UCLouvain

epl

École polytechnique de Louvain

Démonstrateur électromécanique pour de la simulation Hardware-In-the-Loop:

Conception mécatronique

Auteurs: **Guillaume Hisette, Sylvain Bruyr**
Promoteurs: **Paul Fisette, Nicolas Docquier**
Lecteurs: **Bruno Dehez, Marine Boxus**
Année académique 2025–2026
Master [120] : ingénieur civil électromécanicien

Remerciements

La réalisation de ce mémoire n'aurait pas été possible sans l'aide et les conseils d'un grand nombre de personnes. Nous profitons pour les en remercier chaleureusement.

Tout d'abord, un grand merci à notre promoteur, Paul Fisette, et à son assistante, Marine Boxus, qui nous ont épaulés et guidés tout au long de l'année. Leurs réunions quasi hebdomadaires ont permis d'obtenir des retours constructifs sur l'avancement de notre travail. Nous les remercions également pour la confiance qu'ils nous ont accordée ainsi que pour la relecture d'une partie de ce mémoire.

Nous adressons aussi nos remerciements à Nicolas Docquier pour ses conseils lors de nos réunions du premier quadrimestre.

Merci à Bastien Debroux pour s'être occupé de la conception et de la fabrication mécanique du démonstrateur, en bénéficiant des conseils de Benoît Herman et de Julien Guiaux Brinon, ainsi qu'à l'usineur des pièces mécaniques, Alban Maton.

Nous tenons à remercier l'ensemble des techniciens du CREDEM : Frank Hesbois pour ses conseils dans la réalisation du coffret électrique, Thierry Daras pour ses recommandations ainsi que pour la mise à disposition de certains capteurs de son stock, et Simon De Jaeger pour nous avoir permis d'accéder au Makin'Pro et à tout le matériel disponible.

Nous tenons également à remercier Alex Berthollet pour ses conseils et son expertise concernant les chaînes d'instrumentation et les capteurs de force, ainsi que pour nous avoir aidés à calibrer notre conditionneur et à gérer l'achat de ces composants.

Nos remerciements vont aussi à Sébastien Timmermans pour sa présence lors de l'une de nos réunions ainsi que pour son aide sur le fonctionnement en temps réel de Robotran.

Nous voulons également remercier Astrid Leduc, Michèle Deneef et Isabelle Snaecken pour la gestion et le suivi de toutes nos commandes en ces temps compliqués de changement de systèmes de facturation.

Enfin, nous tenons à remercier nos familles pour la relecture finale de notre mémoire ainsi que pour le soutien qu'elles nous ont apporté.

Nous profitons de cette page pour informer le lecteur que l'intelligence artificielle a été utilisée exclusivement pour la reformulation, la correction orthographique et l'amélioration de la structure de ce mémoire.

Table des matières

Table des figures **vii**

Liste des tableaux **viii**

1 Contexte du HIL **3**
1.1 Revue de littérature . . . 3
1.1.1 Applications du HIL dans l'industrie . . . 3
1.1.1.1 HIL dans le processus de design . . . 3
1.1.1.2 HIL hors du scope du mémoire . . . 4
1.1.2 Implémentations du HIL dans le domaine académique . . . 4
1.1.2.1 Development of a Biomechanical Multibody Model for the Hardware-in-the-Loop Simulation of Total Hip Endoprostheses 4
1.1.2.2 Test bench design for motorcycle suspensions HIL test . . . 4
1.2 Moteur de simulation : formalisme multi-corps . . . 5
1.2.1 Architecture de Robotran . . . 5
1.2.2 Formalisme multi-corps . . . 5
1.2.3 Simulation en temps réel . . . 7
1.2.4 Simulation Hardware in the loop . . . 8

2 Cahier des Charges **9**
2.1 Limites d'utilisation du démonstrateur HIL . . . 9
2.2 Graphe des fonctions . . . 9
2.3 Cahier des charges . . . 10

3 Conception du démonstrateur **15**
3.1 Actionnement . . . 16
3.1.1 Choix de la morphologie d'actionnement . . . 16
3.1.1.1 Actionneur servo-hydraulique . . . 17
3.1.1.2 Servomoteur linéaire . . . 18
3.1.1.3 Actionneur linéaire à vis . . . 18
3.1.1.4 Morphologie retenue . . . 19
3.1.2 Dimensionnement . . . 19
3.1.2.1 Couple de friction $M_{friction}$ . . . 20
3.1.2.2 Couple effectif M_{eff} . . . 20
3.1.2.3 Inertie du vérin I_{verin} . . . 21
3.1.2.4 Simulation . . . 21
3.1.3 Modèle thermique . . . 22
3.1.4 Choix du protocole de communication . . . 23
3.1.4.1 CANopen sur bus CAN . . . 24
3.1.4.2 PROFINET . . . 24
3.1.4.3 EtherCAT . . . 24
3.1.4.4 Résumé des protocoles de communication . . . 24

3.1.5 Choix final de l'actionnement . . . 24
3.2 Instrumentation et chaîne d'acquisition . . . 26
3.2.1 Mesure de force . . . 26
3.2.1.1 Étude des différentes technologies de capteurs de force . . . 27
3.2.1.2 Comparaison et choix des capteurs à jauge de contrainte . . . 28
3.2.1.3 Conditionnement du signal analogique . . . 30
3.2.2 Mesure de position . . . 31
3.2.3 Conversion analogique-numérique . . . 31
3.2.4 Résumé de la précision de la mesure théorique . . . 33
3.3 Électronique globale . . . 33
3.3.1 Système informatique . . . 33
3.3.2 Chaîne d'instrumentation et acquisition de données . . . 34
3.3.3 Chaîne d'actionnement et puissance . . . 35
3.3.4 Sécurité électrique . . . 35

4 Architecture logicielle et implémentation de la simulation HIL 36
4.1 ROS 2 (Robot Operating System 2) . . . 36
4.1.1 Architecture logicielle du banc de test . . . 37
4.1.1.1 adc_node . . . 38
4.1.1.2 FastDDS . . . 38
4.1.1.3 actuator_node . . . 38
4.1.1.4 ros2_control . . . 39
4.1.1.5 Nœud Robotran . . . 39
4.2 Choix du système d'exploitation temps réel . . . 41
4.3 Initialisation d'une simulation . . . 41
4.3.1 Procédure de retour à l'origine . . . 41
4.3.2 Recherche de la longueur neutre du transmetteur de force . . . 43

5 Performance et caractérisation 44
5.1 Performance de la chaîne d'instrumentation . . . 44
5.1.1 Calibration du conditionneur . . . 44
5.1.2 Conversion de la mesure de tension du potentiomètre . . . 46
5.1.3 Méthode d'acquisition et performances de la chaîne d'instrumentation en statique . . . 46
5.1.4 Impact du filtre interne du conditionneur CPJ en statique . . . 47
5.2 Caractérisation de la déformation du capteur et de la structure . . . 48
5.3 Caractérisation de la fonction de transfert géométrique . . . 49
5.3.1 Erreur de la fonction . . . 51
5.3.2 Erreur liée à la position d'origine . . . 51
5.4 Caractérisation du modèle thermique . . . 52
5.5 Mesure de la raideur des Transmetteurs de Force considérés . . . 53
5.6 Simulation Hardware In the Loop . . . 53
5.6.1 Simulation avec deux nœuds . . . 53
5.6.2 Driven Motion Sinus . . . 54
5.6.3 Système masse-ressort classique avec amortissement . . . 55
5.6.4 Simulation d'un VTT . . . 57
5.7 Délai d'actionnement . . . 60

A Démonstration de l'impact du filtre de moyennage sur la dynamique 64

B Options considérées pour le choix de l'actionneur **65**
B.1 Actionneur servo-hydraulique 65
B.2 Servomoteur linéaire 65
B.3 Actionneur à vis 65
B.3.1 Vérins 65
B.3.2 Moteurs 66
B.3.3 Combinaisons étudiées 66

C Pré-prototype **67**

D Analyse du flambement du corps rigide pour le teste de déformation du démonstrateur **68**
D.1 Schéma electrique du circuit de sécurité 69

Table des figures

1 HIL : intersection entre la simulation et l'expérimental, traduction de l'illustration de la Fig.2 de [1] . . . 1

1.1 Environnement Robotran - Figure tiré de [2] . . . 5
1.2 Schéma-bloc du solveur Robotran - Figure inspirée et simplifiée de [2] . . . 6
1.3 Répartition temporelle d'un pas de calcul pour une simulation en temps réel . 7
1.4 Schéma-bloc du solveur Robotran avec du HIL - Figure inspirée et simplifiée de [3] . . . 8

2.1 Graphe des fonctions - Séquence de mise en place . . . 10
2.2 Graphe des fonctions - Boucle de test . . . 10

3.1 Vue globale du banc de test . . . 15
3.2 Schéma simplifié du banc d'essai . . . 16
3.3 Schéma hydraulique d'un actionneur linéaire servo-hydraulique à distributeur proportionnel [4] . . . 17
3.4 Vue en coupe d'un servomoteur linéaire cylindrique [5] . . . 18
3.5 Couple de friction dynamique en fonction de la vitesse linéaire du vérin pour différents pas de vis (5-10-16 mm/tour) [6] . . . 20
3.6 Couple effectif en fonction de la force linéaire du vérin pour différents pas de vis (5-10-16 mm/tour) [6] . . . 21
3.7 Évolution du point de fonctionnement d'un moteur durant une courte simulation 22
3.8 Point de fonctionnement RMS, couple max et vitesse max de l'ensemble des combinaisons de moteurs et vérins considérés . . . 22
3.9 Festo ESBF-BS-40-200-10P . . . 25
3.10 Vue en coupe du kit de montage axial reliant le vérin au servomoteur . . . 25
3.11 Stepperonline A6M60-400H2A1-M17 . . . 25
3.12 Courbes de couple/vitesse du servomoteur choisi [7] . . . 26
3.13 Photo du contrôleur . . . 26
3.14 Pont de Wheatstone avec 4 jauges de contrainte [8] . . . 27
3.15 Structure interne d'un capteur piézoélectrique [9] . . . 28
3.16 Schéma équivalent du capteur de force sous un système masse-ressort - Figure tirée de [10] . . . 29
3.17 Distribution des masses impliquant la dynamique - Figure tirée de [10] . . . 29
3.18 Conditionneur analogique CPJ. . . . 31
3.19 Convertisseur MCC 128 [11] . . . 32
3.20 Précision de la chaîne de mesure de force . . . 33
3.21 Schéma électrique global du démonstrateur . . . 34
3.22 Définition des signaux du capteur S9M - Figure tiré de [12] . . . 35

4.1 Fonctionnement d'un topic sous ROS 2 - Figure inspirée de [13] . . . 36
4.2 Fonctionnement d'un service sous ROS 2 - Figure inspirée de [13] . . . 37
4.3 Fonctionnement d'un mécanisme d'action sous ROS 2 - Figure inspirée de [13] 37

4.4 Architecture ROS2 du banc de test . . . 38
4.5 Vérification du suivie de la vitesse en temps réel . . . 40
4.6 Conception du nœud Robotran. . . . 40
4.7 Distribution des temps de réponse du service entre le noeud Robotran et actionneur en fonction du noyau Linux . . . 41
4.8 Procédure de retour à l'origine, le signal HSW est celui du capteur . . . 42
4.9 Capteur SICK MHT15-P2247 . . . 42
4.10 Montage du capteur de retour à l'origine (2) devant le bras de levier (1) . . . 42
4.11 Transmetteur de force étudié avec, en rouge, les composants dont la masse est ignorée lors de la mise à zéro du capteur de force . . . 43

5.1 Éléments mécaniques dont le poids est annulé par le conditionneur (propre poids, fixations et guidage externe). . . . 45
5.2 Application d'une masse de 1 kg sur le capteur de force . . . 46
5.3 Force du ressort 2 en fonction de la distance et calcul de la raideur. . . . 46
5.4 Distribution du bruit de la chaîne d'instrumentation en statique (comparaison des modes d'acquisition) . . . 48
5.5 Distribution du bruit d'une mesure statique, centrée sur sa moyenne respective, avec et sans filtrage analogique du conditionneur. . . . 49
5.6 Évolution de la déformation mesurée du capteur et de la structure en fonction de la force appliquée en compression. . . . 50
5.7 Évolution de la déformation mesurée du capteur et de la structure en fonction de la force appliquée en traction. . . . 50
5.10 Longueurs caractérisant la fonction de transfert géométrique . . . 50
5.8 Illustration de la déformation de la plaque. . . . 51
5.9 Vue globale du démonstrateur. . . . 51
5.11 Position du transmetteur de force par rapport à la position du vérin . . . 51
5.12 Erreur de position maximale du transmetteur de force par rapport à l'erreur d'installation du capteur de retour à l'origine pour les 3 positions possibles du transmetteur de force . . . 52
5.13 Temps de fonctionnement avant le déclenchement de la sécurité thermique en fonction du taux de charge du moteur [7] . . . 53
5.14 Force du ressort 1 en fonction de la distance et calcul de la raideur. . . . 54
5.15 Force du ressort 2 en fonction de la distance et calcul de la raideur. . . . 54
5.16 Systèmes masse ressort considéré . . . 54
5.17 Simulation d'un système masse-ressort-amortisseur régi par la gravité. . . . 54
5.18 Résultat de la simulation avec une amplitude d'environ 15 mm, le ressort étant compressé lorsque sa position diminue . . . 55
5.19 Résultat de la simulation avec une amplitude d'environ 20 mm . . . 55
5.20 Simulation HIL du système masse-ressort avec amortissement (filtre CPJ activé). 56
5.21 Zoom sur les oscillations du système (filtre CPJ activé). . . . 57
5.22 Simulation HIL du système masse-ressort avec amortissement (filtre CPJ désactivé). . . . 57
5.23 Zoom sur les oscillations du système (filtre CPJ désactivé). . . . 58
5.24 Modèle du VTT simulé [14] . . . 58
5.25 Dimensions du modèle de casse vitesse utilisé dans la simulation du VTT (en m pour la vue de côté) . . . 59
5.26 Évolution du déplacement du transmetteur de force pour les deux simulations 59
5.27 Évolution de la force appliquée au transmetteur de force pour les deux simulations 59
5.28 Mesure du délai de l'actionnement (Δt) . . . 60

A.1 Impact de la fréquence du signal sur l'amplitude de la dynamique . . . 64

C.1 Pré-prototype composé du contrôleur Technosoft (1), de l'interface série-USB (2), du servo linéaire Faulhaber (3) et du ressort testé 67

Liste des tableaux

3.1 Valeurs du cahier des charges selon le transmetteur de force et l'actionneur . . 16
3.2 Comparaison des caractéristiques des vis à billes et vis mères filetées 19
3.3 Comparaison des différentes morphologies d'actionneurs 19
3.4 Caractéristiques techniques du capteur de force [12] 24
3.5 Comparaison des capteurs à jauge de contrainte 29
3.6 Synthèse des paramètres de l'analyse dynamique des capteurs 30
3.7 Caractéristiques principales du CPJ [15] 31
3.8 Caractéristiques principales du potentiomètre linéaire HLP129 [16] 31
3.9 Quanta de quantification pour les différents capteurs 32
3.10 Caractéristiques principales du MCC 128 [11] 32

5.1 Récapitulatif des équivalences des données pour le capteur de force et le conditionneur 45
5.2 Données géométriques et tensions associées du potentiomètre HLP129. 46
5.3 Comparaison statistique de l'acquisition de l'ADC avec et sans le filtre interne du conditionneur. 48
5.4 Temps de déclenchement de la sécurité thermique mesuré et fourni par le fabricant à 150 et 200% de charge. 53
5.5 Temps de calcul de la boucle de simulation et délai de communication HIL. . 54

Introduction

Le propre d'une simulation en ingénierie est de refléter le plus précisément possible le comportement réel de ce qui est étudié. Pour ce faire, les chercheurs ont développé des modèles mathématiques, pour toute sorte de pièces utilisées dans des systèmes mécaniques, capables de capturer le comportement d'intérêt pour l'application développée. Bien qu'un modèle analytique permette de mettre en évidence les paramètres importants à l'œuvre dans le phénomène étudié, son obtention peut s'avérer être une tâche longue de par le nombre de pièces à modéliser et fastidieuse pour les phénomènes les plus complexes. Par conséquent, très tôt dans l'histoire des simulations sur ordinateur, les chercheurs ont tenté de lier leurs simulations au réel par d'autres moyens que des modèles mathématiques.

L'une des méthodes utilisées est le Hardware-In-the-Loop (HIL). Ceci consiste à intégrer une pièce réelle à la boucle de simulation virtuelle. L'équipement testé est placé dans un instrument de test capable, en temps réel, de l'actionner et d'effectuer des mesures pour alimenter la simulation. Ce type d'expérimentation se place donc à l'intersection entre la simulation informatique et l'expérimentation réelle (Fig.1).

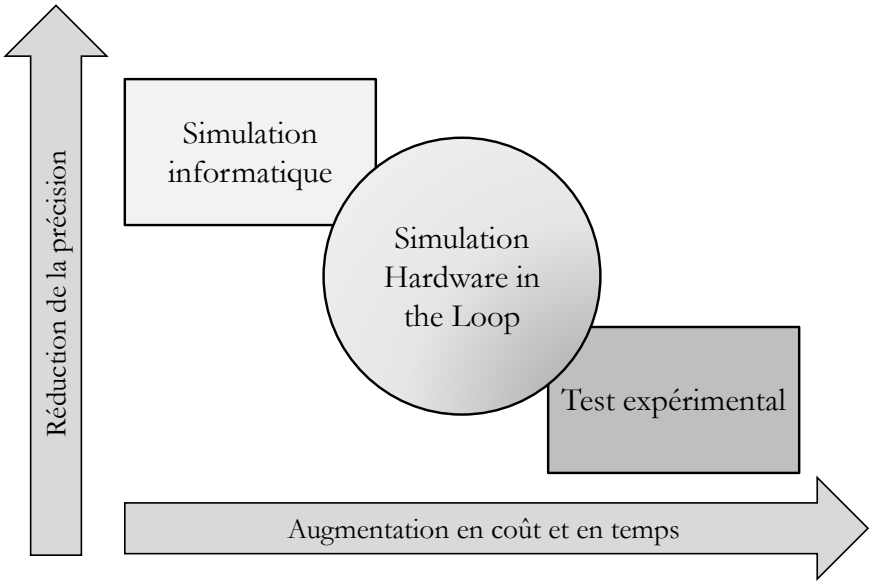


FIGURE 1 – HIL : intersection entre la simulation et l'expérimental, traduction de l'illustration de la Fig.2 de [1]

Dans le milieu académique, le HIL est intéressant pour simuler des mécanismes qu'il est difficile de modéliser. En effet, certains champs de recherche proposent des modèles simples pour comprendre des phénomènes physiques mais limités dans leur précision. Par exemple, la loi de Coulomb pour le frottement mécanique est une loi très facile à implémenter mathématiquement dans une simulation mais beaucoup plus complexe à mettre en œuvre en pratique. En effet, le coefficient de frottement doit être calibré pour chaque couple de matériaux utilisés et la loi ne prend pas en compte les comportements non linéaires.

Le HIL offre une solution intéressante à ce problème car il permet de prendre en compte tous les comportements réels du composant étudié sans devoir développer aucun modèle. La précision du résultat obtenu dépendra alors de la précision des appareils de mesure et d'actionnement utilisés. Évidemment, ce principe d'utilisation peut être retourné. Le HIL peut être également utilisé pour des composants dont le modèle a déjà été développé et consensuellement validé mais auquel des modifications ou altérations ont été intentionnellement ajoutées. Les résultats du modèle mathématique de la pièce d'origine peuvent donc être comparés aux résultats obtenus avec le test HIL de la pièce modifiée.

Dans l'industrie, le HIL est déjà massivement utilisé pour faciliter la validation d'un design. Avec le HIL, il n'est en effet plus nécessaire de fabriquer le système complet dans lequel la pièce sera installée. Lors de l'étape de design, la machine développée n'est, par définition, pas encore produite et n'est par conséquent pas disponible pour effectuer des tests en situation réelle sur une pièce en particulier. Avec le HIL en revanche, le système peut être simulé dans une simulation multi-corps à partir d'un modèle plus simple disponible à l'étape de design et seule la pièce à tester doit être fabriquée. La pièce peut alors être testée en situation quasi-réelle à moindre coût.

La conception complète de ce projet constituant un travail d'envergure, elle a été scindée en deux parties distinctes : la conception électromécanique, développée par Guillaume Hisette et Sylvain Bruyr dans le présent document, et la conception mécanique, présentée dans le mémoire de Bastien Debroux [17]. Le projet global étant commun aux deux travaux, les chapitres d'introduction, d'état de l'art et de clarification du projet sont rigoureusement identiques dans les deux manuscrits.

Chapitre 1

Contexte du HIL

Ce chapitre vise à donner une vision d'ensemble du *Hardware-in-the-Loop*. Dans un premier temps, une revue de littérature décrit les utilisations du HIL dans l'industrie. Elle décrit également deux implémentations du HIL dans le domaine académique. Dans un second temps, ce chapitre décrit le modèle mathématique utilisé comme moteur de simulation pour le démonstrateur développé au cours de ce mémoire.

1.1 Revue de littérature

1.1.1 Applications du HIL dans l'industrie

Dans l'industrie, les ingénieurs tirent parti des nombreux avantages du HIL au quotidien. Cette section se base sur les conclusions de [1] sur l'utilité du HIL dans l'industrie. En effet, cet article liste les avantages du HIL traduits ci-dessous dans le design et le développement de systèmes :

1. Test et développement d'algorithmes et de logiciels en présence de conditions plus réalistes
2. Test et développement de sous-systèmes d'électronique embarqué
3. Étude précise des opérations d'actionneurs et de capteurs
4. Étude de logiciels et matériel embarqué et étude de leur effets mutuels
5. Vérification et validation des opérations de logiciels ou matériel
6. Évaluation des performances d'un système dans des conditions proches des conditions réelles, dans un environnement de laboratoire
7. S'exercer dans des conditions réelles pour l'utilisateur et détecter des sources d'erreurs
8. Évaluer la tolérance aux erreurs d'un système par injection de différentes erreurs dans chaque sous-système
9. Augmentation considérable de la fiabilité et de la probabilité de succès
10. Réduction de la probabilité d'un dommage financier destructif ou d'un dommage de santé
11. Réduction du coût global du projet par détection des modes de défaillance
12. Diminution du temps écoulé entre le design et la production

1.1.1.1 HIL dans le processus de design

Aujourd'hui, le HIL est largement utilisé pour ses avantages dans le domaine du design. Selon [1], les principaux facteurs qui donne au HIL son intérêt dans le design de systèmes sont :

— La possibilité d'effectuer des tests destructifs ou coûteux du système réel
— L'existence d'éléments hautement non-linéaires et difficiles à modéliser précisément, comme des actionneurs ou des capteurs
— La complexité des logiciels de contrôle, l'interaction entre un contrôleur implémenté et le matériel.

Dans le secteur automobile, le processus de design en V est souvent utilisé. Dans la branche de droite du V, les concepteurs doivent valider par étapes successives les pièces conçues. Le HIL permet de faire des tests en situation quasi-réelle. Sans installation HIL dans les laboratoires de l'entreprise, si une nouvelle suspension a été développée et doit être testée en conditions réelles, ils faut organiser un test sur circuit, adapter un véhicule à la nouvelle suspension, trouver un moyen de récupérer les données d'intérêt, ... Avec le HIL, la suspension est placée dans le banc de test et la voiture est simulée. Toutes les données d'intérêt, nécessaires à la validation de la pièce, sont disponibles grâce à la simulation multi-corps, sans capteurs additionnels. Tout cela sans devoir ni fabriquer un véhicule ni en adapter un existant. Des applications similaires sont trouvées dans les secteurs de l'aérospatial, de l'énergie, de la robotique et de la marine [1].

1.1.1.2 HIL hors du scope du mémoire

Les points 1, 2 et 4 concernent des bancs d'essai HIL très distincts de celui développé ici. Ils testent des systèmes logiciels et électroniques (puces et PCB). Les caractéristiques du système de test sont alors très différentes d'un banc conçu pour tester une pièce mécanique.

Le point 7 concerne des bancs dans la catégorie des *Pilot-in-the-Loop*. D'un point de vue historique, les premières applications du HIL étaient destinées à des simulateurs. Ces implémentations ne correspondent pas réellement à l'application développée dans ce mémoire car les données mesurées sont produites par un humain et non une pièce mécanique. La technologie est toutefois similaire (simulation - action de l'environnement - prise de mesure - réaction de la simulation). Les premières applications ont vu le jour dans le secteur aéronautique et spatial, notamment pour les simulations pour l'entraînement des pilotes à l'atterrissage lunaire dans le programme Apollo de la NASA. Dans le secteur automobile, les simulateurs automobiles sont autant utilisés par les professionnels de sport moteur que par les particuliers au travers du *simracing*.

1.1.2 Implémentations du HIL dans le domaine académique

Deux applications qui entrent dans le scope de ce mémoire sont décrites ci-dessous. Ces deux papiers de recherche ont servi de base de réflexion.

1.1.2.1 Development of a Biomechanical Multibody Model for the Hardware-in-the-Loop Simulation of Total Hip Endoprostheses

La première application est le développement d'un modèle multi-corps pour de la simulation HIL d'une prothèse de hanche [18]. Comme dans notre application, un lien entre un logiciel de simulation multi-corps et un setup expérimental est établi.

1.1.2.2 Test bench design for motorcycle suspensions HIL test

La seconde application est la conception d'un banc de test pour étudier le comportement d'une suspension arrière de moto grâce à une simulation HIL [19]. A nouveau, un lien entre les mesures prises sur un système physique et une simulation informatique est établi.

1.2 Moteur de simulation : formalisme multi-corps

Comme son nom l'indique, une simulation HIL repose en partie sur la modélisation numérique d'un système. Dans le cadre de ce mémoire, une simulation multi-corps est mise en œuvre afin d'alimenter le démonstrateur en données physiques.

Le logiciel utilisé est Robotran, développé par le groupe de recherche en Mécatronique, Énergie Électrique et Systèmes Dynamiques (MEED) de l'Université catholique de Louvain (UCLouvain). Robotran constitue aujourd'hui un environnement de modélisation complet, capable de réaliser diverses analyses multi-corps. Il permet notamment d'effectuer des analyses d'équilibre, des analyses modales, de la dynamique directe et inverse, ainsi que de la simulation en temps réel.

1.2.1 Architecture de Robotran

L'environnement Robotran s'articule autour de trois modules complémentaires, illustrés à la figure 1.1 :

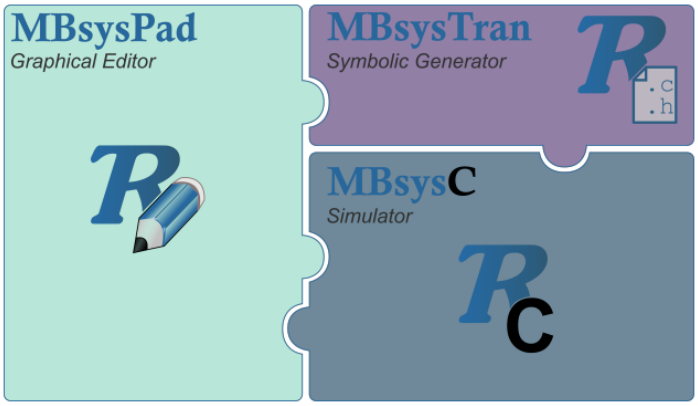


FIGURE 1.1 – Environnement Robotran - Figure tiré de [2]

— **MBsysPad** : Il s'agit de l'éditeur graphique permettant de décrire la topologie du système. C'est via cette interface que l'utilisateur définit la géométrie, les masses, centre de masse et les inerties de tous les corps. Les corps sont eux reliés par une articulation à un degré de liberté de type translation (prismatique) ou rotation (révolute). Les mouvements relatifs plus complexes sont modélisés par une succession de ces articulations élémentaires, séparées par des corps fictifs sans dimension ni masse.
— **MBsysTran** : Il s'agit du générateur symbolique qui traduit, à partir de la description du système, la forme analytique des équations cinématiques et dynamiques du modèle multi-corps. Ces équations, générées en une seule étape, sont ensuite exploitées par les modules de simulation. Cette approche analytique permet d'obtenir des simulations nettement plus rapides qu'avec des méthodes purement numériques.
— **MBsysC** : Il s'agit de l'environnement de simulation. L'utilisateur garde un contrôle total sur la simulation en implémentant directement ses propres lois constitutives (forces externes, couples, forces articulaires, déplacements) agissant dans les corps et articulations.

