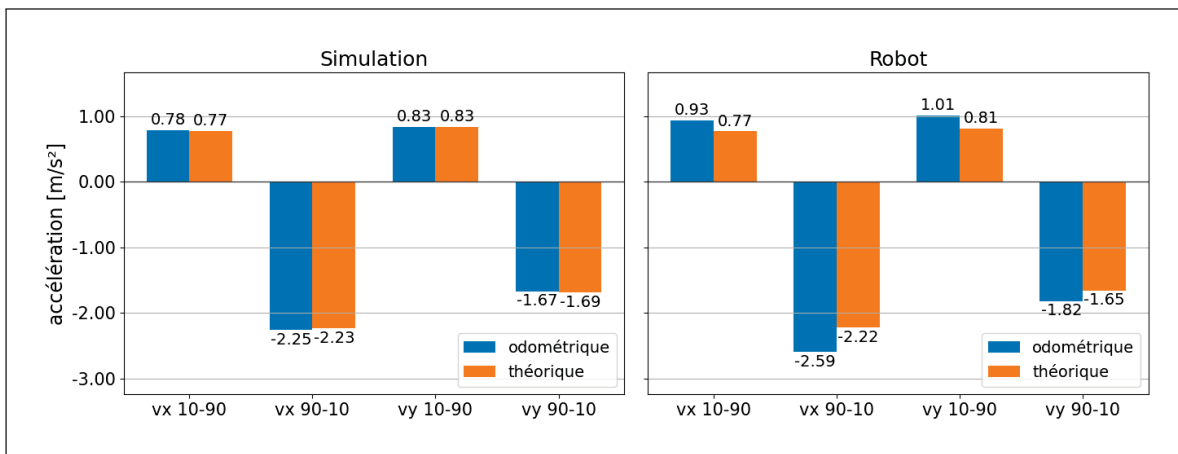


Figure 3.16 Comparaison entre la simulation et le robot réel avec la séquence d'échelons sur les axes. Pour chaque cas, les estimations odométrique et théorique de l'accélération et de la décélération moyennes en v_x et en v_y sont comparées

L'objectif n'est pas d'obtenir une superposition parfaite point par point. Le réglage retenu vise plutôt à placer la dynamique simulée dans le même ordre de grandeur que celle du robot sur les

phases de montée et de descente. Cet essai confirme un réglage comportemental. Il ne constitue pas une identification dynamique complète de la base réelle.

3.6.7 Synthèse de la validation

Pour chacun de ces essais, les profils de commande utilisés expérimentalement sont rejoués en simulation. Les résultats portent sur quatre effets observables. La garde de braquage régularise la forme des coins. La représentation minimale pour les déplacements selon Y uniquement limite les écarts parasites lors des inversions. Le test de transition entre secteurs confirme que le plafond de vitesse dépend du braquage et doit être reproduit dans le modèle. Le réglage des rampes place la dynamique simulée dans le même ordre de grandeur que celle du robot sur les transitoires utiles.

Cette validation n'établit pas une identité complète entre la simulation et la base réelle. Elle montre plutôt une cohérence suffisante à la frontière du modèle retenu pour développer, tester et comparer les fonctions du projet dans une logique *sim-to-real*.

3.7 Discussion

Dans l'ensemble, le jumeau numérique développé dans ce chapitre est d'abord un outil de développement et de validation. Les essais retenus restent ciblés. Ils isolent des comportements précis et justifient des choix de conception, plutôt que d'établir une équivalence parfaite entre la simulation et le robot réel.

Le jumeau numérique ne cherche pas à reproduire toute la logique interne de la base. Il reconstruit plutôt la partie commune de la chaîne de commande, reportée aussi sur le robot réel jusqu'à la commande intermédiaire, puis compare les effets observables produits des deux côtés.

Les résultats sur la garde de braquage montrent surtout un gain d'homogénéité dans les virages. Sans cette contrainte, les coins du parcours carré ne présentent pas tous la même forme. Certains restent arrondis, tandis que d'autres font apparaître un bref départ en sens opposé, suivi d'un

freinage mécanique. La réduction de ces asymétries rend le comportement plus prévisible pour un opérateur humain et limite des chocs indésirables sur le robot réel. La contrainte de représentation minimale pour les déplacements selon Y uniquement agit sur un autre type d'écart. En limitant les retournements inutiles des roues directrices, elle réduit les petits déplacements parasites visibles lors des inversions de sens. Cet effet est particulièrement utile lorsque la base doit conserver un déplacement latéral propre, par exemple en téléopération, en calibration ou dans des phases de positionnement précis.

Le plafonnement par secteur de braquage et le réglage des rampes jouent un rôle plus structurel. Le premier montre que la vitesse admissible de la base dépend de l'orientation des roues. Le second place la dynamique simulée dans le même ordre de grandeur que celle du robot, sans chercher à reproduire toute sa dynamique interne. Des écarts demeurent, en particulier dans les phases transitoires et dans les effets odométriques observés sur le robot réel près des frontières entre secteurs de braquage. Le retard visible sur la première transition illustre ce comportement. Le jumeau numérique ne cherche toutefois pas à reproduire ce comportement. Comme expliqué à la sous-section 3.5.3, celui-ci peut être évité par la contrainte de plafond commun de vitesse `common_cap_linear_vel_for_odom` lorsque nécessaire. Ces écarts restent compatibles avec l'usage visé du jumeau numérique dans le mémoire.

Le chapitre définit aussi une structure de travail réutilisable. Le même processus automatisé d'évaluation peut servir à d'autres essais que ceux retenus ici. Lorsqu'un comportement du robot doit être reproduit ou encadré, une logique ou une contrainte peut être ajoutée dans la commande intermédiaire, puis évaluée par un essai ciblé dans le même processus. L'apport du chapitre tient aussi à cette organisation : elle fournit une méthode commune pour ajouter, isoler et vérifier de nouvelles règles de commande.

3.8 Conclusion

Ce chapitre a présenté le jumeau numérique du RanGen et la validation ciblée de son comportement. Il établit une chaîne de commande cohérente pour la base mobile et une validation

centrée sur le comportement observable. Le résultat obtenu n'est pas un double exact du système physique, mais il fournit un jumeau numérique utilisable pour la suite du travail.

L'approche ne se limite pas aux essais présentés ici. La logique portée par `ros2_control` peut être reprise dans un autre environnement de simulation, et le processus automatisé d'évaluation peut être réutilisé pour vérifier de nouveaux comportements de la base mobile.

À plus long terme, une migration vers Isaac Sim pourrait améliorer la fidélité visuelle et permettre des environnements plus complexes. Pour ce mémoire, Gazebo demeure toutefois un choix adapté aux besoins du projet.

Ce jumeau numérique fournit ainsi une base commune aux développements des chapitres suivants, où la mesure iGPS est intégrée au bras puis utilisée comme source globale dans la transition de localisation.

CHAPITRE 4

BRAS ROBOTIQUE

4.1 Objectif, portée et chaîne instrumentée

Ce chapitre présente l'instrumentation iGPS ajoutée au bras robotique et la chaîne de transformations qui relie la mesure au poignet aux repères du bras, de l'outil et de la base mobile. Il traite de la conception de la couronne, de son intégration ROS 2, des procédures de calibration et des essais d'évaluation.

Le travail établit une chaîne géométrique cohérente entre la mesure iGPS au poignet, la base du bras et le repère de la sonde *i5* tenue dans la pince. Les calibrations portent sur cette instrumentation ajoutée, sans modifier le modèle géométrique interne du bras Kinova. Les repères calibrés servent à localiser la base du bras, à reconstruire la pose de la base mobile dans la zone iGPS, à définir les poses de manipulation et à quantifier l'exactitude de positionnement obtenue. L'essai de régularité compare le suivi d'un même chemin de référence pour deux montages : le bras fixé sur table et le bras embarqué sur la base mobile. Il évalue l'effet du montage mobile sur les oscillations du résidu cartésien.

4.2 Conception de la couronne

L'installation des capteurs iGPS sur le dessus de la base mobile les placerait dans une zone fortement exposée aux occlusions. À cette hauteur, la table, le bras et les objets manipulés peuvent masquer une partie du champ de mesure. La base mobile n'est donc pas retenue comme support d'instrumentation.

Les capteurs iGPS sont placés au poignet du bras. Ce montage améliore leur visibilité et rapproche la mesure de l'outil. La pose de la base du bras est reconstruite par cinématique directe. Cette reconstruction dépend du modèle géométrique du bras, dont les erreurs se propagent dans la pose reconstruite. Ces erreurs ne sont pas mesurées séparément de la chaîne de reconstruction. Elles restent incluses dans le résidu de validation de la pose reconstruite de la base du bras,

synthétisé au tableau 4.3 à partir d'une référence *i6* indépendante. Cette dépendance est acceptée, car le montage au poignet permet de choisir une posture favorable au signal iGPS pendant la navigation et de maintenir les capteurs au-dessus de la table pendant la manipulation.

La solution retenue est une couronne fixée au poignet. Elle porte quatre capteurs iGPS issus de deux sondes *i5* démontées et regroupés en deux paires. Chaque paire reprend ainsi le principe d'une sonde *i5*. Le système iGPS ne fournit une mesure exploitable que si ses deux capteurs sont visibles simultanément. La couronne n'est pas intégrée au modèle de collision du robot. Les interférences entre le bras et ce support ajouté au poignet ne sont donc pas prises en compte lors de la manipulation.

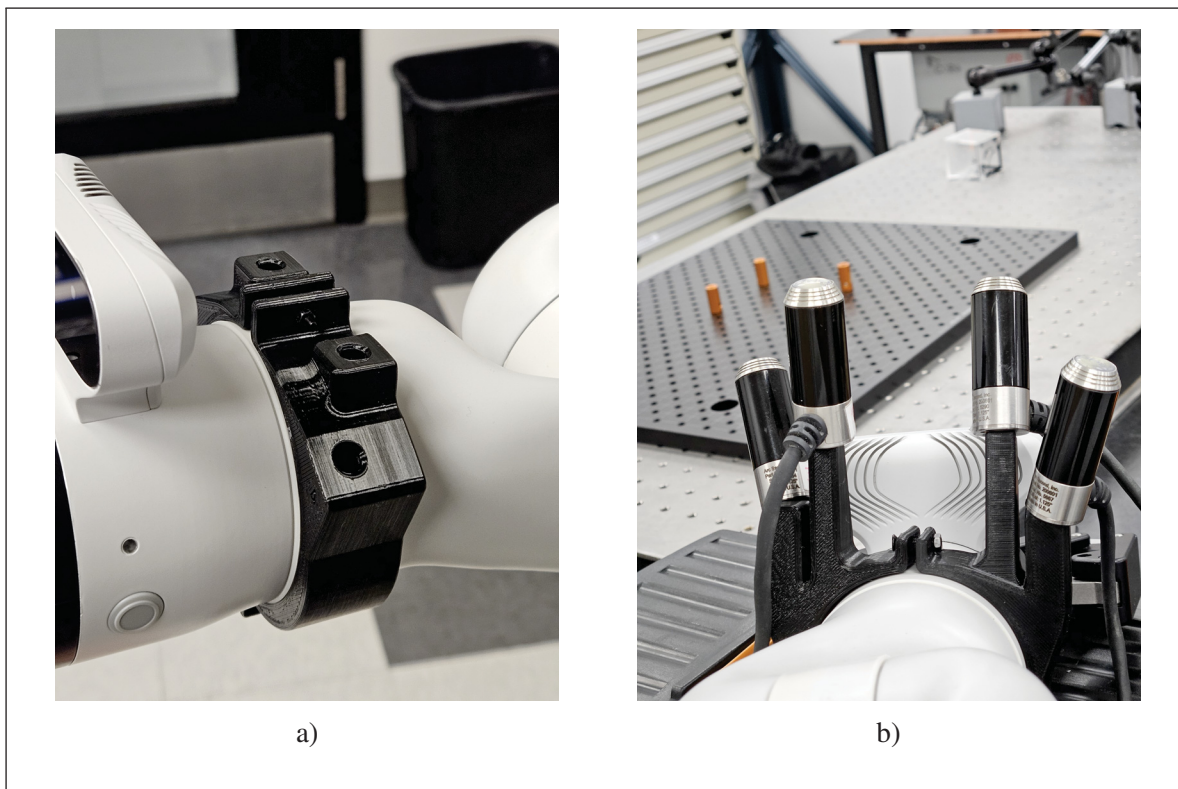


Figure 4.1 Configurations compactes étudiées pour la couronne : les deux capteurs d'une même paire sont orientés différemment afin d'élargir la couverture autour du poignet

Les géométries compactes de la figure 4.1 cherchent à couvrir davantage de directions autour du poignet en orientant différemment les capteurs d'une même paire. Cette stratégie améliore la

couverture individuelle, mais réduit le domaine de visibilité commun aux deux capteurs d'une paire. Or la mesure iGPS dépend de cette visibilité simultanée. Avec une plage d'incidence utile d'environ 45° , une inclinaison du poignet peut rendre une paire inexploitable.

La géométrie retenue conserve une orientation commune afin de préserver le domaine de visibilité partagé. Comme le montre la figure 4.2, les capteurs sont disposés en arc autour du poignet, avec des hauteurs décalées. Cette disposition privilégie la plage d'inclinaisons du poignet dans laquelle les deux capteurs restent visibles simultanément.



Figure 4.2 Couronne finale adaptée au poignet du bras Kinova

La couronne finale est imprimée en 3D et fixée autour du poignet par deux demi-anneaux assemblés par boulons. Une fois équipée de ses capteurs, la couronne a une masse de 637 g.

Le serrage sur une surface cylindrique et le montage démontable peuvent produire deux écarts de montage : un décalage axial et une rotation autour de l'axe du poignet. La figure 4.3 illustre

ces deux écarts, notés δx et $\delta \phi$. La calibration du montage de la couronne se limite à ces deux paramètres mécaniquement interprétables.

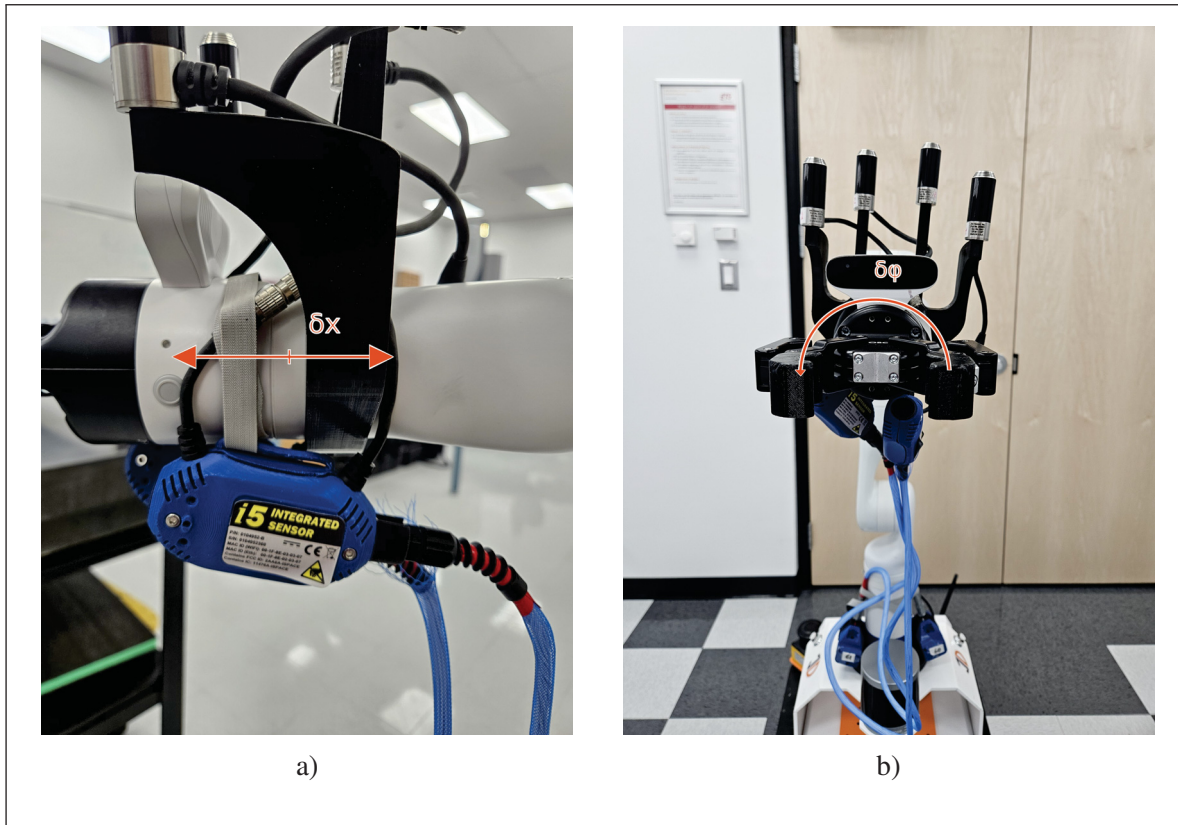


Figure 4.3 Écarts de montage retenus pour la calibration du montage de la couronne : (a) décalage axial δx , (b) roulis $\delta \phi$

4.3 Conception des supports de préhension

La couronne rend la mesure iGPS disponible au poignet. Pour les essais, une sonde *i5* est aussi saisie dans la pince afin de mesurer la position atteinte par l'effecteur. La transformation entre l'outil et cette sonde doit donc être calibrée. La position de la sonde dans la pince Robotiq doit aussi rester répétable.

Deux éléments imprimés en 3D sont ajoutés à la configuration expérimentale : des doigts adaptés à la préhension cylindrique et un support fixé sous la sonde *i5*.

La sonde *i5* est munie d'une interface de fixation située sous la sonde. Le support imprimé s'y fixe. Sa géométrie comprend deux cylindres coaxiaux : un cylindre serré par les doigts de la pince et un cylindre de plus grand diamètre formant un épaulement. Ces éléments sont présentés à la figure 4.4.

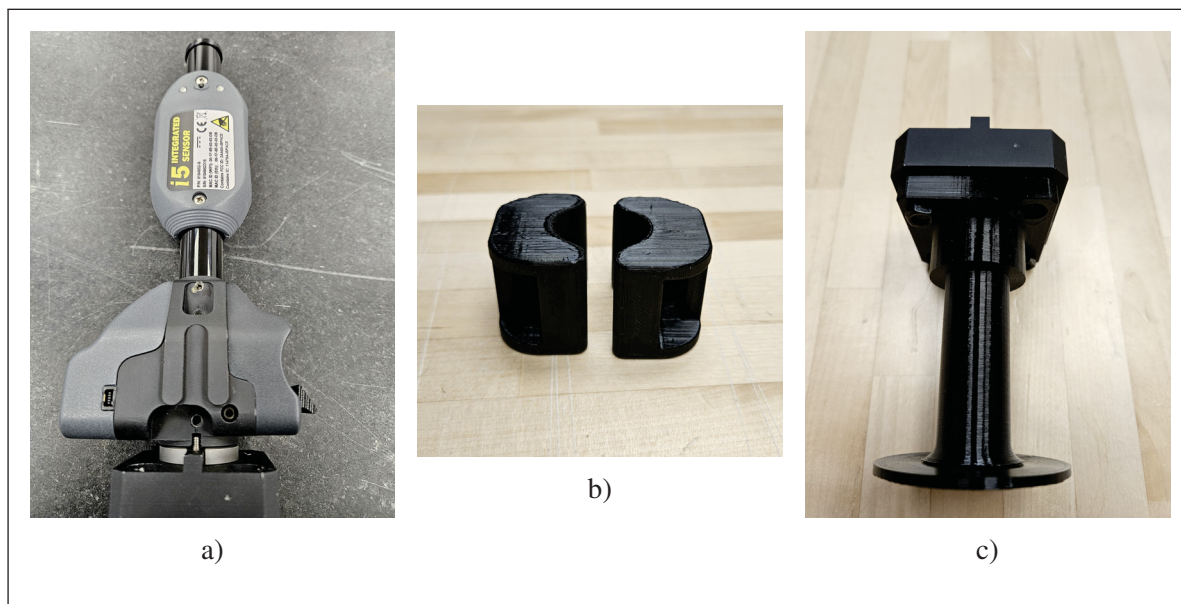


Figure 4.4 Sonde *i5* et éléments de préhension utilisés dans la pince Robotiq pendant les calibrations et les essais : (a) sonde *i5* et interface de fixation, (b) doigts de préhension imprimés, (c) support cylindrique imprimé

L'ensemble formé par la sonde *i5*, son interface de fixation et les pièces imprimées de préhension a une masse de 593 g.

À la fermeture de la pince, les empreintes demi-cylindriques des doigts centrent le cylindre saisi. Elles limitent les translations latérales et deux rotations, tandis que l'épaulement fixe la position axiale. Le serrage limite aussi la rotation autour de l'axe saisi. Les imprécisions géométriques de cette prise, notamment liées aux tolérances d'impression et à l'assemblage des pièces, sont prises en compte par la calibration terminale présentée à la section 4.7.