1.2.2 Formalisme multi-corps

Les systèmes multi-corps se divisent en deux familles distinctes : les systèmes en arbre et les systèmes comportant des boucles cinématiques. L'équation générale 1.1 régit la dynamique directe de ces systèmes sous forme matricielle :

$$M(q)\ddot{q} + c(q, \dot{q}, \delta, frc, trq, g) = Q(q, \dot{q}) + J(q)^t\lambda \tag{1.1}$$

Où :

— $M \in \mathbb{R}^{n\times n}$ est la matrice de masse généralisée symétrique du système

— $q, \dot{q}, \ddot{q} \in \mathbb{R}^{n\times 1}$ désignent respectivement les coordonnées, vitesses et accélérations généralisées relatives aux articulations.

— $c \in \mathbb{R}^{n\times 1}$ est le vecteur dynamique non linéaire contenant les termes gyroscopiques, centripètes et gravitationnels, ainsi que la contribution des forces (frc) et couples (trq) externes.

— $\delta \in \mathbb{R}^{10n\times 1}$ regroupe les paramètres dynamiques du système (masses, positions des centres de masse et matrice d'inertie).

— $Q \in \mathbb{R}^{n\times 1}$ contient les n forces et couples généralisés agissant dans les articulations.

— $J \in \mathbb{R}^{m\times n}$ est la matrice jacobienne des contraintes.

— $\lambda \in \mathbb{R}^{m\times 1}$ représente le multiplicateur de Lagrange associé aux contraintes.

Dans le cas d'un système en arbre, le terme de force lié aux contraintes est nul ($J(q)^t\lambda = 0$).

Afin de fermer complètement les boucles du système multi-corps, il convient d'ajouter les contraintes géométriques en position h, ainsi que leurs dérivées temporelles en vitesse et en accélération :

$$h(q) = 0 \tag{1.2}$$

$$\dot{h}(q, \dot{q}) = J(q)\dot{q} = 0 \tag{1.3}$$

$$\ddot{h}(q, \dot{q}, \ddot{q}) = J(q)\ddot{q} + \dot{J}(q)\dot{q} = 0 \tag{1.4}$$

Ce système d'équations différentielles-algébriques (DAE) peut être transformé en équations différentielles ordinaires (ODE) afin d'en permettre l'intégration numérique. Cette réduction repose sur une procédure mathématique de partitionnement consistant à séparer les coordonnées généralisées (q) en variables indépendantes (q_u) et dépendantes (q_v). L'équation 1.1 est ensuite réduite par le biais de permutations matricielles et d'opérations algébriques, suivant la méthodologie détaillée dans le document de référence *Robotran basics* [2]. Cela permet d'obtenir en sortie un système dont la seule inconnue est l'accélération des variables indépendantes ($\ddot{q}_u$), afin de la transmettre à l'intégrateur numérique (Runge-Kutta, Euler implicite, Euler explicite, etc.). Un schéma bloc illustrant cette résolution est présenté à la figure 1.2.

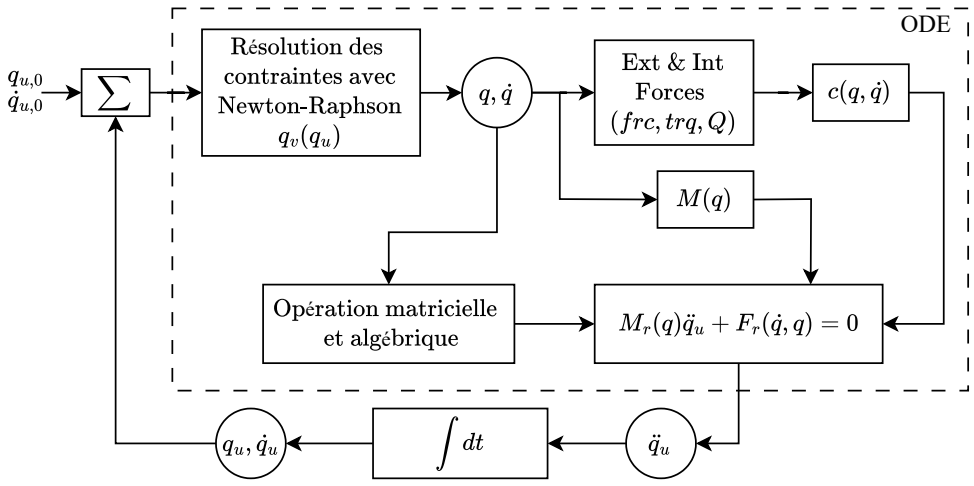


FIGURE 1.2 – Schéma-bloc du solveur Robotran - Figure inspirée et simplifiée de [2]

Modélisation des efforts et lois de comportement : Il est également possible d'imposer le mouvement de certaines coordonnées. Celles-ci sont appelées variables commandées (q_c). Bien que ces variables interviennent dans le calcul de la matrice de masse et du vecteur dynamique c, elles ne sont pas traitées par l'intégrateur numérique puisque leur évolution temporelle est connue et imposée par l'utilisateur.

Trois types de forces peuvent être intégrés au modèle multi-corps qui sont calculés sur base de la cinématique du mouvement $q, \dot{q}$:

- **Forces et couples articulaires :** Directement intégrés au vecteur Q, ils modélisent les efforts internes aux articulations, tels que des forces produites par du frottement, un ressort ou un amortisseur.
- **Efforts externes :** Ils représentent les interactions entre un corps et son environnement (forces de contact pneu-sol, résistance aérodynamique, ...). Ces efforts sont regroupés dans une force résultante frc et un couple résultant trq appliqués au centre de masse du corps considéré.
- **Liens (Links) :** Utilisés pour modéliser des transmetteurs d'efforts entre deux points (par exemple une suspension arrière de VTT), ils ne sont applicables que si la masse de l'élément de liaison est négligeable. Le solveur Robotran les traite comme des forces d'action-réaction externes s'exerçant entre deux corps.

1.2.3 Simulation en temps réel

Le logiciel Robotran offre également la possibilité d'exécuter des simulations en temps réel. La figure 1.3 illustre de manière conceptuelle la répartition temporelle au sein d'un pas de simulation. Le temps total d'une itération se décompose en trois phases distinctes :

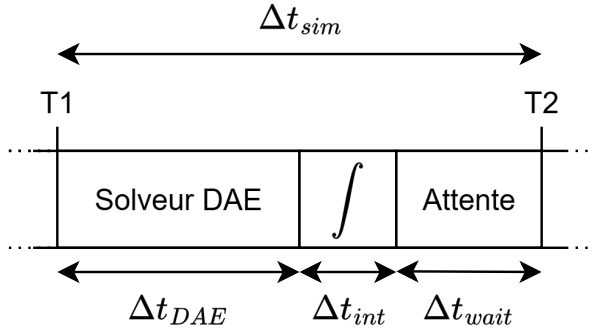


FIGURE 1.3 – Répartition temporelle d'un pas de calcul pour une simulation en temps réel

- **Le temps d'évaluation du système DAE (Δt_{DAE})** : Cette phase correspond au temps de calcul nécessaire pour évaluer la dynamique du système. À titre d'exemple, l'utilisation d'une méthode d'intégration de type Runge-Kutta d'ordre 4 (RK4) requiert quatre appels au système pour évaluer les accélérations $\ddot{q}_u$ pour un pas de temps. Le terme Δt_{DAE} représente alors la somme des temps d'exécution de ces appels successifs. L'utilisation d'un générateur symbolique permet de réduire ce temps de calcul.
- **Le temps d'intégration (Δt_{int})** : Il s'agit du temps processeur requis pour effectuer les opérations mathématiques inhérentes au schéma d'intégration numérique (tel que le calcul des sommes pondérées, la division du pas de temps, etc.).
- **Le temps d'attente (Δt_{wait})** : Afin de garantir le strict respect du temps réel, le système effectue une période de temporisation. Cette attente permet d'ajuster la durée totale du pas de calcul afin qu'elle corresponde exactement à l'écoulement du temps physique, assurant ainsi la condition de synchronisation $\Delta t_{sim} = \Delta t_{réel}$.

Une simulation est considérée en temps réel uniquement si $\Delta t_{DAE} + \Delta t_{int} < \Delta t_{sim}$.

1.2.4 Simulation Hardware in the loop

Dans le cas d'une simulation HIL, la simulation envoie les données de position et de vitesse au démonstrateur, qui transmet en retour l'effort associé afin de permettre le calcul du vecteur c. Ce processus est illustré à la figure 1.4. Cette interaction introduit par ailleurs un délai temporel Δt_{HIL} englobant le temps de communication avec le démonstrateur, le contrôle et l'acquisition de données. Dès lors, il devient nécessaire de respecter la condition $\Delta t_{DAE} + \Delta t_{int} + \Delta t_{HIL} < \Delta t_{sim}$ afin de garantir le respect du temps réel.

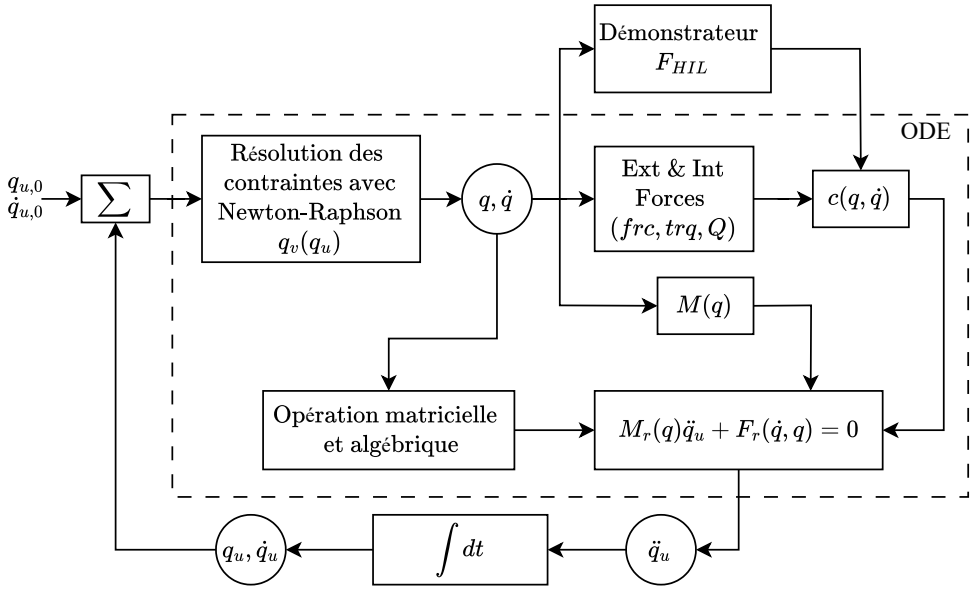


FIGURE 1.4 – Schéma-bloc du solveur Robotran avec du HIL - Figure inspirée et simplifiée de [3]

Chapitre 2

Cahier des Charges

2.1 Limites d'utilisation du démonstrateur HIL

Le démonstrateur développé ici sera utilisé comme *Proof of Concept* dans la thèse d'une doctorante membre du groupe de recherche MEED. Le projet se limitera par conséquent pour cette année à un actionnement d'une pièce mécanique selon un degré de liberté en translation. Dans ce qui suit, il sera fait référence à la pièce testée par le nom générique de transmetteur de force (TF). Ce nom englobe toute sorte de pièces que l'on peut trouver dans des mécanismes. Les exemples les plus courants sont les ressorts et amortisseurs. Les types de TF que le banc doit pouvoir tester sont décrits dans le cahier des charges. Cependant, il a été décidé de retenir l'étude d'une suspension arrière de vélo comme transmetteur de force de référence pour ce mémoire.

2.2 Graphe des fonctions

Le démonstrateur est utilisé en parallèle d'une simulation *Robotran*. Son utilisation se découpe en deux séquences. La première est représentée horizontalement dans le graphe des fonctions et concerne la mise en place du test (Fig.2.1). Le transmetteur de force est monté dans la machine et maintenu en place dans sa position neutre. Ensuite la phase de test peut démarrer. Après le test, le démonstrateur doit revenir à l'arrêt en position neutre.

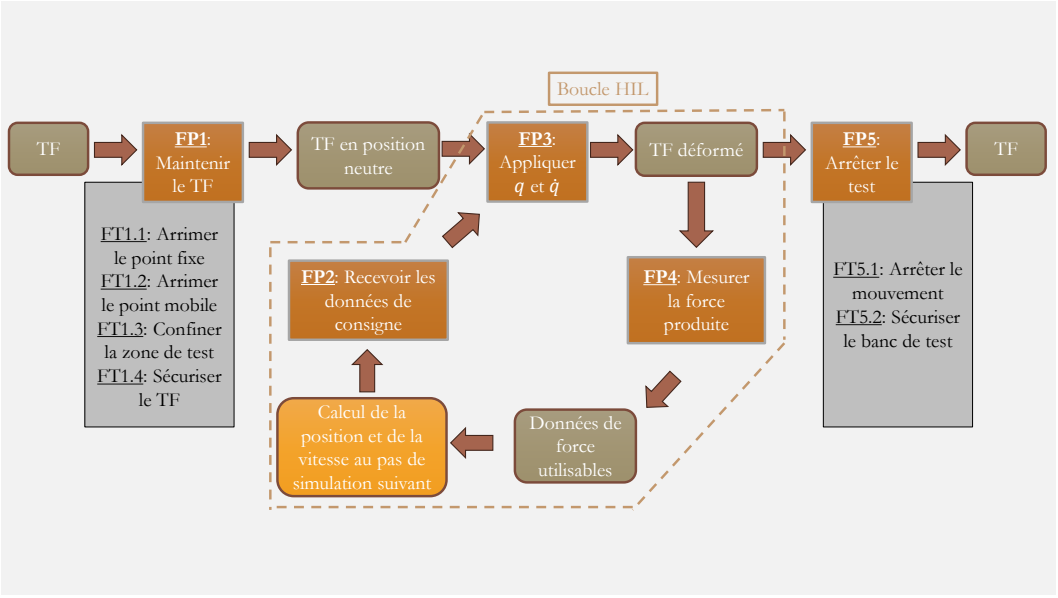


FIGURE 2.1 – Graphe des fonctions - Séquence de mise en place

La phase de test (Fig.2.2) est une boucle qui commence avec le transmetteur de force en position neutre. Ensuite, la simulation multi-corps donne la position à laquelle la machine doit amener le transmetteur. Une fois cette position atteinte, la force produite par le transmetteur dans cette position est mesurée et les données transformées en une forme utilisable par la simulation. La boucle pour le premier pas de temps est alors complétée. Avec la mesure de la force, la simulation calcule la position à appliquer sur le TF au pas de temps suivant. Ces données sont transmises au banc de test et la boucle continue pour chaque pas de temps jusqu'à la fin de la simulation. Après cela, la séquence principale reprend pour terminer le test.

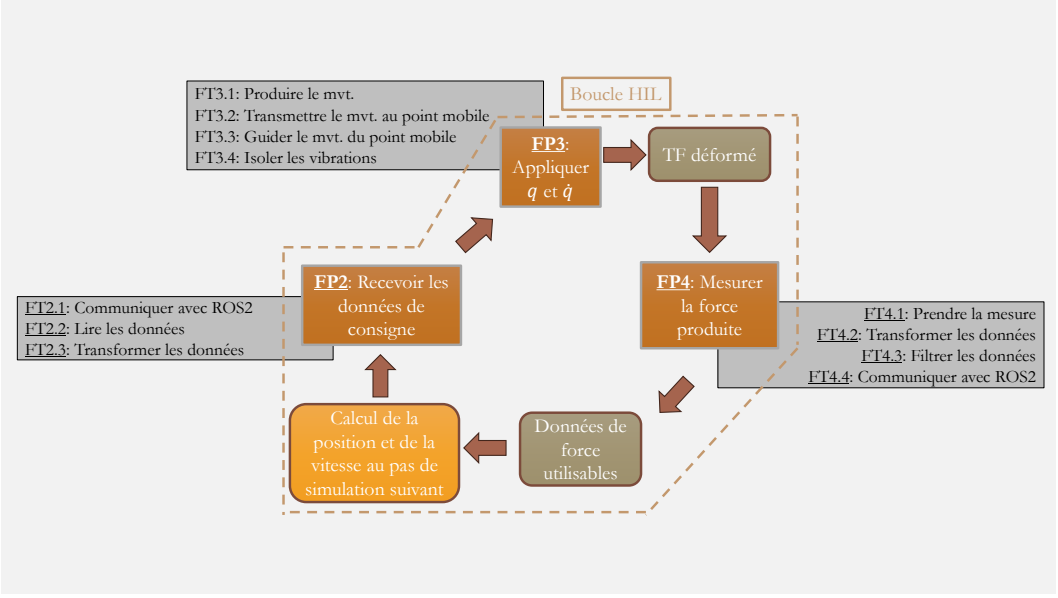


FIGURE 2.2 – Graphe des fonctions - Boucle de test

2.3 Cahier des charges

Cahier des charges

Auteur : Bastien Debroux
Promoteurs : Paul Fisette et Nicolas Docquier
Assistante : Marine Boxus

Date : 02-05-26
Version : 3.0
Modifications apportées : Version finale

Contexte : Conception d'un démonstrateur électromécanique pour de la simulation HIL. Le logiciel Robotran, développé par l'équipe de recherche en ingénierie mécanique de l'école polytechnique de Louvain, est un logiciel de simulation dynamique d'un système multi-corps. Le but de la thèse de Marine Boxus est d'inclure le comportement mécanique réel d'un transmetteur de force à Robotran au moyen d'un banc de test dans lequel le transmetteur de force est substitué à la fonction décrivant son équation constitutive dans le modèle mathématique simulé. La conception de ce banc de test est le sujet du mémoire présent.

Groupe 1 : Bastien Debroux (conception mécanique)

Groupe 2 : Guillaume Hisette et Sylvain Bruyr (conception électromécanique)

Date	Origine	Groupe à charge	Fonctions Principales (FP)
13-10-25		1 et 2	FP1 : Appliquer une position et une vitesse au point d'ancrage d'une extrémité du transmetteur de force
13-10-25		1 et 2	FP2 : Mesurer la force produite par le transmetteur de force
13-10-25		2	FP3 : Extraire en temps réel de la simulation multi-corps les données de consigne (position et vitesse)
13-10-25		2	FP4 : Transmettre en temps réel à la simulation multi-corps les forces mesurées
13-10-25		1	FP5 : Maintenir et guider le transmetteur de force
			Exigences Fonctionnelles (EF), critères et valeurs
23-09-25	Restriction initiale		EF 1.1 : Appliquer le déplacement et la vitesse selon l'unique degré de liberté du transmetteur dans l'axe de celui-ci.

07-10-25	Restriction [1][1]		EF 1.2 : Fréquence d'excitation max. = 20 Hz
13-10-25	Objectif de performance[2]		EF 1.2 : Précision déplacement = 0.1 mm
13-10-25	Restriction [1]		EF 1.3 : Course max. = 75 mm
07-10-25	Restriction [1]		EF 1.4 : Temps de simulation max. = 15 min
13-10-25	Restriction [1]		EF 2.1 : Force max. = 3000 N
13-10-25	Exigence de précision dynamique[3]		EF 2.2 : Mesure de la force en moins de 0.1 ms
13-10-25	Imposé par les promoteurs		EF 3.1 : Les données de consigne sont calculées par le logiciel Robotran et manipulées via ROS
13-10-25	Imposé par les promoteurs		EF 4.1 : La force mesurée est prise en compte par Robotran et les données sont manipulées via ROS
13-10-25	Restriction [1]		EF 5.1 : Accepter des transmetteurs de force de type assemblage ressort + amortisseur
13-10-25	Restriction [1]		EF 5.2 : Accepter des transmetteurs mesurant, en pleine extension, au maximum 250 mm et au minimum 100 mm
02-05-26			EF 5.2 : Le transmetteur de force est composé de trois parties, représentées à la figure 1. Au centre, le moyen de produire la force est considéré comme une boite noire (ressort, amortisseur, guidage, ...). À chaque extrémité, un tenon d'arrimage percé d'un alésage permet la fixation du transmetteur de force. La force produite est appliquée aux centres des alésages le long de l'axe les reliant.
			Contraintes (C)
07-10-25	Voir contexte		C1 : Utilisation aisée à des fins de recherche
07-10-25	Imposé par les promoteurs		C2 : Stockage et utilisation dans le laboratoire Stevin a047 de la plateforme CREDEM
07-10-25			C3 : Garantir la sécurité de l'utilisateur
07-10-25			C4 : Garantir l'intégrité fonctionnelle du transmetteur de force
07-10-25			C5 : Prendre en compte des possibilités d'évolution
13-10-25			C6 : Assurer un cycle de vie durable
13-10-25			C7 : Maintenance aisée
13-10-25			C8 : Entrer dans le budget alloué au projet

[1] Restriction [1] : basée sur le cas d'étude type supposé du banc de test : test d'une suspension de vélo. Le tableau 1 reprend une gamme de produit type de suspension arrière de vélo. La fréquence d'actionnement équivalente correspond aux impuretés sur la route (bosses, graviers, pavés, ...).

[2] Imposé par les promoteurs

[3] Si le temps de mesure est trop important, la mesure sera faussée et ne représentera plus la valeur dynamique réelle.

			Exigences de Contrainte (EC), critères et valeurs
07-10-25	Imposé par les promoteurs		EC 1.1 : Utilisation exclusivement réservée à une chercheuse de l'équipe
07-10-25	Permettre de réaliser une batterie de test sans une trop longue interruption		EC 1.2 : Temps de cycle faible (< 5min entre deux tests)
07-10-25			EC 1.3 : Changement rapide et aisé du transmetteur de force en moins de 30min
13-10-25			EC 2.1 : Reste monté en permanence sur le banc de test du laboratoire
13-10-25			EC 3.1 : Stopper automatiquement tout mouvement des actionneurs lorsque l'utilisateur est exposé au danger par manipulation du dispositif
18-10-25			EC 3.2 : Intégrer une fonction d'arrêt d'urgence manuelle
13-10-25			EC 3.3 : Respecter les normes de sécurité de l'université conformément au document Procédure d'implantation d'un prototype dans un bâtiment UCL[4]
13-10-25			EC 4.1 : Empêcher toute déformation non souhaitée du transmetteur de force par des vérifications redondantes
13-10-25			EC 5.1 : Pouvoir installer ou monter deux démonstrateurs identiques pour tester plusieurs transmetteurs en simultané
13-10-25			EC 6.1 : Rendre le dispositif démontable et le démantèlement possible
23-09-25			EC 6.2 : Inclure autant que possible le matériel déjà à disposition
13-10-25			EC 7.1 : Rendre accessible au démontage toute pièce critique ou nécessitant une action de maintenance
13-10-25			EC 8.1 : Budget disponible pour l' ensemble du mémoire = 7000 €

[4] https://www.uclouvain.be/fr/myuclouvain/serp/les-prototypes-et-autres-installations-experimentales

Figure 1

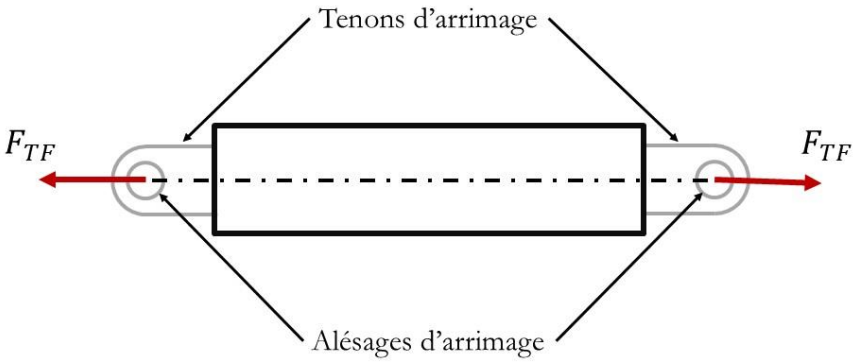


Tableau 1

Marque	Modèle	Longueur [mm]	Course [mm]	Type de ressort
RockShox	Vivid Ultimate RC2T	210 230 250	55 57.5/60/62.5/65 70/72.5/75	Pneumatique
Fox Racing Shox	Float X Evol LV 2Pos Factory	190 210 230	45 50/52.5/55 57.5/60/62.5/65	Pneumatique
Fox Racing Shox	DHX2 2Pos Factory	210 230	50/55 60/65	Ressort hélicoïdal
Öhlins	TTX 22M Coil	210 230 250	50/55 57.5/60/65 70/75	Ressort hélicoïdal
Marzocchi	Bomber CR Trunnion	205	60	Ressort hélicoïdal
Riverside	Amortisseur arrière à ressort pour 28''	234	52	Ressort hélicoïdal (53.8 N/m)
Rockrider	UFIT (900g)	190	51	Ressort hélicoïdal
Riverside	Amortisseur arrière ressort btwin7	150	31	Ressort hélicoïdal (750 Lbs/In)

Chapitre 3

Conception du démonstrateur

L'objectif de ce chapitre est de décrire et d'expliquer la phase de conception du démonstrateur pour la partie mécatronique. Il est structuré autour de trois axes principaux : les systèmes d'actionnement, la chaîne d'instrumentation et enfin le réseau électrique global. La conception purement mécanique est, quant à elle, détaillée dans le mémoire de Bastien Debroux [17].

Pour offrir une première vision d'ensemble, la réalisation globale du démonstrateur est illustrée sur la figure 3.1.

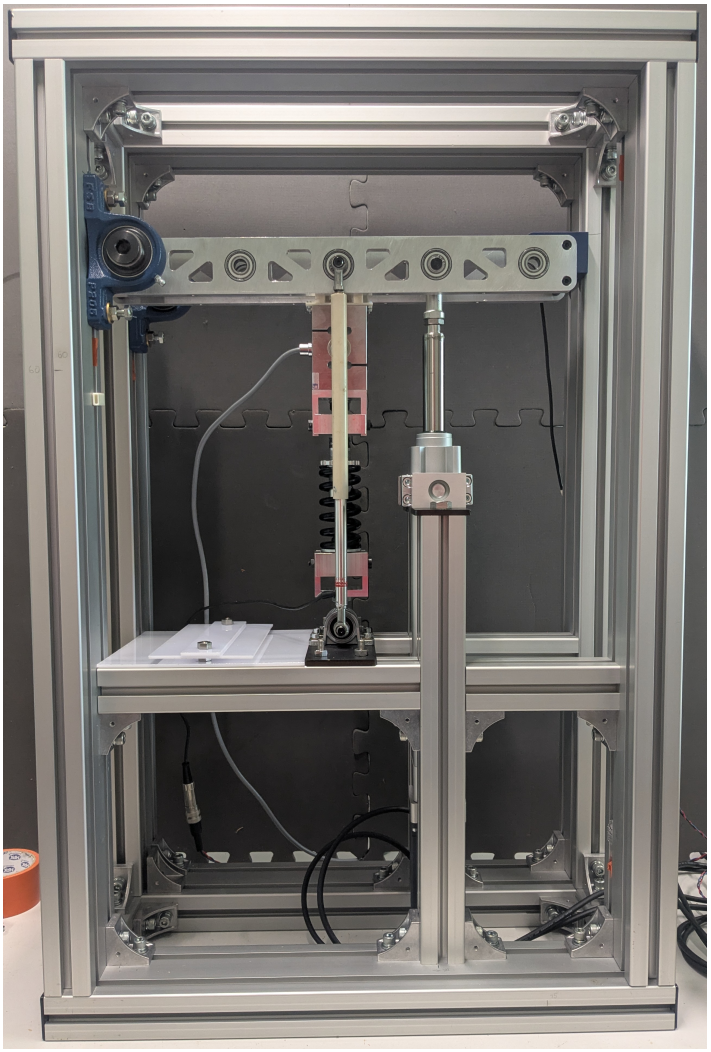


FIGURE 3.1 – Vue globale du banc de test

3.1 Actionnement

Afin d'appliquer un déplacement au transmetteur de force étudié, une technique d'actuation doit être sélectionnée. Elle doit permettre d'appliquer une distance précise entre les extrémités du bloc **A** du schéma simplifié du prototype représenté sur la Figure 3.2 avec une précision de $100\mu m$ et à des vitesses élevées.
Le banc d'essai peut accepter un transmetteur de force à 3 positions différentes (blocs 1 à 3 sur la Figure 3.2) permettant de sacrifier de la course et de la vitesse pour une force maximale plus importante et inversement. Néanmoins, les objectifs du cahier des charges étant exprimés pour le transmetteur de force étudié à l'emplacement 2, il existe une relation géométrique permettant de connaître les contraintes sur l'actionneur en liant le déplacement du transmetteur de force à celui de l'actionneur. Plus de détails relatifs à cette relation sont disponibles dans la section 5.3. Le Tableau 3.1 contient les contraintes sur l'actionneur.

	TF (2)	**Actionneur (A)**
Course [mm]	75	110
Force [N]	3000	2040
Vitesse [m/s]	0.2	0.29

TABLEAU 3.1 – Valeurs du cahier des charges selon le transmetteur de force et l'actionneur

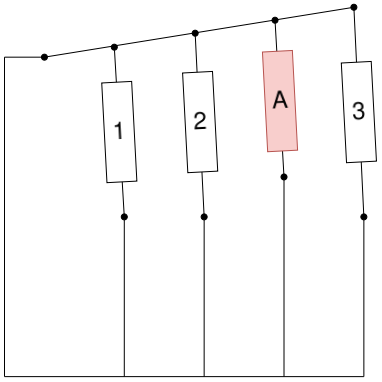


FIGURE 3.2 – Schéma simplifié du banc d'essai

3.1.1 Choix de la morphologie d'actionnement

L'objectif de cette section est de détailler les différentes techniques d'actionnement considérées lors de la conception du démonstrateur dans le but d'identifier la solution la plus appropriée. Des critères supplémentaires à ceux imposés par le cahier des charges doivent être considérés pour des raisons de performances, mais également pour garantir la conception d'un prototype fonctionnel dans le délai et avec les moyens disponible. Ces critères sont les suivants :

- Le jeu mécanique de l'actionneur
- L'encombrement de l'actionneur
- Le coût de la solution
- Le délai de livraison
- La prise en main nécessaire
- La maintenance nécessaire
- La longévité du dispositif

Chaque morphologie est analysée en détail dans le but d'en identifier les avantages et inconvénients. Un dimensionnement sommaire de chaque actionneur a par ailleurs été effectué afin de pouvoir chiffrer le coût de chaque solution. La solution choisie sera, quant à elle, dimensionnée de manière rigoureuse.
Les morphologies considérées sont les suivantes :

1. Actionneur servo-hydraulique
2. Servomoteur linéaire
3. Actionneur linéaire à vis

3.1.1.1 Actionneur servo-hydraulique

Cet actionneur génère une force via un vérin hydraulique. Un moteur électrique entraine une pompe qui met de l'huile sous pression, le vérin est alimenté via un distributeur proportionnel (ou plusieurs vannes proportionnelles pour les systèmes les plus avancés) permettant de moduler le débit d'huile entrant et sortant du vérin à tout instant. Un capteur de position mesure le déplacement de la tige du vérin et permet un contrôle en boucle fermée de la position du vérin. La Figure 3.3 contient le schéma hydraulique d'un actionneur de ce type.

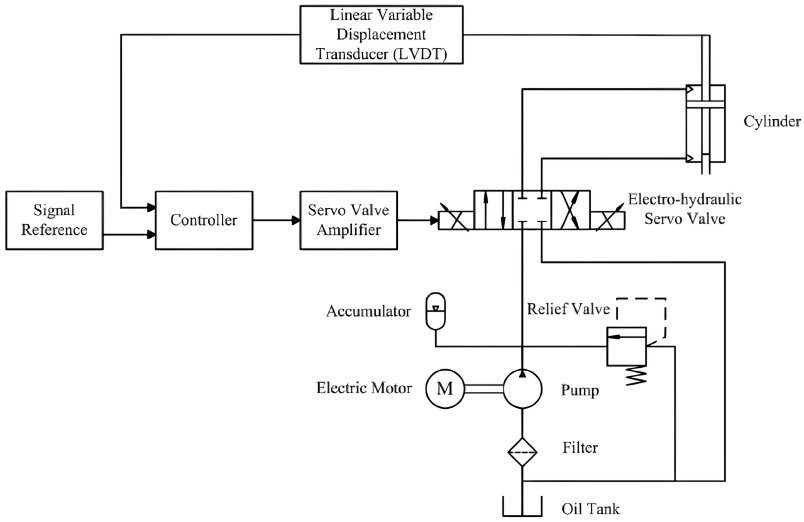


FIGURE 3.3 – Schéma hydraulique d'un actionneur linéaire servo-hydraulique à distributeur proportionnel [4]

Certains modèles fonctionnent d'une manière légèrement différente en utilisant une pompe volumétrique entrainée par un servomoteur. L'utilisation d'une pompe volumétrique permet de relier la quantité d'huile déplacée à la rotation du moteur, ce qui rend l'utilisation d'un capteur de position facultative.

Ces actionneurs ont pour avantage de fournir des forces très importantes avec un haut niveau de dynamisme. En revanche, ce sont des dispositifs très coûteux et dont la prise en main n'est pas aisée sans formation préalable.

Dimensionnement Pour le critère de vitesse de $0.29m/s$, la vitesse maximale du vérin (v_{max} [m/s]) est généralement fournie dans sa documentation, dans certain cas, il est cependant nécessaire de la calculer sur base du débit maximal de la pompe (Q_{max} $[m^3/s]$) et de la surface du piston du vérin (A $[m^2]$) :

$$v_{max} = \frac{Q_{max}}{A} \tag{3.1}$$

Il est important de noter que la surface du piston (A $[m^2]$) est différente suivant la direction du déplacement à cause de la tige du vérin.

Pour calculer la force nécessaire, il est indispensable de prendre en considération la masse du piston lors des phases d'accélération :

$$F_{tot} = F_{eff} + m \cdot a_{max} \tag{3.2}$$

Où :

— F_{tot} : la force totale du vérin nécessaire [N]

— F_{eff} : la force effective appliquée au bout du vérin et transmise au TF [N]
— m : la masse du piston [kg]
— a_{max} : l'accélération linéaire maximale du piston $[m/s^2]$

La force maximale du vérin (F_{max} [N]) est généralement fournie dans la documentation mais peut également être calculée sur base de la pression de la pompe (p [Pa]) et de la surface du vérin (A $[m^2]$) tout en notant, comme précédament, que la surface varie en fonction de la direction d'application de la force :

$$F_{max} = p \cdot A \tag{3.3}$$

3.1.1.2 Servomoteur linéaire

Les servomoteurs linéaires convertissent l'énergie électrique directement en un mouvement linéaire en se passant de composants mécaniques tels que des courroies, pignons, etc. L'absence d'éléments de transmission offre d'excellentes performances en dynamique ainsi qu'une absence de jeu et un rendement excellent. Ils sont constitués d'un coulisseau contenant des aimants permanents qui coulisse dans un stator. Celui-ci contient le bobinage moteur ainsi que des capteurs de position, généralement à effet Hall, permettant le contrôle en position du coulisseau. La Figure 3.4 montre une vue en coupe d'un servomoteur linéaire cylindrique.

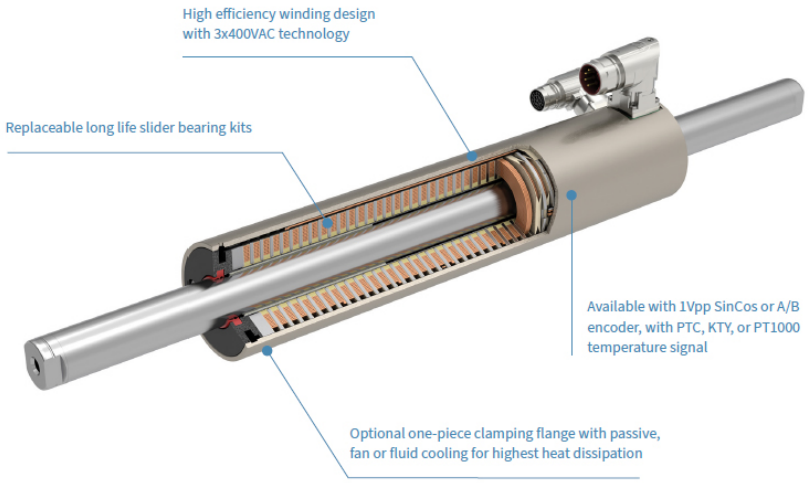


FIGURE 3.4 – Vue en coupe d'un servomoteur linéaire cylindrique [5]

Dimensionnement La vitesse maximale est donnée dans la documentation des servomoteurs considérés. Pour la force produite, comme précédemment, il faut prendre en considération la force requise pour l'accélération de la partie mobile de l'actionneur :

$$F_{tot} = F_{eff} + m \cdot a_{max} \tag{3.4}$$

Où :
— F_{tot} : la force fournie par le vérin [N]
— F_{eff} : la force effective appliquée au bout du coulisseau [N]
— m : la masse du piston [kg]
— a_{max} : l'accélération linéaire maximale du coulisseau $[m/s^2]$

3.1.1.3 Actionneur linéaire à vis

Ce type d'actionneur utilise une vis pour transformer le mouvement rotatif d'un servomoteur en un mouvement linéaire. La position angulaire du servomoteur peut donc être traduite en une position linéaire précise.

Il existe deux types de vis, la vis à billes et la vis mère filetée, le Tableau 3.2 met en évidence les principales différences entre ces deux technologies.

Caractéristique	**Vis à billes**	**Vis mère**
Jeu mécanique	$< 30\mu m$	$< 100\mu m$
Endurence	$> 10000km$	$< 1500km$
Coût	$> 1000€$	$< 600€$
Rendement mécanique	$> 90\%$	$< 70\%$

Tableau 3.2 – Comparaison des caractéristiques des vis à billes et vis mères filetées

Il existe deux manières d'implémenter un actionneur à vis, la première consiste à choisir un vérin complet qui assure le guidage ainsi que l'actionnement, la deuxième consiste à concevoir une solution sur mesure implémentant une vis ainsi que son guidage. Les deux possibilités sont étudiées dans l'annexe B.

3.1.1.4 Morphologie retenue

Le Tableau 3.4 dresse une comparaison des différentes morphologies d'actionneurs, pour chaque critère de sélection, chacune des solutions est classée sur la base d'un score de 1 à 3. La solution dont la somme des scores est la plus élevée est la mieux adaptée à notre application.

Le choix final s'est donc porté sur un actionneur à vis (voir section 3.1.1.3). La vis mère affichant un jeu mécanique plus important, ainsi qu'une durée de vie plus faible a orienté le choix vers la vis à bille. La solution consistant à concevoir un actionneur sur mesure à base d'une vis à billes ne représentant qu'un faible avantage économique tout en allongeant le délai de livraison, le choix s'est porté sur l'achat d'un vérin complet assurant le guidage et l'actionnement.

Critère	**Actionneur servo-hydraulique**	**Servomoteur linéaire**	**Actionneur à vis**
Jeu mécanique	2	2	1
Encombrement	1	1	2
Coût	1	2	3
Délai de livraison	1	1	2
Longévité	2	2	1
Complexité	1	3	2
Prise en main	1	2	3
Total	**9**	**13**	**14**

Tableau 3.3 – Comparaison des différentes morphologies d'actionneurs

3.1.2 Dimensionnement

Le dimensionnement est basé sur les données de la série des vérins **Festo ESBF** [6], Afin de dimensionner la vis et le moteur l'entrainant, la vitesse de rotation du moteur peut être calculée grâce à l'équation suivante :

$$N = \frac{60v}{l} \tag{3.5}$$

Où :

— N = Vitesse de rotation du moteur [RPM]
— v = Vitesse linéaire de la vis [m/s]
— l = Pas de la vis [m/tour]

Le couple moteur nécessaire peut être calculé via l'équation suivante :

$$M_{mot} = M_{friction} + M_{eff} + I_{verin} \cdot \alpha + I_{mot} \cdot \alpha \tag{3.6}$$

Où :

- M_{mot} = Couple moteur [Nm]
- $M_{friction}$ = Couple dû à la friction au sein du vérin [Nm]
- M_{eff} = Couple effectif appliqué à la vis et transformé en force [Nm]
- I_{verin} = Inertie du vérin $[kg \cdot m^2]$
- I_{mot} = Inertie du moteur $[kg \cdot m^2]$
- α = Accélération angulaire du moteur $[rad/s^2]$

3.1.2.1 Couple de friction $M_{friction}$

Le couple de friction du vérin est représenté par l'équation suivante :

$$M_{friction} = M_{no-load} + M_V \tag{3.7}$$

Les deux termes représentent respectivement la friction statique du vérin ($M_{no-load}$) et le couple de friction dynamique M_V qui augmente avec la vitesse. Dans le cas des vérins Festo, la valeur de $M_{no-load}$ est de 0.2 Nm pour l'ensemble des pas de vis considérés, celle de M_V dépend de la vitesse et est représentée à la Figure 3.5.

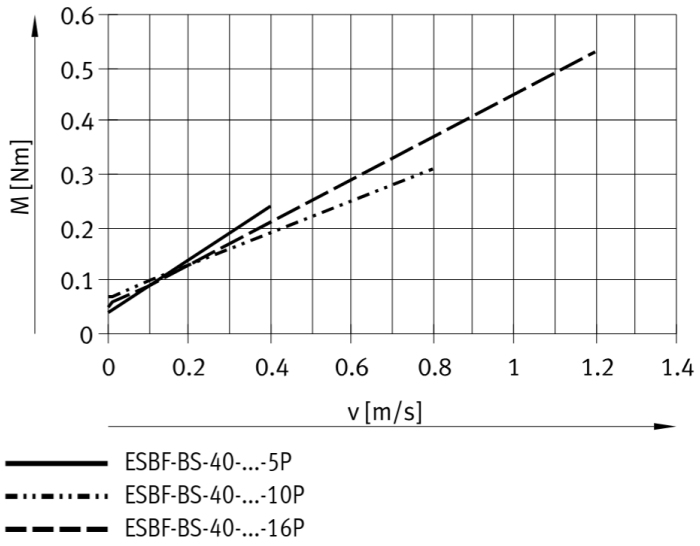


FIGURE 3.5 – Couple de friction dynamique en fonction de la vitesse linéaire du vérin pour différents pas de vis (5-10-16 mm/tour) [6]

3.1.2.2 Couple effectif M_{eff}

Le couple effectif est calculé sur base de la force F souhaitée au bout du vérin, cette force est exprimée par l'équation suivante :

$$M_{eff} = \frac{F \cdot l}{2\pi \cdot \eta} \tag{3.8}$$

Où :

- M_{eff} = Couple effectif appliqué à la vis et transformé en force [Nm]
- F = Force linéaire au bout du vérin [N]
- l = Pas de la vis [m/tour]
- η = Rendement mécanique de la vis (typiquement entre 0.85 et 0.95) [/]

Dans le cas des vérins *Festo*, la relation entre la force et le couple est représentée à la Figure 3.6.

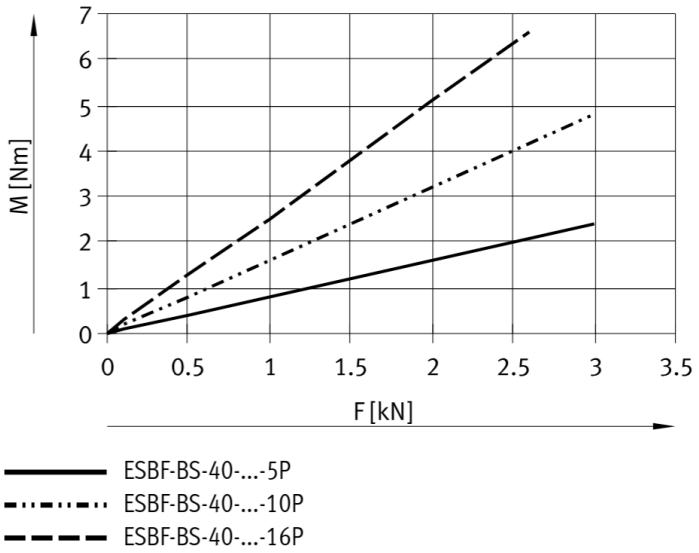


FIGURE 3.6 – Couple effectif en fonction de la force linéaire du vérin pour différents pas de vis (5-10-16 mm/tour) [6]

3.1.2.3 Inertie du vérin I_{verin}

L'inertie du vérin dépend de la longueur de la vis et donc de la course du vérin choisi. Elle est donnée par l'équation suivante :

$$I_{verin} = I_0 + I_H \cdot L \tag{3.9}$$

Où :

- I_0 = Inertie de base du vérin $[kg \cdot m^2]$
- I_H = Inertie par mètre de course du vérin choisi $[kg \cdot m^2]$
- L = Course du vérin [m]

Les valeurs d'inertie I_0 et I_H sont données dans la documentation du vérin [6] tandis que la course L est de 0.2 m pour répondre à la contrainte de 0.11 m.

3.1.2.4 Simulation

Le couple moteur nécessaire dépend donc non seulement de la force et de la vitesse exercée sur le transmetteur de force, mais également de son accélération. De plus, la relation reliant le déplacement appliqué au transmetteur de force à celui du vérin n'est pas triviale, un programme de simulation a donc été mis au point afin de calculer le couple et la vitesse du moteur tout au long d'une simulation. Une fois un moteur sélectionné, ce programme permet d'évaluer la capacité du démonstrateur à réaliser une simulation HIL sur base des données d'une simulation purement numérique. La figure 3.7 représente le résultat d'une courte simulation arbitraire, le couple ainsi que la vitesse maximale du moteur sont facilement identifiables, tout comme son point de fonctionnement RMS. Si celui-ci se situe sous son couple nominal, le moteur est supposé capable de faire tourner la simulation sans s'arrêter. Dans le cas où le couple RMS dépasse le couple nominal du moteur, il est nécessaire de se référer au modèle thermique abordé à la sous-section 3.1.3.

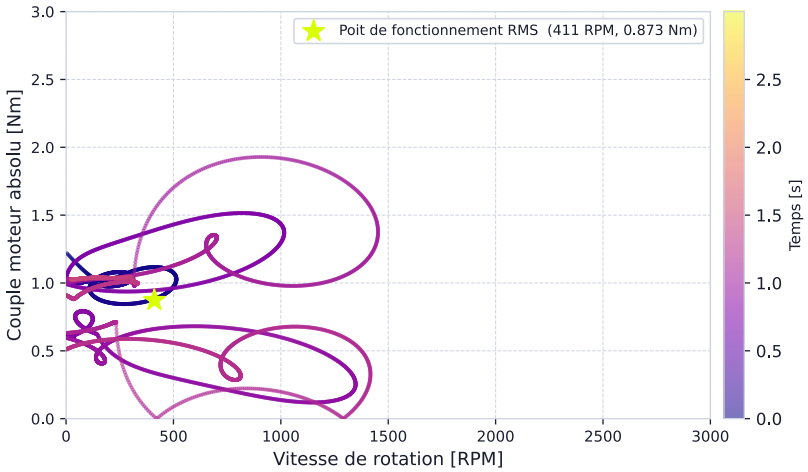


FIGURE 3.7 – Évolution du point de fonctionnement d'un moteur durant une courte simulation

La Figure 3.8 reprend le point de fonctionnement RMS ainsi que le couple et la vitesse maximale de plusieurs combinaisons de moteurs et vérins. Les moteurs de fabricants différents, mais suffisamment similaires en termes de performances ont été regroupés (voir annexe B). Ces valeurs ont été générées sur base de la simulation du VTT ayant servi à l'élaboration du cahier des charges.

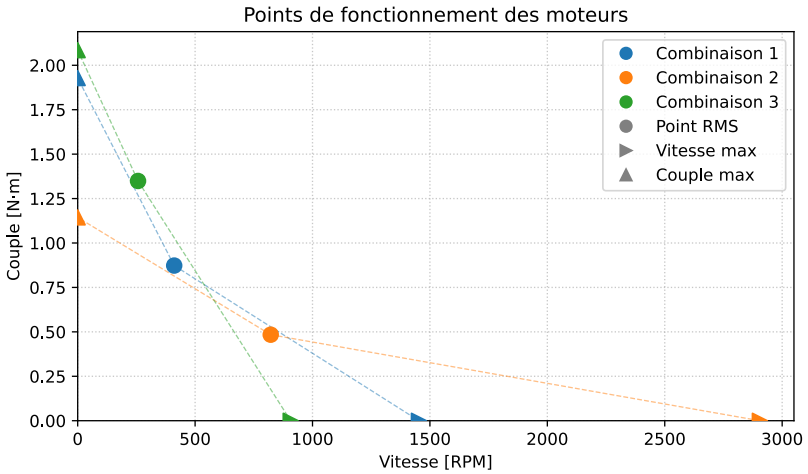


FIGURE 3.8 – Point de fonctionnement RMS, couple max et vitesse max de l'ensemble des combinaisons de moteurs et vérins considérés

La combinaison 2 s'approche fort de la vitesse nominale de 3000 rpm tandis que la combinaison 3, utilisant un moteur pas-à-pas dont le couple disponible chute rapidement avec la vitesse, est à la limite de sont envelope à plusieurs reprises durant la simulation. La combinaison 1 a donc été retenue, c'est celle qui offre le plus de marge tant en termes de couple maximal que de vitesse maximale.

3.1.3 Modèle thermique

Au cours de son fonctionnement, tout moteur électrique subit un échauffement causé principalement par les pertes Joules liées au passage du courant dans son bobinage. Si celui-ci n'est pas maintenu sous sa température limite, la durée de vie du moteur peut se voir raccourcie dans le meilleur des cas, mais le moteur peut être mis hors service immédiatement dans le pire des cas.

La température du bobinage est donc souvent le facteur qui limite le couple développé par le moteur durant sa phase d'utilisation. Dans le contexte de ce démonstrateur, il est souhaité d'utiliser le moteur selon un cycle de 15 minutes de fonctionnement suivies de 5 minutes de refroidissement. Pour chacun des moteurs, l'analyse thermique se base sur le couple RMS obtenu suite à la simulation (M_{RMS}) de la sous-section 3.1.2 qui permet de calculer le courant RMS traversant le bobinage grâce à la constante de couple (k_t [Nm/A]) de chaque moteur :

$$I_{RMS} = \frac{M_{RMS}}{k_t} \tag{3.10}$$

La puissance des pertes Joules peut être calculée sur la base de la résistance électrique du bobinage $R_{bobinage}$:

$$P = R_{bobinage} \cdot I_{RMS}^2 \tag{3.11}$$

Enfin, grâce à la constante de temps thermique du moteur τ_{th}, la température lors de l'échauffement et du refroidissement peut être calculée :

$$T_{échauffement}(t) = T_{init} + (R_{th} \cdot P) \cdot (1 - e^{-\frac{t}{\tau_{th}}}) \tag{3.12}$$

$$T_{refroidissement}(t) = T_{amb} + (T_{init} - T_{amb}) \cdot e^{-\frac{t}{\tau_{th}}} \tag{3.13}$$

Où :

- T_{init} = Température initiale du cycle (d'échauffement ou refroidissement) [°C]
- R_{th} = Résistance thermique du moteur [K/W], dans le cas où elle ne serait pas fournie dans la documentation, elle peut être calculée grâce à $\tau_{th} = R_{th} \cdot C_{th}$
- T_{amb} = Température ambiante [°C]

Il est donc possible de modéliser numériquement la température du moteur au cours de ses cycles de fonctionnement et de refroidissement. Cependant, en pratique, les contrôleurs disposent d'une sécurité thermique qui coupe l'alimentation du moteur avant de laisser le bobinage surchauffer. Cette sécurité se base sur un modèle thermique qui n'est généralement pas accessible par l'utilisateur tout en étant assez conservateur, il est donc nécessaire de caractériser cette sécurité dans le cas où le moteur opérerait régulièrement au-delà de son couple nominal. Cette caractérisation est disponible dans la section 5.4.

3.1.4 Choix du protocole de communication

Lors de la simulation, le contrôleur moteur reçoit une nouvelle consigne de position toutes les millisecondes, la communication avec le contrôleur doit donc être en mesure de transmettre des consignes avec un minimum de latence à une fréquence de 1kHz.

Au préalable de la conception du banc d'essai, un pré-prototype à plus petite échelle a été mis au point (plus d'informations sur ce dispositif sont disponibles à l'annexe C). Ce prototype utilisait une communication série (RS-232) pour la communication entre un ordinateur et un contrôleur, le transfert d'une instruction prenait alors plusieurs millisecondes. Cela a souligné la nécessité de porter une attention particulière au choix du protocole de communication reliant l'ordinateur au contrôleur du moteur.

Les protocoles retenus, sur la base des contrôleurs existants sur le marché, sont les suivants :

- CANopen sur bus CAN
- PROFINET
- EtherCAT

Au-delà de la fréquence de transfert des instructions de position, d'autres critères ont également leur importance, notamment la facilité d'intégration avec le système d'exploitation et le middleware (ROS2). Le coût du hardware supplémentaire nécessaire à la communication (carte interface, câbles spécifiques, etc) est aussi pris en compte.

3.1.4.1 CANopen sur bus CAN

Le CANopen est une couche applicative qui peut s'appuyer sur différentes couches matérielles, notamment sur le bus de terrain CAN (Controller Area Network). Bien que le bus CAN soit théoriquement en mesure d'assurer une communication à 1000 instructions par seconde, son utilisation requiert une carte interface. Celle-ci peut communiquer avec un PC via différents protocoles (USB, SPI, série, ...) dont le choix est important car il peut limiter la fréquence de transfert des données sous le seuil de 1 kHz recherché [20]. En termes de support, l'interface avec ROS2 est très bien supportée et l'offre de contrôleurs utilisant le CANopen est vaste.

3.1.4.2 PROFINET

PROFINET est un protocole de communication industriel développé par Siemens, la couche applicative PROFIdrive vient s'y ajouter pour les applications de contrôle de mouvement. Cette combinaison est tout à fait en mesure d'assurer la fréquence d'instructions de 1kHz et même de la dépasser, cependant l'intégration avec ROS2 est incomplète, la couche PROFIdrive n'étant pas implémentée. De plus, l'implémentation de la couche PROFINET nécessite l'achat de matériel et logiciel propre à Siemens [21].

3.1.4.3 EtherCAT

L'EtherCAT est un protocole ouvert développé par Beckhoff [22], pour le contrôle de mouvement, il implémente la couche CANopen en apportant de nombreux avantages par rapport au bus CAN. D'abord, en termes de performances, le protocole est capable de dépasser largement les 1000 instructions par secondes et sera généralement limité par les périphériques du réseau, lesquels, dans une application de contrôle de mouvement, sont capables d'atteindre 1kHz. Concernant l'intégration avec ROS2, un driver open-source est disponible, et permet d'utiliser n'importe quelle interface réseau Ethernet pour se connecter au contrôleur du moteur.