4.4 Intégration logicielle iGPS dans ROS 2

La couronne et les supports constituent la partie mécanique de l'instrumentation. Les poses iGPS doivent être publiées dans ROS 2 pour être utilisées par la chaîne logicielle du robot. L'infrastructure iGPS est décrite en méthodologie. Cette section précise son usage avec le bras Kinova.

L'application Surveyor du système iGPS fonctionne sur un ordinateur Windows séparé. Le NUC du robot l'interroge par une connexion TCP sur le réseau Wi-Fi. Dans les essais du chapitre, le nœud peut interroger jusqu'à cinq repères : la couronne, la sonde *i5* tenue dans la pince, le repère virtuel de la base du bras, la sonde *i5* manipulée par l'opérateur et la sonde *i6*. La couronne est interrogée systématiquement pour exploiter la mesure iGPS au poignet. Les autres repères sont ajoutés selon les besoins des protocoles : référence externe, repère terminal ou désignation de cibles.

La fréquence de publication de la couronne dépend du nombre de repères interrogés par cycle et de la commande Surveyor utilisée. La commande `GetFrameData` retourne l'ensemble des données disponibles pour le repère demandé, tandis que `GetFrameDataLite` retourne seulement sa pose. L'objectif est de rester proche de 40 Hz, soit la fréquence de fonctionnement recommandée dans l'application Surveyor. L'exemple Python du fournisseur sert seulement de point de comparaison. L'interface ROS 2 est écrite en C++. Elle utilise la commande allégée pour limiter les données retournées par Surveyor et reconstruit les réponses JSON successives, car Surveyor les transmet sans délimiteur explicite. La même reconstruction est ajoutée à l'exemple Python du fournisseur afin de comparer les deux interfaces dans des conditions équivalentes.

Le tableau 4.1 compare la fréquence de publication de la couronne lorsque le nombre de repères interrogés augmente. Dans l'interface ROS 2 C++, la commande allégée maintient une fréquence presque constante, entre 37,8 et 38,0 Hz, du cycle minimal au cycle incluant les cinq repères. Avec la commande complète, cette fréquence diminue lorsque davantage de repères sont interrogés. Le tableau II-1, placé en annexe, détaille la taille des données retournées par chaque commande et explique cet écart. Pour la suite du chapitre, l'interface C++ utilise la commande

allégée et n'interroge que les repères nécessaires afin de maintenir une fréquence proche de 40 Hz.

Tableau 4.1 Fréquence de publication de la couronne iGPS selon l'interface utilisée, la commande Surveyor et les repères iGPS interrogés. Les commandes *allégée* et *complète* correspondent respectivement à `GetFrameDataLite` et `GetFrameData`. Les symboles de la première colonne désignent les repères ajoutés cumulativement : couronne (c), sonde *i5* tenue dans la pince (p), référence de base du bras (b), sonde *i5* opérateur (o) et sonde *i6* (i6). Les valeurs sont les moyennes des essais répétés

Repères interrogés	Python allégé [Hz]	Python complet [Hz]	C++ allégé [Hz]	C++ complet [Hz]
c	36,5	28,7	38,0	38,0
c + p	34,2	23,8	37,8	35,8
c + p + b	33,6	22,4	37,8	35,3
c + p + b + o	31,8	19,2	37,8	31,0
c + p + b + o + i6	27,9	13,5	37,9	23,3

4.5 Repères, notations et organisation expérimentale

Les sections suivantes regroupent trois volets : la calibration de la couronne et de la chaîne terminale, la commande cartésienne et ses paramètres, puis l'évaluation de l'exactitude de positionnement et de la régularité du suivi de chemin.

Le tableau 4.2 fixe la nomenclature de la chaîne bras-iGPS utilisée dans la suite du mémoire.

Les transformations rigides sont notées T_a^b . L'indice désigne le repère dans lequel la pose est exprimée et l'exposant désigne le repère dont la pose est décrite. Par exemple, $T_{map}^{igps_sensor}$ représente la pose du repère `igps_sensor` exprimée dans `map`. La notation t_a^b désigne la composante de translation de la transformation T_a^b . Les notations $t_a^{b,x}$, $t_a^{b,y}$ et $t_a^{b,z}$ désignent ses composantes scalaires selon les axes du repère a .

La figure 4.5 situe les repères principaux et les deux sous-chaînes géométriques utiles à la campagne.

La figure 4.6 isole les cinq paramètres géométriques estimés dans le chapitre. Elle sépare les corrections associées au montage de la couronne des paramètres locaux de la chaîne terminale.

Tableau 4.2 Correspondance entre la nomenclature de la chaîne bras-iGPS utilisée dans la suite du mémoire et les repères ROS associés

Notation simplifiée	Notation ROS	Description
map	map	repère global de l'iGPS
arm_base	gen3_base_link	repère de base du bras
bracelet	gen3_bracelet_link	repère du poignet
igps_sensor	igps_sensor_link	repère iGPS de la couronne
ee	gen3_end_effector_link	repère de l'effecteur
gripper_base	gen3_robotiq_85_base_link	repère de base de la pince
tool	gen3_robotiq_85_tool_link	repère outil de la pince
i5	i5_link	repère de la sonde <i>i5</i> saisie

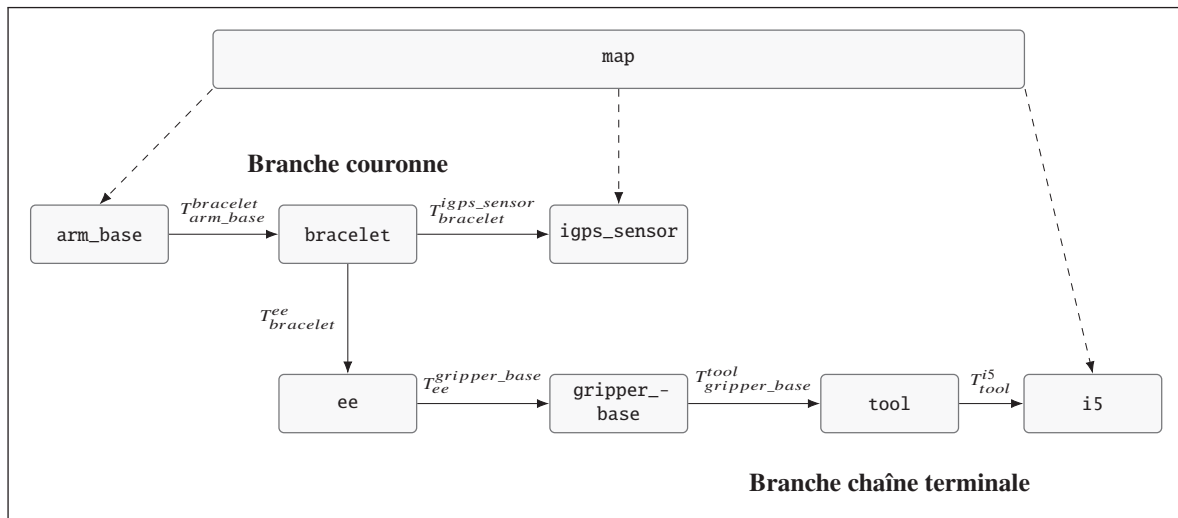


Figure 4.5 Repères principaux de la chaîne instrumentée. La branche supérieure relie la base du bras, le poignet et la couronne iGPS. La branche inférieure relie le poignet, la pince et la sonde *i5* saisie comme repère terminal. Les traits pleins représentent les transformations géométriques connues ou calibrées. Les traits pointillés indiquent les poses iGPS disponibles dans map

Dans les équations de calibration, un chapeau indique une grandeur reconstruite ou prédite. Une barre indique une mesure de référence.

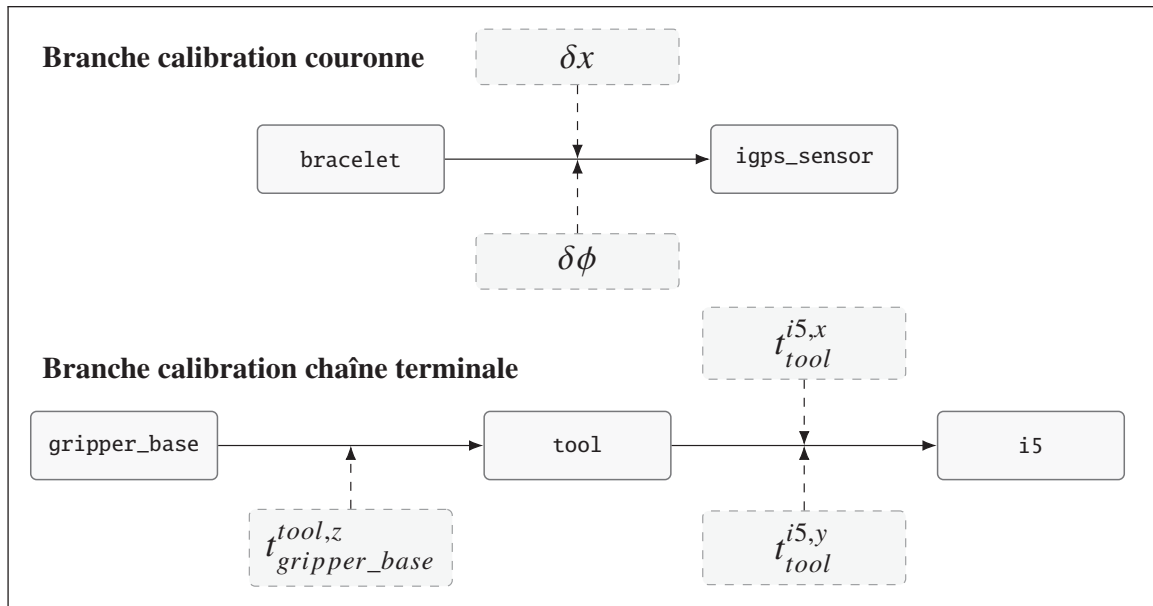


Figure 4.6 Paramètres géométriques calibrés dans le chapitre. Les corrections du poignet ($\delta x, \delta \phi$) sont présentées séparément des composantes de translation locales de la chaîne terminale ($t_{tool}^{i5,x}, t_{tool}^{i5,y}, t_{gripper_base}^{tool,z}$)

4.6 Calibration et validation du montage de la couronne

4.6.1 Formulation du problème

La couronne fournit la pose iGPS du repère `igps_sensor` dans `map`. Cette mesure situe la couronne, mais pas directement le poignet du bras. La transformation entre le repère `bracelet` et ce repère iGPS dépend du montage réel de la couronne. Le problème de calibration consiste à estimer les deux corrections géométriques qui rendent cette transformation cohérente avec le modèle géométrique du bras.

La pose $T_{map}^{igps_sensor}$ fournie par Surveyor repose sur une calibration iGPS préalable de l'assemblage de capteurs. Les positions nominales des capteurs sont d'abord saisies à partir du modèle CAO, puis affinées dans Surveyor. Cette étape définit le repère iGPS de la couronne utilisé comme entrée de la chaîne géométrique. La calibration présentée ici ne remplace pas

cette calibration interne au système iGPS. Elle estime seulement les corrections géométriques du montage de la couronne par rapport au poignet.

Pour rendre ces corrections observables, la mesure au poignet est ramenée à la base du bras par la cinématique directe. La pose reconstruite de `arm_base` est ensuite comparée à une référence indépendante.

La référence est établie avec la sonde *i6*, qui mesure directement dans `map` le repère physique de la base du bras. Elle définit un repère virtuel de base, indépendant de la couronne et de la cinématique du poignet. Un écart de montage de la couronne apparaît alors comme un biais sur la pose reconstruite. La relation de reconstruction est

$$T_{map}^{arm_base} = T_{map}^{igps_sensor} \left(T_{arm_base}^{igps_sensor} \right)^{-1}. \quad (4.1)$$

La correction du montage de la couronne modélise les deux écarts définis à la figure 4.3. Elle est paramétrée par

$$\theta = \begin{bmatrix} \delta x \\ \delta \phi \end{bmatrix}, \quad (4.2)$$

où δx est une correction de translation selon l'axe x du repère `igps_sensor`. Le paramètre $\delta \phi$ correspond à une correction de roulis autour de ce même axe.

La transformation nominale entre `bracelet` et `igps_sensor` est notée $T_{bracelet}^{igps_sensor, nom}$. La correction appliquée à cette transformation s'écrit :

$$T_{corr}(\theta) = \begin{bmatrix} R_x(\delta \phi) & t(\delta x) \\ 0 & 1 \end{bmatrix}, \quad (4.3)$$

où

$$t(\delta x) = \begin{bmatrix} \delta x & 0 & 0 \end{bmatrix}^T, \quad (4.4)$$

et

$$R_x(\delta\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\delta\phi) & -\sin(\delta\phi) \\ 0 & \sin(\delta\phi) & \cos(\delta\phi) \end{bmatrix}. \quad (4.5)$$

La transformation corrigée entre bracelet et igps_sensor est

$$T_{bracelet}^{igps_sensor, corr}(\theta) = T_{bracelet}^{igps_sensor, nom} T_{corr}(\theta). \quad (4.6)$$

La calibration est évaluée sur un ensemble de poses du poignet. Pour l'échantillon i , la relation de reconstruction de l'équation (4.1) est appliquée avec la transformation corrigée $T_{bracelet}^{igps_sensor, corr}(\theta)$. La pose prédite de la base du bras s'écrit :

$$\hat{T}_{map}^{arm_base}(i, \theta) = T_{map}^{igps_sensor}(i) \left(T_{arm_base}^{bracelet}(i) T_{bracelet}^{igps_sensor, corr}(\theta) \right)^{-1}. \quad (4.7)$$

Cette pose prédite est comparée à la pose du repère virtuel de base $\bar{T}_{map}^{arm_base}(i)$.

Les résidus de translation et de rotation sont définis par

$$e_t(i, \theta) = \hat{t}_{map}^{arm_base}(i, \theta) - \bar{t}_{map}^{arm_base}(i), \quad (4.8)$$

$$e_r(i, \theta) = \text{Log} \left(\left(\bar{R}_{map}^{arm_base}(i) \right)^{-1} \hat{R}_{map}^{arm_base}(i, \theta) \right), \quad (4.9)$$

et la fonction objectif normalisée est

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N \left(\frac{\|e_t(i, \theta)\|^2}{s_t^2} + \frac{\|e_r(i, \theta)\|^2}{s_r^2} \right), \quad (4.10)$$

où s_t et s_r normalisent les résidus de translation et de rotation dans le coût total. Les valeurs retenues dans la campagne sont $s_t = 10$ mm et $s_r = 5^\circ$ (0,0873 rad).

Le résidu est évalué sur la pose complète, mais l'optimisation ne corrige que les deux écarts de montage retenus pour la couronne. Les écarts qui ne peuvent pas être représentés par ces

deux paramètres restent dans le résidu. Comme la pose reconstruite de `arm_base` utilise aussi la cinématique directe du bras, l'optimisation peut absorber une partie des biais systématiques compatibles avec ces deux corrections. Cette compensation reste toutefois limitée au montage de la couronne. Elle ne modifie pas le modèle géométrique interne du bras.

4.6.2 Protocole de calibration et résolution numérique

Le protocole repose sur une série de poses faisant varier le roulis du poignet. Quatre poses de base, espacées dans le volume de travail compatible avec les contraintes de visibilité iGPS, sont exécutées chacune avec cinq décalages de roulis, compris entre -30° et 30° par pas de 15° . Ce choix fournit une discrétisation régulière de l'ensemble des poses retenues pour la campagne de calibration. Il augmente ainsi le nombre d'observations autour de poses géométriquement distinctes, tout en conservant une visibilité suffisante pour les capteurs de la couronne. La campagne produit vingt échantillons exploitables pour estimer les deux paramètres optimisés, δx et $\delta \phi$. Le problème est donc largement surdéterminé, ce qui contraint suffisamment l'optimisation.

Chaque échantillon combine trois informations synchronisées : $T_{map}^{igps_sensor}(i)$, $T_{arm_base}^{bracelet}(i)$ et la référence externe du repère `arm_base`, $\bar{T}_{map}^{arm_base}(i)$.

La résolution numérique minimise la fonction objectif de l'équation (4.10) par recherche déterministe dans l'espace $(\delta x, \delta \phi)$. L'espace de recherche est limité à deux paramètres bornés. Il peut donc être échantillonné directement, sans choisir de point initial ni calculer de gradient. Une exploration grossière est effectuée sur $\delta x \in [-50, 50]$ mm par pas de 5 mm et sur $\delta \phi \in [-90, 90]^\circ$ par pas de 2° .

Le meilleur candidat est ensuite raffiné localement par subdivisions successives des pas. Après chaque subdivision, le raffinement est poursuivi seulement si la diminution du coût normalisé dépasse 10^{-3} . Ce seuil correspond, avec $s_t = 10$ mm, à une amélioration équivalente à environ 0,3 mm sur l'échelle de normalisation translatrice, soit l'ordre de grandeur de l'incertitude dynamique indiquée pour le système iGPS dans la méthodologie.

La surface de coût obtenue présente un minimum net. Le minimum obtenu correspond à une correction légère, compatible avec un écart de montage de la couronne. Les coupes de la surface de coût, reportées en annexe à la figure II-1, montrent que le coût est surtout sensible au roulis pour les valeurs retenues de s_t et s_r .

4.6.3 Protocole de validation et résultats

La validation utilise cinq poses de base distinctes de celles employées pour la calibration. Elles sont choisies manuellement afin de s'écarter des poses de calibration et de limiter un biais de validation. Elles suivent ensuite le même balayage en roulis du poignet. L'ensemble de validation est évalué avant et après calibration.

Tableau 4.3 Comparaison des métriques avant/après sur l'ensemble d'optimisation de la couronne et sur l'ensemble de validation indépendant. Les écarts-types σ_t et σ_r sont calculés sur les normes d'erreur de translation et de rotation, échantillon par échantillon

Ensemble	État	N	J	RMS trans. [mm]	σ_t [mm]	RMS rot. [rad]	σ_r [rad]
Optimisation	Avant	20	19,55	43,82	2,02	0,0513	0,0021
Optimisation	Après	20	0,43	6,27	2,95	0,0157	0,0051
Validation	Avant	25	17,21	41,10	4,14	0,0494	0,0034
Validation	Après	25	0,52	6,99	2,99	0,0144	0,0048

Le tableau 4.3 compare les ensembles d'optimisation et de validation avant et après calibration. Les erreurs avant calibration sont proches pour les deux ensembles. Après calibration, l'erreur de translation passe d'environ 41 mm à 7 mm, et l'erreur de rotation d'environ 0,049 rad à 0,014 rad. Ces résultats montrent que la correction ne se limite pas aux poses d'optimisation et se retrouve aussi sur les poses de validation.

Sur cet ensemble de validation indépendant, la correction réduit l'erreur de reconstruction de la base du bras par rapport à la référence externe. Sa validité reste toutefois liée à l'état de montage de la couronne. Dès que la couronne est démontée, desserrée ou repositionnée, la calibration doit être reprise.

4.7 Calibration et validation de la chaîne terminale

4.7.1 Formulation géométrique

Une fois le lien entre la couronne et le poignet corrigé, la calibration terminale prolonge la chaîne jusqu'à la sonde *i5* tenue dans la pince. Elle estime uniquement la géométrie locale de cette prise. Cette séparation évite de reporter un écart résiduel du montage au poignet dans les paramètres terminaux.

Trois translations locales sont estimées : la longueur utile entre *gripper_base* et l'outil, ainsi que les décalages en *x* et en *y* entre l'outil et la sonde *i5*.

Une sonde *i5* saisie dans la pince sert à cette estimation. Aucune recherche numérique n'est nécessaire. Une fois $T_{bracelet}^{igps_sensor, corr}$ connue, la translation de la sonde *i5* exprimée dans *bracelet* suffit à déterminer les trois inconnues.

Le modèle géométrique exprime la pose de la sonde *i5* à partir de *bracelet* :

$$T_{bracelet}^{i5} = T_{bracelet}^{ee} T_{ee}^{gripper_base} T_{gripper_base}^{tool} T_{tool}^{i5}. \quad (4.11)$$

La mesure iGPS fournit une expression indépendante de cette même pose. Elle donne la pose relative de la sonde par rapport au repère iGPS de la couronne, en appliquant la même relation d'inversion de chaîne que l'équation (4.1) :

$$T_{igps_sensor}^{i5} = \left(T_{map}^{igps_sensor} \right)^{-1} T_{map}^{i5}. \quad (4.12)$$

La correction du montage de la couronne permet d'exprimer cette même pose dans *bracelet* :

$$T_{bracelet}^{i5} = T_{bracelet}^{igps_sensor, corr} T_{igps_sensor}^{i5}. \quad (4.13)$$

Le passage du modèle complet à l'identification des trois paramètres repose sur des relations géométriques connues. Les repères `ee` et `gripper_base` sont confondus, ce qui donne $T_{ee}^{gripper_base} = I$. La transformation entre `bracelet` et `ee` impose quant à elle une rotation fixe $R_{bracelet}^{ee} = R_x(\pi)$. Comme les paramètres cherchés décrivent uniquement la position de la sonde dans la chaîne terminale, seule la partie translation de $T_{bracelet}^{i5}$ est utilisée :

$$t_{bracelet}^{i5} = t_{bracelet}^{ee} + R_x(\pi) \begin{bmatrix} t_{tool}^{i5,x} \\ t_{tool}^{i5,y} \\ t_{gripper_base}^{tool,z} \end{bmatrix}, \quad (4.14)$$

où $t_{bracelet}^{ee}$ provient de la transformation connue entre `bracelet` et `ee`.