3.1.4.4 Résumé des protocoles de communication

Le Tableau 3.4 dresse une comparaison des différents protocoles de communication, pour chaque critère de sélection, chacun des protocoles est classé en obtenant un score de 1 à 3. La solution dont la somme des scores est la plus grande est donc la plus adaptée.

Critère	CANopen sur bus CAN	PROFINET	EtherCAT
Fréquence de transfert	1	2	3
Coût lié au matériel supplémentaire	2	1	3
Facilité d'intégration avec ROS2	3	1	2
Total	**6**	**4**	**8**

Tableau 3.4 – Caractéristiques techniques du capteur de force [12]

Le protocole EtherCAT s'est démarqué des autres en offrant des performances largement suffisantes pour cette application tout en étant suffisamment simple à intégrer à ROS2.

3.1.5 Choix final de l'actionnement

Vérin Le vérin choisi est le **Festo ESBF-BS-40-200-10P**, il offre une solution prête à l'emploi pour l'implémentation d'un actionneur à vis à billes.

Le vérin (représenté à la Figure 3.9) possède les caractéristiques principales suivantes :

- Course : 200 [mm]
- Pas de vis : 10 [mm/tour]
- Force linéaire maximale : 3000 [N]
- Vitesse linéaire maximale : 0.8 [m/s]
- Couple moteur maximal : 5.6 [N]
- Vitesse de rotation maximale : 4800 [rpm]
- Jeu mécanique : < 0.03 [mm]

FIGURE 3.9 – Festo ESBF-BS-40-200-10P

Le kit de montage parallèle **Festo EAMM-A-D40-60G**, représenté à la Figure 3.10, permet le montage d'un servomoteur axialement au vérin. Il inclut un accouplement flexible afin de ne pas créer de problème d'hyperstaticité. L'absence de courroie pas rapport à d'autres solution d"accouplement permet de supprimer une source d'imprécision angulaire.

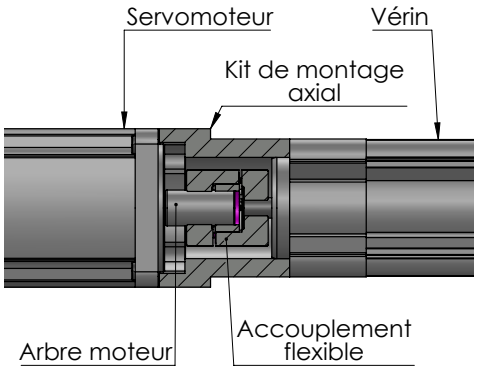


FIGURE 3.10 – Vue en coupe du kit de montage axial reliant le vérin au servomoteur

Servomoteur Le servomoteur sélectionné est le **Stepperonline A6M60-400H2A1-M17**, c'est un moteur synchrone à aimants permanents muni d'un encodeur offrant une résolution de 17 bits (131072 positions par tour). Il est important de noter que cet encodeur ne conserve pas sa position en cas de perte d'alimentation, il est donc indispensable d'implémenter une procédure de retour à l'origine qui sera exécutée lors de chaque redémarrage (voir section 4.3.1). L'état d'avancement de l'intégration à ROS2 de ce moteur et de son contrôleur au moment de la décision, ainsi que le délai de livraison très rapide et son faible coût sont les principaux facteurs ayant orienté ce choix.

Le moteur (représenté à la Figure 3.11) possède les caractéristiques principales suivantes :

- Puissance nominale : 400 [W]
- Couple nominal : 1.27 [Nm]
- Couple maximal : 4.45 [Nm]
- Vitesse nominale : 3000 [rpm]
- Vitesse maximale : 6000 [rpm]
- Inertie du rotor : 0.6 $[kg \cdot cm^2]$
- Tension nominale : 220 [V]

FIGURE 3.11 – Stepperonline A6M60-400H2A1-M17

Les courbes de couple par rapport à la vitesse d'un moteur sont une caractéristique importante et offrent des informations supplémentaires aux valeurs nominales et maximales présentées précédemment. La Figure 3.12 représente ces courbes pour le moteur sélectionné.

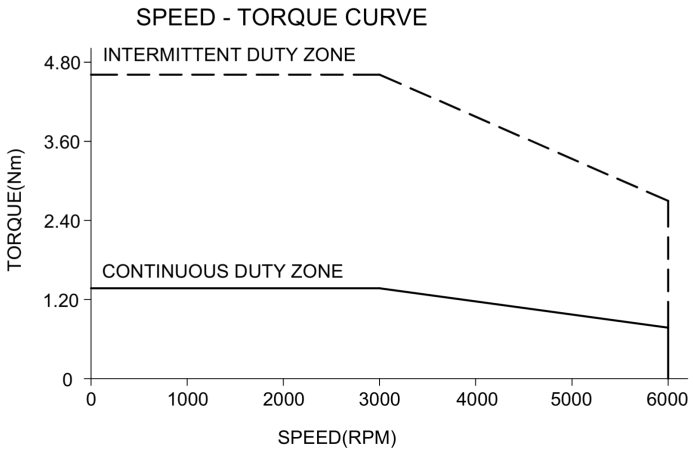


FIGURE 3.12 – Courbes de couple/vitesse du servomoteur choisi [7]

Contrôleur Le contrôleur choisi pour piloter le servomoteur est le **Stepperonline A6-400EC**, il s'agit du contrôleur conseillé par le fabricant du moteur. Il prend en charge la communication EtherCAT, ce protocole ayant été choisi principalement pour sa vitesse de transfert importante ainsi que pour son intégration aisée avec ROS2.

Le contrôleur (représenté à la Figure 3.13) possède les caractéristiques principales suivantes :

- Puissance nominale : 400 [W]
- Courant nominal : 2.8 [A]
- Courant maximal : 10.1 [Nm]
- Tension d'alimentation : 200-240 [VAC]

FIGURE 3.13 – Photo du contrôleur

3.2 Instrumentation et chaîne d'acquisition

Le démonstrateur intègre des capteurs permettant de mesurer la force appliquée sur le TF ainsi que son déplacement. Le choix de ces capteurs est détaillé dans la suite de cette section.

3.2.1 Mesure de force

Afin de résoudre les équations de la simulation multi-corps et de calculer le pas de temps suivant, la force appliquée sur le transmetteur (TF) doit être mesurée puis réinjectée dans le modèle multi-corps.

Pour garantir l'adéquation de ce système avec les exigences du cahier des charges (voir chapitre 2), le capteur et sa chaîne d'acquisition doivent présenter les caractéristiques suivantes :

- **Fréquence d'excitation mécanique maximale du système** ($f_{\text{exc,max}}$) : 20 Hz

— **Force maximale** ($F_{\max}$) : 3000 N
— **Direction de la force** : un seul degré de liberté
— **Mode de sollicitation** : fonctionnement bidirectionnel (traction et compression).

3.2.1.1 Étude des différentes technologies de capteurs de force

Dans l'industrie, différentes technologies de capteurs existent pour mesurer une force. Chacune d'elles possède ses propres avantages et inconvénients. Les deux plus grandes technologies sont les capteurs à jauges de contrainte et les capteurs piézoélectriques.

Capteurs à jauge de contrainte

Principe de fonctionnement : Ils sont constitués d'un corps d'épreuve métallique (acier, aluminium) conçu pour se déformer de manière élastique sous l'application d'une charge. Des jauges résistives de déformation sont collées sur ce corps, leur déformation entraînant une variation de leur résistance électrique. Lors qu'elles sont soumises à de la traction, le fil s'allonge et sa section s'amincit, ce qui accroît sa résistance au passage des électrons (la résistance électrique augmente et inversement en compression). En connectant ces jauges en un pont de Wheatstone, il est possible de convertir ces infimes variations de résistance en un signal de tension exprimé en mV. Une illustration de ce montage est présentée à la Figure 3.14[8].

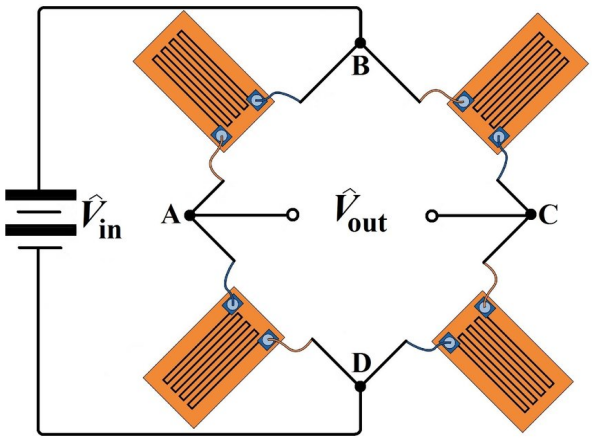


FIGURE 3.14 – Pont de Wheatstone avec 4 jauges de contrainte [8]

Avantages :
— Ils ne présentent aucune dérive temporelle, ce qui les rend très précis et efficaces pour l'analyse de charges statiques.
— Le montage en pont de Wheatstone permet de compenser les effets de variation des résistances dus à la température.
— Il existe de nombreux types de montages différents afin de mesurer les efforts souhaités.

Inconvénients :
— Pour l'application d'une force dynamique, la réponse en fréquence maximale du capteur est directement liée à la fréquence fondamentale du corps d'épreuve mécanique, ce qui les rend peu efficaces pour de la haute dynamique.
— Le signal de sortie est très faible et nécessite un amplificateur externe.
— Le signal de sortie est directement proportionnel à la tension d'alimentation du capteur (ou ponts de Wheatstone). Celle-ci doit être la plus stable possible.

Capteurs piézoélectriques

Principe de fonctionnement : Ces capteurs utilisent l'effet piézoélectrique en intégrant des monocristaux tels que le quartz ou le phosphate de gallium. Lorsqu'une force est appliquée directement sur le cristal, celui-ci subit une déformation infime et génère une charge électrique directement proportionnelle à la force exercée. Il est ensuite nécessaire d'utiliser un amplificateur de charge pour exploiter cette mesure [23]. Une illustration de cette technologie de capteur est présentée à la figure 3.15.

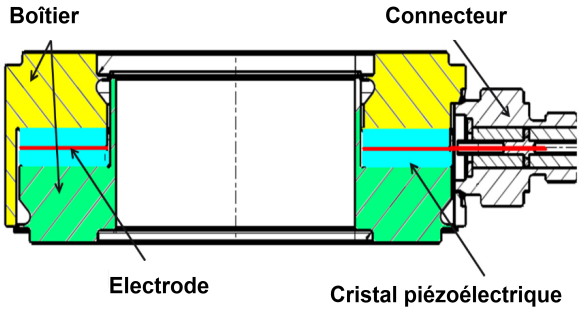


FIGURE 3.15 – Structure interne d'un capteur piézoélectrique [9]

Avantages :
- Ils sont particulièrement adaptés aux analyses requérant une grande dynamique (comme les analyses vibratoires), car la rigidité extrême des cristaux leurs donne une réponse en fréquence très élevée.
- La robustesse des cristaux leurs donne une grande résistance aux surcharges et diminue le risque de l'endommager.

Inconvénients :
- Ils présentent une dérive temporelle due à une fuite de charge dans le circuit (environ 1 N/min [9]), ce qui les rend inadaptés aux mesures purement statiques.
- Ils nécessitent une chaîne d'amplification de charge complexe.

Conclusion : Dans le cadre d'une application HIL, la mesure de force doit être parfaitement stable et fiable afin d'être réinjectée dans la boucle de simulation. Toute instabilité du capteur fausserait le comportement dynamique du modèle et pourrait compromettre la stabilité globale, y compris en régime statique. La technologie à jauge de contrainte a donc été retenue, ses performances en dynamique s'avérant parfaitement adaptées à la dynamique maximale visée par le cahier des charges (voir section 3.2.1.2).

3.2.1.2 Comparaison et choix des capteurs à jauge de contrainte

Le choix du capteur a été effectué selon les critères comparatifs présentés dans le tableau 3.5. L'effort maximal requis par le cahier des charges est fixé à 3000 N. Néanmoins, la géométrie mécanique du démonstrateur induit un facteur d'amplification de 3 (dans le cas le plus critique) sur la force développé par le vérin (La force maximal délivrable par le vérin est de 3000 N). Afin de ne pas restreindre les performances du démonstrateur dans le cas d'utilisation d'une simulation nécessitant une force plus élevée, un capteur ayant une pleine échelle de minimum 9000 N est donc nécessaire.

Le point négatif concernant les capteurs de force à jauge de contrainte réside dans leur fréquence de réponse maximale admissible lors de l'application d'une charge dynamique. Certaines informations relatives à leur réactivité en dynamique ne sont pas présentes dans les fiches techniques des fournisseurs. Une étude dynamique peut néanmoins être réalisée sur la base des informations connues.

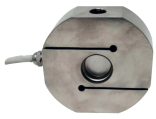
Capteurs	HBK - S9M [12]	HBK - U2B [24]	Scaime MTRZ [25]
Aperçu			
Force nominale F_N	10 kN	10 kN	10 kN
Force maximale dynamique	100 % F_N	100 % F_N	100 % F_N
Précision	0,02 % F	0,1 % F	0,03 % F
Déformation à charge nominale D_{NO}	0,2 mm	0,047 mm	0,3 mm
Fréquence fondamentale f_0	2,1 kHz	17,5 kHz	/
Prix (TTC)	643,72 €	1 663,32 €	556,6 €
Disponibilité (Commande mi-Mars) [1]	En stock	En stock	8 semaines

TABLEAU 3.5 – Comparaison des capteurs à jauge de contrainte

Étude de la dynamique : Le capteur de force se déformant sous l'application d'une charge, il est judicieux de l'assimiler à un système masse-ressort (voir la Figure 3.16) afin de déterminer sa fréquence fondamentale (fréquence à laquelle le capteurs oscille naturellement lorsqu'il reçois une impulsion) et sa fréquence de réponse maximale admissible. La masse M_1 correspond à la masse de l'extrémité active du capteur et la masse M_2 correspond à la masse ajoutée des dispositifs de fixation. Lors de la phase de conception, la masse M_2 a été estimée à 0,4 kg. (Cette valeur est largement supérieure à la masse réelle du dispositif et permet de s'octroyer une marge de sécurité en cas de changement de TF et donc, potentiellement, de système de fixation). La masse active M_1 des capteurs n'est pas fournie dans les fiches techniques, mais elle peut être déterminée sur la base de la fréquence fondamentale f_0 et de la raideur équivalente K du capteur. Cette raideur est calculée à partir de la déflexion en régime nominal (D_{NO}) :

$$K = \frac{F_N}{D_{NO}} \tag{3.14}$$

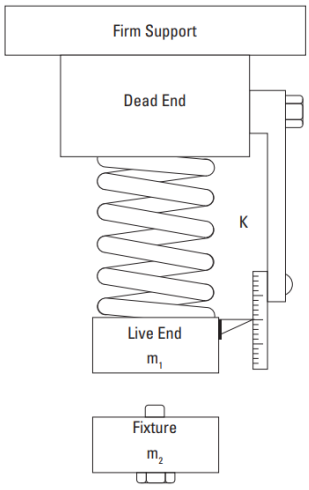


FIGURE 3.16 – Schéma équivalent du capteur de force sous un système masse-ressort - Figure tirée de [10]

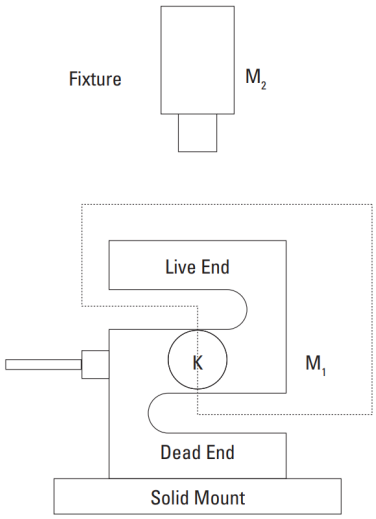


FIGURE 3.17 – Distribution des masses impliquant la dynamique - Figure tirée de [10]

Il faut maintenant calculer l'impact sur la fréquence fondamentale dû à la fixation d'une masse M_2 supplémentaire. Cela se calcule grâce à l'expression suivante [10] :

$$f_s = f_0 \sqrt{\frac{M_1}{M_1 + M_2}} \tag{3.15}$$

Le Tableau 3.6 récapitule les valeurs numériques pour les trois capteurs de force étudiés :

Capteurs	HBK - S9M	HBK - U2B	Scaime MTRZ
Fréquence fondamentale f_0	2,1 kHz	17,5 kHz	/
Déformation à charge nominale D_{NO}	0,2 mm	0,047 mm	0,3 mm
Raideur équivalente K	50 MN/m	212,77 MN/m	33,33 MN/m
Masse active calculée M_1	287,2 g	17,6 g	/
Fréquence fondamentale avec fixation (f_s)	1,36 kHz	3,59 kHz	/
Fréquence maximale conseillée	136 Hz	359 Hz	/

TABLEAU 3.6 – Synthèse des paramètres de l'analyse dynamique des capteurs

La valeur de la fréquence fondamentale du capteur Scaime MTRZ n'étant pas fournie par le fabricant, il est impossible de déterminer sa fréquence maximal. Cependant, ce capteur étant relativement similaire en taille et en masse au capteur S9M, et qu'il a une déformation à charge nominal plus grande que le S9M, on peut en conclure qu'il a une moins bonne performance dynamique que le capteur S9M.

Une limitation d'un facteur 10 de la fréquence fondamentale du système (f_s) est généralement préconisée afin de diminuer l'impact sur l'amplitude et la phase du signal de sortie [26]. La fréquence maximale conseillée d'utilisation des capteurs est bien supérieure aux 20 Hz définis dans le cahier des charges.

Conclusion : Au vu de l'ensemble de ces informations, le choix s'est porté sur le modèle S9M. Il représente le meilleur compromis entre le coût et la précision de la mesure de force, bien que cela se fasse au détriment d'une déformation plus élevée ($60\mu m$ pour une charge de 3000 N). Il permet par ailleurs de garantir un délai de livraison optimal afin de concevoir ce démonstrateur avant la fin de l'année et s'avère parfaitement capable de fonctionner en dynamique pour les fréquences demandées.

3.2.1.3 Conditionnement du signal analogique

Comme mentionné précédemment, le signal de sortie du capteur de force est directement proportionnel à sa tension d'alimentation en raison de sa configuration en pont de Wheatstone. Ce signal étant de l'ordre du millivolt, il est nécessaire de l'amplifier afin de pouvoir le convertir en données numériques à l'aide d'un convertisseur analogique-numérique (ADC).

Le conditionneur CPJ du fabricant Scaime [15] permet de répondre précisément à ces deux exigences. De plus, cet appareil intègre un filtre passe-bas du deuxième ordre avec une fréquence de coupure de 10 Hz, permettant d'atténuer les composantes de bruit à haute fréquence sur le signal mesuré et amplifié. Néanmoins, ce filtre peut être désactivé dans le cas où une application nécessiterait une bande passante plus large pour atteindre les 20 Hz requis par le cahier des charges. Ses caractéristiques principales sont synthétisées dans le tableau 3.7.

Ce conditionneur a été calibré afin d'obtenir une tension de sortie de ± 10 V pour une plage de charge de ± 5000 N, ce qui sera expliqué dans la section 5.1.1. Cette charge maximale de calcul reste inférieure à la valeur souhaitée par le cahier des charges (3000 N), ce qui permet d'optimiser et d'obtenir une meilleure résolution lors de la quantification par l'ADC.

Caractéristique	Valeur
Précision	0,05 %
Alimentation capteur	3, 5 ou 10 VDC
Sortie en tension	±10 V
Bande passante	≤ 20 kHz
Filtre passe-bas (commutable)	10 Hz

TABLEAU 3.7 – Caractéristiques principales du CPJ [15]

FIGURE 3.18 – Conditionneur analogique CPJ.

3.2.2 Mesure de position

Afin de vérifier le déplacement imposé au niveau du TF par la géométrie et la chaîne d'actionnement, un potentiomètre linéaire a été installé en parallèle de ce dernier. Son objectif principal est de valider le déplacement relatif du TF. Un capteur de type HLP129 de chez Penny & Giles [16] a été sélectionné à partir du stock disponible dans la réserve de Thierry Daras. Ses caractéristiques principales sont synthétisées dans le Tableau 3.8.

Caractéristique	Valeur
Course électrique	150 mm
Résistance	6 kΩ
Linéarité indépendante (typique)	± 0,07 %
Hystérésis (répétabilité)	< 0,01 mm
Vitesse maximale du curseur	10 m/s

TABLEAU 3.8 – Caractéristiques principales du potentiomètre linéaire HLP129 [16]

Ce capteur fonctionnant exclusivement en mode diviseur de tension, le circuit d'acquisition doit présenter une impédance élevée afin de ne pas perturber la mesure. La piste du modèle de 150 mm possédant une résistance de 6 kΩ, le canal du convertisseur analogique-numérique (ADC) doit présenter une résistance d'entrée d'au moins $0,6$ MΩ selon la documentation [16]. Une impédance d'entrée insuffisante induirait un effet de charge par la mise en parallèle du circuit de mesure. Cela fausserait la valeur lue par rapport à la position physique réelle.

3.2.3 Conversion analogique-numérique

Afin d'exploiter et de traiter les données issues des capteurs, une étape de conversion analogique-numérique (ADC) est indispensable. Ce convertisseur doit être capable de transmettre les informations reçues via les deux signaux analogiques au calculateur embarqué du démonstrateur (un Raspberry Pi) (La justification de l'utilisation d'un RPi est fournie à la section 3.3.1).

De plus, la résolution de cet ADC doit être rigoureusement sélectionnée pour être inférieure à la précision du capteur de force, garantissant ainsi la numérisation de l'ensemble des variations de charge. Le capteur de force utilisé présentant une précision de 0,02 % de la charge appliquée, le cas le plus critique correspond à la charge nominale de 10 kN, pour laquelle l'incertitude absolue s'élève à 2 N.

Le quantum de quantification pour les différents capteurs utilisés dans le banc de test est calculé selon l'équation 3.16 (où ΔX représente la plage de mesure considérée). Les résultats sont repris dans le tableau 3.9.

$$q = \frac{\Delta X}{2^{16}} \tag{3.16}$$

Quantum en tension q_V	**Quantum en force q_F**	**Quantum en position q_d**
0,305 mV	0,305 N	4,55 μm

TABLEAU 3.9 – Quanta de quantification pour les différents capteurs

Ce quantum de 0,305 N pour la force étant largement inférieur à la limite d'incertitude de 2 N, la numérisation ne dégradera pas la précision du capteur.

Sur la base de l'ensemble des critères énoncés ci-dessus, le choix s'est porté sur la carte d'acquisition MCC 128 de Measurement Computing (distribuée par Digilent) [11]. Ce module d'extension se connecte directement sur le connecteur GPIO (General Purpose Input/Output) du Raspberry Pi (voir la figure 3.19) afin de lui fournir une tension d'alimentation et de communiquer grâce au protocole SPI directement vers le RPi. De plus, les entrées analogiques du convertisseur permettent de numériser des signaux sur une plage de ±10 V.

Caractéristique	**Valeur**
Résolution	16 bits
Nombre de canaux	8 SE ou 4 DIFF
Fréquence d'échantillonnage	100 kS/s
Temps de conversion	9.8 μs
Précision absolue	±6 mV à pleine échelle
Bruit de canal	350 μV_{RMS} (plage ±10 V)
Impédance d'entrée	> 1 GΩ

TABLEAU 3.10 – Caractéristiques principales du MCC 128 [11]

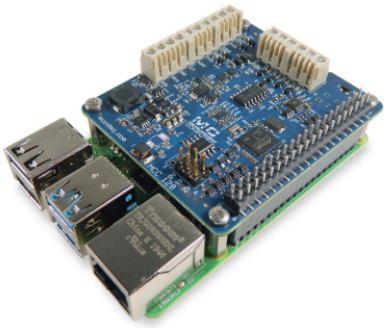


FIGURE 3.19 – Convertisseur MCC 128 [11]

Comme l'indique le tableau 3.10, cet ADC possède une impédance d'entrée largement supérieure à la valeur limite critique imposée par le potentiomètre linéaire ($0,6$ MΩ). Par ailleurs, sa fréquence d'échantillonnage maximale est parfaitement compatible avec les exigences du calcul en temps réel, dont le pas de temps cible est fixé à $\Delta t_{sim} = 10^{-3}$ s (soit une fréquence de rafraîchissement de 1 kHz).

De plus, ce convertisseur dispose d'une bibliothèque open source en C/C++ dédiée à l'acquisition des données [27]. Celle-ci propose deux modes principaux d'acquisition :

— **Lecture de valeur unique (Single-value reading)** : Ce mode permet d'interroger le convertisseur pour obtenir une mesure ponctuelle sur un canal donné. Cette opération peut être intégrée au sein d'une boucle logicielle afin de cadencer l'acquisition à la fréquence désirée de 1 kHz, correspondant au pas de temps de la simulation en temps réel. La donnée, lue à l'instant précis de la requête, est immédiatement transmise au Raspberry Pi.

— **Lecture en mode continu (Continuous mode)** : L'ADC est configuré pour acquérir un nombre N de données à une fréquence d'échantillonnage choisie, puis transmet un bloc complet de mesures. Le stockage temporaire s'effectue dans la mémoire tampon (*buffer*) de l'ADC, et les données sont ensuite envoyées par paquets au Raspberry Pi en SPI. Cette méthode est particulièrement adaptée si l'on considère que, sur un intervalle de 1 ms, la force appliquée sur le capteur présente des variations négligeables (La vérification de l'hypothèse est éffectuée dans l'Annexe A), la dynamique maximale du phénomène physique étant limitée à 20 Hz. Il est ainsi possible d'effectuer

un sur-échantillonnage durant cette milliseconde pour acquérir N échantillons, puis d'appliquer un filtrage numérique afin d'atténuer le bruit et de stabiliser la mesure.

Ces deux méthodes d'acquisition de données sont comparées dans la section 5.1.3.

3.2.4 Résumé de la précision de la mesure théorique

L'évaluation de l'incertitude théorique globale repose sur les caractéristiques de l'ensemble de la chaîne d'instrumentation. Pour rappel, celle-ci est calibrée pour délivrer une tension de 10 V en sortie sous une charge de 5 kN. Afin de permettre une comparaison directe, la précision de chaque maillon a été convertie en newtons (N) sur la base de cette charge maximale de 5 kN, ce qui représente le cas le plus défavorable. Ces valeurs individuelles sont synthétisées sur la figure 3.20.

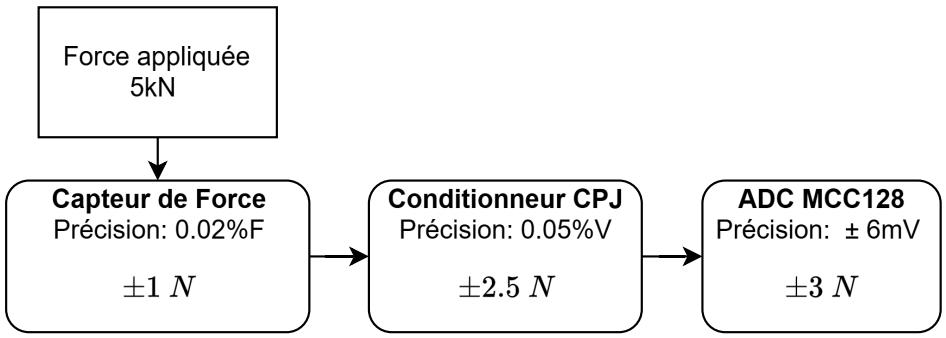


FIGURE 3.20 – Précision de la chaîne de mesure de force

Ces incertitudes étant strictement indépendantes les unes des autres, l'incertitude globale de la chaîne d'instrumentation est calculée par la méthode de la somme quadratique (RSS) :

$$E_{\text{tot}} = \sqrt{E_{\text{capteur}}^2 + E_{\text{CPJ}}^2 + E_{\text{MCC128}}^2} \approx \pm 4{,}03 \text{ N} \tag{3.17}$$

Ce résultat met en évidence que la chaîne d'instrumentation présente une incertitude maximale théorique de ±4,03 N à pleine échelle.

3.3 Électronique globale

Maintenant que l'entièreté des composants ont été sélectionnés, il est possibles d'établir l'électronique globale du démonstrateur (Voir figure 3.21)

3.3.1 Système informatique

Le schéma électrique de la Figure 3.21 intègre deux systèmes informatiques distincts : d'un côté, un ordinateur fixe et, de l'autre, un Raspberry Pi 4.

Le concept initial de cette séparation était de disposer d'un démonstrateur totalement autonome pour l'actionnement et l'acquisition de données, tandis que l'ordinateur fixe se dédiait exclusivement à la gestion de la simulation multi-corps. Cependant, le Raspberry Pi ne disposant que d'un unique port Ethernet déjà utilisé pour la communication EtherCAT avec le driver moteur, une interface USB-Ethernet a d'abord été employée pour relier le PC fixe au Raspberry Pi.

Lors des phases d'expérimentation, un délai d'environ 1,6 ms a été observé en raison de l'utilisation de cette interface. Afin d'assurer un simulation en temps réel avec un $\Delta t_{\text{sim}} = 10^{-3}$ s, un tel délai n'est pas acceptable.

La conception a donc été adaptée : le port Ethernet natif du Raspberry Pi est désormais dédié à la communication avec le PC fixe, tandis que la liaison avec le driver moteur est à

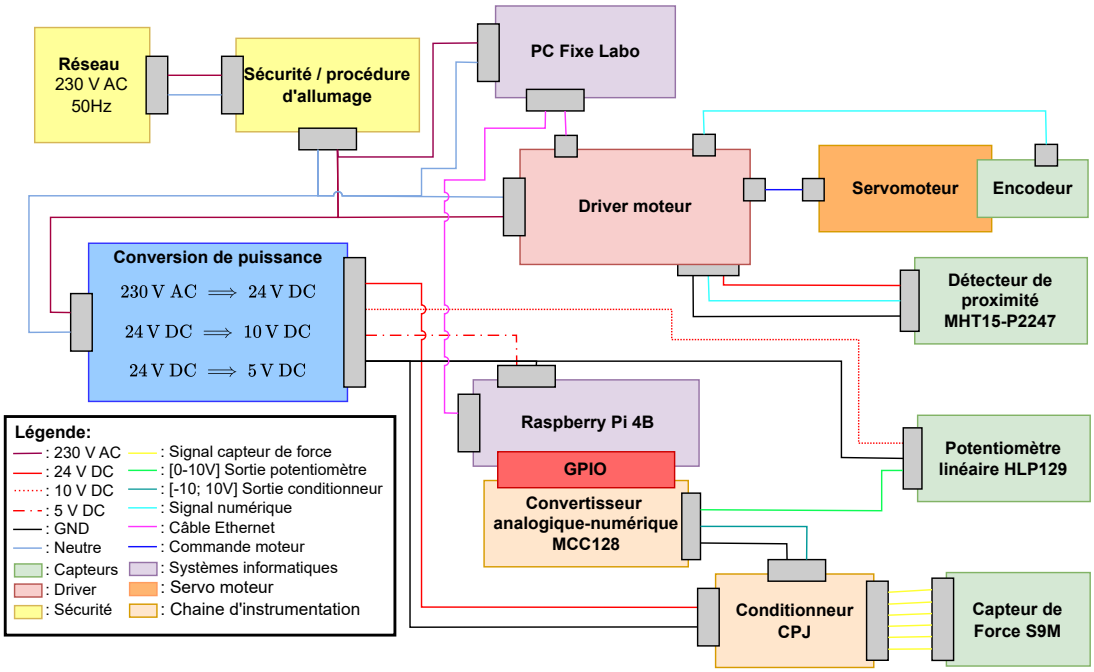


FIGURE 3.21 – Schéma électrique global du démonstrateur

présent gérée par le PC fixe (ce dernier possédant plusieurs ports Ethernet). Ce changement de conception permet d'atteindre les temps de communication requis (inférieurs à la milliseconde) vers le driver et le Raspberry Pi via deux câbles Ethernet distincts. En contrepartie, cela supprime l'objectif initial consistant à séparer physiquement la simulation du démonstrateur. La figure 3.21 prend en compte cette nouvelle configuration.

Le Raspberry Pi est alimenté par du 5 Volt DC provenant du bloc conversion de puissance. L'utilisation d'un Rpi est justifiée afin de simplifier l'acquisition des données des capteurs via les entrées GPIO permettant de communiquer grâce au protocole SPI avec le MCC128

3.3.2 Chaîne d'instrumentation et acquisition de données

Le capteur de force S9M utilise un raccordement à six fils. Le signal d'excitation ($V_{\mathrm{Ex},+}$, $V_{\mathrm{Ex},-}$), qui assure l'alimentation du capteur, est une tension stable de 5 V fournie par le conditionneur CPJ. Un signal de contre-réaction ($V_{\mathrm{sens},+}$, $V_{\mathrm{sens},-}$) est également renvoyé au conditionneur afin de compenser la chute de potentiel subie par le signal d'alimentation dû à la résistivité des câbles.

Enfin, le signal différentiel mesuré par le pont de Wheatstone ($V_{\mathrm{sig},+}$, $V_{\mathrm{sig},-}$) présente une amplitude comprise entre -10 mV et 10 mV, la sensibilité nominale du capteur étant de 2 mV/V. Ce signal est transmis au conditionneur CPJ afin d'y être amplifié. L'ensemble de ces câbles est placé au sein d'un blindage afin de limiter les interférences électromagnétiques générées par la partie puissance de l'actionnement. Les connexions internes du pont de Wheatstone sont illustrées à la figure 3.22. Elles sont configurées de manière à obtenir une tension positive (et donc une valeur de force positive) lorsque le capteur est sollicité en compression.

Le conditionneur CPJ, alimenté en 24 V, amplifie le signal V_{sig} afin d'obtenir une tension de sortie comprise entre -10 V et 10 V, laquelle est ensuite envoyée vers l'ADC.

Le potentiomètre linéaire est quant à lui alimenté en 10 V. Sa piste de mesure faisant office de diviseur résistif, il fournit une tension comprise entre 0 V et 10 V qui est directement

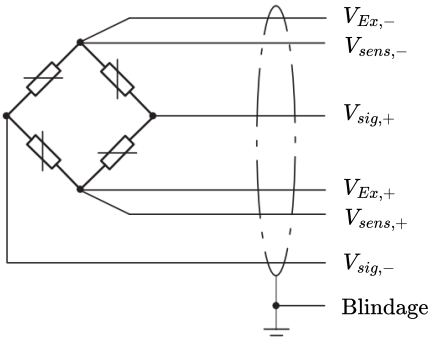


FIGURE 3.22 – Définition des signaux du capteur S9M - Figure tiré de [12]

connectée à l'ADC.

3.3.3 Chaîne d'actionnement et puissance

Le servo-contrôleur est alimenté en 230VAC et est relié au moteur via 2 câbles propriétaires, l'un envoyant la puissance vers les trois bobinages du moteur, l'autre permettant de lire les données de l'encodeur. Il est également relié au PC via un câble Ethernet permettant la communication EtherCAT. Un détecteur de proximité y est également connecté via trois câbles pour son alimentation et pour faire parvenir le signal de proximité au contrôleur (12V, GND, SIG).

Une alimentation DC se charge de convertir les 230VAC en 24VDC, 2 convertisseurs buck abaissent cette tension de 24V en 10V et 5V respectivement.

3.3.4 Sécurité électrique

La sécurité est assurée par un circuit composé d'un relais d'auto-maintient, il est interrompu par un bouton d'arrêt d'urgence ainsi que par deux interrupteurs se déclenchant lors de l'ouverture de la porte (le schéma électrique de ce circuit est disponible à l'annexe D.1).

Chapitre 4

Architecture logicielle et implémentation de la simulation HIL

Ce chapitre décrit l'implémentation logicielle du démonstrateur et détaille les procédures nécessaires à son initialisation ainsi qu'à la réalisation d'une simulation HIL. Depuis quelques années, le logiciel Robotran offre la possibilité d'être utilisé avec ROS (*Robot Operating System*). Le cahier des charges stipule ainsi l'utilisation de ROS 2 pour assurer la communication entre la simulation numérique et le démonstrateur physique.

4.1 ROS 2 (Robot Operating System 2)

ROS 2 (*Robot Operating System 2*) est une suite logicielle *open-source* conçue pour simplifier le développement d'applications robotiques. L'architecture de ROS 2 s'articule autour de nœuds (*nodes*), qui sont chacun responsables d'une tâche unique (par exemple : l'exécution d'une simulation, l'acquisition des données d'un capteur ou l'actionnement d'un moteur). Chaque nœud a la possibilité d'échanger des données avec les autres nœuds présents sur le réseau grâce à trois mécanismes de communication distincts : les topics, les services et les actions [13].

Les topics Ils permettent un échange continu de données entre les nœuds et fonctionnent selon un modèle de publication/abonnement (*Publisher/Subscriber*). Le nœud éditeur (*publisher*) envoie ses données sur le *topic* de manière asynchrone. Un nœud abonné (*subscriber*) peut alors s'y connecter afin de recevoir automatiquement les données dès qu'elles sont publiées. Un même *topic* peut être partagé par plusieurs nœuds éditeurs et abonnés. Un schéma explicatif de ce fonctionnement est présenté à la figure 4.1.

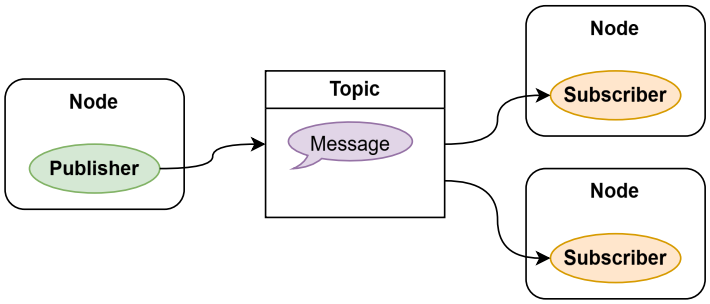


FIGURE 4.1 – Fonctionnement d'un topic sous ROS 2 - Figure inspirée de [13]

Les services Ils permettent de réaliser un échange ponctuel de données entre deux nœuds, sur la base d'un modèle client/serveur. Le nœud client transmet une requête au nœud serveur, qui traite l'information et lui renvoie une réponse. Il s'agit d'un mode de communication synchrone : le nœud client bloque son exécution et attend la réponse du serveur avant de poursuivre sa tâche. Un schéma explicatif de ce fonctionnement est présenté à la figure 4.2.

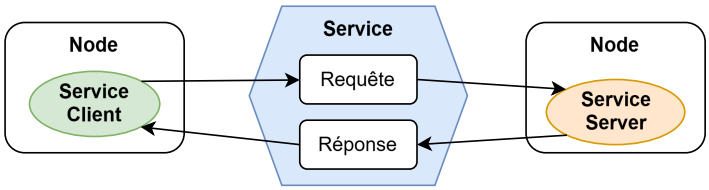


FIGURE 4.2 – Fonctionnement d'un service sous ROS 2 - Figure inspirée de [13]

Les actions Une action combine les concepts de topic et de service. Le nœud *action client* transmet un objectif (*goal*) au *action server* et s'assure de sa bonne réception par le biais d'un premier service dédié (*Goal Service*, Étape 1). Le nœud *action server* exécute ensuite la tâche et publie périodiquement l'avancement de son exécution (*feedback*) via un *topic* (Étape 2). Enfin, le *action client* demande le résultat final auprès du *action server*, qui lui répond par l'intermédiaire d'un second service (*Result Service*) uniquement lorsque l'objectif est atteint ou validé (Étape 3).

Par exemple, pour un robot devant se déplacer vers une coordonnée précise, *action client* envoie la position cible. Le *action server* renvoie alors sa position réelle tout au long du déplacement grâce au *feedback*. Une fois l'objectif atteint, le *action server* retourne le résultat final via le service de résultat, ce qui clôture l'action.

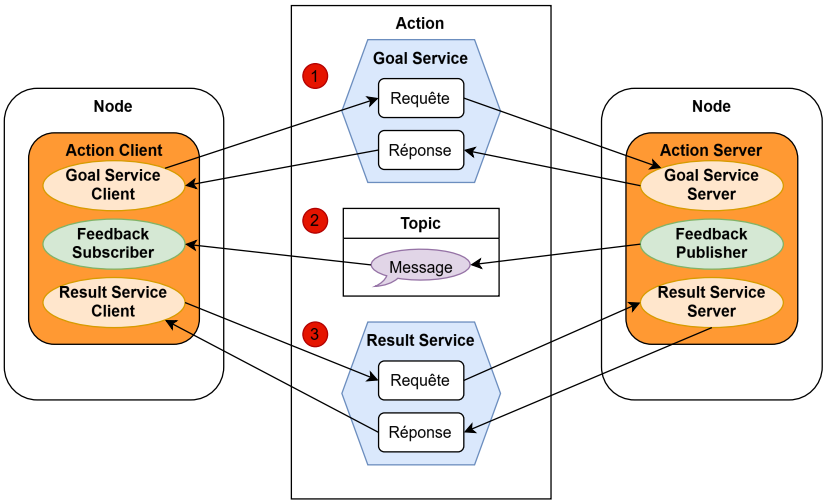


FIGURE 4.3 – Fonctionnement d'un mécanisme d'action sous ROS 2 - Figure inspirée de [13]

La distribution ROS 2 Jazzy Jalisco est utilisée pour ce mémoire. Cette version est supportée jusqu'en mai 2029, ce qui permet de couvrir la fin de la thèse de Marine Boxus. Cependant, après cette date, l'utilisation de la distribution restera possible, mais plus aucun support du code source de ROS2 Jazzy ne sera assuré [28].

4.1.1 Architecture logicielle du banc de test

La Figure 4.4 représente l'architecture ROS2 du banc de test.

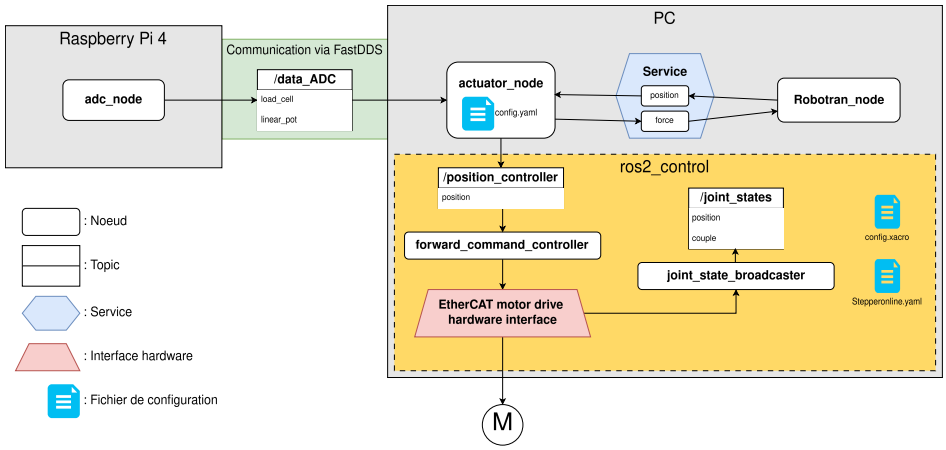


FIGURE 4.4 – Architecture ROS2 du banc de test

4.1.1.1 adc_node

Le nœud ADC a pour tâche de lire la tension mesurée par l'ADC et de la transmettre au nœud actionneur. La tension est lue depuis l'ADC via l'interface SPI, avant d'être transmise au PC via la connexion FastDDS sur le topic /data_ADC.

4.1.1.2 FastDDS

FastDDS est une implémentation open-source du standard DDS (*Data Distribution Service*), elle assure la communication en temps réel entre le Raspberry Pi et le PC via une connexion Ethernet. Les topics et services créés par le Raspberry et le PC peuvent être accédés de manière totalement transparente par les deux machines avec une latence inférieure à 0.2 ms.

4.1.1.3 actuator_node

Le rôle du nœud actionneur est de traduire la consigne de position venue de la simulation en une force. À chaque pas de temps de simulation (toutes les millisecondes), il reçoit donc une requête de position à laquelle il répond par la force mesurée.

Avant le lancement de la simulation, durant une phase d'initialisation, il se charge de trouver la position neutre du transmetteur de force (position à laquelle la force mesurée est nulle), ensuite, durant la simulation, un appel du service se déroule comme suit :

1. La requête de position provenant de la simulation, à appliquer au transmetteur de force, est traduite en position à atteindre par le vérin via la relation géométrique (voir sous-section 5.3)
2. La position que doit atteindre le vérin est publiée sur le topic **/position_controller** afin d'être envoyée au contrôleur du moteur
3. La tension du capteur de force (V_{adc} [V]), mesurée par l'ADC, est lue sur le topic **data_ADC** et traduite en force (F [N]) via la fonction de transfert :

$$F = gain \cdot (V_{adc} - offset) \tag{4.1}$$

4. La force obtenue est envoyée à la simulation en réponse au service.

Le fichier de configuration *config.yaml* contient les informations suivantes relatives au banc d'essai :

— Les paramètres *gain* et *offset* de la fonction de transfert (Équation 4.1) reliant la tension mesurée par l'ADC à la force appliquée au capteur

— La position du transmetteur de force dans le banc de test (1, 2 ou 3)
— La tolérance de force lors de la recherche de la longueur neutre du transmetteur de force (voir section 4.3.2).

4.1.1.4 ros2_control

ros2_control est un framework conçu pour faciliter la communication avec le hardware d'une application robotique. Il permet d'abstraire le matériel en fournissant une interface pouvant directement être utilisée par d'autres nœuds. Les éléments de ros2_control utilisés pour ce banc d'essai et détaillés dans les paragraphes suivants sont une interface hardware et deux contrôleurs :

forward_command_controller Ce contrôleur est le plus simple des contrôleurs disponibles dans le framework ros2_control. Il crée un topic (/position_controller) sur lequel d'autres noeuds peuvent publier des instructions de position à destination du moteur. Le contrôleur transmet ces instructions à l'interface hardware sans les modifier, laquelle se charge de les communiquer au servo-contrôleur.

joint_state_broadcaster Ce contrôleur lit les valeurs provenant des interfaces d'état de l'interface hardware et les publie sur le topic /joint_states. Celà permet d'avoir accès à la position réelle du moteur et au couple qu'il développe, il n'est donc pas utilisé lors d'une simulation mais s'avère utile à des fins de débugage.

EtherCAT motor drive hardware interface L'interface hardware prend en charge l'envoi et la réception des paquets EtherCAT entre le PC et le servo-contrôleur nécessaires à la transmission des instructions de position et à la lecture de la position réelle et du couple moteur. Elle se charge aussi de l'initialisation du servo-contrôleur et de son redémarrage en cas d'erreur. La communication d'une interface hardware avec le reste du framework ros2_control s'effectue via des interfaces de commandes et des interfaces d'état, les interfaces utilisées dans le cas du banc de test sont les suivantes :
— Interface de commande : **position** : reçoit des consignes de position provenant d'un contrôleur et les transmet via la connexion EtherCAT
— Interface d'état : **position** : lit la position réelle du moteur et l'expose aux contrôleurs
— Interface d'état : **effort** : lit le couple développé par le moteur et l'expose aux contrôleurs.

Cette interface est composée de deux modules, le premier est le driver EtherCAT, développé par *ICube Robotics* [29], il assure la création des paquets EtherCAT ainsi que l'initialisation du servo-contrôleur. Le second contient les informations propres au servo-contrôleur ainsi qu'au vérin grâce aux fichiers de configuration suivants :
— config.xacro : définit l'axe prismatique correspondant au vérin, avec notamment ses limites de position
— Stepperonline.yaml : contient les informations propres au servo-contrôleur, les adresses correspondant aux valeurs échangées (position, couple) ainsi que les limites de couple et de vitesse de rotation. Le facteur de conversion permettant de traduire la position de l'encodeur en millimètres au bout du vérin y est également renseigné. Avec l'encodeur offrant une résolution de 17 bits par tour et une vis ayant un pas de 10 mm, le facteur transformant la position en mètre en la position de l'encodeur est $\frac{2^{17}}{10} \cdot 1000 = 13107200$

4.1.1.5 Nœud Robotran

Ce nœud permet d'exécuter une simulation Robotran (agissant comme le client du service) et de transmettre les informations de position au nœud actionneur (serveur du service). Cette

communication sous forme de service permet de garantir une force en retour nécessaire au calcul du vecteur c pour la résolution du système différentiel algébrique (DAE) (voir Figure 1.4).

Comme mentionné ci-dessus, seule la position relative est envoyée via le service (dans le cas d'un ressort, la position envoyée $x = 0$ correspond à une force $F = 0$). Le cahier des charges stipulait d'appliquer à la fois une position et une vitesse au niveau du transmetteur de force. Cependant, la simulation fonctionnant en temps réel, la consigne de vitesse n'est pas nécessaire : le suivi des positions à haute fréquence garantit implicitement celui de la vitesse. La figure 4.5 compare la vitesse calculée par Robotran de la dérivée numérique de la position envoyée par Robotran.

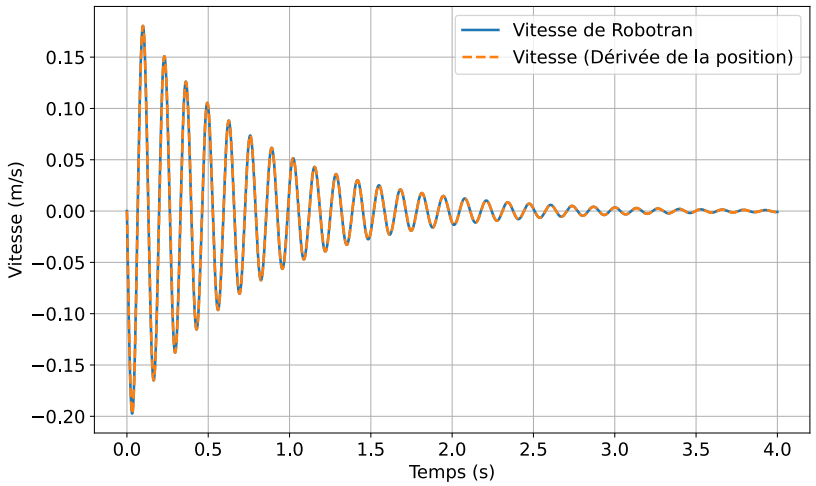


FIGURE 4.5 – Vérification du suivie de la vitesse en temps réel

D'un point de vue logiciel, un projet Robotran est écrit en C tandis que ROS 2 est développé en $C++$. Il a donc fallu intégrer le code Robotran au sein d'un projet ROS 2 et modifier son processus de compilation. Un tutoriel détaillant cette procédure est disponible dans la documentation de Robotran [30].

La figure 4.6 illustre la distribution des fichiers présents dans le nœud Robotran. Un fichier externe au projet original (`force_client.cpp`) permet d'accéder aux fonctions d'initialisation du nœud ROS 2 afin de les exécuter dans le `main.cpp` de Robotran et de transmettre les positions via le service. Ce nœud est initialisé au sein d'une variable globale afin d'être accessible par l'ensemble des fichiers du projet. Enfin, les fichiers `user_LinkForce.c` et `user_JointForce.c` font appel au service à chaque pas de simulation.

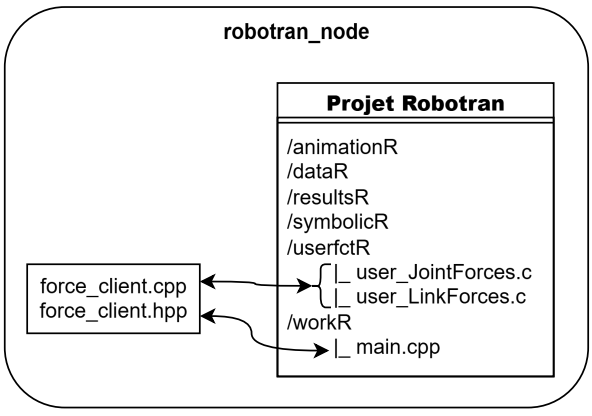


FIGURE 4.6 – Conception du nœud Robotran.

4.2 Choix du système d'exploitation temps réel

L'utilisation d'un Raspberry Pi dans le banc d'essai a justifié le choix de Linux sur celui-ci ainsi que sur le PC. Le comportement de l'OS (*Operating System*) est critique lors d'applications en temps réel. En effet, dans le cadre d'une simulation Hardware In the Loop, si le service reliant le noeud Robotran et le noeud actionneur vient à dépasser le temps qui lui est alloué (voir sous-section 1.2.4), cela crée un blocage qui empêche la simulation de suivre le temps réel. Il a donc été choisi d'utiliser le noyau **linux-rt** qui est conçu pour offrir un déterminisme temporel amélioré par rapport au noyau Linux classique, en dépit de performances générales diminués.
La Figure 4.7 compare la distribution des temps de réponse du service pour le noyau Linux classique et temps réel. Comme attendu, la distribution des temps de réponse avec le noyau real time est beaucoup plus concentrée et le temps de réponse maximal est de 0.405 ms, bien inférieur au temps alloué. En revanche, pour le noyau Linux classique, la distribution est plus large, et le maximum est de plus de 4 ms, ce qui correspond à plus de 4 pas de temps de simulation, ce qui n'est pas acceptable.

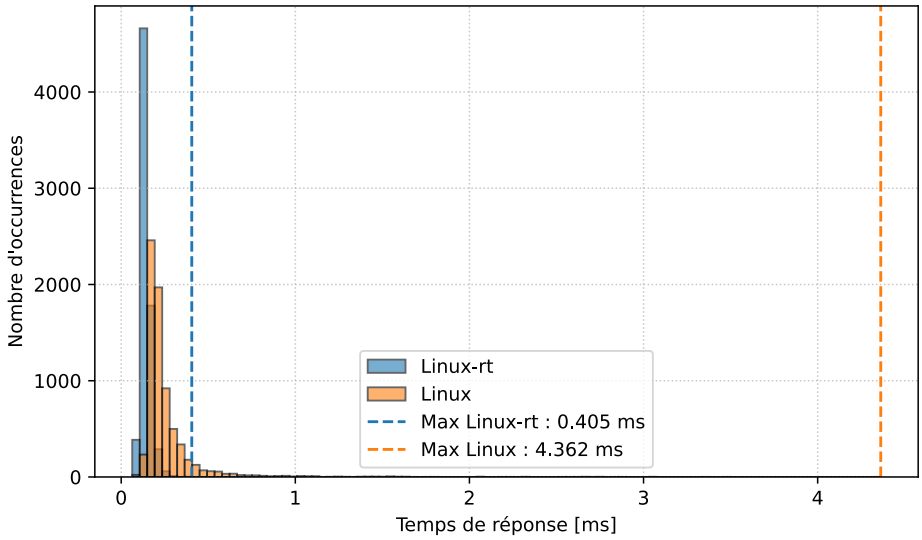


FIGURE 4.7 – Distribution des temps de réponse du service entre le noeud Robotran et actionneur en fonction du noyau Linux

4.3 Initialisation d'une simulation

4.3.1 Procédure de retour à l'origine

Comme précisé dans la section 3.1.5 présentant l'actionneur, l'encodeur du servomoteur "oublie" sa position en cas de coupure d'alimentation. Il est donc nécessaire d'établir une procédure de retour à l'origine afin de pouvoir traduire de manière fiable l'extension du vérin à une valeur de l'encodeur du moteur.