En développant la rotation $R_x(\pi)$, avec $t_{bracelet}^{ee,x} = t_{bracelet}^{ee,y} = 0$, les composantes de $t_{bracelet}^{i5}$ donnent :

$$\begin{aligned} t_{tool}^{i5,x} &= t_{bracelet}^{i5,x} \\ t_{tool}^{i5,y} &= -t_{bracelet}^{i5,y} \\ t_{gripper_base}^{tool,z} &= t_{bracelet}^{ee,z} - t_{bracelet}^{i5,z}. \end{aligned} \quad (4.15)$$

Ces trois relations déterminent les paramètres de la chaîne terminale à partir de la position mesurée de la sonde *i5*.

4.7.2 Valeurs retenues et validation

La validation utilise cinq nouveaux placements de la sonde *i5* dans la pince. La principale incertitude est la répétabilité du serrage. Le protocole vérifie que les paramètres calibrés restent cohérents après remise en place de la sonde.

À partir d'un candidat $(t_{tool}^{i5,x}, t_{tool}^{i5,y}, t_{gripper_base}^{tool,z})$, on reconstruit une transformation $T_{bracelet}^{i5,cand}$, puis la pose prédite de la sonde *i5* dans le repère global :

$$\hat{T}_{map}^{i5} = T_{map}^{igps_sensor} \left(T_{bracelet}^{igps_sensor, corr} \right)^{-1} T_{bracelet}^{i5,cand}. \quad (4.16)$$

Cette prédiction est comparée à $\bar{T}_{map}^{i5}$, la mesure iGPS de la sonde $i5$ utilisée comme référence. L'erreur de position vaut :

$$e_{pos} = \|\hat{t}_{map}^{i5} - \bar{t}_{map}^{i5}\|_2. \quad (4.17)$$

Chaque nouveau placement permet aussi de réestimer les trois paramètres à partir des relations de l'équation (4.15).

Tableau 4.4 Valeurs calibrées de la chaîne terminale, réestimations et erreur de position après cinq remises en place de la sonde $i5$

Quantité	Calibré	Moyenne ($N = 5$)	Écart-type
$t_{tool}^{i5,x}$ [mm]	-0,32	-0,44	0,09
$t_{tool}^{i5,y}$ [mm]	56,22	56,23	0,16
$t_{gripper_base}^{tool,z}$ [mm]	160,85	160,90	0,11
e_{pos} [mm]	—	0,24	0,10

Le tableau 4.4 rassemble les valeurs calibrées, les réestimations obtenues sur cinq nouveaux placements de la sonde $i5$ et l'erreur de position associée. La dispersion des trois paramètres réestimés caractérise la répétabilité du serrage après remise en place de la sonde.

L'erreur moyenne de position reste inférieure au millimètre. Elle est du même ordre de grandeur que l'incertitude de mesure du système iGPS et ne révèle pas de biais géométrique dominant dans la chaîne terminale. La chaîne calibrée s'étend ainsi jusqu'à la position de la sonde $i5$ saisie. Cette calibration permet donc d'utiliser la position iGPS de la sonde $i5$ saisie comme mesure externe dans les protocoles d'évaluation.

4.8 Commande cartésienne en vitesse et paramètres retenus

Les calibrations précédentes rendent la mesure iGPS exploitable comme retour de pose pour le bras. Les essais comparent cette source au retour interne Kinova en conservant une même commande cartésienne en vitesse. Cette loi de commande est utilisée dans les deux protocoles.

Dans le protocole d'exactitude de positionnement, elle sert au suivi des points de passage évalués après stabilisation. Dans le protocole de régularité, elle sert au suivi d'une suite de poses cartésiennes définissant un chemin de référence. Les gains et le calcul des vitesses articulaires restent identiques dans les deux cas.

Cette commande forme une erreur de pose à six coordonnées dans le repère de base du bras. Les trois premières composantes décrivent l'écart de position. Les trois suivantes décrivent l'écart d'orientation sous forme axe-angle. Selon le mode évalué, la pose courante est fournie par la chaîne iGPS ou par le retour de pose Kinova. Pour une position désirée p_d et une position courante p , cette erreur et la vitesse cartésienne désirée sont définies par

$$e_c = \begin{bmatrix} p_d - p \\ \theta u \end{bmatrix}_{\text{arm_base}}, \quad v_c = K_p e_c + K_d \dot{e}_c + K_i \int e_c dt. \quad (4.18)$$

Dans cette expression, e_c est l'erreur cartésienne. Elle regroupe l'écart de position $p_d - p$ et l'erreur d'orientation θu . Les symboles p_d et p désignent les positions désirée et courante. Le terme θ est l'angle d'erreur, et u l'axe unitaire associé. Les termes $\dot{e}_c$ et $\int e_c dt$ sont la dérivée et l'intégrale temporelle de e_c . Les matrices K_p , K_d et K_i regroupent les gains proportionnels, dérivés et intégraux. Ces gains sont appliqués coordonnée par coordonnée.

La loi est écrite sous forme continue et implantée comme une commande périodique à 100 Hz. À chaque période, la dérivée est estimée par différence finie et l'intégrale est mise à jour par accumulation de l'erreur. La sortie v_c est la vitesse cartésienne désirée dans `arm_base`. Elle sert d'entrée à la conversion différentielle vers les vitesses articulaires.

À chaque itération, les états articulaires définissent la configuration courante du modèle du Gen3 chargé depuis l'URDF du fournisseur. Pinocchio calcule alors la jacobienne de l'effecteur pour cette configuration (Carpentier *et al.*, 2019). La conversion vers l'espace articulaire s'effectue ensuite au niveau des vitesses, avec la pseudo-inverse amortie de cette jacobienne :

$$\dot{q} = J_\lambda^+ v_c, \quad J_\lambda^+ = J^\top (J J^\top + \lambda^2 I)^{-1}. \quad (4.19)$$

Le terme λ régularise l'inversion et limite les pics de vitesse articulaire près des singularités ou des configurations mal conditionnées, avec un léger compromis sur le suivi instantané. La même valeur est utilisée dans tous les protocoles et tous les modes de rétroaction. Elle fixe donc le comportement numérique de la conversion, sans constituer un paramètre de comparaison.

Dans cette formulation, l'erreur doit être exprimée dans `arm_base`, car la jacobienne du modèle Kinova est définie dans ce repère. Une erreur définie dans `map` doit donc être convertie avant le calcul de v_c .

Ce choix conserve une loi de commande identique pour tous les modes évalués. Une formulation dans le repère de l'outil n'est pas retenue, car elle modifierait simultanément la loi de commande et le rôle des sources de rétroaction.

Les gains sont ajustés manuellement par essais successifs, sans optimisation automatique ni analyse formelle de stabilité du contrôleur. Dans la même configuration expérimentale que le protocole d'exactitude de positionnement décrit à la sous-section 4.9.1, les jeux de gains candidats sont soumis à une séquence dédiée de 18 mouvements cartésiens élémentaires. Ils sont comparés qualitativement en recherchant une convergence rapide vers la consigne, avec un dépassement limité, sans saturation prolongée des commandes et avec une faible erreur résiduelle de position mesurée par l'iGPS. Les valeurs utilisées dans les essais sont regroupées dans le tableau 4.5 : gains PID cartésiens, coefficient de régularisation de la jacobienne et limite de vitesse articulaire. Le terme intégral est fixé à zéro. Les essais exploratoires avec un gain intégral non nul ne réduisent pas l'erreur externe. Les gains très faibles restent sous l'échelle de mesure. Les gains plus élevés accumulent une erreur pendant l'approche et le maintien. Ils dégradent alors la convergence finale. Ce réglage évite un comportement dépendant du gain intégral. La comparaison reste centrée sur la source de pose utilisée par la commande.

Ce réglage fixe les paramètres de commande. Il n'est pas interprété comme une évaluation de performance. L'exactitude de positionnement et la régularité du suivi sont évaluées par les protocoles présentés dans la suite du chapitre.

Tableau 4.5 Paramètres du contrôleur cartésien retenus pour l'ensemble des essais du chapitre. Les gains sont regroupés en un triplet commun pour les trois translations et un second pour les trois rotations. La régularisation et la limite de vitesse articulaire sont maintenues fixes pendant la campagne de réglage

Type	Élément	Paramètre	Valeur	Unité
Gain PID	translation x, y, z	K_p	0,90	s^{-1}
Gain PID	translation x, y, z	K_d	0,03	[–]
Gain PID	translation x, y, z	K_i	0,00	s^{-2}
Gain PID	rotation roulis, tangage, lacet	K_p	0,40	s^{-1}
Gain PID	rotation roulis, tangage, lacet	K_d	0,06	[–]
Gain PID	rotation roulis, tangage, lacet	K_i	0,00	s^{-2}
Réglage commun	régularisation	λ	0,05	[–]
Réglage commun	vitesse articulaire maximale	$\dot{q}_{max}$	2,5	rad/s

4.9 Évaluation expérimentale

L'évaluation porte sur deux axes : l'apport de la rétroaction iGPS à l'exactitude de positionnement et la régularité du suivi de chemin. La figure 4.7 présente les montages expérimentaux associés à ces protocoles.

Les deux protocoles sont réalisés avec l'instrumentation complète en place. La couronne équipée de ses capteurs et l'ensemble formé par la sonde *i5* et les pièces de préhension ajoutent respectivement 637 g et 593 g au bras. Ces masses se répartissent entre le poignet et la pince. Elles restent toutefois significatives par rapport à la charge utile continue du Kinova Gen3, limitée à 2 kg sur toute l'enveloppe de travail et à 4 kg à mi-portée (Kinova Robotics, 2024).

4.9.1 Exactitude de positionnement

Le protocole d'exactitude de positionnement est réalisé avec le bras embarqué sur la base mobile à l'arrêt. La sonde *i5* reste saisie dans la pince Robotiq. Chaque essai comprend trois parcours des 9 positions cibles de la sonde *i5*. Chaque consigne est maintenue 15 s avant l'échantillonnage. Sur l'ensemble des essais et des modes retenus, la norme moyenne de la vitesse cartésienne commandée reste sous le seuil de 1 mm s^{-1} pendant la dernière seconde de maintien. Les 27

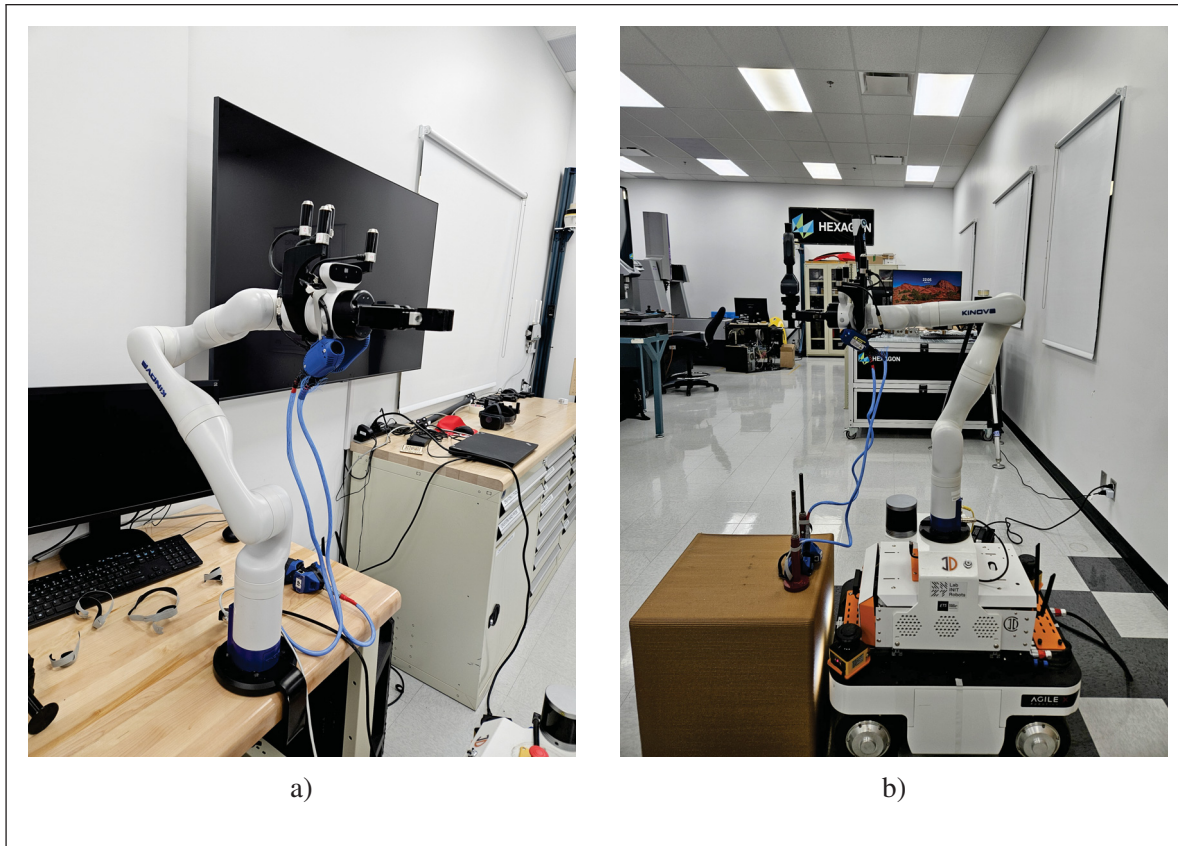


Figure 4.7 Montages expérimentaux utilisés pour l'évaluation : (a) bras fixé sur table, utilisé pour la comparaison du suivi de chemin, (b) bras embarqué sur la base mobile, utilisé pour le protocole d'exactitude de positionnement et pour cette comparaison

échantillons de chaque essai sont donc considérés comme prélevés en régime stabilisé. Avec trois essais par mode, chaque mode regroupe 81 erreurs de position 3D mesurées par l'iGPS.

Les modes évalués combinent deux choix : la source de rétroaction de la commande cartésienne et l'ancrage global de arm_base . Cet ancrage désigne la transformation $T_{\text{map}}^{\text{arm_base}}$ qui situe la base du bras dans map. Il sert à convertir les consignes définies dans map vers arm_base , où la loi de commande est calculée.

Les modes iGPS et TF/Kinova avec ancrage externe forment la comparaison principale. Ils utilisent la même référence externe pour $T_{\text{map}}^{\text{arm_base}}$, figée avant l'essai comme lors de la calibration de la couronne. Cet ancrage ne dépend pas de la mesure iGPS au poignet. La comparaison isole

donc la source de retour de pose. Le premier mode utilise le retour issu de la chaîne iGPS. Le second utilise le retour de pose Kinova, reconstruit par la cinématique directe à partir des états articulaires.

Les trois autres modes explorent l'effet de l'ancrage. L'ancrage dynamique met à jour $T_{map}^{arm_base}$ pendant l'essai. L'ancrage initial capture cette transformation avant la manipulation à partir de la mesure iGPS au poignet et de la cinématique directe du bras, puis la conserve. Ces modes décrivent les cas où la tâche ne dispose pas d'une nouvelle mesure externe de arm_base au moment de la manipulation.

Dans tous les modes, l'évaluation reste externe. Pour l'échantillon k , l'erreur de position vaut

$$e_k = p_{k,i5}^{mes} - p_{k,i5}^{ref}. \quad (4.20)$$

Dans cette expression, $p_{k,i5}^{mes}$ et $p_{k,i5}^{ref}$ sont respectivement la position mesurée et la position cible de la sonde $i5$, exprimées dans map . Le terme e_k est l'erreur de position 3D associée.

La métrique principale est la RMSE 3D :

$$RMSE_{3D} = \sqrt{\frac{1}{N} \sum_{k=1}^N \|e_k\|_2^2}. \quad (4.21)$$

Ici, N désigne le nombre d'erreurs de position regroupées pour le mode et l'essai considérés.

Le tableau 4.6 synthétise les résultats par mode. Pour chaque essai, les erreurs sont regroupées sur l'ensemble des cibles. Les valeurs reportées sont les médianes des métriques sur les trois essais. La RMSE reste la métrique principale. La médiane et l'écart type décrivent la distribution globale des erreurs 3D, et la répétabilité décrit la dispersion radiale autour de chaque groupe de cibles.

Tableau 4.6 Synthèse des erreurs de positionnement de la sonde *i5* en régime stabilisé. Les trois essais comprennent 27 échantillons retenus chacun. Pour une série d'essais, les valeurs numériques correspondent à la médiane inter-essais des métriques calculées séparément par essai. L'écart type est calculé, pour chaque essai, sur les normes $\|e_k\|_2$. La répétabilité correspond à la dispersion radiale moyenne autour de chaque groupe de cibles

Mode	Erreur de la sonde <i>i5</i> mesurée par l'iGPS [mm]			
	RMSE	Médiane	Écart type	Répétabilité
iGPS, ancrage externe	1,74	1,73	0,16	0,08
TF/Kinova, ancrage externe	7,69	6,73	2,23	0,28
TF/Kinova, ancrage initial	9,11	8,56	2,75	0,14
iGPS, ancrage dynamique	5,84	4,21	2,27	0,57
iGPS, ancrage initial	1,73	1,69	0,16	0,07

Le tableau met en évidence l'effet du retour de pose. Pour les deux modes avec ancrage externe, la RMSE passe d'environ 7,7 mm avec TF/Kinova à 1,7 mm avec iGPS. La même tendance apparaît pour les modes avec ancrage initial : la RMSE reste d'environ 1,7 mm avec iGPS, contre 9,1 mm avec TF/Kinova. Le mode iGPS avec ancrage dynamique se situe entre ces deux groupes, avec une RMSE d'environ 5,8 mm.

La figure 4.8 complète cette synthèse par une lecture par cible. Chaque groupe *C1* à *C9* correspond à une position cible de la séquence de points de passage. Les groupes restent séparés afin de montrer si l'erreur dépend de la position de travail dans la zone locale de manipulation.

Les modes iGPS avec ancrage externe et ancrage initial conservent une erreur faible et presque constante sur les neuf cibles. Les modes TF/Kinova et le mode iGPS avec ancrage dynamique présentent une variation plus marquée selon la cible.

Pour la manipulation locale avec base arrêtée dans la zone iGPS, le mode iGPS avec ancrage initial est retenu parmi les modes évalués.

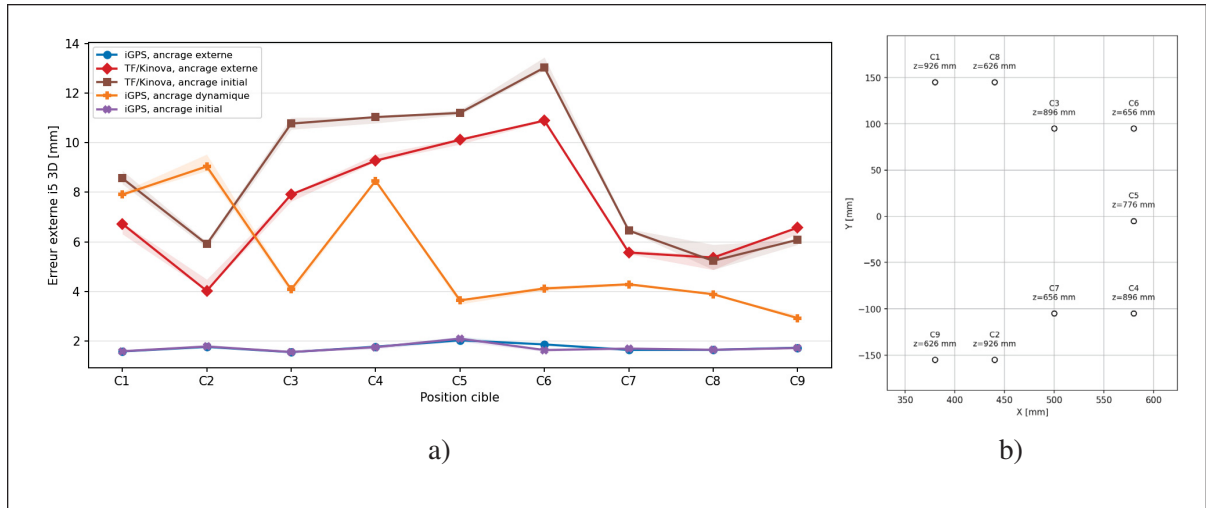


Figure 4.8 Lecture par cible de l'erreur externe 3D de la sonde *i5*. Dans le panneau (a), chaque point correspond, pour un groupe de cibles, à la médiane inter-essais de l'erreur médiane. La bande indique l'étendue min-max inter-essais. Le panneau (b) situe les cibles dans le plan *XY* de *arm_base* et indique leur coordonnée *Z*

4.9.2 Régularité du suivi de chemin

Ce protocole compare le suivi d'un même chemin de référence sur deux montages : le bras fixé sur table et le bras monté sur la base mobile à l'arrêt. Dans les deux cas, la sonde *i5* tenue dans la pince suit un chemin en forme de huit dans des plans *YZ* de *arm_base*, décalés en *X*. La suite de poses cibles est identique dans le repère du bras pour les deux montages. Le protocole est répété cinq fois pour chacun d'eux.