Dans ce but, le contrôleur du moteur dispose de plusieurs procédures préprogrammées, la plus simple consistant à déplacer le vérin jusqu'en butée (complètement rétracté ou étendu). Le contrôleur détecte cette butée en mesurant le couple nécessaire au déplacement qui augmente brusquement lors du contact avec la butée. Cette solution nécessite que le vérin puisse atteindre la fin de sa course dans une des deux directions, ce qui n'est malheureusement pas garanti lorsqu'un transmetteur de force est installé dans le banc d'essai, il est donc nécessaire d'opter pour une solution plus développée.

Une autre solution consiste à utiliser un capteur positionné de manière à détecter la position horizontale du bras de levier. Ce capteur doit changer d'état lorsque l'angle du bras de levier passe de 0° à un angle inférieur, il est donc actif lorsque l'angle se situe entre 0° et une valeur supérieure arbitraire. Celà permet donc au moteur de trouver sa position de deux manières :

1. Si le capteur est inactif, le vérin s'étend à grande vitesse jusqu'à activer le capteur, puis se rétracte lentement pour le désactiver avant de s'étendre à nouveau à basse vitesse. Lors de ce déplacement, le moteur s'arrête dès que le capteur s'active à nouveau et réinitialise l'encodeur à 0.
2. Si le capteur est actif, le vérin se rétracte lentement jusqu'à désactivation du capteur avant de s'étendre à nouveau à basse vitesse. Lors de ce déplacement, le moteur s'arrête dès que le capteur s'active à nouveau et réinitialise l'encodeur à 0.

Cette procédure est représentée graphiquement à la Figure 4.8.

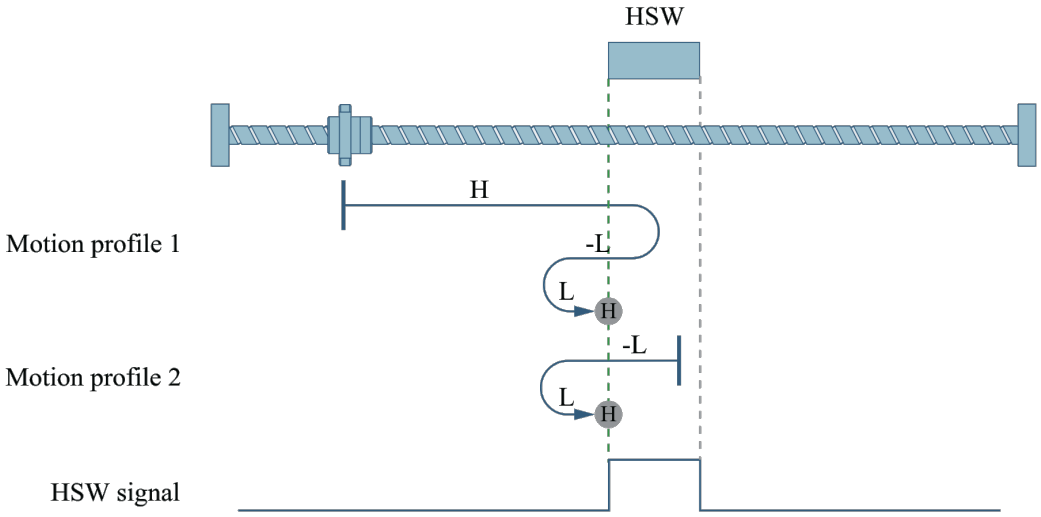


FIGURE 4.8 – Procédure de retour à l'origine, le signal HSW est celui du capteur

Le capteur utilisé est le **SICK MHT15-P2247** représenté sur la Figure 4.9, il s'agit d'un capteur de proximité photoélectrique. Il émet un rayon infrarouge et en détecte la réflexion lorsqu'un objet lui est suffisamment proche. Il est positionné face au bras de levier de sorte qu'il s'active lorsque celui-ci lui fait face, comme illustré par la photo de la Figure 4.10

FIGURE 4.9 – Capteur SICK MHT15-P2247

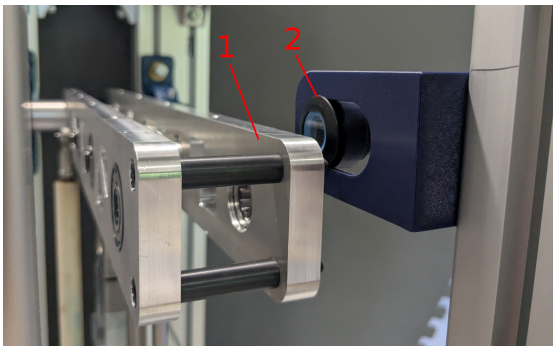


FIGURE 4.10 – Montage du capteur de retour à l'origine (2) devant le bras de levier (1)

4.3.2 Recherche de la longueur neutre du transmetteur de force

Cette étape consiste à trouver la position du transmetteur de force à laquelle aucune force n'y est mesurée. Il est donc important de définir à quoi correspond une force mesurée de 0N. La procédure de calibration du capteur pour cette référence de 0N est expliquée dans la section 5.1.1. Dans le cas du transmetteur de force testé, il s'agit de la force mesurée lorsque le ressort est complètement relâché, la masse de la partie en rouge à la Figure 5.16 est donc ignorée.

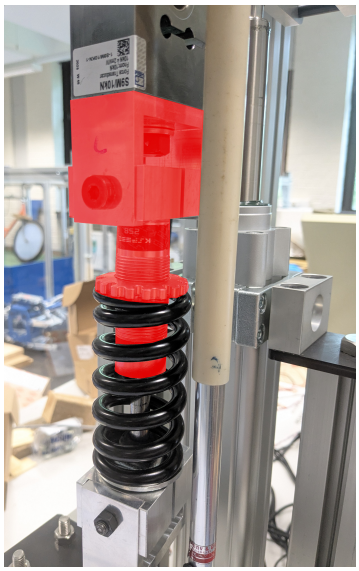


FIGURE 4.11 – Transmetteur de force étudié avec, en rouge, les composants dont la masse est ignorée lors de la mise à zéro du capteur de force

Comme la fonction reliant la force en fonction de la position du ressort est considérée comme une fonction monotone, la longueur neutre est trouvée via une recherche dichotomique. Pour commencer, une position de départ ainsi qu'une largeur de fenêtre de recherche sont fournies, le vérin se déplace aux deux extrémités de la fenêtre ainsi qu'à son centre et mesure la force à ces 3 points. L'opération est répétée sur la moitié de fenêtre contenant le 0 jusqu'à atteindre une valeur suffisamment proche de 0N, cette tolérance est fournie dans les fichiers de configuration.

Chapitre 5

Performance et caractérisation

Ce chapitre est divisé en plusieurs sections visant à évaluer les performances de l'ensemble des sous-systèmes du démonstrateur. Il détaille ensuite les différentes étapes franchies ainsi que les analyses menées pour aboutir à la réalisation d'une simulation Hardware-in-the-Loop appliquée à un VTT.

5.1 Performance de la chaîne d'instrumentation

Cette section a pour objectif d'expliquer la calibration de la chaîne d'instrumentation ainsi que d'en analyser ses performances.

5.1.1 Calibration du conditionneur

Le conditionneur CPJ possède trois réglages majeurs : la sensibilité associée au capteur, le gain d'amplification ainsi que le zéro.

Tout d'abord, le réglage du CPJ a été choisi de manière à obtenir une tension maximale (10 V) lors de l'application d'une force de 5 kN. Cela permet d'obtenir une résolution plus fine au niveau de l'ADC (convertisseur analogique-numérique), tout en restant dans les limites des forces maximales applicables définies par le cahier des charges (3 kN).

Le réglage de la sensibilité sur le CPJ peut être effectué sur base de la sensibilité du capteur fourni par le fabricant ($S_{\text{capteur, 10 kN}} = 1{,}999$ mV/V). Il est dès lors possible de calculer la sensibilité équivalente pour l'application d'une charge maximale de 5 kN à l'aide de la relation suivante :

$$S_{\text{capteur, 5 kN}} = \frac{5\text{ kN}}{10\text{ kN}} \times S_{\text{capteur, 10 kN}} = 0{,}995\text{ mV/V} \tag{5.1}$$

Cela permet de régler la gamme de sensibilité du conditionneur.

Les deux derniers réglages peuvent être déterminés en faisant varier la force appliquée au capteur à l'aide de masses étalonnées. Cependant, le capteur pouvant supporter une force maximale de 10 kN, cela nécessiterait des poids considérables pour effectuer cette calibration.

La calibration du CPJ a donc été effectuée en utilisant un simulateur de signal conçu par Alex Berthollet, dont le but est de simuler le comportement d'un capteur de force à jauges de contrainte. Ce simulateur permet de faire varier la sensibilité par pas de 0,2 mV/V, simulant ainsi l'application de différentes charges physiques sur le capteur. Un récapitulatif de l'équivalence entre sensibilité, signal de sortie du capteur, tension de sortie du CPJ et Force équivalente est comprise dans le Tableau 5.1 :

La procédure de calibration du gain est la suivante :

1. **Ajustement du zéro :** Calibrer le zéro du CPJ (0 V en sortie) lors de la simulation d'une charge de 0 N.

Sensibilité (mV/V)	Sortie capteur (mV)	Sortie Conditionneur (V)	Force (N)
0	0	0	0
0.2	1	2	1000
0.4	2	4	2000
0.6	3	6	3000
0.8	4	8	4000

TABLEAU 5.1 – Récapitulatif des équivalences des données pour le capteur de force et le conditionneur

2. **Application d'une charge intermédiaire :** Appliquer une charge connue au simulateur (0,8 mV/V), ce qui correspond à 4 kN et doit délivrer une tension de sortie de 8 V. Le gain doit alors être ajusté pour atteindre cette valeur.
3. **Optimisation de la précision :** Afin d'être le plus précis possible, il convient de mesurer la tension d'excitation du capteur ($V_{\text{exc}} = 5{,}0061$ V) et le signal de sortie ($V_{\text{sig}} = 4{,}008$ mV). En effet, comme mentionné lors de la phase de conception, le signal de sortie est directement proportionnel à la tension d'excitation. Il est alors possible de faire correspondre la sensibilité exacte du capteur pour une charge de 4 kN (soit 8 V en sortie du CPJ, voir tableau 5.1), ce qui donne $S_{\text{capteur, 4 kN}} = 0{,}7996$ mV/V. Le gain peut ensuite être ajusté jusqu'à obtenir en sortie :

$$V = \frac{\frac{V_{\text{sig}}}{V_{\text{exc}}}}{S_{\text{capteur, 4 kN}}} \times 8 \text{ V} = 8{,}010235 \text{ V} \tag{5.2}$$

Cette méthode permet d'ajuster le gain en compensant l'impact de la tension d'alimentation et en tenant compte de la sensibilité réelle du capteur de force. Le gain du CPJ est désormais calibré.

Une fois le gain calibré, le simulateur est remplacé par le capteur de force afin de procéder au calibrage de son zéro. Pour ne pas fausser les mesures par des efforts parasites (ne provenant pas uniquement du transmetteur de force), le zéro a été ajusté après le montage complet du capteur dans le démonstrateur, en s'assurant que le ressort se trouvait parfaitement au repos (ne transmettant ainsi aucune force au capteur). Cette procédure garantit que la force injectée dans la simulation multi-corps provient exclusivement de la réponse élastique du ressort.
La figure 5.1 illustre les différents éléments mécaniques dont le poids propre est compensé et annulé par le capteur lors de cette étape (le poids du capteur lui-même, les systèmes de fixation, ainsi que le guidage externe du transmetteur de force).

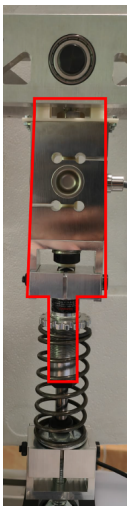


FIGURE 5.1 – Éléments mécaniques dont le poids est annulé par le conditionneur (propre poids, fixations et guidage externe).

Une vérification de la calibration a, par la suite, été effectuée sur base d'une masse de 1kg. L'ajout de cette masse sollicite le capteur en traction, ce qui explique le signe négatif (Voir Figure 5.2, 5.3).

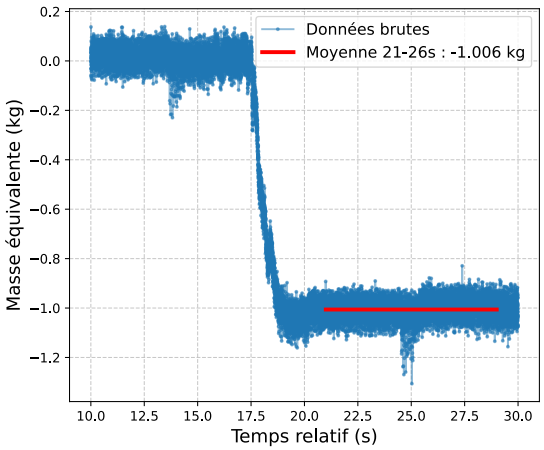


FIGURE 5.2 – Application d'une masse de 1 kg sur le capteur de force

FIGURE 5.3 – Force du ressort 2 en fonction de la distance et calcul de la raideur.

5.1.2 Conversion de la mesure de tension du potentiomètre

Le potentiomètre linéaire HLP129 fait office de diviseur de tension sur une plage de 0 à 10 V. L'ADC opérant sur une plage d'entrée de ±10 V, seule la moitié de sa dynamique admissible est exploitée. La course de ce capteur présentant une caractéristique linéaire, la relation entre la tension mesurée et la position peut être modélisée par l'équation d'une droite :

$$d = m \cdot V + b \tag{5.3}$$

Les paramètres de cette droite sont déterminés à partir des caractéristiques de distance géométrique lorsque le potentiomètre est totalement rétracté ou totalement déployé, données fournies par la fiche technique du fabricant [16]. Ces distances limites ainsi que les mesures de tension correspondantes lues sur l'ADC sont répertoriées dans le tableau 5.2.

Configuration	**Distance entre les rotules d [mm]**	**Tension lue sur l'ADC V [V]**
Totalement rétracté	$d_r = 298{,}6$	$V_r = 10{,}043$
Totalement déployé	$d_d = 448{,}6$	$V_d = -0{,}015$

TABLEAU 5.2 – Données géométriques et tensions associées du potentiomètre HLP129.

La pente m de la droite est calculée par la relation suivante :

$$m = \frac{d_d - d_r}{V_d - V_r} = \frac{448{,}6 - 298{,}6}{-0{,}015 - 10{,}043} = -14{,}913 \text{ mm/V} \tag{5.4}$$

L'ordonnée à l'origine b est ensuite déduite :

$$b = d_d - (m \cdot V_d) = 448{,}6 - (-14{,}913 \cdot (-0{,}015)) = 448{,}376 \text{ mm} \tag{5.5}$$

Cette loi de comportement linéaire est par la suite implémentée pour convertir la tension brute du potentiomètre mesurée par l'ADC en une distance inter-axes.

5.1.3 Méthode d'acquisition et performances de la chaîne d'instrumentation en statique

Des mesures statiques d'une durée de 15 secondes ont été réalisées afin de comparer les deux méthodes d'acquisition détaillées à la section 3.2.3.

La méthode de lecture à valeur unique est cadencée par une boucle temporelle à 1 kHz. Cette configuration permet d'acquérir une donnée sur chaque canal toutes les millisecondes et est ensuite publiée sur le topic `data_ADC`.

La méthode de lecture en continu est configurée pour acquérir 40 échantillons par canal (force et position) avec une fréquence d'échantillonnage de 40 kHz par canal. L'acquisition s'effectue en alternance entre les deux canaux, moyennant un temps de mesure fixe de 9.8 μs par conversion. Cette approche génère un échantillon de 40 mesures de force et 40 mesures de position par milliseconde. Ces échantillons subissent ensuite un moyennage numérique avant leur publication sur le topic `data_ADC`.

La figure 5.4 illustre la distribution du bruit autour de la moyenne pour ces deux modes de lecture :

— **Canal de Force :** Le mode à lecture unique présente une dispersion très importante, avec des valeurs atteignant −30 mV. L'écart-type s'élève à 7.12 mV (soit une incertitude de 3.56 N) pour une amplitude d'erreur crête-à-crête de plus de 21 N. En revanche, l'approche continue transforme cette distribution en une courbe gaussienne. L'écart-type est réduit à 0.83 mV (0.42 N), divisant ainsi le bruit par un facteur d'environ 8.5. De plus, 98.5 % des mesures acquises dans le mode continu se situent dans un intervalle de ±1 N autour de la valeur moyenne, ce qui garantira une fiabilité lors de la procédure de recherche du zéro de force.

— **Canal de Position :** Le constat est similaire pour le capteur de distance. L'écart-type chute drastiquement de 22.21 mV (0.33 mm) en lecture unique, à seulement 2.41 mV (0.04 mm) en lecture continue, témoignant d'une réduction du bruit par un facteur 9.

Cette diminution significative confirme la nécessité d'utiliser une méthode d'acquisition continue (sur-échantillonnage suivi d'un moyennage) pour filtrer efficacement le bruit. L'hypothèse selon laquelle ce moyennage a un impact négligeable sur l'amplitude et le retard du signal, dont la dynamique maximale est de 25 Hz, est vérifiée dans l'Annexe A via l'analyse de la réponse impulsionnelle du filtre. Le décalage temporel théorique s'élève à 0,4875 ms pour une atténuation de l'amplitude du signal de seulement 0,1%.

5.1.4 Impact du filtre interne du conditionneur CPJ en statique

Comme expliqué dans le chapitre de conception (voir section 3.2.1.3), le conditionneur CPJ possède un filtre analogique passe-bas d'ordre 2 doté d'une fréquence de coupure à 10 Hz. L'impact de ce filtre doit être étudié, notamment dans le cas où le démonstrateur HIL serait utilisé pour une simulation présentant une dynamique supérieure à cette fréquence de 10 Hz.

La distribution des données autour de leurs moyennes respectives est présentée sur la figure 5.5. Les deux jeux de données ont été acquis de manière consécutive avec la même configuration du démonstrateur. On observe une différence de 0,78 N entre les moyennes de ces deux séries. Cela s'explique par le fait que le CPJ utilise un filtre analogique : les tolérances des composants réels utilisés pour le filtrage peuvent impacter l'amplitude finale du signal. Par conséquent, si une utilisation du démonstrateur nécessitant une fréquence supérieure à 10 Hz, un recalibrage préalable sera indispensable pour compenser ces variations.

Le tableau 5.3 répertorie les caractéristiques statistiques calculées sur les mesures. L'écart-type est divisé par deux grâce au filtrage : 97,81% des données se situent alors dans un intervalle de ±1 N autour de la valeur moyenne, contre seulement 68,97% dans le cas non filtré.

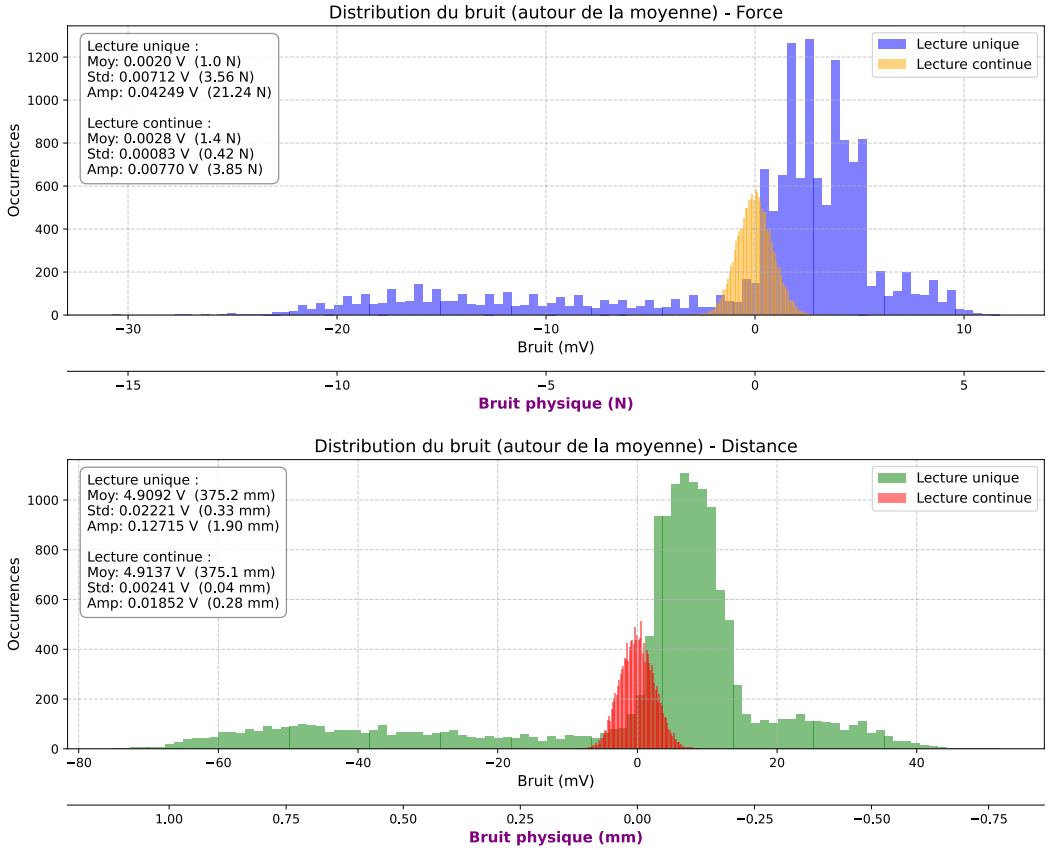


FIGURE 5.4 – Distribution du bruit de la chaîne d'instrumentation en statique (comparaison des modes d'acquisition)

Statistique	**Filtre 10 Hz**	**Sans Filtre**
Moyenne	1.68 N	0.9 N
Écart-type	0.44 N	0.99 N
Intervale de tolérance $\pm 0.25\ N$	42.07 %	19.75%
Intervale de tolérance $\pm 0.5\ N$	74.27 %	38.27%
Intervale de tolérance $\pm 0.75\ N$	91.52 %	54.94 %
Intervale de tolérance $\pm 1.0\ N$	97.81 %	68.97 %

TABLEAU 5.3 – Comparaison statistique de l'acquisition de l'ADC avec et sans le filtre interne du conditionneur.

5.2 Caractérisation de la déformation du capteur et de la structure

L'objectif de cette démarche est de garantir la plus haute précision possible concernant le déplacement imposé au niveau du transmetteur de force (TF), puisque ce déplacement dicte directement l'effort appliqué. Afin d'isoler et de caractériser déformation propre du capteur et de la structure mécanique du démonstrateur, le TF est remplacé par un corps rigide. L'essai est ensuite mené en augmentant progressivement la charge appliquée.

Avant toute application d'effort, il convient de s'assurer que ce corps rigide ne risque pas de flamber, un tel phénomène pouvant induire des efforts parasites susceptibles d'endommager le capteur. La charge maximale admissible sur ce corps rigide, présentant une longueur de 300 mm entre rotules, est de 10 643 N. Le détail de ce calcul est présenté dans l'Annexe D.

La déformation globale du système a été mesurée pour la position centrale du transmetteur

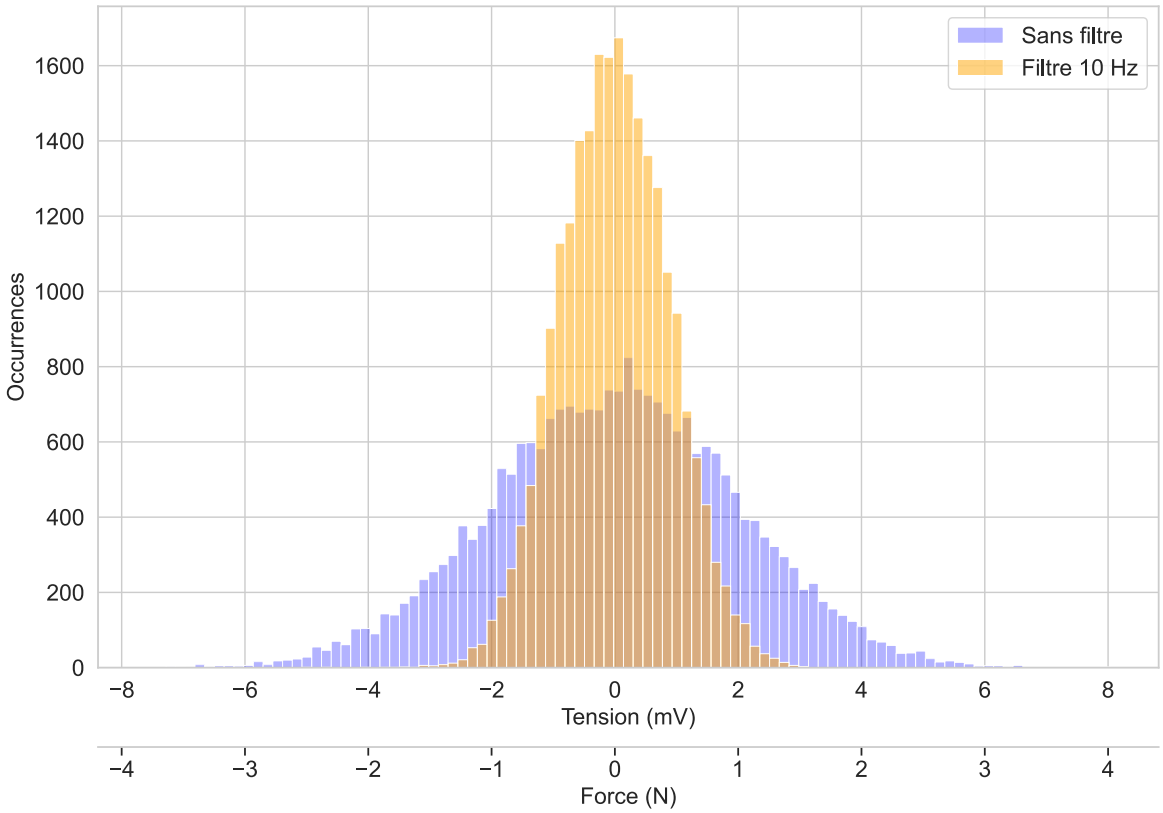


FIGURE 5.5 – Distribution du bruit d'une mesure statique, centrée sur sa moyenne respective, avec et sans filtrage analogique du conditionneur.

de force (TF). Lors de cet essai, les mesures de position ont été enregistrées, d'une part, via le potentiomètre linéaire monté en parallèle du TF et du capteur, celui-ci étant fixé entre les deux axes (voir Figure 5.9), et d'autre part, via l'encodeur du moteur en appliquant la fonction de transfert permettant de transposer le déplacement du vérin à celui du TF.

Des tests en compression et en traction ont été effectués sur le corps rigide ; les résultats sont présentés respectivement sur les Figures 5.6 et 5.7. On remarque un écart de près de 1 mm entre les déformations mesurées par les deux méthodes. Cela est dû au fait que l'élément critique de cette chaîne cinématique est la plaque supportant l'axe inférieur du TF. La déformation mesurée via l'encodeur moteur dans le plan d'application de la force est donc plus importante qu'au niveau de l'extrémité de l'axe où est fixé le potentiomètre linéaire (voir Figure 5.8).

À titre de comparaison, la déformation théorique du seul capteur de force n'est que de 0,09 mm sous une charge de 3 000 N. Cette valeur est bien inférieure à la déformation globale mesurée lors des deux essais. Afin de limiter l'impact de cette déformation structurelle, le test pourrait être reconduit après avoir rigidifié la plaque incriminée. Une telle modification permettrait d'améliorer drastiquement les performances du banc, bien qu'elle n'ait pas pu être implémentée dans le cadre de ce mémoire.

5.3 Caractérisation de la fonction de transfert géométrique

La fonction de transfert permet de traduire la commande de position devant être appliquée au transmetteur de force en la position devant être atteinte par le vérin. Toute erreur lors de cette conversion introduit donc une erreur entre le déplacement demandé par la simulation et celui réellement appliqué au transmetteur de force étudié.

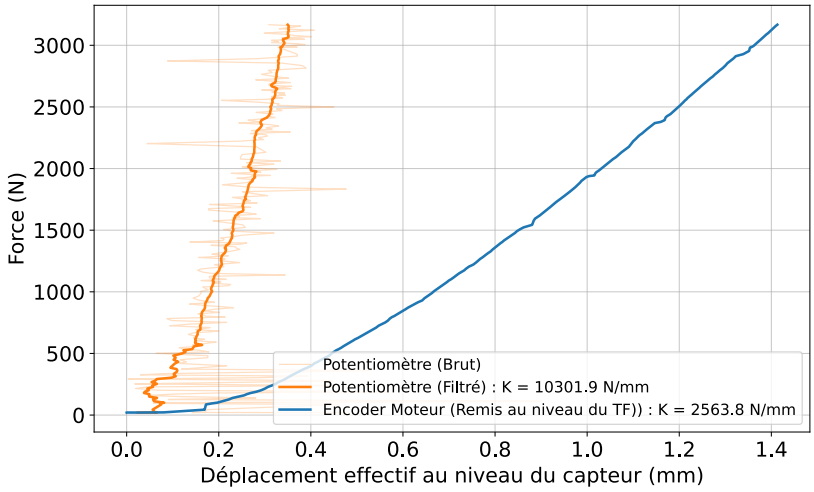


FIGURE 5.6 – Évolution de la déformation mesurée du capteur et de la structure en fonction de la force appliquée en compression.

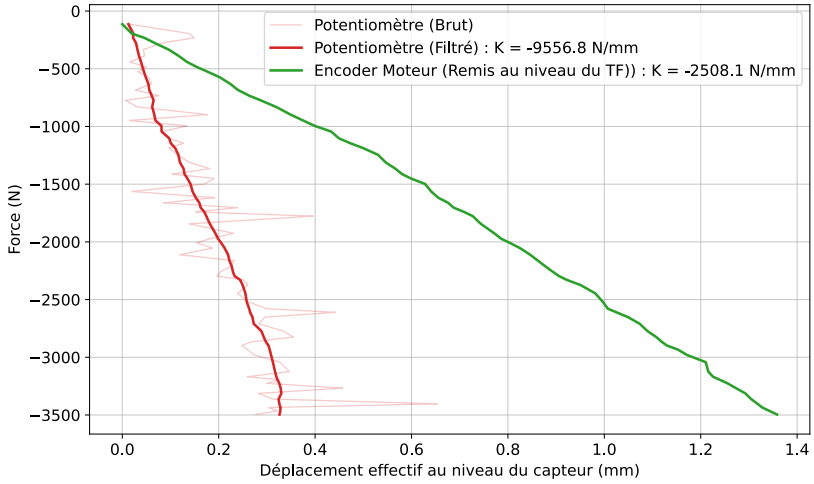


FIGURE 5.7 – Évolution de la déformation mesurée du capteur et de la structure en fonction de la force appliquée en traction.

La Figure 5.10 décrit les dimensions du banc utilisées dans les équations de la fonction de transfert. La longueur du vérin est definie par L_v et celle du transmetteur de force par L_{tf}, L_v peut alors être trouvée grâce aux équations suivantes :

$$\Delta = \frac{L_{tf}^2 - h_{tf}^2 - 2d_{tf}^2}{2d_t f} \tag{5.6}$$

$$R = \sqrt{h_{tf}^2 + d_{tf}^2} \tag{5.7}$$

$$\Theta = \arctan\left(\frac{d_{tf}}{h_{tf}}\right) + \arcsin\left(\frac{\Delta}{R}\right) \tag{5.8}$$

$$L_v = \sqrt{2d_v^2(1 - \cos\Theta) + h_v^2 + 2d_v h_v \sin\Theta} \tag{5.9}$$

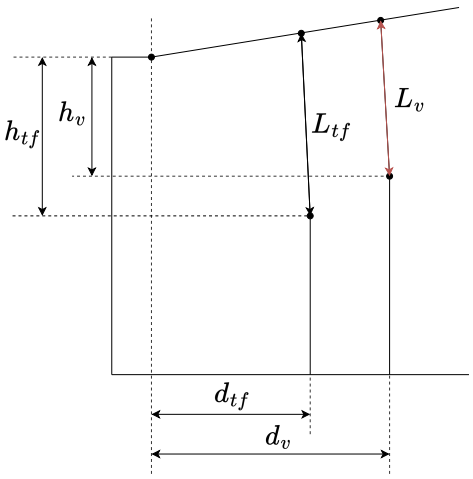


FIGURE 5.10 – Longueurs caractérisant la fonction de transfert géométrique

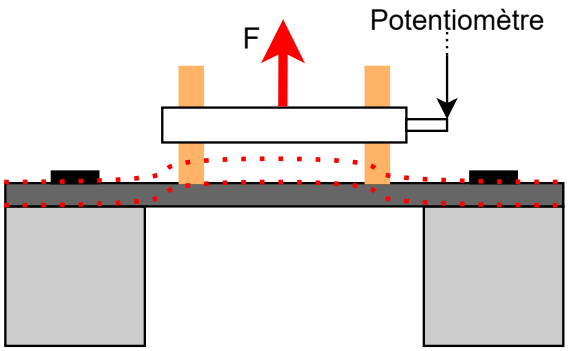


FIGURE 5.8 – Illustration de la déformation de la plaque.

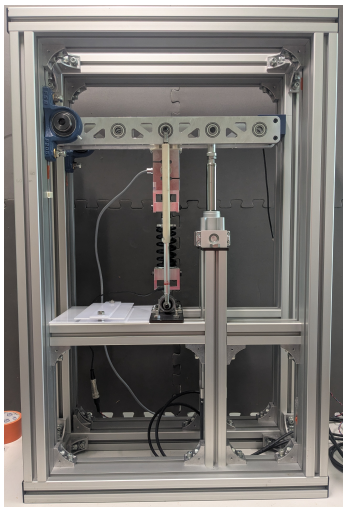


FIGURE 5.9 – Vue globale du démonstrateur.

5.3.1 Erreur de la fonction

Grâce au potentiomètre installé entre les deux axes entraînant le transmetteur de force, il est possible de mesurer cette distance et la comparer au résultat de la fonction théorique. La Figure 5.11 représente la valeur mesurée et théorique pour les trois positions du transmetteur de force au sein du banc de test.

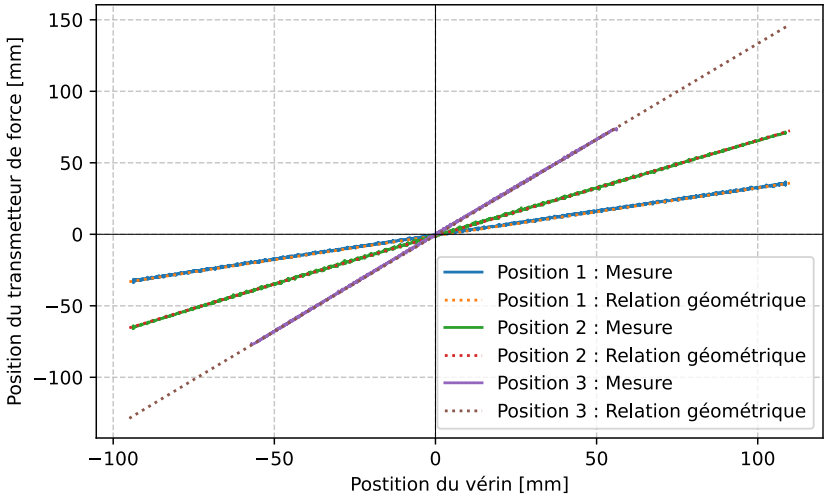


FIGURE 5.11 – Position du transmetteur de force par rapport à la position du vérin

Le bruit affectant les mesures provenant du potentiomètre ne permet pas de formellement affirmer que l'erreur ne dépasse jamais 0.05 mm, cependant, la moyenne du signal ne mesure jamais d'erreur supérieure à 0.05 mm pour les trois positions du transmetteur de force.

5.3.2 Erreur liée à la position d'origine

Comme décrit dans la section 4.3.1, la position d'origine du vérin, correspondant à la position horizontale du bras de levier, est trouvée grâce à un capteur optique. Il est donc important de mesurer l'erreur de ce capteur, ainsi que l'impact de cette erreur sur la position appliquée au transmetteur de force étudié.

Le premier test consiste donc à mesurer la distance entre les deux axes actionnant le transmetteur de force grâce au potentiomètre après plusieurs itérations de la procédure de retour à l'origine. Après 10 itérations, toutes les mesures sont contenues dans une fenêtre de 0.04 mm. La répétabilité de la procédure de retour à l'origine s'avère donc bien suffisante par rapport aux autres sources d'erreur du système.

Cependant, comme la fixation du capteur ne permet pas de garantir qu'il ne soit pas déplacé accidentellement, il est pertinent de mesurer l'erreur de position induite par une erreur de position d'origine. Comme les déplacements appliqués au transmetteur de force sont relatifs à sa longueur neutre, qui est elle-même mesurée avant chaque simulation, cette erreur modifie donc seulement les paramètres de la fonction de transfert. La Figure 5.12 présente donc l'erreur maximale induite par l'erreur de position du capteur sur l'ensemble de la course du vérin, on y observe que, afin de maintenir l'erreur sous les 0.05 mm, l'écart de position du capteur de retour à l'origine doit être maintenu sous les 2 mm. Un simple repère visuel sur le banc d'essai est jugé suffisant pour garantir la bonne installation du capteur.

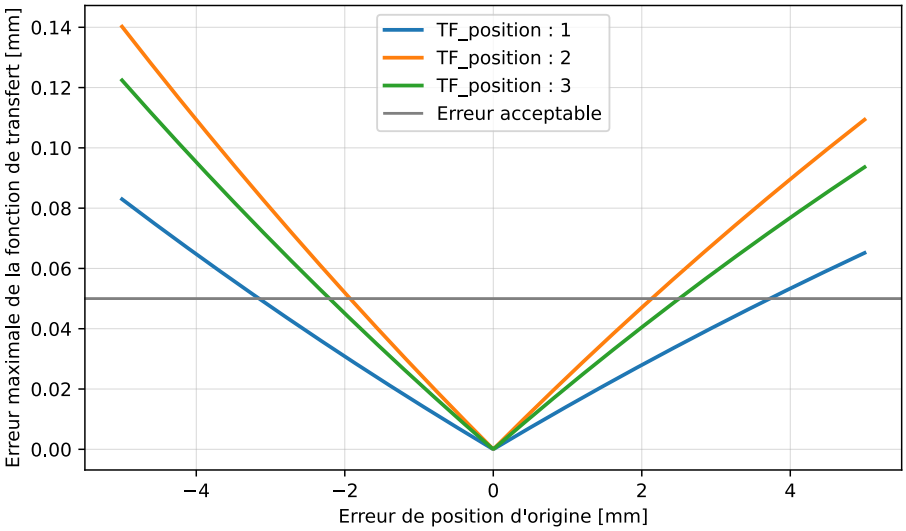


FIGURE 5.12 – Erreur de position maximale du transmetteur de force par rapport à l'erreur d'installation du capteur de retour à l'origine pour les 3 positions possibles du transmetteur de force

5.4 Caractérisation du modèle thermique

Le servo-contrôleur possède une protection interne contre la surchauffe. Celle-ci se base sur un modèle thermique qui n'est pas accessible par l'utilisateur, le fabricant se contente de fournir la courbe représentée sur la Figure 5.13 donnant le temps de fonctionnement avant le déclenchement de la sécurité en fonction du taux de charge du moteur. Ce taux de charge est défini comme le pourcentage du couple nominal développé par le moteur.

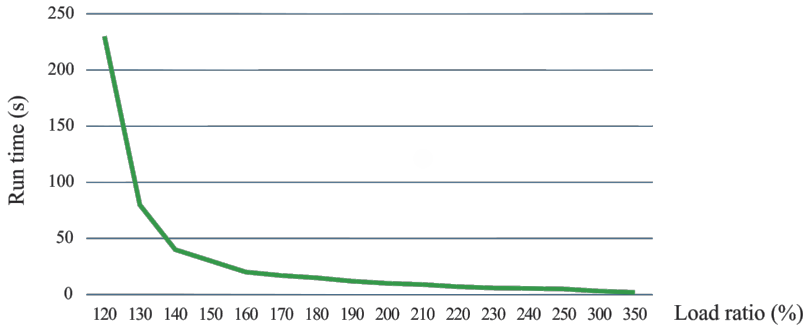


FIGURE 5.13 – Temps de fonctionnement avant le déclenchement de la sécurité thermique en fonction du taux de charge du moteur [7]

Afin de vérifier la véracité de la courbe fournie, une consigne de couple peut être appliquée au moteur bloqué, le temps avant le déclenchement de la sécurité thermique peut alors être mesuré. Ce test a été effectué à 150% et 200% du couple nominal et à cinq reprises tout en laissant le moteur refroidir cinq minutes entre chaque itération. Les moyennes des temps mesurés sont rassemblées dans le Tableau 5.4

Taux de charge appliqué [%]	**Temps mesuré [s]**	**Temps fourni [s]**
150	28.1	29.6
200	8.9	10

TABLEAU 5.4 – Temps de déclenchement de la sécurité thermique mesuré et fourni par le fabricant à 150 et 200% de charge.

Les valeurs mesurées sont légèrement inférieures à celles fournies, mais restent suffisamment proches pour pouvoir tout de même se référer à la courbe du fabricant à titre indicatif.

5.5 Mesure de la raideur des Transmetteurs de Force considérés

De manière à pouvoir comparer les simulations HIL avec des simulations classiques, il est nécessaire de connaître avec précision la raideur des ressorts utilisés dans le transmetteur de force. Le démonstrateur a été exploité pour identifier expérimentalement cette raideur réelle. Les figures 5.14 et 5.15 présentent les données obtenues. En effectuant une régression linéaire, l'équation de la droite correspondante est déterminée. Sa pente donne la raideur de chacun des deux ressorts ($K_1 = 3826{,}6$ N/m et $K_2 = 113294{,}19$ N/m).

5.6 Simulation Hardware In the Loop

5.6.1 Simulation avec deux nœuds

Afin de valider l'utilisation et l'intégration de ROS 2 ainsi que sa communication avec Robotran, une simulation de système masse-ressort-amortisseur a été étudiée (Voir Figure 5.16). Le comportement du démonstrateur a été simulé par un nœud jouant le rôle du serveur de service, renvoyant la force $F = K \cdot x$. Les paramètres du système considéré sont $K = 3826{,}6$ N/m, $C = 50$ N $\cdot$ s/m, $m = 50$ kg et $L_0 = 0{,}5$ m. La figure 5.17 présente la force et la position du ressort dans l'articulation. Aucun décalage temporel n'est observé entre la valeur maximale de la force et la valeur minimale de la position. Le nœud Robotran attend la réception de la force avant de poursuivre sa simulation.

Sur la base de cette simulation, le tableau 5.5 détaille le temps de calcul d'une boucle de simulation Robotran ($\Delta t_{DAE} + \Delta t_{\text{int}}$) ainsi que le délai de communication lié au service

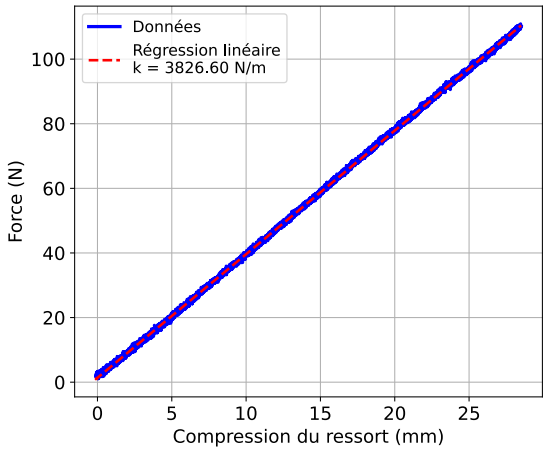


FIGURE 5.14 – Force du ressort 1 en fonction de la distance et calcul de la raideur.

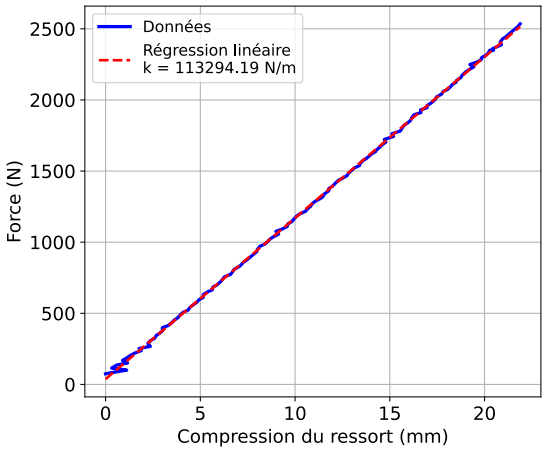


FIGURE 5.15 – Force du ressort 2 en fonction de la distance et calcul de la raideur.

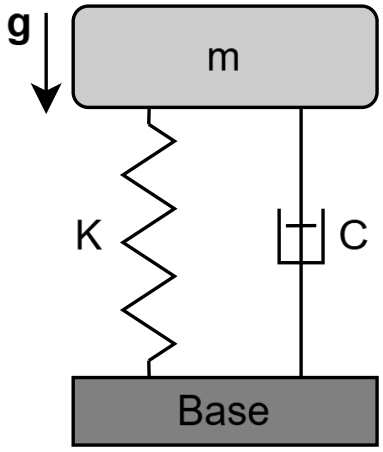


FIGURE 5.16 – Systèmes masse ressort considéré

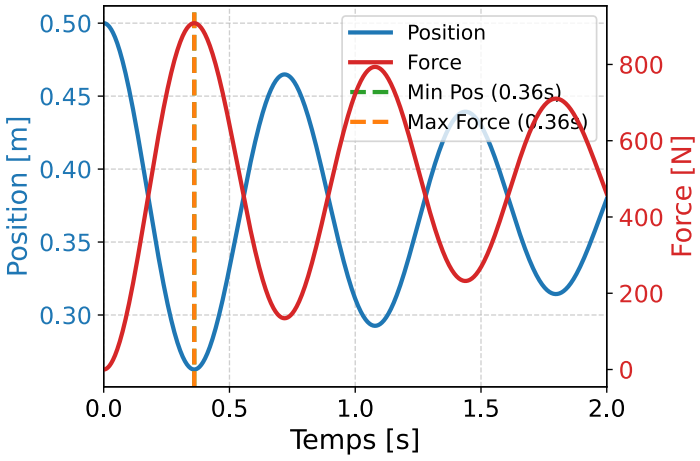


FIGURE 5.17 – Simulation d'un système masse-ressort-amortisseur régi par la gravité.

(Δt_{HIL}), qui correspond au temps écoulé entre l'envoi d'une coordonnée de position et la réception de la force.

$\Delta t_{DAE} + \Delta t_{\text{int}}$	Δt_{HIL}
30,15 μs	0,347 ms

TABLEAU 5.5 – Temps de calcul de la boucle de simulation et délai de communication HIL.

5.6.2 Driven Motion Sinus

Ce test vise à valider l'envoi au moteur d'instructions de position depuis la simulation ainsi que la réception, par la simulation, de la force mesurée. Pour cela, un mouvement sinusoïdal est imposé au ressort, la force correspondant est récupérée par la simulation, mais n'est donc pas utilisée, le mouvement étant imposé.

La Figure 5.18 représente les données provenant d'une simulation dans laquelle l'amplitude du mouvement sinusoïdal est d'environ 15 mm. L'effet du bruit sur la mesure de force est assez significatif, car la faible raideur du ressort implique des forces assez faibles. Lors du changement

de direction après la compression, il est possible d'observer l'effet du guidage introduisant une force parasite qui éloigne la force d'un sinus pur.

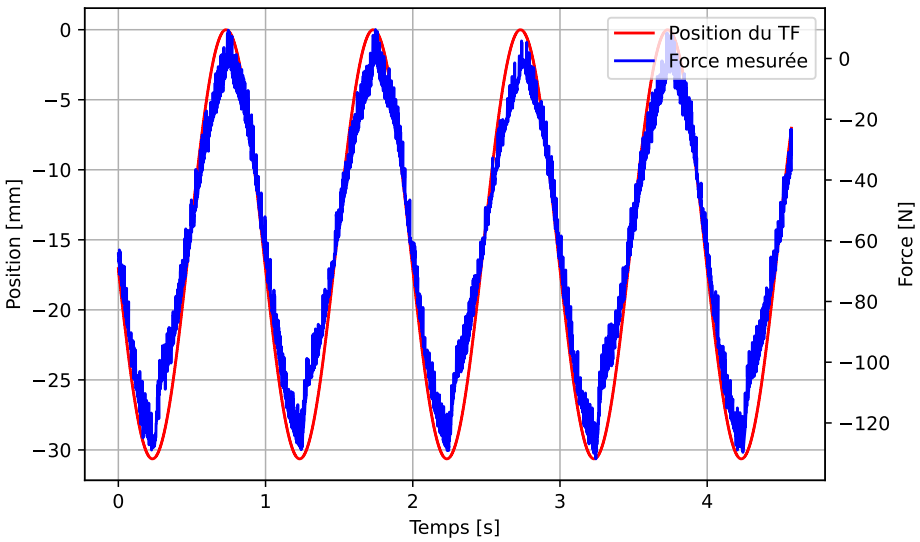


FIGURE 5.18 – Résultat de la simulation avec une amplitude d'environ 15 mm, le ressort étant compressé lorsque sa position diminue

Sur la Figure 5.19, la même simulation est effectuée avec, cette fois, une amplitude d'environ 20 mm. Cela met en évidence la butée en caoutchouc du transmetteur de force, lorsque le ressort est comprimé suffisamment loin, un élément en caoutchouc agit comme un ressort plus raide placé en parallèle du ressort principal. Les forces de ces deux ressorts s'additionnent ce qui provoque un pic de force lorsque la butée est atteinte.

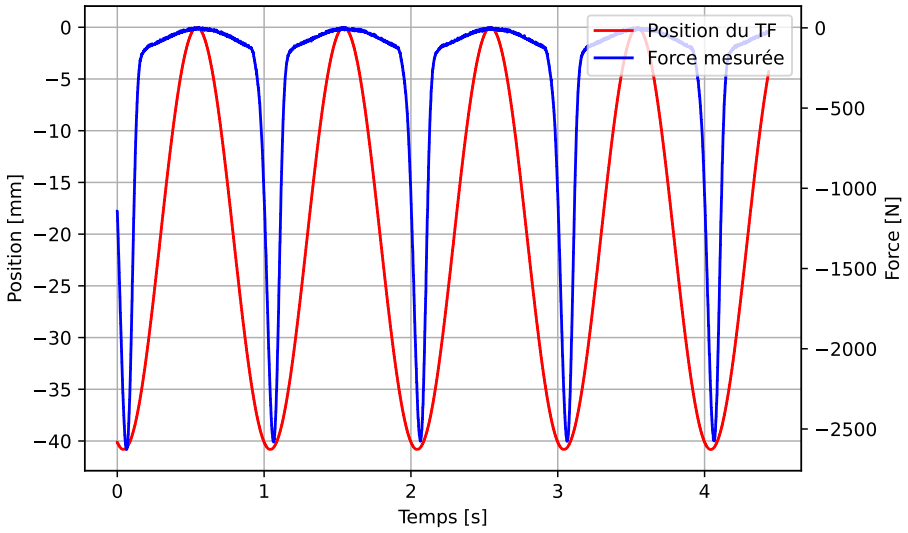


FIGURE 5.19 – Résultat de la simulation avec une amplitude d'environ 20 mm

5.6.3 Système masse-ressort classique avec amortissement

Une simulation de système masse-ressort avec un amortisseur en parallèle, régi par la gravité, a été étudiée. L'ajout de cet amortisseur a pour objectif de réduire la dynamique du système. Il est intégré dans la simulation en parallèle du ressort via un liens (*Link*),

afin d'obtenir un comportement le plus proche possible du régime quasi-statique. La figure 5.16 représente le système considéré, dont les paramètres sont : $K = 113\,294{,}19$ N/m, $C = 10\,000$ N $\cdot$ s/m et $m = 50$ kg.

Les résultats de cette simulation HIL sont présentés sur la Figure 5.20, pour laquelle le filtre analogique interne du conditionneur CPJ était activé. Les courbes en pointillés rouge et bleu clair représentent le comportement du système masse-ressort issu de la simulation numérique pure (référence). La courbe verte correspond à la force appliquée par le transmetteur de force (TF) sur le capteur et renvoyée à la simulation HIL, tandis que la courbe orange trace le déplacement réel du TF.

Une divergence de 0,56 mm est mesurée entre la position simulée et le déplacement du TF. Cet écart peut s'expliquer par le guidage du transmetteur de force, dont la longueur neutre identifiée présente un décalage par rapport à la position d'amorçage de la compression du ressort. On remarque par ailleurs que lorsque la simulation atteint son équilibre, la simulation HIL se met à osciller autour de sa valeur stationnaire. Le zoom de la Figure 5.21 montre que l'oscillation du déplacement présents une amplitude de 64 μm, ce qui engendre une fluctuation de la force d'une amplitude de 48 N à une fréquence d'environ 12 Hz. Un décalage temporel d'environ 15 ms est également visible entre ces deux signaux.

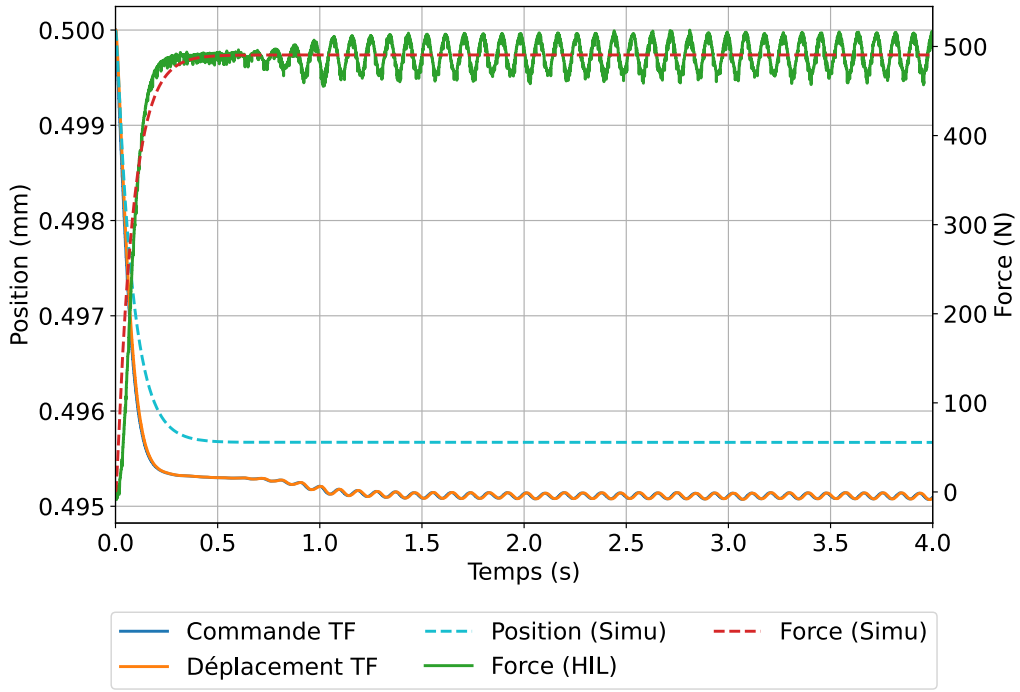


FIGURE 5.20 – Simulation HIL du système masse-ressort avec amortissement (filtre CPJ activé).