Les transitions entre plans et les temps de stabilisation sont exclus du calcul. Les métriques portent seulement sur les segments actifs et répétables du mouvement.

L'analyse utilise le résidu cartésien par rapport à la position de référence échantillonnée. En notant $p_{\text{mes}}[n]$ la position mesurée de la sonde *i5* et $p_{\text{ref}}[n]$ la position de référence au même instant, ce résidu est

$$e[n] = p_{\text{mes}}[n] - p_{\text{ref}}[n]. \quad (4.22)$$

La comparaison fréquentielle porte sur la norme de ce résidu. Elle utilise une densité spectrale de puissance (DSP) estimée par Welch (Welch, 1967).

La borne inférieure de la bande est déterminée à partir de la DSP groupée des essais sur les montages fixe et mobile. Cette courbe est lissée, puis analysée en échelle log-log entre 0,2 et 6 Hz. La fréquence retenue correspond au point de la courbe le plus éloigné de la droite reliant les points situés à ces deux fréquences. Elle vaut $f_c = 0,71$ Hz.

La borne supérieure est fixée à 10 Hz. Elle respecte la fréquence de mesure de la sonde *i5*. Sur les segments actifs, la fréquence moyenne de mesure vaut environ 35,6 Hz pour le montage fixe et 31,0 Hz pour le montage mobile. Cette borne demeure sous la limite de Nyquist du cas le plus lent.

La bande fréquentielle retenue s'étend donc de f_c à 10 Hz. Elle écarte les variations lentes du suivi et conserve les oscillations plus rapides du résidu.

Pour chaque essai et chaque plan *YZ*, la DSP de Welch est intégrée dans cette bande. Le ratio mobile/fixe associé est

$$R_X = \frac{\int_{f_c}^{10\text{ Hz}} S_{\text{mobile},X}(f) df}{\int_{f_c}^{10\text{ Hz}} S_{\text{fixe},X}(f) df}, \quad (4.23)$$

où $S_{\text{mobile},X}$ et $S_{\text{fixe},X}$ sont les DSP de Welch de la norme du résidu pour un même essai et un même plan.

La figure 4.9 montre que les DSP des deux montages restent très proches dans la bande retenue. Les ratios de l'équation (4.23) restent près de l'unité pour les quatre plans *YZ*, entre 0,96 et 1,01. Le montage mobile n'augmente pas la puissance spectrale du résidu dans cette bande par rapport au montage fixe. Les figures II-2 à II-4 détaillent le choix de f_c et la sensibilité aux limites de bande. Une analyse complémentaire du suivi, qui quantifie l'erreur spatiale par la distance géométrique au chemin de référence, est présentée à la figure II-5.

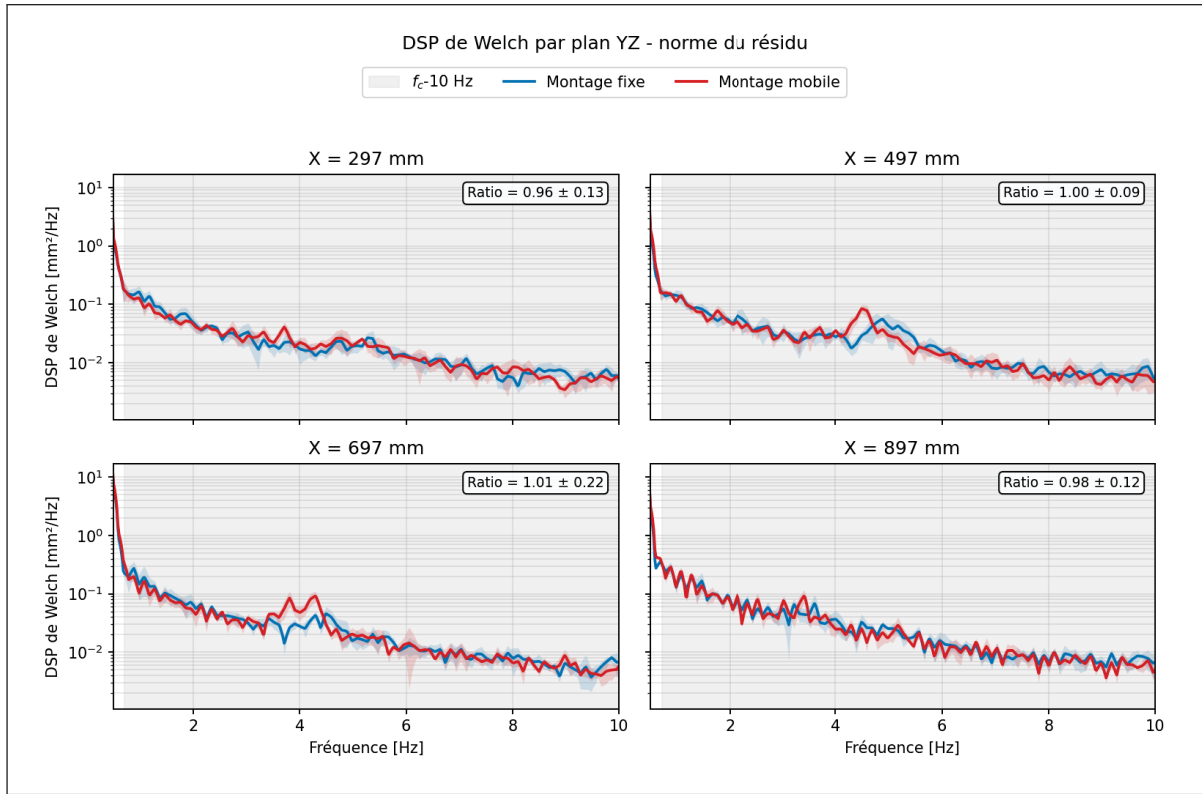


Figure 4.9 DSP de Welch de la norme du résidu cartésien pour les montages fixe et mobile. Chaque courbe correspond à la moyenne inter-essais des DSP pour un plan YZ, et les bandes colorées indiquent l'écart type inter-essais. La zone grisée indique la bande f_c-10 Hz intégrée dans l'équation (4.23)

Cette comparaison inclut aussi les différences mécaniques visibles à la figure 4.7. Le bras mobile repose sur une base potentiellement moins rigide, mais il est boulonné à sa structure. Le bras fixe repose sur une table plus rigide, mais sa base est seulement serrée par une bride frontale.

4.10 Synthèse

Ce chapitre rend la mesure iGPS portée par le bras exploitable dans le système robotique. La calibration du montage de la couronne fixe la transformation entre son repère iGPS et le poignet du bras. La calibration terminale prolonge cette chaîne jusqu'à la sonde *i5* saisie. Le système dispose ainsi d'une mesure cartésienne externe pour la manipulation et la localisation.

Les essais d'exactitude de positionnement quantifient l'apport de la mesure iGPS en rétroaction cartésienne sur la base mobile. Avec un ancrage externe commun, la rétroaction iGPS présente une erreur nettement plus faible que le retour de pose Kinova. Le mode iGPS avec ancrage initial conserve ce niveau d'exactitude lorsque la base mobile est arrêtée avant la manipulation. Ce résultat appuie la configuration reprise au chapitre suivant, où la manipulation est réalisée après l'arrêt du robot dans la zone iGPS. Les essais de suivi de chemin évaluent l'influence du montage mobile sur la régularité du suivi. Avec les métriques retenues, aucune hausse mesurable de la puissance spectrale du résidu n'est observée sur le montage mobile dans les conditions d'essai.

Ces résultats restent liés aux conditions expérimentales du chapitre. La calibration du montage de la couronne dépend de son état de montage et doit être reprise après démontage, desserrage ou repositionnement. La calibration terminale dépend surtout de la répétabilité du placement de la sonde *i5* dans la pince. Le protocole d'exactitude de positionnement est mené avec le bras embarqué sur la base mobile, dans une zone locale de manipulation. L'analyse du suivi de chemin reste liée aux métriques choisies et aux plans *YZ* retenus, qui fixent l'extension du bras pendant le mouvement.

La chaîne obtenue est reprise dans le chapitre suivant. Dans la zone iGPS, la mesure portée par la couronne permet de reconstruire la pose globale de la base mobile et de l'utiliser comme source de localisation à la place de la localisation SLAM. Cette continuité rend possible la transition vers l'iGPS avant la manipulation locale de la démonstration intégrée.

CHAPITRE 5

NAVIGATION INTÉGRÉE AVEC IGPS

5.1 Objectif et portée

Ce chapitre intègre l'instrumentation iGPS du chapitre précédent à la navigation mobile du RanGen. La couronne fixée au poignet fournit une mesure externe. La chaîne géométrique du robot transforme cette mesure en pose globale de la base mobile. L'enjeu central du chapitre est d'en faire une source globale compatible avec Nav2 après une transition contrôlée depuis la localisation autonome.

Le scénario étudié relie une zone hors couverture iGPS au laboratoire de métrologie. Il s'appuie sur la précision de l'iGPS dans la zone instrumentée et conserve une localisation autonome issue des capteurs embarqués hors couverture.

La difficulté tient à l'initialisation hors couverture iGPS du repère de carte utilisé par Nav2. La transformation entre ce repère et le repère iGPS n'est donc pas connue au démarrage. Elle doit être estimée dans une zone où les deux sources de pose sont disponibles.

La contribution du chapitre concerne la transition entre ces deux sources. Elle remplace la source globale utilisée en navigation, tout en conservant une carte publiée et un arbre TF compatibles avec Nav2.

Le chapitre commence par la chaîne autonome retenue et l'architecture de transition. Il détaille ensuite la calibration nécessaire à la transition vers l'iGPS, puis évalue l'effet de cette source aux poses cibles Nav2. Il se termine par une démonstration intégrée combinant récupération d'outil, transition vers l'iGPS et manipulation locale au poste opérateur.

5.2 Chaîne autonome retenue pour la base mobile

La localisation autonome fournit une pose utilisable hors couverture iGPS. Elle reste active en parallèle du mode iGPS pour conserver une source indépendante. Elle sert de repli en cas d'indisponibilité de l'iGPS, par perte de signal, coupure de connexion ou occlusion prolongée.

Les contraintes de l'application visée limitent la comparaison aux méthodes de localisation sur carte. Une carte préexistante est nécessaire, ce qui oriente l'évaluation vers SLAM Toolbox en mode *localization* et AMCL. La comparaison repose sur des essais réalisés avec le robot réel, durant lesquels les commandes envoyées à la base mobile et les données des capteurs sont enregistrées. Ces enregistrements sont ensuite rejoués pour chaque méthode évaluée, afin de soumettre les différentes stratégies aux mêmes conditions expérimentales.

Seul le choix final de la chaîne autonome est repris dans ce chapitre. La comparaison détaillée des méthodes et des résultats est présentée à l'annexe III.

La chaîne retenue sépare l'estimation locale de la localisation globale. Un filtre de Kalman étendu (EKF), configuré avec `robot_localization`, fusionne l'odométrie des roues et l'IMU pour estimer la pose locale dans le repère `odom`. SLAM Toolbox fonctionne en mode *localization* sur une carte préexistante. Il utilise cette estimation locale comme prédiction, aligne les scans du LiDAR 2D sur la carte et fournit la pose globale de la base dans le repère `map`. Dans la campagne résumée en annexe, cette configuration obtient les plus faibles RMSE de translation et d'orientation, soit 0,075 m et 2,943°.

Cette chaîne constitue donc le mode autonome retenu pour l'architecture de localisation développée dans ce chapitre.

5.3 Architecture de transition SLAM-iGPS

L'architecture de transition isole Nav2 du changement de source globale. Elle utilise la pose globale et la carte fournies par SLAM Toolbox, ainsi que la pose de la base reconstruite à partir

de la mesure iGPS. Elle maintient les mêmes interfaces ROS exposées à la navigation : la carte publiée sur `/map` et l'arbre TF contenant T_{map}^{odom} .

Le nœud `map_mux` assure cette continuité. Il sélectionne la source globale active, publie T_{map}^{odom} et fournit la carte utilisée par la navigation.

Dans cette architecture, `map_mux` complète l'arbre TF avec :

$$T_{map}^{odom} = T_{map}^{base} \left(T_{odom}^{base} \right)^{-1}. \quad (5.1)$$

La pose T_{map}^{base} représente la pose de la base dans le repère global actif, tandis que T_{odom}^{base} est l'estimation locale fournie par l'EKF. Avant d'être transmise à `map_mux`, la mesure iGPS de la couronne est convertie en pose de la base par la chaîne géométrique calibrée au chapitre 4.

La continuité de cette interface exige que la pose globale sélectionnée et la carte publiée soient exprimées dans un même repère global. Les différentes façons d'assurer cette cohérence sont comparées à la section 5.5.1.

5.4 Calibration de la transformation entre les repères

Le repère `map` est initialisé dans le laboratoire `InitRobots`, où aucune mesure iGPS n'est disponible. La transformation entre ce repère et le repère iGPS n'est donc pas connue au démarrage. Elle est estimée dans la zone de transition, à l'entrée du laboratoire de métrologie, où les poses SLAM et iGPS sont disponibles simultanément.

La calibration consiste à estimer la transformation plane $T_S^I \in SE(2)$, qui exprime dans le repère SLAM les positions mesurées dans le repère iGPS. Elle comprend deux translations dans le plan et une rotation autour de z . Pour contraindre ces trois degrés de liberté, le robot suit un trajet en L pendant lequel `map_mux` collecte des positions correspondantes dans les deux repères. Ces repères et les positions correspondantes sont illustrés à la figure 5.1.

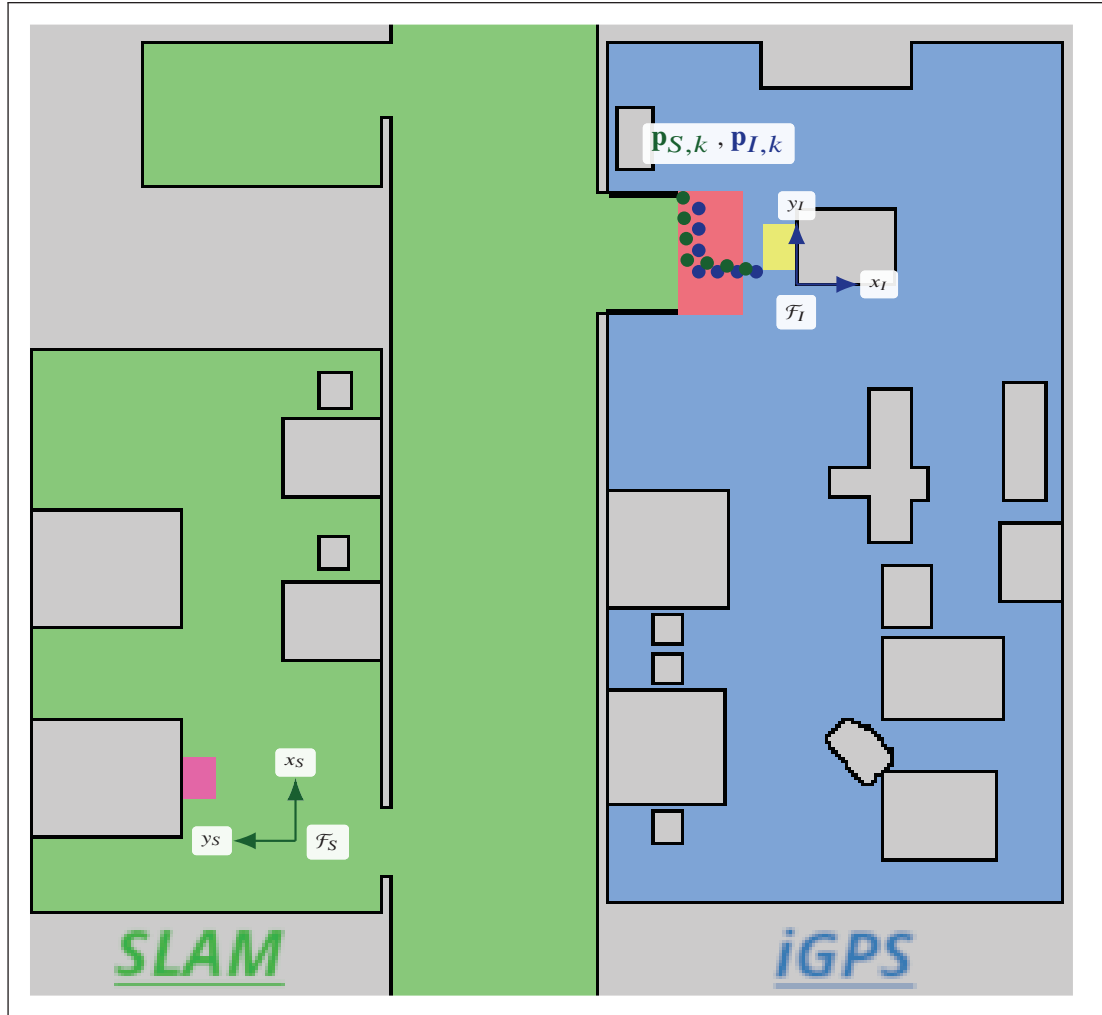


Figure 5.1 Repères de la carte SLAM $\mathcal{F}_S$ et de l'iGPS $\mathcal{F}_I$. Les deux ensembles de points correspondent aux mêmes poses collectées, mais exprimées dans chacun des deux repères

On note $\mathbf{p}_{I,k}$ la position k mesurée dans le repère iGPS et $\mathbf{p}_{S,k}$ la position correspondante dans le repère SLAM. Les paramètres $\hat{R}$ et $\hat{\mathbf{t}}$ de la transformation sont estimés par :

$$\hat{R}, \hat{\mathbf{t}} = \underset{R \in \text{SO}(2), \mathbf{t} \in \mathbb{R}^2}{\text{argmin}} \sum_{k=1}^N \left\| R \mathbf{p}_{I,k} + \mathbf{t} - \mathbf{p}_{S,k} \right\|^2. \quad (5.2)$$

Le problème est résolu par une estimation rigide en moindres carrés. Les deux ensembles de points sont d'abord centrés autour de leur moyenne. La matrice de corrélation est ensuite décomposée par SVD :

$$H = \sum_{k=1}^N (\mathbf{p}_{I,k} - \bar{\mathbf{p}}_I) (\mathbf{p}_{S,k} - \bar{\mathbf{p}}_S)^T = U \Sigma V^T. \quad (5.3)$$

La rotation et la translation sont alors données par :

$$\hat{R} = V U^T, \quad \hat{\mathbf{t}} = \bar{\mathbf{p}}_S - \hat{R} \bar{\mathbf{p}}_I. \quad (5.4)$$

Si $\det(\hat{R}) < 0$, la transformation obtenue ne correspond pas à une rotation physique, mais à une réflexion. Elle aligne les points en conservant les distances, mais inverse l'orientation du plan, comme un effet miroir. Elle ne peut donc pas représenter le changement de repère entre les repères SLAM et iGPS, qui doit rester une rotation propre. On inverse alors le signe de la deuxième colonne de V , associée à la plus petite valeur singulière, afin de supprimer cette réflexion tout en modifiant le moins possible l'ajustement. La rotation recalculée vérifie alors $\det(\hat{R}) = +1$.

Les résidus associés à cette solution sont calculés pour chaque correspondance :

$$e_k = \|\hat{R} \mathbf{p}_{I,k} + \hat{\mathbf{t}} - \mathbf{p}_{S,k}\|. \quad (5.5)$$

Les résidus associés à une première estimation servent à écarter les correspondances aberrantes. La transformation est ensuite recalculée sur l'ensemble filtré. La composante verticale n'est pas estimée par cette optimisation plane. Sa valeur est fixée à partir de la différence sur z entre la pose de base issue de SLAM et celle reconstruite à partir de l'iGPS.

On appelle erreur de calibration l'écart en translation et en lacet entre la transformation estimée $\hat{T}_S^I$ et la transformation réelle entre les deux repères. Cette erreur résulte de deux contributions. La première provient des erreurs des sources de localisation utilisées pour former les correspondances : la pose SLAM peut notamment avoir accumulé une erreur avant ou pendant la collecte, tandis que la mesure iGPS comporte sa propre incertitude. La seconde est liée à l'estimation de la transformation à partir d'un ensemble fini de correspondances, notamment à la géométrie du trajet, au filtrage et à l'ajustement. Les résidus après filtrage reflètent l'effet combiné de ces contributions et servent d'indicateur de la qualité de la calibration.

5.5 Bascule vers la localisation iGPS

Une fois la transformation entre les repères SLAM et iGPS calibrée, il reste à définir son usage dans Nav2 pendant le changement de source globale. La bascule doit exploiter cette transformation calibrée tout en conservant les interfaces de navigation pendant le passage vers l'iGPS.

5.5.1 Choix de la stratégie de transition

La transformation calibrée entre les repères SLAM et iGPS peut être utilisée de deux façons. En mode *fuse*, le repère de la carte SLAM reste le repère global actif et la pose iGPS y est exprimée. En mode *switch*, le repère iGPS devient le repère global actif et la carte SLAM publiée est réexprimée dans ce repère. La figure 5.2 compare le cheminement de la pose et de la carte dans les deux cas.

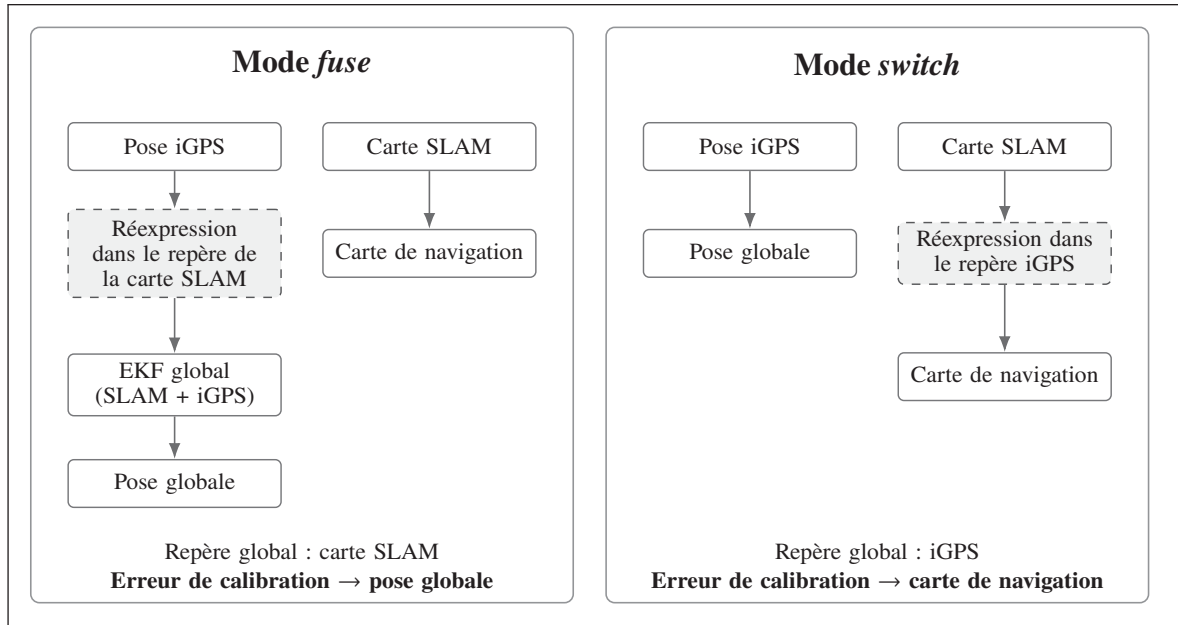


Figure 5.2 Comparaison des deux utilisations de la transformation entre les repères SLAM et iGPS. Le contour en pointillés indique l'opération qui applique la transformation calibrée entre les repères.

Le choix dépend de l'endroit où se reporte l'erreur de calibration. En mode *fuse*, elle affecte la pose iGPS exprimée dans la carte, puis la transformation globale T_{map}^{odom} . En mode *switch*, elle se reporte sur la carte publiée sur `/map`. La figure met ainsi en évidence que la même erreur de calibration ne dégrade pas la même sortie de l'architecture.

Le mode *fuse* conserve une interprétation unique des coordonnées de carte, ce qui permet de définir les poses cibles une seule fois. Il reste toutefois moins adapté à la zone de métrologie, car cette erreur limite directement l'exactitude de la localisation globale.

Le mode *switch* est retenu, car il place l'erreur de calibration sur la carte publiée plutôt que sur la pose globale. L'iGPS peut ainsi être utilisé directement comme référence dans le laboratoire de métrologie. Ce compromis reste acceptable, car la carte réexprimée n'est utilisée que dans cette zone, pour des trajets courts où la navigation repose surtout sur les observations locales.

5.5.2 Réexpression de la carte en mode switch

Dans ce mode, `map_mux` publie une copie de la grille d'occupation SLAM dans le repère iGPS. La translation entre les repères est prise en compte par la position de l'origine de la grille. En revanche, Nav2 n'applique pas le lacet de cette origine aux cellules de la grille. Les données de la grille doivent donc être recalculées après rotation.

Les cellules libres, occupées et inconnues sont alors reconstruites dans la grille résultante. La nouvelle carte est publiée avec une origine translatée et une orientation identifiée.

SLAM Toolbox conserve la carte originale publiée sur `/map_lidar`. Cette séparation évite de perturber la localisation SLAM en arrière-plan. Elle permet aussi de revenir vers la localisation autonome sans reconstruire son état à partir d'une carte modifiée.

La commande de bascule vers l'iGPS est autorisée uniquement lorsque la transformation entre le repère de la carte SLAM et le repère iGPS est disponible. Lors de la bascule, `map_mux` republie immédiatement `/map` et déclenche le nettoyage de la carte de coût globale.

5.6 Erreur de pose aux cibles Nav2

Le protocole quantifie l'erreur de pose des deux sources globales aux cibles Nav2, par rapport à une référence externe. Le robot répète cinq fois une série de seize poses. À chaque pose atteinte, les estimations des deux sources et la référence externe sont enregistrées après immobilisation de la base. L'analyse porte donc sur l'erreur de pose une fois le robot arrêté. Cette erreur est interprétée comme une mesure d'exactitude, tandis que la dispersion entre répétitions est rapportée séparément. Le protocole ne caractérise pas l'erreur pendant le déplacement.

La référence externe fournit une pose plane de la base mobile. Elle est reconstruite à partir de deux points mesurés avec la sonde *i6*. Ces points de contact sont situés sur le dessus des LiDAR 2D. Leur géométrie fournit la position et le lacet de `base_link`.

La même référence sert à comparer deux sources globales : source SLAM et source iGPS. L'odométrie locale reste identique dans les deux reconstructions.

Pour une source globale $s \in \{\text{SLAM}, \text{iGPS}\}$ et une pose cible k , l'erreur de position dans le plan est

$$e_{xy,k}^s = \begin{bmatrix} x_k^s - x_k^{\text{ref}} \\ y_k^s - y_k^{\text{ref}} \end{bmatrix}, \quad d_{xy,k}^s = \|e_{xy,k}^s\|_2. \quad (5.6)$$

L'erreur de lacet est l'écart angulaire absolu :

$$d_{\theta,k}^s = \left| \text{atan2}\left(\sin(\theta_k^s - \theta_k^{\text{ref}}), \cos(\theta_k^s - \theta_k^{\text{ref}})\right) \right|. \quad (5.7)$$

Les RMSE sont calculées sur les $N = 16$ poses cibles d'un essai :

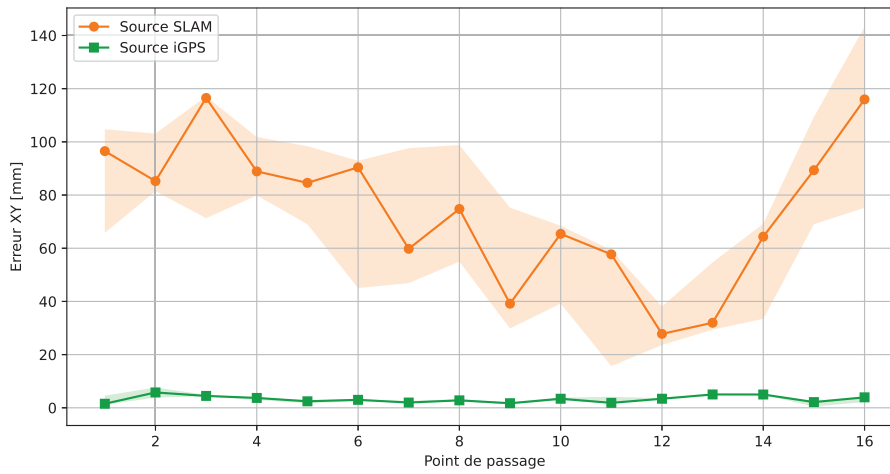
$$\text{RMSE}_{xy}^s = \sqrt{\frac{1}{N} \sum_{k=1}^N \left(d_{xy,k}^s\right)^2}, \quad \text{RMSE}_{\theta}^s = \sqrt{\frac{1}{N} \sum_{k=1}^N \left(d_{\theta,k}^s\right)^2}. \quad (5.8)$$

Tableau 5.1 Synthèse des erreurs de pose aux cibles Nav2 pour les deux sources globales. Chaque essai comprend 16 poses cibles retenues. La RMSE, la médiane et l'écart type sont calculés séparément pour chaque essai sur ces 16 poses. Les valeurs numériques correspondent à la médiane inter-essais de ces métriques

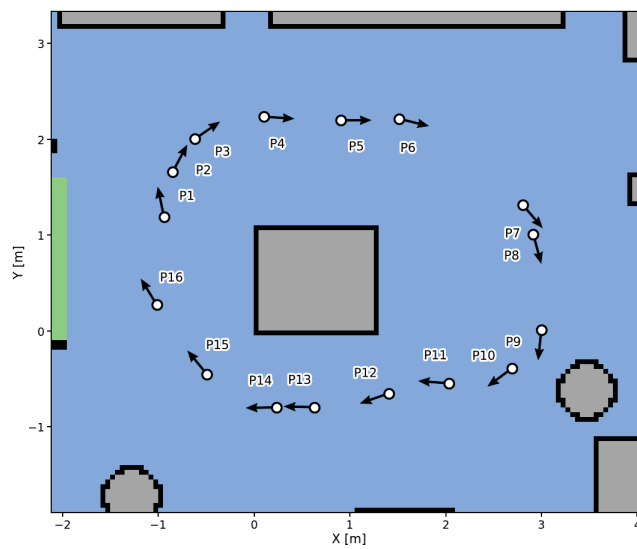
Source globale	Erreur XY [mm]			Erreur absolue de lacet [°]		
	RMSE	Médiane	Écart type	RMSE	Médiane	Écart type
Source SLAM	78,0	73,3	30,1	2,17	1,52	1,09
Source iGPS	3,5	3,3	1,5	0,24	0,12	0,13

Le tableau 5.1 quantifie l'effet du remplacement de source globale. La RMSE médiane en position passe de 78,0 mm avec la source SLAM à 3,5 mm avec la source iGPS. L'erreur de lacet suit la même tendance, de 2,17° à 0,24°. L'écart type XY de la source iGPS est de 1,5 mm, soit environ la moitié de sa médiane. Rapportée à la médiane, cette dispersion n'est pas négligeable, mais elle reste millimétrique et nettement inférieure aux 30,1 mm de la source SLAM.

La figure 5.3 localise ces écarts sur le trajet. Avec SLAM, l'erreur dépend de la zone du trajet. Cette variabilité est cohérente avec l'alignement des scans LiDAR sur la carte. Sa qualité dépend



a)



b)

Figure 5.3 Erreur XY aux poses cibles Nav2 pour les sources globales SLAM et iGPS. Le panneau (a) compare, pour chaque pose cible, la médiane sur les cinq essais. Les zones ombrées indiquent l'intervalle interquartile. La pose évaluée est reconstruite en combinant la source globale sélectionnée, SLAM ou iGPS, avec la même odométrie locale, selon la méthode décrite à la section 5.3. Le panneau (b) situe ces poses sur la carte. Les flèches indiquent l'orientation de la base et de la couronne

de la géométrie observée, de l'orientation de la base et des écarts locaux entre la salle réelle et la

carte. Les erreurs iGPS restent, à l'inverse, à l'échelle de quelques millimètres pour les seize poses, malgré les changements d'orientation de la couronne.

La référence externe ne constitue pas une vérité terrain 3D complète. Elle dépend de la mesure iGPS de la sonde *i6*, de la calibration des deux points de contact et de la répétabilité du contact manuel. Cette limite doit donc être prise en compte dans l'interprétation des résultats iGPS, car les erreurs mesurées sont elles-mêmes de l'ordre de quelques millimètres.

5.7 Démonstration intégrée

La démonstration intégrée présente l'enchaînement des fonctions du système dans une tâche de service à l'opérateur. Sur demande de l'opérateur, le robot récupère un outil, revient au poste d'assemblage, puis exécute une manipulation locale après activation de la source globale issue de l'iGPS. Une vidéo complémentaire montre le déroulement complet de la mission. Cette section en retient les étapes structurantes.

L'opérateur conserve la supervision de la mission. Depuis une interface, il sélectionne l'outil requis, désigne le bac de réception comme cible de dépôt et déclenche le départ du robot. Il suit ensuite la progression et valide les étapes importantes de calibration et de manipulation. La figure 5.4 présente cette interface.

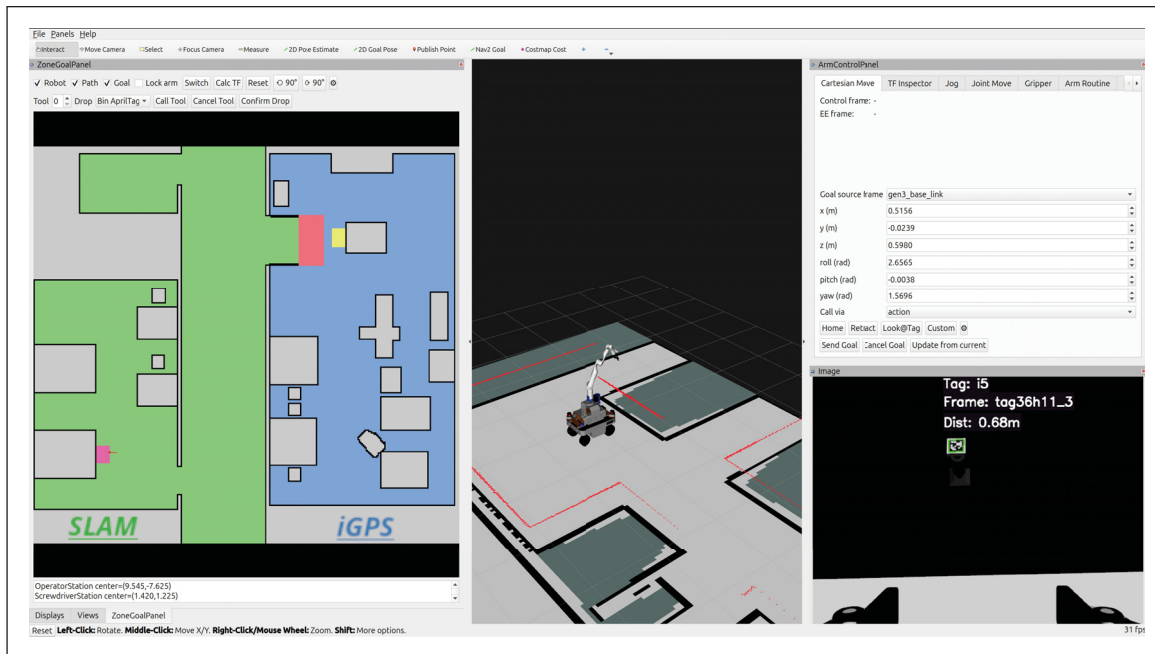


Figure 5.4 Interface RViz utilisée pour lancer et superviser la démonstration intégrée depuis le poste opérateur. Le panneau de gauche regroupe la carte, les zones du scénario et la commande de départ du robot. La scène 3D centrale affiche la plateforme mobile, le bras et les repères utilisés pendant la calibration SLAM-iGPS. Le panneau de droite sert au suivi des séquences du bras, notamment pour le dépôt de l'outil et la manipulation des vis

Pour récupérer l'outil, le robot se déplace du poste opérateur, dans le laboratoire de métrologie, vers l'emplacement de prise de l'outil situé dans InitRobots. Ce trajet utilise la chaîne de navigation autonome. Nav2 guide la base jusqu'à une pose cible définie devant cet emplacement. Après immobilisation, le bras utilise une détection AprilTag locale pour ajuster la pose de prise.

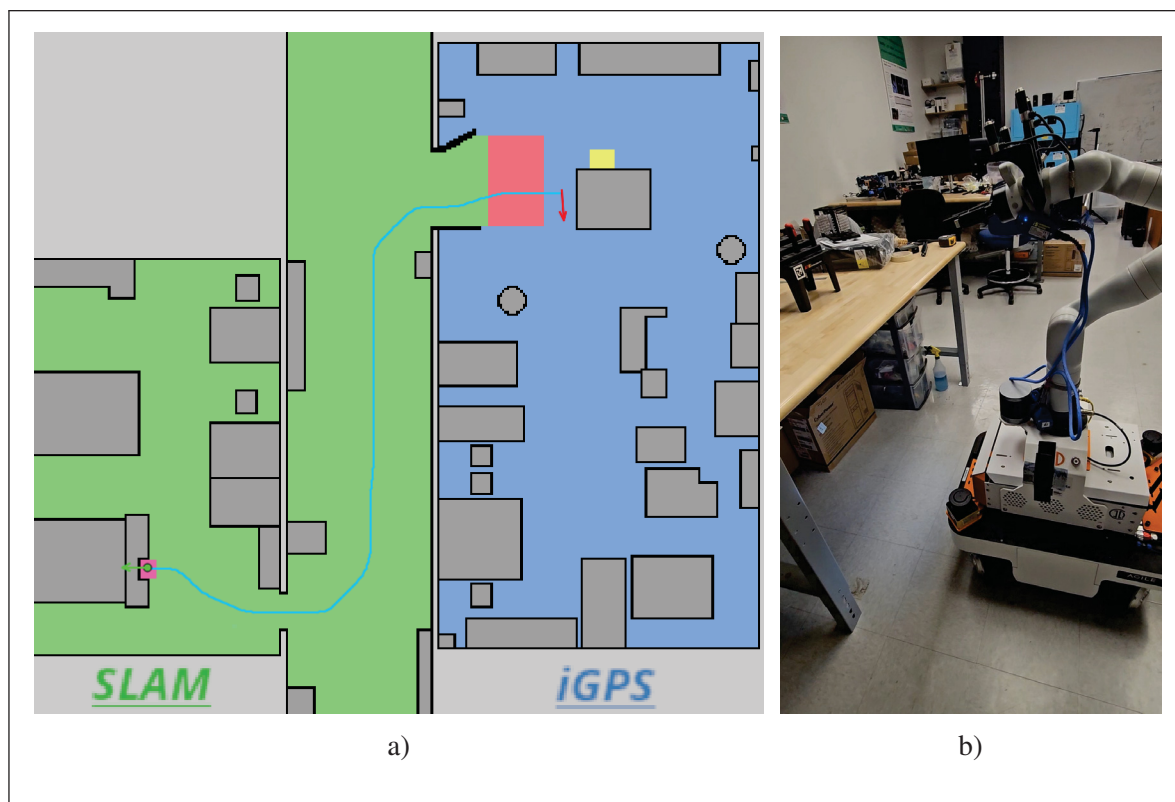


Figure 5.5 Récupération de l'outil : (a) départ du poste opérateur et navigation autonome vers l'emplacement de prise de l'outil, (b) arrivée de la base devant cet emplacement

Après la prise de l'outil, le robot revient vers le poste opérateur sous localisation autonome. La base rejoint d'abord une pose de transition. À cette pose, les mesures iGPS deviennent disponibles pour calibrer le lien entre les repères. Après cette calibration, la source globale bascule vers l'iGPS, puis la navigation reprend jusqu'au poste opérateur avec cette source de localisation.

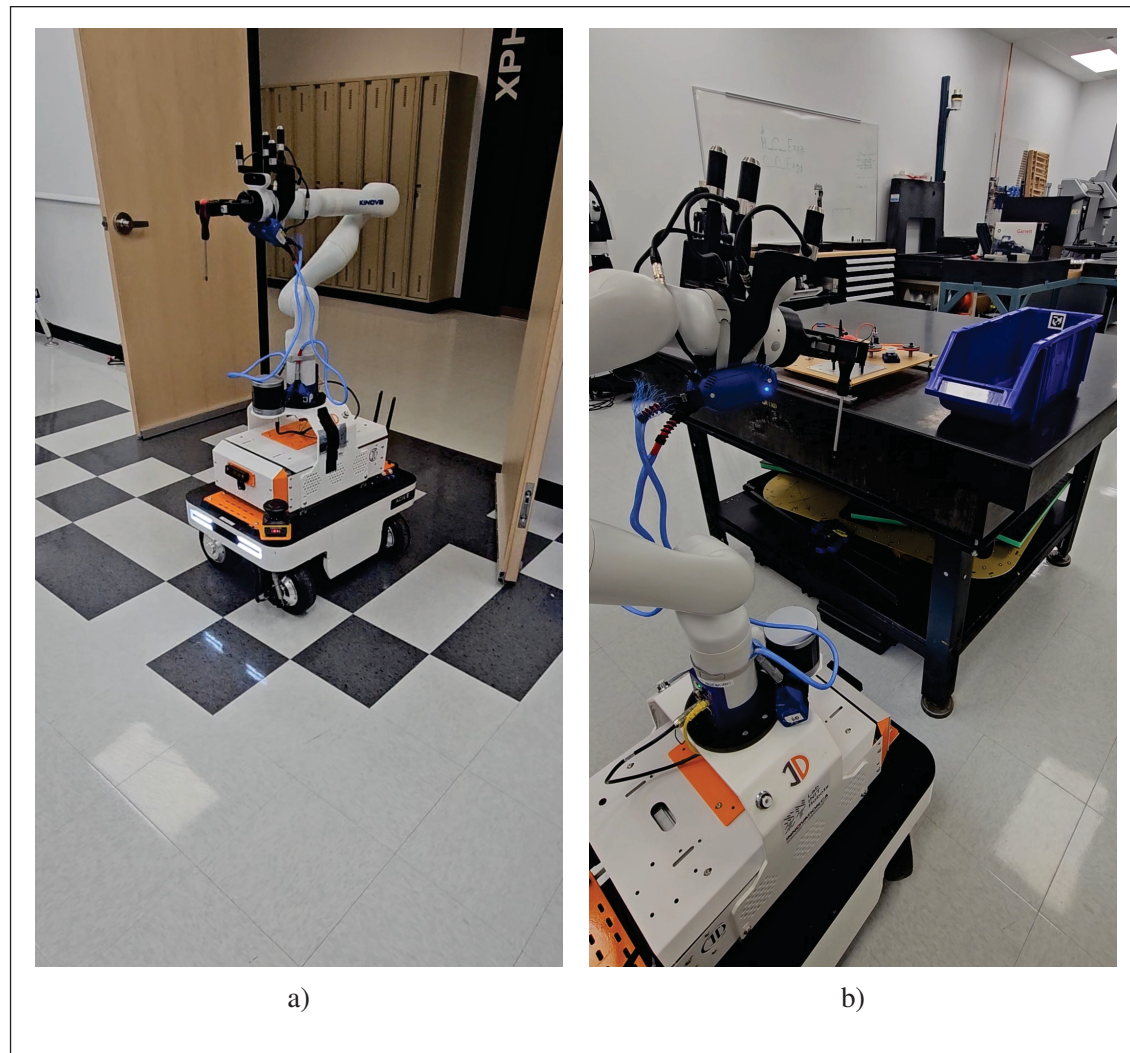


Figure 5.6 Retour vers le poste opérateur avec changement de source globale : (a) pose de transition pour la calibration, (b) retour au poste opérateur après bascule vers l'iGPS

La pose de la base est ensuite figée pour fournir une référence stable à la manipulation locale. Cet ancrage reprend le principe évalué au chapitre précédent. Le robot dépose l'outil transporté dans le bac de réception identifié par AprilTag. La pose estimée du bac est conservée pour la suite de la mission. Si le repère du bac n'est pas disponible, la sonde *i5* peut servir de mode de repli pour désigner la cible de dépôt.

Après la livraison, l'opérateur utilise l'outil au poste d'assemblage et prépare la manipulation des vis à retirer. Pour chaque vis, l'opérateur place la sonde *i5* sur le point de prise et enregistre la position mesurée dans l'interface. Une fois les trois cibles enregistrées, le bras saisit les vis un par un et les dépose dans le bac de réception.

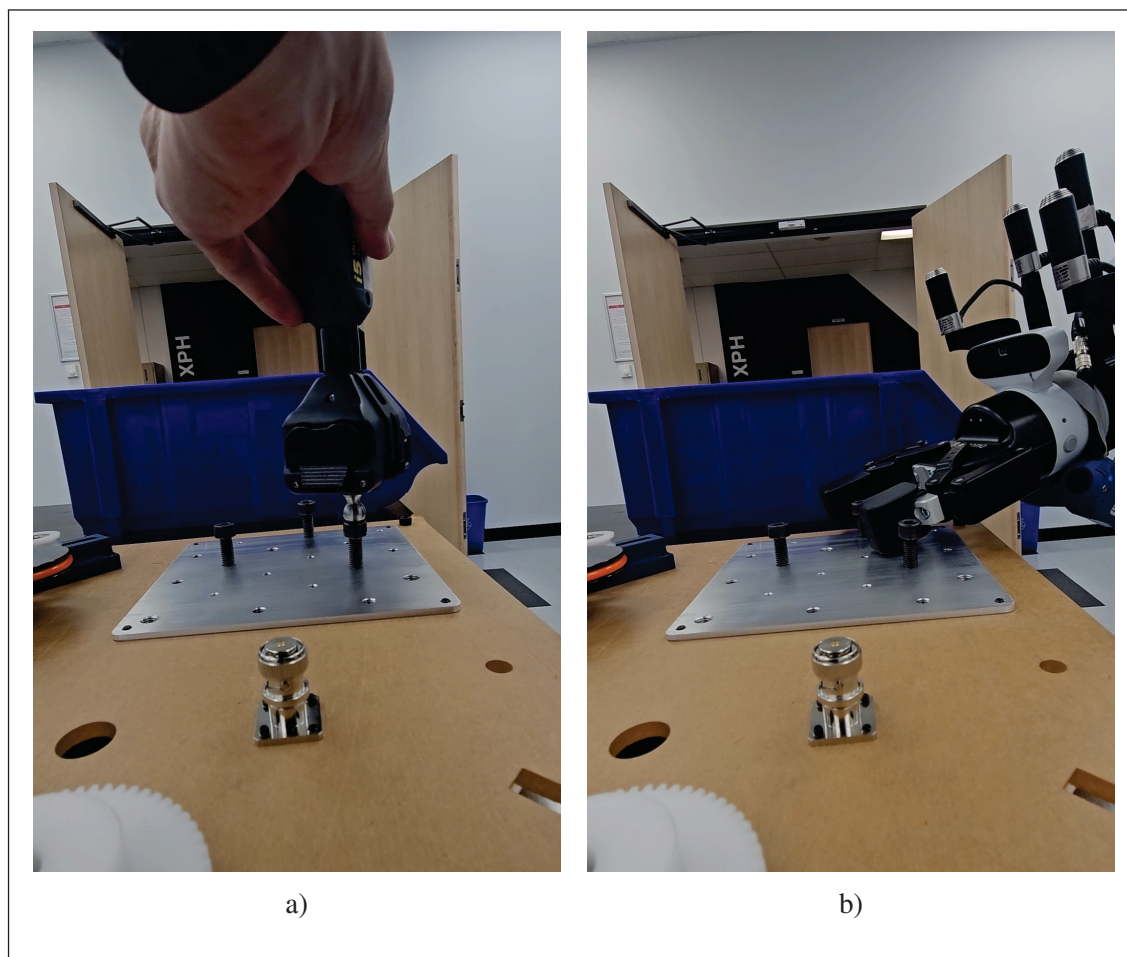


Figure 5.7 Manipulation locale au poste opérateur : (a) désignation d'une cible de prise avec la sonde *i5*, (b) saisie d'une vis avec le bras

Cette séquence constitue donc une démonstration fonctionnelle de la transition. Elle ne constitue pas une nouvelle mesure de performance. Elle montre plutôt que la localisation autonome, la bascule vers l'iGPS et la manipulation locale peuvent être mobilisées dans une tâche continue supervisée par l'opérateur.

CONCLUSION ET RECOMMANDATIONS

Ce mémoire présente l'étude de la transition de localisation du RanGen, un robot mobile manipulateur instrumenté par un système iGPS. Les travaux réalisés ont conduit à la conception et à l'évaluation d'une architecture qui conserve une localisation autonome hors couverture iGPS, bascule vers une source globale issue de l'iGPS dans la zone instrumentée, puis maintient des repères cohérents pour la navigation finale et la manipulation locale.

Le mémoire répond à la problématique initiale par une architecture de transition validée expérimentalement. Elle organise la simulation, la localisation autonome, la mesure iGPS, les calibrations géométriques, la navigation et la manipulation autour de repères, d'interfaces et de protocoles cohérents.

6.1 Bilan des contributions

La méthodologie a établi la base commune des chapitres de validation. Elle a défini la plateforme RanGen, les sources de localisation, l'infrastructure iGPS, l'architecture ROS 2 et les protocoles d'évaluation. Cette organisation a permis de traiter trois axes techniques complémentaires : la cohérence *sim-to-real*, l'instrumentation iGPS du bras manipulateur, puis la transition vers une localisation assistée par iGPS pour la base mobile.

Le jumeau numérique a servi de base de développement et de validation pour les fonctions nécessaires à la transition. Il ne cherche pas à reproduire exactement le robot réel, mais à offrir des interfaces de commande communes et un comportement observable comparable. Les essais ciblés ont fait ressortir plusieurs contraintes de commande de la base mobile : la garde de braquage, la représentation minimale de la translation latérale et la limitation de vitesse selon le secteur de braquage. Le processus automatisé d'évaluation mis en place peut servir à vérifier de nouvelles règles de commande sans redéfinir toute la campagne d'essai.

L'instrumentation iGPS du bras a rendu la mesure externe exploitable pour la manipulation. La couronne montée au poignet a été conçue et intégrée dans ROS 2. La transformation géométrique entre la couronne et le poignet a été calibrée dans le repère du bras. La validation a montré que cette calibration corrige les principaux écarts de montage. La calibration terminale a relié la mesure iGPS à la sonde *i5* tenue dans la pince, avec une erreur moyenne de position inférieure au millimètre. Ensemble, ces deux calibrations forment une chaîne géométrique cohérente entre le poignet, l'outil et la mesure externe.

L'évaluation de la commande cartésienne du bras a quantifié l'apport du retour iGPS. Avec un ancrage externe commun, la RMSE 3D est passée d'environ 7,7 mm avec le retour cinématique interne du bras à 1,7 mm avec le retour iGPS. Le mode iGPS avec ancrage initial conserve ce niveau d'exactitude lorsque la base mobile est immobilisée avant la manipulation. Dans les essais de régularité, la base mobile n'a pas entraîné d'augmentation mesurable de la puissance spectrale du résidu cartésien par rapport au montage fixe. Cette observation reste limitée à la bande d'analyse retenue, de 0,71 à 10 Hz, et au protocole de régularité du mémoire.

L'intégration iGPS a été étendue à la navigation mobile. Hors couverture iGPS, la localisation autonome fournit une estimation de pose globale. Dans le laboratoire de métrologie, la source iGPS peut devenir la source globale active. La transition entre les deux sources repose sur une calibration plane entre le repère de carte et le repère iGPS. La commutation maintient une carte publiée et un arbre TF cohérents pour Nav2. Pour les poses cibles évaluées après immobilisation de la base, la RMSE médiane en position est passée de 78,0 mm avec la source SLAM à 3,5 mm avec la source iGPS. L'erreur de lacet a suivi la même tendance, de 2,17° à 0,24°.

La démonstration intégrée a réuni ces éléments dans un scénario d'assistance à l'opérateur. Elle a montré l'enchaînement fonctionnel obtenu : navigation autonome vers une pose de prise d'outil, retour au poste de métrologie, transition vers la source iGPS, ancrage de la base et

manipulation locale instrumentée. Elle ne constitue pas une cellule industrielle complète, mais montre l'intégration des fonctions nécessaires au scénario préparé.

6.2 Limites

Les résultats restent liés au contexte expérimental du mémoire. La couverture iGPS est limitée au laboratoire de métrologie. Hors de cette zone, la navigation repose sur la localisation autonome. Dans la zone instrumentée, la disponibilité et la précision de la mesure dépendent de la ligne de visée entre capteurs et émetteurs, du placement de ces éléments, des occlusions et du maintien de la géométrie calibrée entre la couronne et le poignet.

Ces calibrations dépendent de leurs conditions de montage. Celle de la couronne est liée à son montage réel et doit être reprise si la couronne est démontée, desserrée ou repositionnée. La calibration terminale dépend de la répétabilité du serrage de la sonde *i5* dans la pince. Comme les erreurs mesurées sont de l'ordre du millimètre, les transformations fixes, les repères du modèle et les petits défauts de montage deviennent des sources d'erreur importantes.

Les protocoles ne couvrent pas tous les modes d'utilisation du robot. L'erreur de pose de la base a été mesurée après immobilisation aux poses cibles, et non pendant le déplacement. La manipulation assistée par iGPS a été réalisée avec la base immobilisée. La démonstration intégrée reste une mission préparée et supervisée par l'opérateur. Les modes de repli restent limités et ne suffisent pas à garantir une autonomie complète.

6.3 Perspectives

Une première perspective consiste à améliorer la robustesse de la mesure iGPS au poignet. Le montage de la couronne pourrait être rendu plus rigide et plus répétable afin que sa calibration reste valable tant que ce montage n'est pas modifié. Une seconde piste serait d'ajouter des capteurs supplémentaires sur la couronne, avec des orientations et des hauteurs complémentaires.

Cette répartition pourrait élargir la plage d'incidence utile et limiter les masquages entre capteurs. Cet ajout devrait toutefois rester compatible avec la charge utile du bras, l'encombrement de la couronne et les dégagements nécessaires aux mouvements du poignet.

Le modèle Gazebo du RanGen pourrait aussi intégrer une dynamique plus complète. Le processus automatisé d'évaluation pourrait alors servir à ajuster les paramètres du simulateur et à vérifier que les comportements simulés restent comparables à ceux du robot réel.

La transition de localisation pourrait être renforcée par une supervision plus explicite de la qualité des mesures et des estimations. L'architecture actuelle maintient la localisation autonome disponible et permet une transition contrôlée vers l'iGPS. Une étape suivante serait d'ajouter des critères de qualité, des seuils de confiance et des règles de décision pour revenir vers la localisation autonome lorsque l'iGPS devient indisponible ou instable.

Une adaptation à une source GNSS constituerait une piste distincte. Le système de transition développé dans ce mémoire pourrait être réutilisé dans une logique proche du mode *fuse*. La carte resterait le repère global, tandis que la source GNSS serait ramenée dans ce repère comme une source globale supplémentaire. Le rôle du système resterait de maintenir, pour Nav2, une carte publiée, une source globale active et un arbre TF cohérents. La difficulté principale serait le rattachement initial du repère cartésien extérieur à cette carte. Une mesure GNSS fournit une position dans un repère géographique, mais elle ne définit pas à elle seule l'orientation et la pose plane utilisées par la navigation. Ce rattachement devrait alors s'appuyer sur une origine géographique, une orientation connue, des points de référence ou une trajectoire de calibration.

ANNEXE I

CONTRÔLE DE COHÉRENCE DE L'ODOMÉTRIE SIMULÉE

Cette annexe complète le chapitre 3 par un contrôle interne de l'odométrie simulée. Elle s'appuie sur la séquence `mode_odom_robustness`, qui soumet le contrôleur à des mouvements en double Ackermann, en parallèle et en pivot, avec des arrêts intermédiaires. L'essai porte uniquement sur la simulation. Il vérifie que l'odométrie publiée par le contrôleur reste cohérente avec la référence Gazebo sur l'ensemble de la séquence.

La figure I-1 présente la chronologie des consignes et les modes attendus pour l'essai.

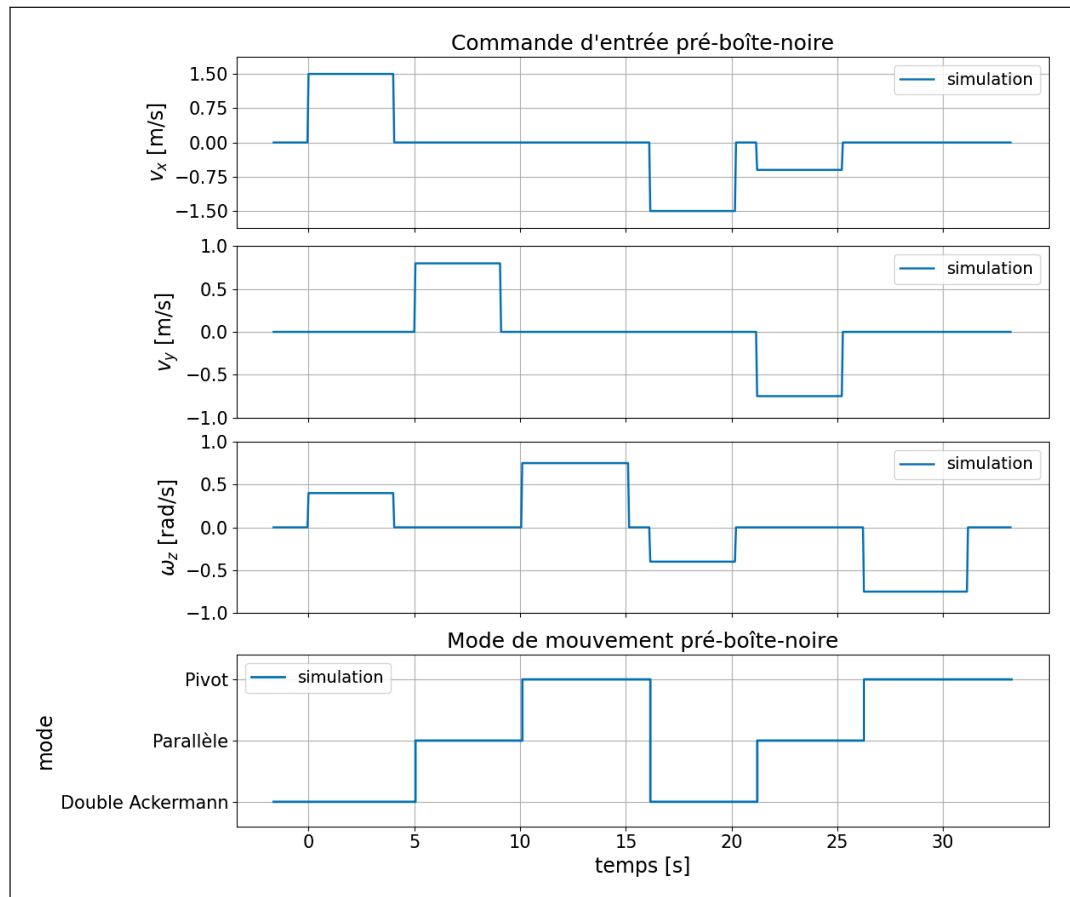


Figure-A I-1 Consignes (v_x , v_y , ω_z) et modes attendus pour le contrôle interne de l'odométrie simulée avec la séquence `mode_odom_robustness`

La figure I-2 montre que l'odométrie brute du contrôleur suit la référence Gazebo sans dérive visible, y compris lors des transitions entre modes.

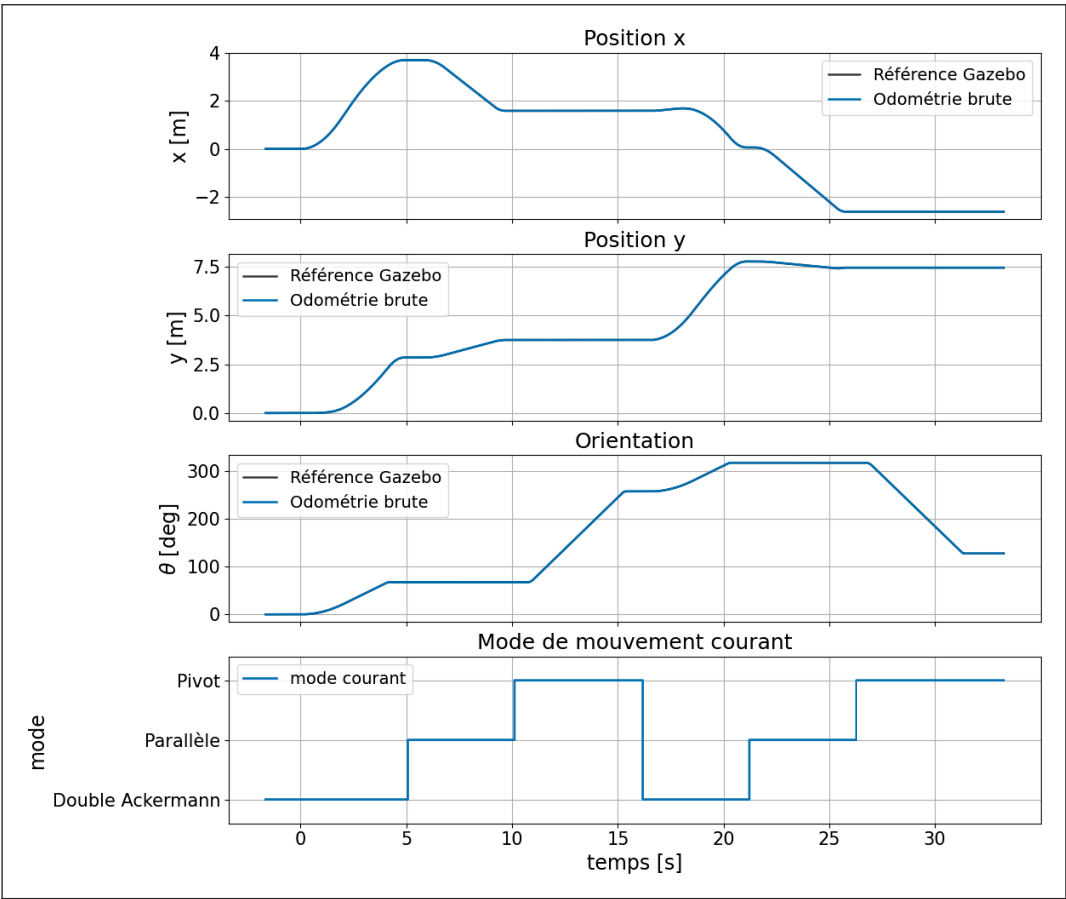


Figure-A I-2 Pose brute du contrôleur comparée à la référence de Gazebo pour la position, l'orientation et le mode courant

ANNEXE II

ANALYSES DÉTAILLÉES DU BRAS ROBOTIQUE

1. Charge des requêtes iGPS

Le tableau II-1 complète la comparaison de fréquence du chapitre. Il isole la charge des requêtes Surveyor. Pour chaque repère, il indique le volume de données retourné et la latence mesurée avec les commandes `GetFrameDataLite` et `GetFrameData`. La charge supplémentaire provient surtout des repères portés par la couronne ou les sondes. Le repère de référence de la base du bras varie peu entre les deux commandes. Ces mesures expliquent pourquoi la commande complète réduit davantage la fréquence de publication lorsque plusieurs repères sont interrogés.

Tableau-A II-1 Comparaison directe, pour chaque repère, de la taille des données retournées par Surveyor et de la latence mesurée avec les commandes allégée et complète. Dans l'en-tête, *Allégé* désigne `GetFrameDataLite` et *Complet* désigne `GetFrameData`. Les tailles et les latences sont les médianes des essais répétés. Les ratios correspondent à la valeur obtenue avec la commande complète divisée par celle obtenue avec la commande allégée

Repère interrogé	Essais	Taille des données retournées [B]			Latence [ms]		
		Allégé	Complet	Ratio	Allégé	Complet	Ratio
Couronne	20	615	46952	76,4	1,30	3,92	2,4
Sonde <i>i5</i> dans la pince	20	608	26020	42,8	0,92	2,11	2,4
Référence base du bras	20	495	490	1,0	0,75	0,72	0,9
Sonde <i>i5</i> opérateur	20	608	26004	42,8	0,76	2,74	3,4
Sonde <i>i6</i>	20	1058	39468	37,3	0,94	3,80	3,6
Total	20	3382	138939	41,1	4,88	14,61	2,6

2. Surface de coût de la calibration de la couronne

La figure II-1 présente les coupes de la surface de coût associées à la calibration de la couronne. Elle complète le résultat du chapitre en localisant le minimum selon la correction de translation δx et la correction de roulis $\delta \phi$.

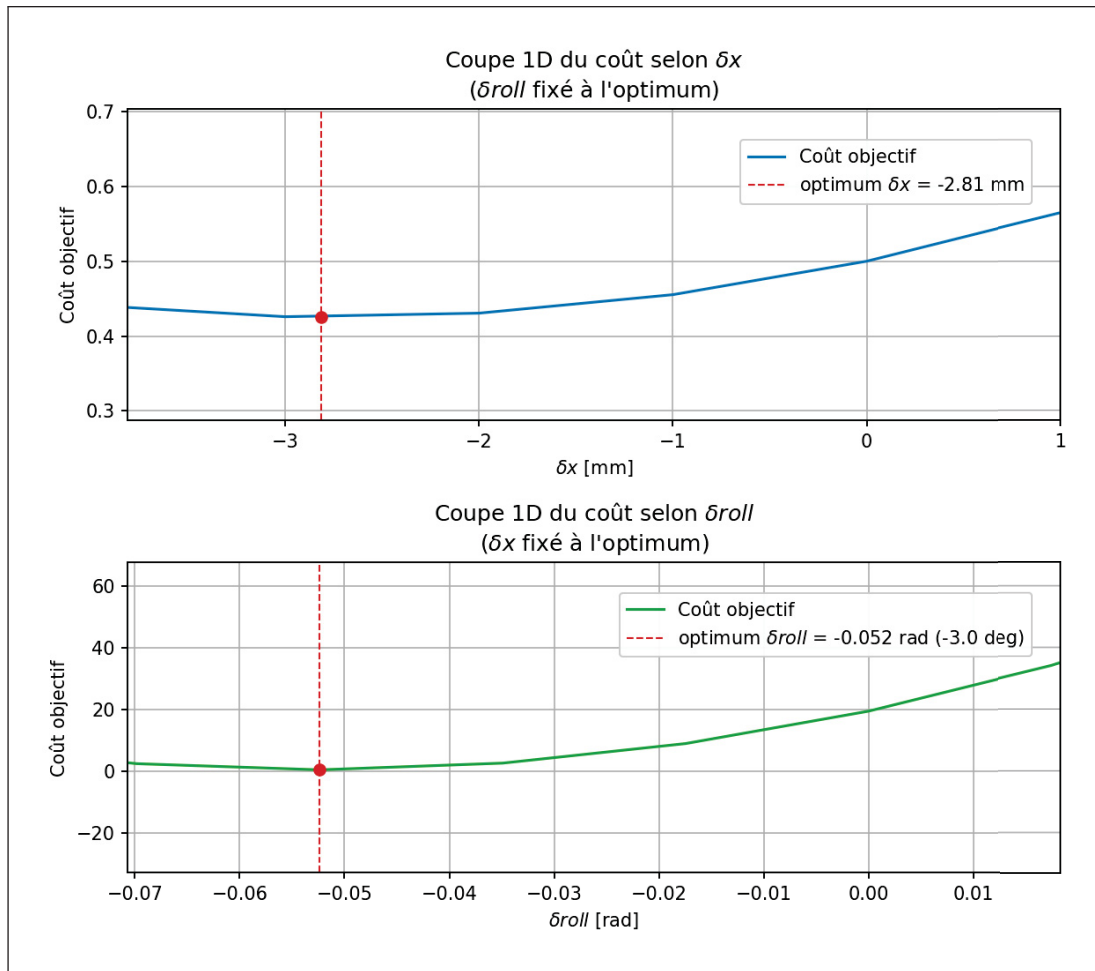


Figure-A II-1 Coupes unidimensionnelles de la surface de coût de calibration de la couronne selon δx et $\delta \phi$. La valeur nulle correspond à l'absence de correction par rapport au montage nominal

3. Choix de la bande fréquentielle d'analyse de stabilité

Les figures II-2 à II-4 documentent les seuils fréquentiels utilisés pour l'analyse de stabilité du chapitre. Elles reprennent le même résidu de trajectoire, analysé au moyen d'une densité spectrale de puissance (DSP) estimée par Welch. La borne $f_c = 0,71$ Hz est choisie à partir de la DSP groupée des essais fixe et mobile. Cette courbe est lissée, puis analysée en échelle log-log entre 0,2 et 6 Hz. La fréquence retenue est celle du point le plus éloigné de la droite reliant ces

deux bornes. La puissance de bande est ensuite intégrée entre f_c et 10 Hz pour chaque couche en X.

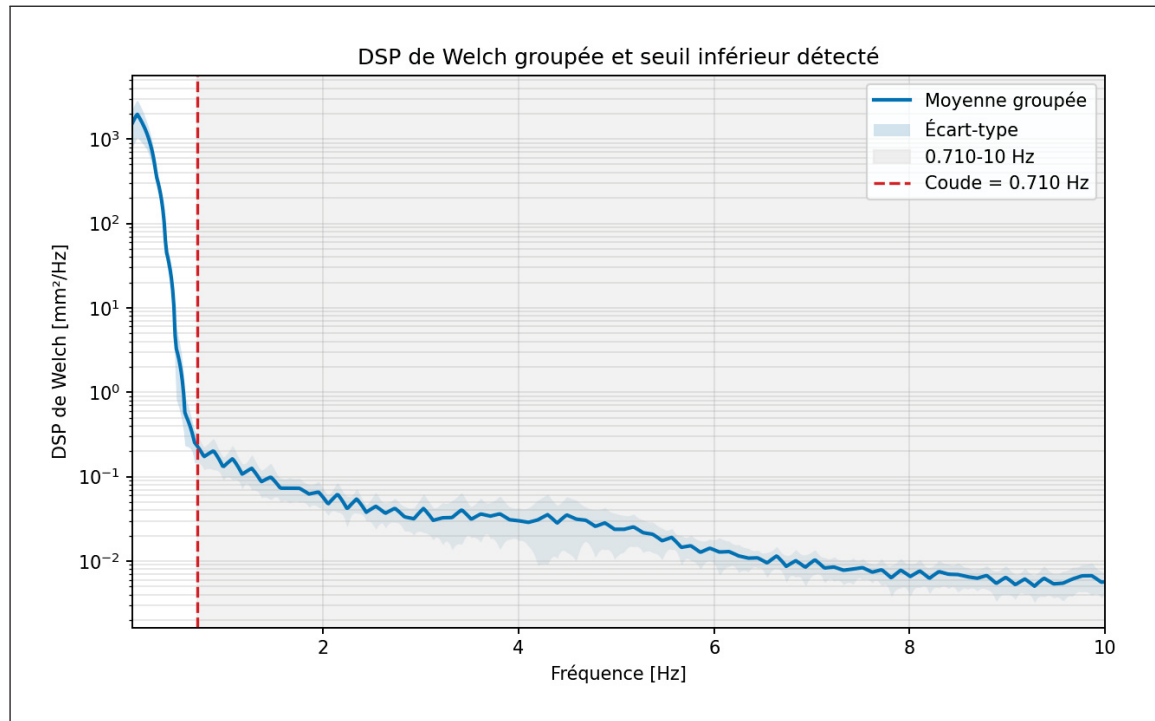


Figure-A II-2 DSP de Welch groupée et seuil inférieur f_c retenu pour la bande vibratoire

Les deux figures de sensibilité montrent que le ratio mobile/fixe reste proche de l'unité pour les seuils voisins des bornes retenues. Elles appuient donc le choix d'une bande d'intégration de f_c à 10 Hz, sans étendre l'analyse vers des fréquences proches de la limite de Nyquist des mesures iGPS.

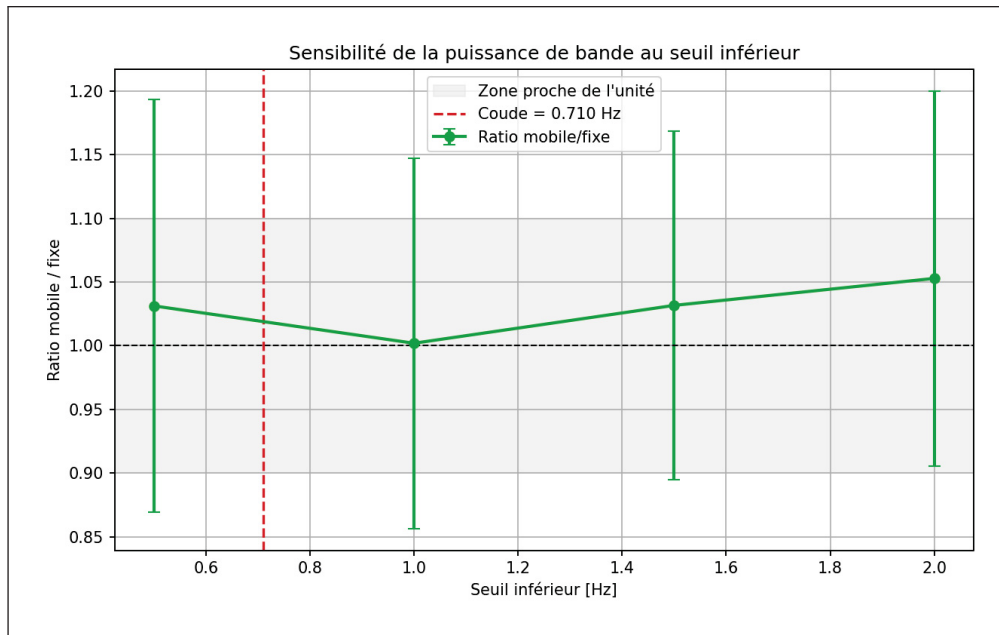


Figure-A II-3 Sensibilité du ratio mobile/fixe de la puissance de bande au seuil inférieur utilisé. Les valeurs testées s'arrêtent à 2 Hz, au-delà de la zone de transition visée

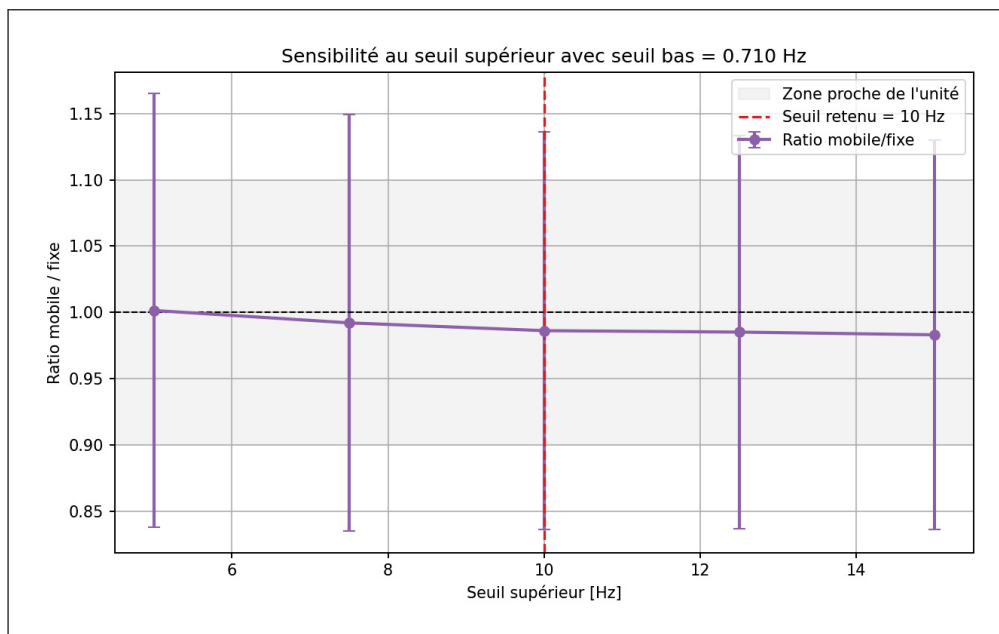


Figure-A II-4 Sensibilité du ratio mobile/fixe de la puissance de bande au seuil supérieur utilisé. Les valeurs testées s'arrêtent à 15 Hz, près de la limite de Nyquist associée à la fréquence moyenne la plus basse des poses iGPS de la sonde *i5*

4. Erreur spatiale de suivi du chemin

L'analyse fréquentielle du chapitre repose sur le résidu entre la position mesurée et la position de référence au même instant. Une métrique complémentaire évalue ici l'écart géométrique sans imposer cette correspondance temporelle. Pour chaque point mesuré pendant le mouvement actif d'un plan YZ , la distance spatiale est définie par

$$d_{sp}[n] = \min_{q \in C_X} \|p_{mes}[n] - q\|_2, \quad (\text{A II-1})$$

où C_X est la polyligne tridimensionnelle du chemin de référence pour le plan considéré. La minimisation est effectuée sur les segments de cette polyligne, et non seulement sur ses points échantillonnés. Elle tient ainsi compte des écarts selon X , Y et Z , sans recalage géométrique ni retrait du biais. Pour chaque essai et chaque plan, la RMSE et le 95^e centile des distances d_{sp} sont calculés.

La figure II-5 montre une augmentation globale de l'erreur avec l'extension du bras pour les deux montages. La RMSE spatiale varie de 15,6 mm à 23,3 mm pour le montage fixe et de 16,1 mm à 24,2 mm pour le montage mobile. Les plages de RMSE restent néanmoins proches pour les deux montages. Le 95^e centile varie de 19,3 mm à 44,6 mm pour le montage fixe et de 21,0 mm à 50,0 mm pour le montage mobile. Les écarts types inter-essais restent compris entre 0,02 et 0,13 mm pour la RMSE et entre 0,15 et 0,33 mm pour le 95^e centile. Cette faible dispersion indique une bonne répétabilité des deux métriques sur les cinq essais. L'écart dans la partie supérieure de la distribution est plus marqué pour le plan le plus éloigné, où ce centile atteint 42,7 mm sur le montage fixe et 50,0 mm sur le montage mobile. Cette métrique complète l'analyse fréquentielle en caractérisant l'écart géométrique au chemin indépendamment de la progression temporelle.

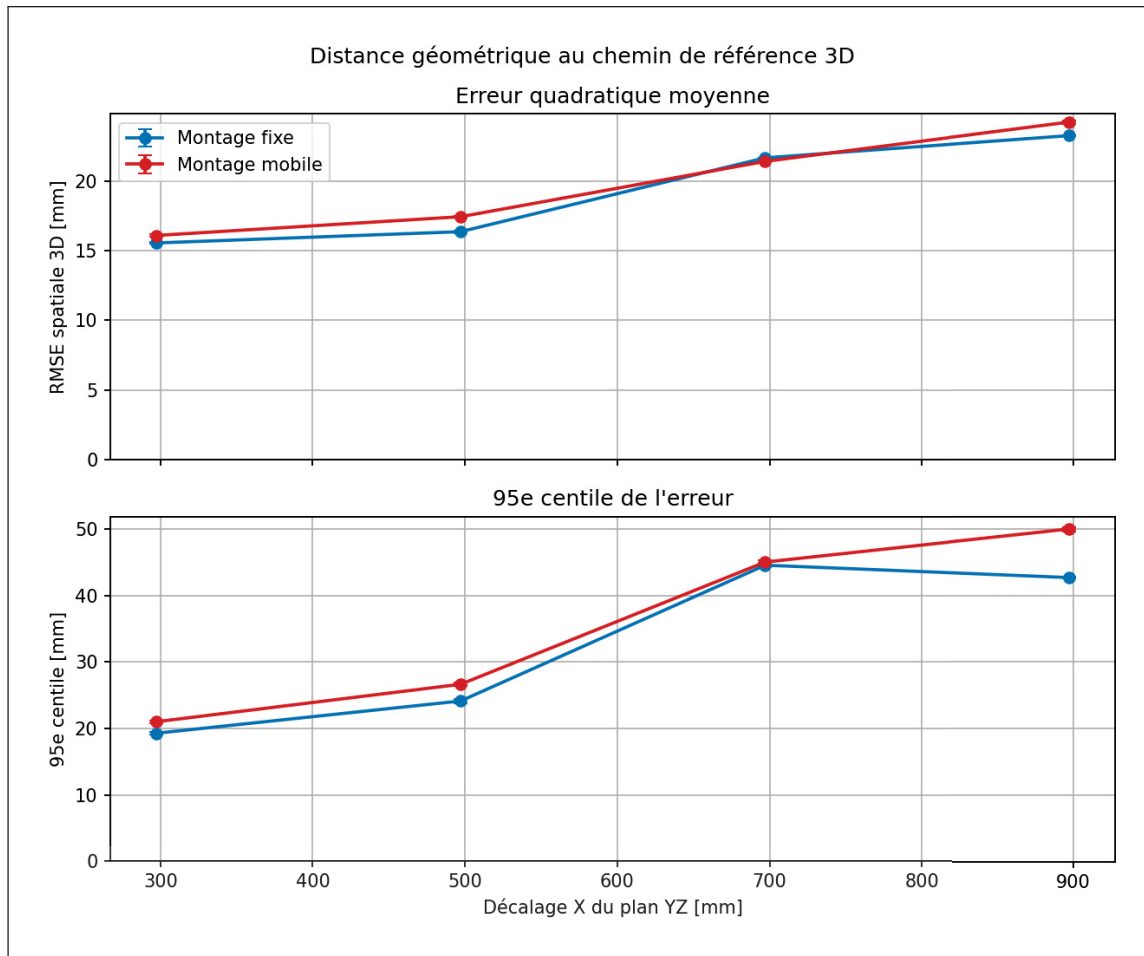


Figure-A II-5 Erreur spatiale tridimensionnelle par rapport au chemin de référence pour les montages fixe et mobile. Chaque point représente la moyenne sur les cinq essais de la métrique calculée pour un plan YZ

ANNEXE III

COMPARAISON DES MÉTHODES DE LOCALISATION AUTONOME DE LA BASE MOBILE

Cette annexe documente la comparaison qui justifie le choix de la chaîne de localisation autonome de la base mobile. Le chapitre 5 en reprend seulement le résultat nécessaire au choix de cette chaîne. Le détail porte sur le protocole, le périmètre retenu, les voies écartées, les résultats et les limites.

1. Protocole de comparaison

Le protocole utilise le robot réel dans l'environnement expérimental défini en méthodologie. Il repose sur une même séquence de données de capteurs, enregistrée dans un rosbag puis rejouée pour évaluer les méthodes de localisation. Ce jeu limite l'effet des variations de conduite entre les méthodes comparées.

La comparaison distingue deux niveaux de pose. Le niveau local estime la pose dans le repère odométrique *odom*. Il oppose l'odométrie par roues seule à un EKF qui fusionne les vitesses de roues et l'IMU. Le niveau global estime la pose dans le repère de carte *map*. Il compare AMCL et SLAM Toolbox à partir de la même estimation locale.

Le périmètre se limite aux méthodes comparables sur le robot réel à partir du même enregistrement. Chaque chaîne doit pouvoir être rejouée sur le même rosbag et utiliser les capteurs retenus pour la chaîne de localisation autonome.

Les chaînes évaluées utilisent l'odométrie des roues, l'IMU et le LiDAR 2D. Le LiDAR 3D et la caméra présents sur la plateforme sont discutés séparément comme voies écartées.

Dans la configuration évaluée, l'EKF fusionne les vitesses v_x , v_y et $\dot{\psi}$ issues de l'odométrie par roues avec le lacet ψ mesuré par l'IMU.

La référence externe sert uniquement à l'évaluation hors ligne et n'appartient pas à la chaîne de localisation autonome. Elle correspond ici à une pose iGPS obtenue avec le montage préliminaire où les capteurs étaient placés sur la base mobile. Sa disponibilité est limitée à la zone couverte par l'iGPS, ce qui restreint la portion du trajet utilisée pour les métriques.

2. Méthodes écartées

Ce périmètre exclut la localisation LiDAR-inertielle en 3D de la campagne quantitative. LIO-SAM et certaines variantes produisaient une pose exploitable sur le robot, mais reposaient sur GTSAM, absent de la configuration du projet après la migration vers ROS 2 Jazzy et Ubuntu 24.04. Le LiDAR 3D posait aussi une contrainte expérimentale, car il perturbait la mesure iGPS dans la zone de métrologie. L'éteindre dans cette zone aurait supprimé la source indépendante que la transition doit conserver en parallèle du système iGPS.

La localisation visuelle reste également hors du périmètre. La caméra Oak-D Pro Wide était disponible dans ROS 2, mais aucune chaîne visuelle n'a été validée pour ce mémoire. Sa fréquence d'acquisition et sa charge logicielle ajoutaient un risque d'intégration finale. La campagne quantitative se limite donc à l'odométrie par roues, à l'IMU et au LiDAR 2D.

3. Résultats

Le tableau III-1 présente les RMSE de translation et de lacet, calculées hors ligne sur les portions du trajet où la référence externe est disponible. Ces métriques comparent respectivement la position plane et le lacet estimés à la référence externe.

L'EKF conserve la même erreur de translation que l'odométrie par roues, mais réduit l'erreur d'orientation. Ce résultat correspond à la configuration évaluée : l'IMU contraint le lacet sans ajouter de mesure absolue de position. L'EKF est donc retenu comme niveau local de la chaîne de localisation autonome.

AMCL réduit les RMSE par rapport aux estimations locales. Sa trajectoire reste cependant plus irrégulière que celle de SLAM Toolbox. Dans cette campagne sur robot réel, SLAM Toolbox est évalué en mode *localization* sur une carte préexistante. Il fournit les RMSE de translation et d'orientation les plus faibles. Ces résultats justifient son choix comme niveau global de la chaîne de localisation autonome.

Tableau-A III-1 RMSE de translation et d'orientation pour les méthodes comparées dans la campagne de localisation autonome

Méthode	RMSE de translation [m]	RMSE de lacet [°]
Odométrie par roues	0,357	6,208
EKF	0,357	4,583
AMCL	0,227	4,146
SLAM Toolbox (<i>localization</i>)	0,075	2,943

La figure III-1 montre la portion du trajet où la référence iGPS permet cette comparaison quantitative. La trajectoire de SLAM Toolbox reste proche de cette référence sur toute la portion représentée. Celle d'AMCL corrige aussi la dérive globale, mais ne suit pas aussi précisément la référence.

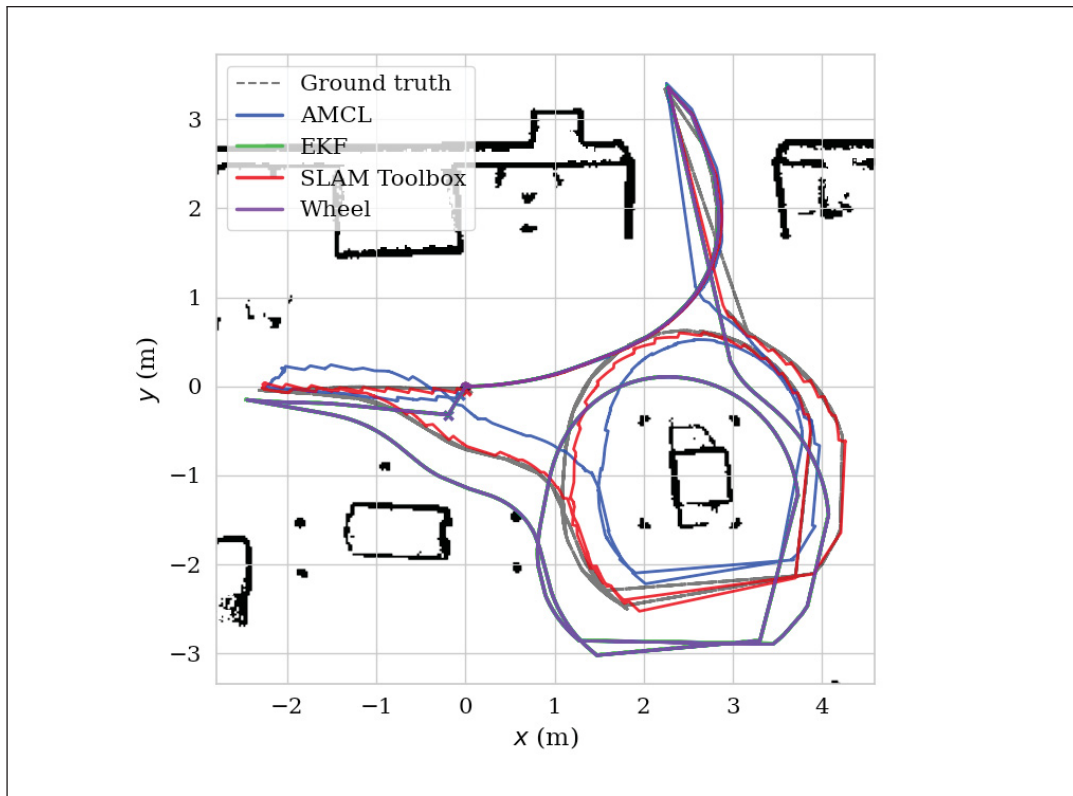


Figure-A III-1 Comparaison des trajectoires reconstruites dans la portion où la référence iGPS est disponible pour le calcul des métriques

Une campagne séparée en simulation complète la comparaison. Le tableau III-2 en présente les résultats principaux. SLAM Toolbox obtient les plus faibles erreurs de translation dans les deux modes évalués. L'écart entre les modes *mapping* et *localization* reste faible lorsqu'une carte préexistante est disponible. LIO-SAM donne l'erreur de lacet la plus faible.

Tableau-A III-2 RMSE de translation et d'orientation pour la campagne simulée

Méthode	RMSE de translation [m]	RMSE de lacet [°]
Odométrie par roues	0,293	2,876
EKF	0,294	0,401
AMCL	0,140	1,169
LIO-SAM	0,255	0,390
SLAM Toolbox (<i>mapping</i>)	0,041	0,513
SLAM Toolbox (<i>localization</i>)	0,043	0,481

4. Limites de la comparaison

Cette comparaison soutient le choix de la chaîne de localisation autonome retenue pour le mémoire. Elle ne constitue pas une étude exhaustive de la localisation mobile. La référence externe ne couvre qu'une partie du trajet. La séquence source est conduite manuellement, même si le rejeu du même rosbag limite cette variabilité. Les paramètres d'AMCL et de SLAM Toolbox n'ont pas été optimisés de manière exhaustive.

Le choix retenu doit donc être lu comme une base autonome robuste plutôt que comme une optimisation exhaustive de la localisation mobile. La transition vers l'iGPS n'impose pas une méthode de localisation particulière : elle suppose surtout une estimation de la base publiée dans la structure TF standard `map-odom-base_link`, avec une carte et une transformation `map-odom` cohérentes pour Nav2. Dans cette configuration, l'association d'un EKF local et de SLAM Toolbox en mode *localization* sur une carte préexistante fournit un point de départ robuste pour le chapitre 5.

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

- 7D Kinematic Metrology. [Page produit officielle consultée le 10 juin 2026]. (2026a). i5IS. Repéré le 2026-06-10 à <https://7dkmetrology.com/7dk-1/i5is-en>.
- 7D Kinematic Metrology. [Page produit officielle consultée le 10 juin 2026]. (2026b). I6 LRP G6. Repéré le 2026-06-10 à <https://7dkmetrology.com/7dk-1/i6-lrp-g6-en>.
- 7D Kinematic Metrology. [Page application officielle consultée le 20 avril 2026]. (2026c). AGV Tracking. Repéré le 2026-04-20 à <https://7dkmetrology.com/7dk-1/agv-tracking>.
- 7D Kinematic Metrology. [Page produit officielle consultée le 11 avril 2026]. (2026d). Products. Repéré le 2026-04-11 à <https://7dkmetrology.com/7dk-1/products>.
- AgileX Robotics. [Dépôt GitHub officiel pour la simulation Gazebo de la base Ranger Mini, consulté le 13 avril 2026]. (2026). UGV Gazebo Sim : Ranger Mini. Repéré le 2026-04-13 à https://github.com/agilexrobotics/ugv_gazebo_sim/tree/master/ranger_mini.
- Arts et Métiers. [Actualité publiée le 26 avril 2018, consultée le 23 juin 2026]. (2018, avril). ColRobot : un projet européen de robotique collaborative porté par Arts et Métiers. Repéré le 2026-06-23 à <https://artsetmetiers.fr/fr/actualites/colrobot-un-projet-europeen-de-robotique-collaborative-porte-par-arts-et-metiers>.
- Bostelman, R. V., Li-Baboud, Y., Yoon, S., Shah, M. & Aboul-Enein, O. Y. (2020). *Towards Measurement of Advanced Mobile Manipulator Performance for Assembly Applications* (Rapport n°2108). Gaithersburg, MD. Repéré à <https://doi.org/10.6028/NIST.TN.2108>.
- Cao, Y., Usai, A., Gu, W. & Rizzo, A. (2025). Localization Handover for Mobile Robots : A Seamless Indoor-Outdoor Approach. *Journal of Intelligent & Robotic Systems*, 111(82), 82. doi : 10.1007/s10846-025-02282-9.
- Carpentier, J., Saurel, G., Buondonno, G., Mirabel, J., Lamiriaux, F., Stasse, O. & Mansard, N. (2019). The Pinocchio C++ Library : A Fast and Flexible Implementation of Rigid Body Dynamics Algorithms and Their Analytical Derivatives. *2019 IEEE/SICE International Symposium on System Integration (SII)*, pp. 614–619. doi : 10.1109/SII.2019.8700380.
- Charles River Analytics. [Dépôt GitHub du paquet ROS robot_localization, consulté le 1 mai 2026]. (2026). robot_localization : Nonlinear State Estimation Nodes for ROS. Repéré le 2026-05-01 à https://github.com/cra-ros-pkg/robot_localization.

- CORDIS. [Results in Brief du projet ColRobot, dernière mise à jour le 26 février 2019, consulté le 23 juin 2026]. (2019, février). Multitasking Robots Work Hand-in-Hand with Operators. Repéré le 2026-06-23 à <https://cordis.europa.eu/article/id/251212-multitasking-robots-work-handinhand-with-operators>.
- Dellaert, F., Fox, D., Burgard, W. & Thrun, S. (1999). Monte Carlo Localization for Mobile Robots. *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, 2, 1322–1328. doi : 10.1109/ROBOT.1999.772544.
- Foote, T. (2013). tf : The Transform Library. *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*, pp. 1–6. doi : 10.1109/TePRA.2013.6556373.
- Gazebo Project. [Consulté le 14 juin 2026]. (2026a). ROS 2 integration overview. Repéré le 2026-06-14 à https://gazebosim.org/docs/latest/ros2_overview/.
- Gazebo Project. [Consulté le 14 juin 2026]. (2026b). ros_gz : Integration between ROS and Gazebo. Repéré le 2026-06-14 à https://github.com/gazebosim/ros_gz.
- Ghodsian, N., Benfriha, K., Olabi, A., Gopinath, V. & Arnou, A. (2023). Mobile Manipulators in Industry 4.0 : A Review of Developments for Industrial Applications. *Sensors*, 23(19), 8026. doi : 10.3390/s23198026.
- Kinova Robotics. [Documentation officielle de l'API KINOVA KORTEX, consultée le 13 avril 2026]. (2022). Kortex API Documentation. Repéré le 2026-04-13 à <https://docs.kinovarobotics.com/>.
- Kinova Robotics. [Fiche produit officielle consultée le 11 avril 2026]. (2024). KINOVA GEN3 Robots. Repéré le 2026-04-11 à <https://www.kinovarobotics.com/product/gen3-robots/>.
- Kritzing, W., Karner, M., Traar, G., Henjes, J. & Sihn, W. (2018). Digital Twin in Manufacturing : A Categorical Literature Review and Classification. *IFAC-PapersOnLine*, 51(11), 1016–1022. doi : 10.1016/j.ifacol.2018.08.474.
- Li, N., Guan, L., Gao, Y., Du, S., Wu, M., Guang, X. & Cong, X. (2020). Indoor and Outdoor Low-Cost Seamless Integrated Navigation System Based on the Integration of INS/GNSS/LIDAR System. *Remote Sensing*, 12(19), 3271. doi : 10.3390/rs12193271.
- Liu, Y., Li, Y., Zhuang, Z. & Song, T. (2020). Improvement of Robot Accuracy with an Optical Tracking System. *Sensors*, 20(21), 6341. doi : 10.3390/s20216341.
- Macenski, S. [Dépôt GitHub du paquet ROS 2 slam_toolbox, consulté le 1 mai 2026]. (2026). slam_toolbox : Slam Toolbox for Lifelong Mapping and Localization in ROS 2. Repéré le 2026-05-01 à https://github.com/SteveMacenski/slam_toolbox.

- Macenski, S. & Jambrecic, I. (2021). SLAM Toolbox : SLAM for the Dynamic World. *Journal of Open Source Software*, 6(61), 2783. doi : 10.21105/joss.02783.
- Macenski, S., Martín, F., White, R. & Ginés Clavero, J. (2020). The Marathon 2 : A Navigation System. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2718–2725. doi : 10.1109/IROS45743.2020.9341207.
- Macenski, S., Foote, T., Gerkey, B., Lalancette, C. & Woodall, W. (2022). Robot Operating System 2 : Design, architecture, and uses in the wild. *Science Robotics*, 7. doi : 10.1126/scirobotics.abm6074.
- Meeussen, W. (2010). REP 105 : Coordinate Frames for Mobile Platforms. Repéré le 2026-06-14 à <https://www.ros.org/reps/rep-0105.html>.
- Moore, T. & Stouch, D. (2014). A Generalized Extended Kalman Filter Implementation for the Robot Operating System. Dans Menegatti, E., Michael, N., Berns, K. & Yamaguchi, H. (Éds.), *Intelligent Autonomous Systems 13* (vol. 302, pp. 335–348). Springer. doi : 10.1007/978-3-319-08338-4_25.
- Muelaner, J. E., Wang, Z., Martin, O., Jamshidi, J. & Maropoulos, P. G. (2012). Verification of the indoor GPS system, by comparison with calibrated coordinates and by angular reference. *Journal of Intelligent Manufacturing*, 23(6), 2323–2331. doi : 10.1007/s10845-010-0488-y.
- Nakamura, Y. & Hanafusa, H. (1986). Inverse Kinematic Solutions With Singularity Robustness for Robot Manipulator Control. *Journal of Dynamic Systems, Measurement, and Control*, 108(3), 163–171. doi : 10.1115/1.3143764.
- National Institute of Standards and Technology. [Consulté le 6 mars 2026]. (2026). Definitions and State of the Art. Repéré le 2026-03-06 à <https://www.nist.gov/digital-twins/definitions-and-state-art>.
- Nicksch, C., Hüttner, A. K. & Schmitt, R. H. (2022a). Transmitter Positioning of Distributed Large-Scale Metrology Within Line-Less Mobile Assembly Systems. Dans Schüppstuhl, T., Tracht, K. & Raatz, A. (Éds.), *Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021* (pp. 15–26). Cham : Springer. doi : 10.1007/978-3-030-74032-0_2.
- Nicksch, C., Sabzehi, M. & Schmitt, R. H. (2022b). Virtual Indoor-GPS for Measurement Uncertainty Determination in Reconfigurable Environments. *Production Engineering*, 16(4), 545–560. doi : 10.1007/s11740-022-01111-4.

- Norman, A. R., Schönberg, A., Gorlach, I. A. & Schmitt, R. (2012). Validation of iGPS as an External Measurement System for Cooperative Robot Positioning. *The International Journal of Advanced Manufacturing Technology*, 64(1–4), 427–446. doi : 10.1007/s00170-012-4004-8.
- NVIDIA Corporation. [Consulté le 14 juin 2026]. (2026a). What Is Isaac Sim ? Repéré le 2026-06-14 à <https://docs.isaacsim.omniverse.nvidia.com/>.
- NVIDIA Corporation. [Consulté le 14 juin 2026]. (2026b). Sensors. Repéré le 2026-06-14 à <https://docs.isaacsim.omniverse.nvidia.com/6.0.0/sensors/index.html>.
- NVIDIA Corporation. [Consulté le 14 juin 2026]. (2026c). Synthetic Data Generation. Repéré le 2026-06-14 à https://docs.isaacsim.omniverse.nvidia.com/6.0.0/synthetic_data_generation/index.html.
- Open Navigation LLC. [Dépôt GitHub du paquet ROS 2 nav2_amcl, consulté le 1 mai 2026]. (2026a). nav2_amcl : Adaptive Monte Carlo Localization for Nav2. Repéré le 2026-05-01 à https://github.com/ros-navigation/navigation2/tree/main/nav2_amcl.
- Open Navigation LLC. [Documentation de configuration des transformations TF dans Nav2, consultée le 23 juin 2026]. (2026b). Setting Up Transformations. Repéré le 2026-06-23 à https://docs.nav2.org/setup_guides/transformation/setup_transforms.html.
- Open Navigation LLC. [Dépôt GitHub du système de navigation ROS 2 Nav2, consulté le 1 mai 2026]. (2026c). Navigation2 : ROS 2 Navigation Framework and System. Repéré le 2026-05-01 à <https://github.com/ros-navigation/navigation2>.
- Ostuni, A. (2022). *Seamless Indoor-Outdoor Autonomous Navigation for Unmanned Ground Vehicles*. (Mémoire de maîtrise, Politecnico di Torino). Repéré à Mémoire de maîtrise en Mechatronic Engineering.
- PickNik Robotics. [Consulté le 14 juin 2026]. (2025). How To Command Simulated Isaac Robot. Repéré le 2026-06-14 à https://moveit.picknik.ai/main/doc/how_to_guides/isaac_panda/isaac_panda_tutorial.html.
- Robotiq. [Documentation officielle consultée le 11 avril 2026]. (2019). 2F-85 and 2F-140 Adaptive Grippers Instruction Manual e-Series. Repéré le 2026-04-11 à https://assets.robotiq.com/website-assets/support_documents/document/2F-85_2F-140_Instruction_Manual_e-Series_PDF_20190206.pdf.
- ros2_control Development Team. [Consulté le 14 juin 2026]. (2026a). gz_ros2_control. Repéré le 2026-06-14 à https://control.ros.org/humble/doc/gz_ros2_control/doc/index.html.

- ros2_control Development Team. [Documentation officielle consultée le 15 mars 2026]. (2026b). Getting Started. Repéré le 2026-03-15 à https://control.ros.org/rolling/doc/getting_started/getting_started.html.
- ros2_control Development Team. [Consulté le 14 juin 2026]. (2026c). Simulator Integrations. Repéré le 2026-06-14 à <https://control.ros.org/master/doc/simulators/simulators.html>.
- Shah, M. (2013). Solving the Robot-World/Hand-Eye Calibration Problem Using the Kronecker Product. *Journal of Mechanisms and Robotics*, 5(3), 031007. doi : 10.1115/1.4024473.
- Thales Alenia Space. [Article publié le 13 mars 2019, consulté le 19 juin 2026]. (2019, mars). ColRobots : on a testé pour vous les Robots Collaboratifs. Repéré le 2026-06-19 à <https://www.thalesaleniaspace.com/fr/news/colrobots-teste-pour-vous-les-robots-collaboratifs>.
- Tsai, R. Y. & Lenz, R. K. (1989). A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration. *IEEE Transactions on Robotics and Automation*, 5(3), 345–358. doi : 10.1109/70.34770.
- Umeyama, S. (1991). Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4), 376–380. doi : 10.1109/34.88573.
- Welch, P. D. (1967). The Use of Fast Fourier Transform for the Estimation of Power Spectra : A Method Based on Time Averaging Over Short, Modified Periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2), 70–73. doi : 10.1109/TAU.1967.1161901.
- Weston Robot. [Documentation Weston Robot pour la plateforme AgileX Ranger Mini 2.0, consultée le 11 avril 2026]. (2026). Ranger Mini 2.0. Repéré le 2026-04-11 à <https://docs.westonrobot.com/robot/ugv/ranger-mini-v2/>.
- Whitney, D. E. (1969). Resolved Motion Rate Control of Manipulators and Human Prostheses. *IEEE Transactions on Man-Machine Systems*, 10(2), 47–53. doi : 10.1109/TMMS.1969.299896.