Afin de réduire ce délai entre la force et la position, le filtre interne du CPJ a été désactivé. Comme le montre la Figure 5.22, la force de la simulation HIL suit désormais plus fidèlement la force de la simulation classique, bien qu'elle soit davantage bruitée en raison du retrait du filtre analogique. Le décalage en position persiste pour les mêmes raisons que mentionnées précédemment. En analysant le zoom de la Figure 5.23, on constate que le déplacement et la force oscillent à une fréquence plus élevée qu'avec le filtre et est d'environ 26 Hz, tandis que le retard temporel entre la force et le déplacement est réduit à seulement 3 ms.

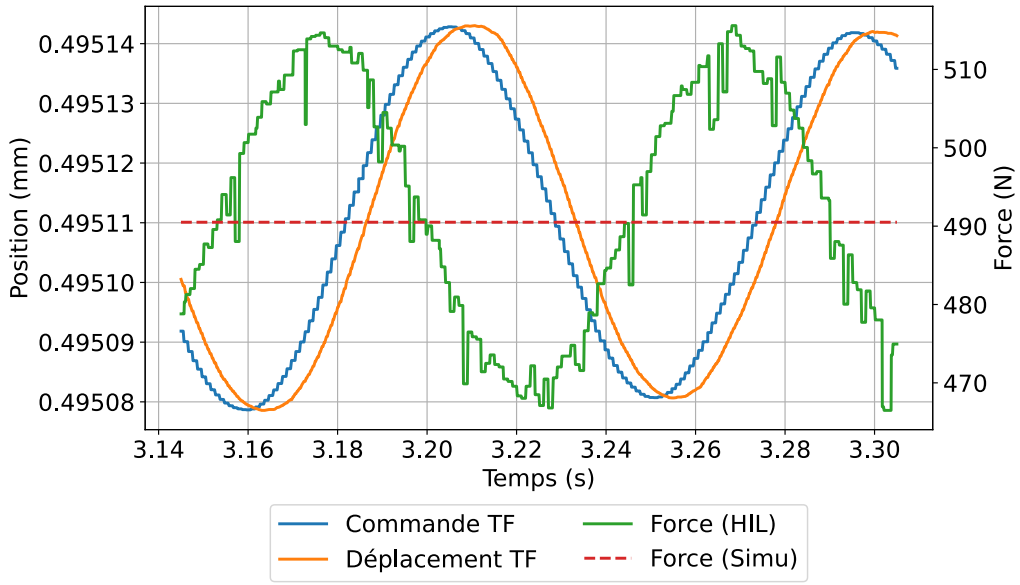


FIGURE 5.21 – Zoom sur les oscillations du système (filtre CPJ activé).

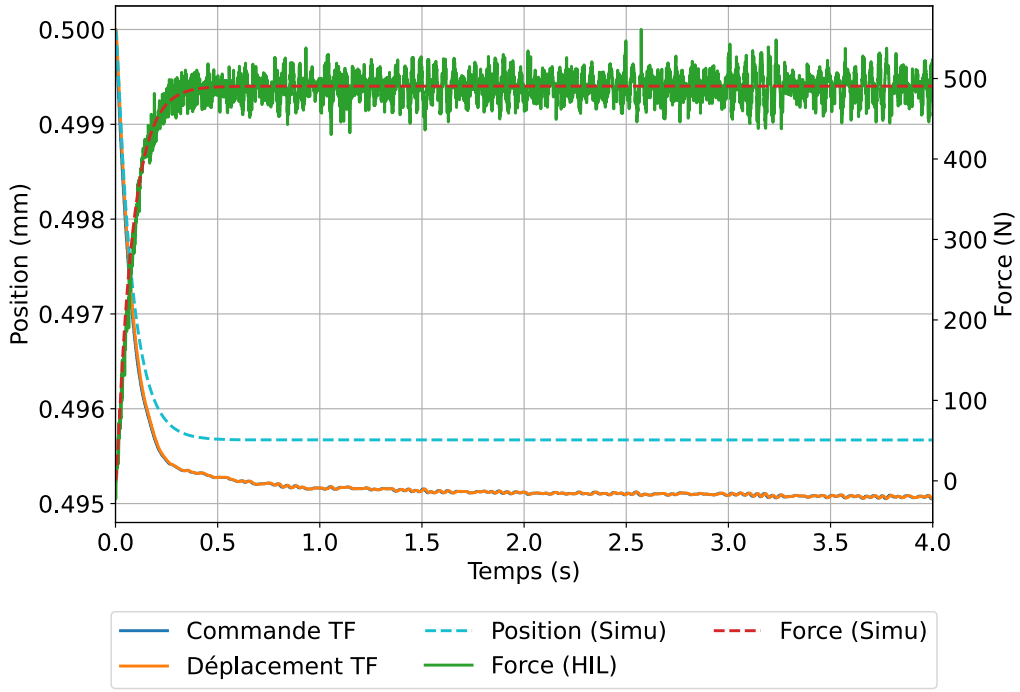


FIGURE 5.22 – Simulation HIL du système masse-ressort avec amortissement (filtre CPJ désactivé).

5.6.4 Simulation d'un VTT

Cette dernière simulation vise à remplacer le modèle du ressort de la suspension arrière d'un VTT en intégrant le transmetteur de force de raideur $K_2 \simeq 113$kN/m à la simulation par une approche Hardware In the Loop. La suspension est donc composée du transmetteur en parallèle d'un amortisseur de coefficient d'amortissement $C = 3000$ kNs/m, la force totale

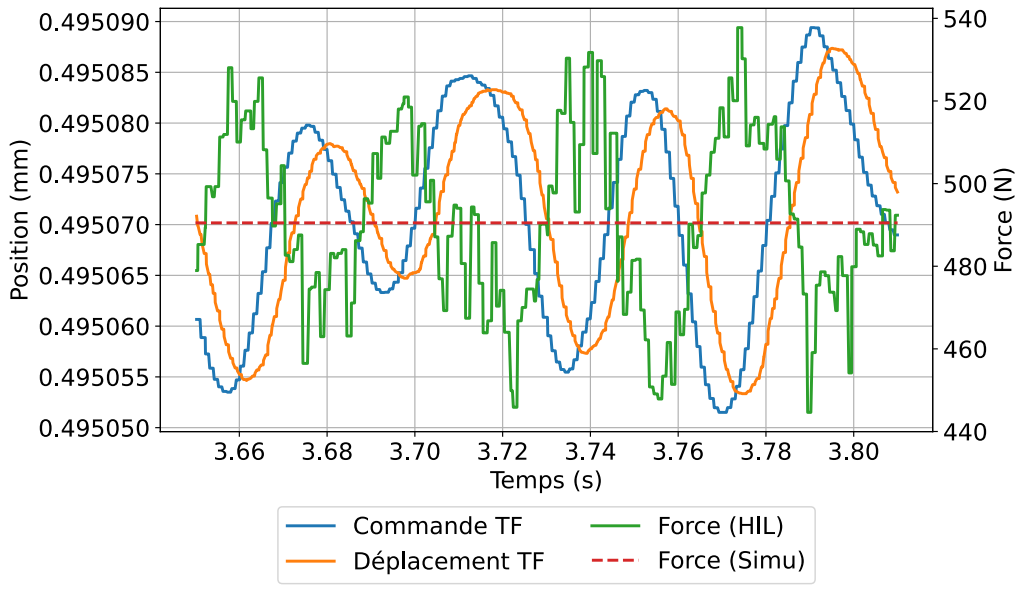


FIGURE 5.23 – Zoom sur les oscillations du système (filtre CPJ désactivé).

appliquée à la suspension est donc la suivante, le terme rouge étant fourni par le banc de test :

$$F_{susp} = K_2 x + C\dot{x} \tag{5.10}$$

La Figure 5.24 représente le modèle de VTT utilisé, sa masse totale est de 70 kg.

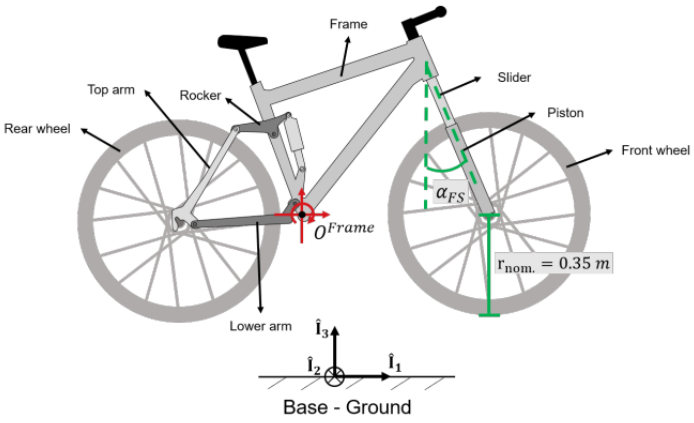


FIGURE 5.24 – Modèle du VTT simulé [14]

La simulation commence avec la suspension à sa longueur neutre, elle est comprimée par l'effet de la gravité, ensuite le vélo avance et passe sur un casse vitesse dont les dimensions sont détaillées sur la Figure 5.25.

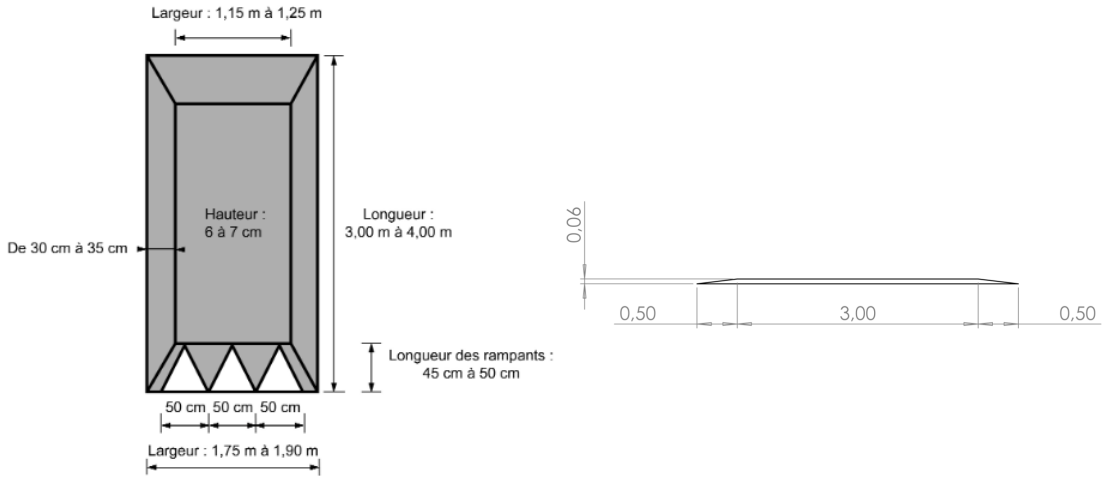


FIGURE 5.25 – Dimensions du modèle de casse vitesse utilisé dans la simulation du VTT (en m pour la vue de côté)

Les Figures 5.26 et 5.27 comparent les résultats de cette simulation avec ceux d'une simulation 100% numérique pour la position du transmetteur de force ainsi que la force qui lui est appliquée. Pour la simulation 100% numérique, le transmetteur de force est modélisé par un simple ressort de raideur $K_2 \simeq 113\text{kN/m}$.

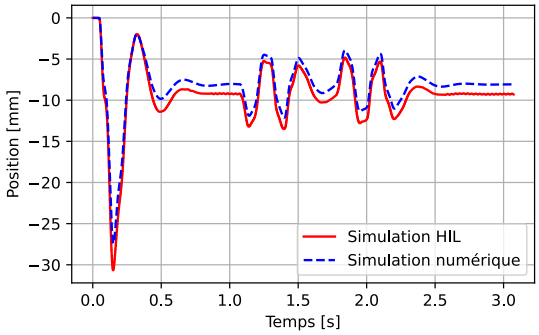


FIGURE 5.26 – Évolution du déplacement du transmetteur de force pour les deux simulations

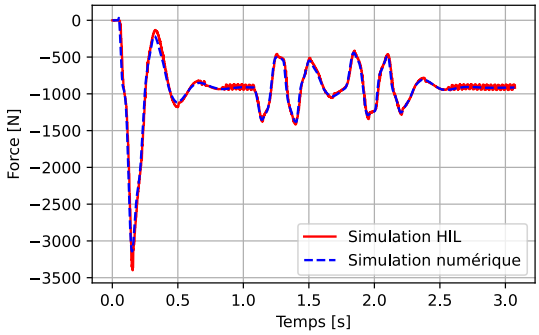


FIGURE 5.27 – Évolution de la force appliquée au transmetteur de force pour les deux simulations

La première compression est causée par la "chute" du cadre lorsque la gravité entre en action tout comme les deux oscillations suivantes. Ensuite, la roue avant du vélo atteint le casse vitesse à $1.1s$ ce qui entraine une rotation du cadre qui comprime la suspension arrière une première fois. Lorsque la roue arrière atteint le casse vitesse à $1.3s$, la suspension est à nouveau comprimée. À $1.8s$, la roue avant commence à descendre du casse vitesse, le même raisonnement que lors de la montée s'applique avec une direction opposée, la suspension arrière se détend à deux reprises lorsque la roue avant puis la roue arrière descendent.

La position HIL est très proche de la position de la simulation numérique, avec certaines différences :

— Oscillations lors des périodes d'équilibre : ces oscillations sont dues à un retard entre la commande de position et son exécution, plus de détails étant disponibles dans la section suivante.

— Décalage de la position : à cause du guidage du transmetteur de force, sa longueur neutre trouvée est décalée par rapport à la position à laquelle le ressort commence à se comprimer

— Écarts d'amplitude de crête : ces écarts lors de pics de compression et de détente du ressort sont moins évidentes à identifier. Le retard précédemment mentionné pourrait y participer, tout comme les non-idéalités engendrées par le guidage du transmetteur de force identifiées dans l'expérience de la section 5.6.2.

La force quant à elle est très proche de celle trouvée par simulation, ce qui est attendu, la masse et géométrie du VTT étant identiques dans les deux cas.

5.7 Délai d'actionnement

Comme cela peut être observé sur la Figure 5.28, il existe un délai d'environ 4 ms entre l'envoi de la consigne de position sur le topic et le moment où le moteur atteint la position instruite.

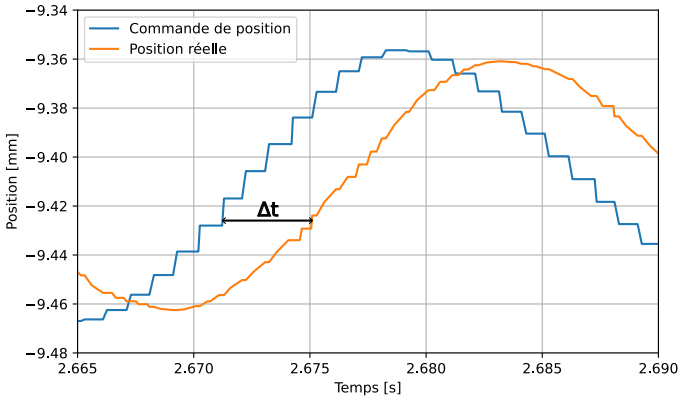


FIGURE 5.28 – Mesure du délai de l'actionnement (Δt)

Le temps écoulé entre la publication de l'instruction sur le topic et l'envoi du paquet EtherCAT correspondant a été mesuré à 0.2 ms, ce délai est le même pour la réception de la position réelle du moteur. Les $\simeq$ 3.6 ms de retard temporel restantes sont donc causées par le contrôleur de position du moteur et est prédictible en se basant sur le gain du contrôleur :

$$\tau \simeq \frac{1}{\omega_c} = \frac{1}{270} = 3.7ms \tag{5.11}$$

Où :

— τ = Retard temporel causé par le contrôleur de position

— ω_c = Pulsation de coupure du contrôleur.

En restant sur un contrôleur purement réactif sans anticipation (sans feed-forward), il est difficile, dans cette application, d'augmenter la bande passante du contrôleur de position sans créer de problème de résonance et de vibrations.

La simulation reçoit donc la conséquence du déplacement imposé en retard ce qui l'amène à régulièrement commander un déplacement trop important. Autour d'une position d'équilibre, cela ce traduit par une oscillation.

Conclusion

Les simulations Hardware-in-the-Loop suscitent un grand intérêt depuis plusieurs années, car elles permettent d'intégrer directement le comportement d'un dispositif physique au sein d'une boucle numérique, sans nécessiter la connaissance préalable de son modèle mathématique. Cette approche permet, d'une part, de mieux représenter certains phénomènes physiques complexes et difficiles à modéliser, tels que les frottements non linéaires dans un guidage. D'autre part, elle offre la possibilité de tester des composants spécifiques à moindre coût, par exemple pour analyser le comportement dynamique d'un véhicule équipé d'une nouvelle suspension.

L'objectif principal de ce mémoire était de concevoir un démonstrateur capable d'actionner un transmetteur de force (TF) et d'en mesurer précisément l'effort appliqué. Ce dispositif est piloté en position et en vitesse à partir des consignes fournies par une simulation multi-corps, à laquelle il réinjecte la force mesurée en temps réel grâce à une architecture de communication reposant sur ROS 2. Ce travail s'est focalisé sur la conception mécatronique du démonstrateur, à savoir la stratégie d'actionnement, la chaîne d'instrumentation ainsi que l'infrastructure logicielle. La partie mécanique est, quant à elle, détaillée dans le mémoire de Bastien Debroux.

Par la suite, les essais de simulation ont été menés afin de valider le démonstrateur. La première étape a consisté à valider l'intégration de ROS 2 au sein d'un projet Robotran. La second a été de piloter le démonstrateur en position sur base d'un mouvement commandé à partir de la simulation numérique. Ce test à par ailleur permis d'analyser des comportements non linéaire du au buté de l'amortisseur considéré. C'est exactement pour prendre en compte ce genre de comportement que les simulations HIL ont vu le jour. Ensuite, la mise en œuvre de la première simulation HIL s'est appuyée sur l'étude d'un système masse-ressort-amortisseur en régime quasi-statique. Celle-ci a permis de déterminer qu'un retard entre la commande moteur et la position réel engendre des oscillations autour de la position d'équilibre. Enfin, la réalisation d'une simulation HIL complète d'un VTT tout-suspendu, dont le ressort de la suspension arrière était physiquement intégré au démonstrateur.

En conclusion, un démonstrateur permettant d'éffectuer des simulation HIL a été réalisé et testé. Néanmoins, plusieurs perspectives d'amélioration peuvent être envisagées afin de repousser les limites des capacités actuelles :

— Rigidification de la plaque de maintien des rotules de l'axe inférieur afin de diminuer sa déformation et d'accroître ainsi la précision de la commande en position.
— Effectuer d'un filtrage numérique supplémentaire de la force car le filtre analogique interne du CPJ a dû être désactivé pour réduire son temps de traitement qui induisait un décalage d'environ 15 ms entre la position réelle du TF et l'effort.
— Intégration d'une commande par anticipation (*feed-forward*) dans le but de compenser le temps de réponse de l'actionnement, actuellement mesuré à 3,7 ms. Cette stratégie permettrait d'atténuer, voire de supprimer, les oscillations observées autour de la position d'équilibre. Pour cela, la consigne de vitesse provenant de la simulation peut

être transmise au servo-contrôleur en même temps que la consigne de position afin de réduire son temps de réponse.

Annexes

Annexe A

Démonstration de l'impact du filtre de moyennage sur la dynamique

D'après le livre de référence de Smith [31], il est possible d'étudier la réponse en fréquence d'un filtre à moyenne mobile dont l'équation est :

$$y[n] = \frac{1}{N} \sum_{k=0}^{N-1} x[n-k] \tag{A.1}$$

Où N est le nombre d'échantillons pris en compte (ici $N = 40$), $y[n]$ la tension filtrée publiée toutes les 1 ms, $x[n]$ les données brutes acquises et k l'indice de sommation. L'acquisition se faisant à une fréquence d'échantillonnage $f_e = 40$ kHz, et en notant f la fréquence du signal mesuré, on définit la pulsation normalisée $\omega = 2\pi\frac{f}{f_e}$. La transformée de Fourier de la réponse impulsionnelle du filtre est :

$$H(\omega) = e^{-j\omega\frac{N-1}{2}} \frac{\sin\left(\omega\frac{N}{2}\right)}{N\sin\left(\frac{\omega}{2}\right)} \tag{A.2}$$

Il est alors possible de déterminer le retard temporel τ induit par le filtrage, qui vaut $\tau = \frac{N-1}{2f_e} = 0,4875$ ms, ainsi que la perte de dynamique (atténuation de l'amplitude) du signal :

$$P_{\%} = (1 - |H(\omega)|) \times 100 \tag{A.3}$$

La figure A.1 illustre la perte d'amplitude du signal due à l'utilisation du filtre. Dans le pire cas de figure où le démonstrateur reçoit un signal à 25 Hz, l'amplitude est impactée de seulement 0,1 %, ce qui est tout à fait négligeable.

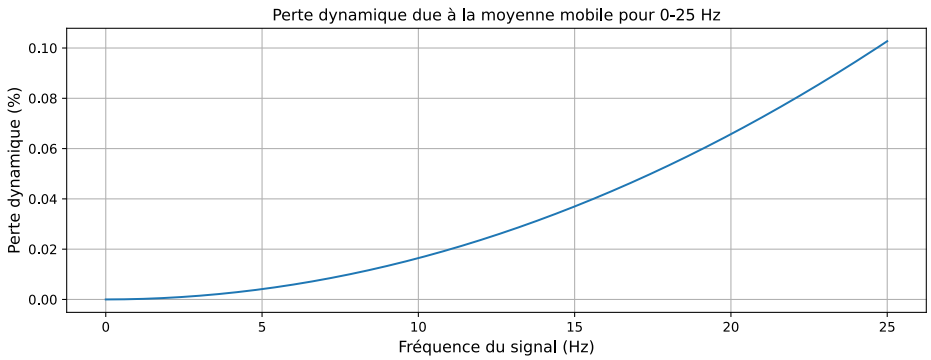


FIGURE A.1 – Impact de la fréquence du signal sur l'amplitude de la dynamique

Annexe B

Options considérées pour le choix de l'actionneur

B.1 Actionneur servo-hydraulique

Description	F_{nom} [N]	v_{max} [Nm]	m [kg]	Prix (€TVAC)	Délai
Rexroth CYTROMOTION 1X/HSF/040/028/200/...	14000	0.232	/	$\approx$ 10000	/

B.2 Servomoteur linéaire

Description	F_{nom} [N]	F_{max} [N]	v_{max} [Nm]	m [kg]	Prix (€TVAC)	Délai
Linmot PS10-70x400	890	2720	3.9	2.6	$>$ 5000	3 semaines
Omron R88L-EC-FW-1115	760	2000	2	2	5800+	6 semaines

B.3 Actionneur à vis

B.3.1 Vérins

Description	F_{max} [N]	M_{max} [Nm]	v_{max} [m/s]	Prix (€TVAC)	Délai
Vérin "homemade"	9200	/	/	820	3 semaines
Festo ESBF-LS-50-100-4P	1600	4.8	0.2	1150	1 mois
Festo ESBF-BS-40-200-5P	3000	3	0.42	1280	2 semaines
Festo ESBF-BS-40-200-10P	3000	5.6	0.83	1340	2 semaines
Festo ESBF-BS-40-200-16P	3000	7.7	1.33	1370	1 mois

B.3.2 Moteurs

Description	M_{nom} (Nm)	M_{max} (Nm)	N_{nom} (rpm)	Inertie $()kg \cdot cm^2$)	Prix (€TVAC)	Délai
Stepperonline A6M60-400H2A1-M17	1.27	4.45	3000	0.6	150	0 jours
Festo EMMT-AS-60-L-LS-RM	1.4	5.6	3000	0.403	2436	1 semaine
Omron R88M-1M40030T-BS2	1.27	4.5	3000	0.4452	2915	1 mois
Schneider R88M-1M40030T-BS2	1.1	4	4000	0.59	2915	3 semaines
Stepperonline 24E1K-30	1.2	3	1500	0.69	85	2 semaines
Stepperonline 34HS39-6004D-E1000	2.6	6.5	1500	0.69	100	1 semaine

B.3.3 Combinaisons étudiées

Combinaison 1

- Festo ESBF-BS-40-200-10P + Stepperonline A6M60-400H2A1-M17
- Festo ESBF-BS-40-200-10P + Schneider R88M-1M40030T-BS2
- Festo ESBF-BS-40-200-10P + Festo EMMT-AS-60-L-LS-RM

Combinaison 2

- Festo ESBF-BS-40-200-5P + Stepperonline A6M60-400H2A1-M17
- Festo ESBF-BS-40-200-5P + Schneider R88M-1M40030T-BS2
- Festo ESBF-BS-40-200-5P + Festo EMMT-AS-60-L-LS-RM

Combinaison 3

- Festo ESBF-BS-40-200-16P + Stepperonline 34HS39-6004D-E1000

Annexe C

Pré-prototype

Le pré-prototype a été conçu durant le premier quadrimestre de cette année académique, à partir de composants de stock qui furent empruntés notamment au CREDEM. Il se compose d'un servomoteur linéaire **Faulhaber LM1247-020-01** appliquant un déplacement sur un petit ressort. Il est contrôlé par un contrôleur **Technosoft IPOS3602 HW-CAN**, celui-ci communique avec le PC via une interface série-USB. Compte tenu de l'absence de capteur de force, celle-ci était calculée sur base du courant traversant le moteur. La Figure C.1 représente l'ensemble de ces éléments.

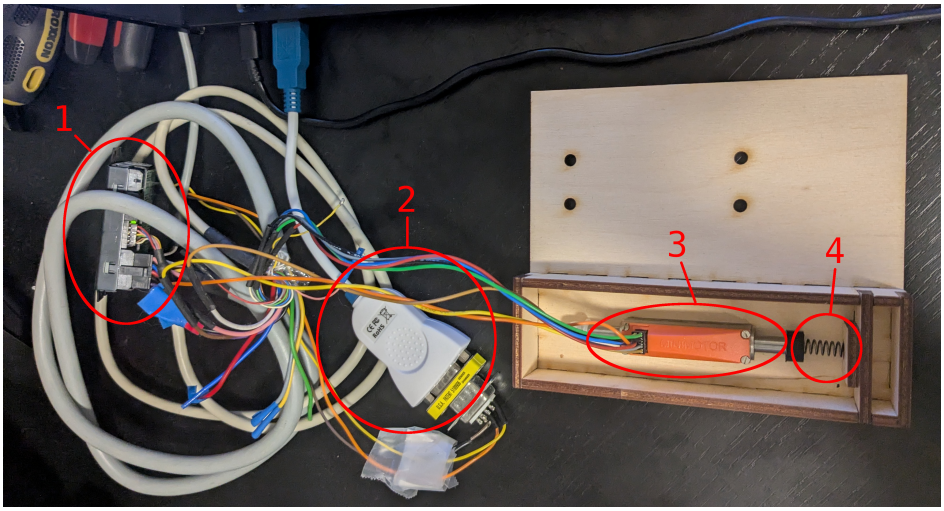


FIGURE C.1 – Pré-prototype composé du contrôleur Technosoft (1), de l'interface série-USB (2), du servo linéaire Faulhaber (3) et du ressort testé

Ce pré-prototype a permis le développement d'une partie du logiciel nécessaire, particulièrement concernant l'intégration de Robotran à ROS2. Cependant, le débit limité et la latence de l'interface série a limité le pas de temps de simulation à 15 ms, ce qui n'a pas permis d'effectuer de simulation complète, mais a néanmoins souligné l'importance du choix de protocole de communication pour le banc d'essai final.

Annexe D

Analyse du flambement du corps rigide pour le teste de déformation du démonstrateur

Le corps rigide utilisé est une tige filetée M12 en acier inoxydable, caractérisée par un module de Young $E = 190$ GPa et un diamètre à fond de filet $d = 10,1$ mm. Son moment quadratique I s'exprime par :

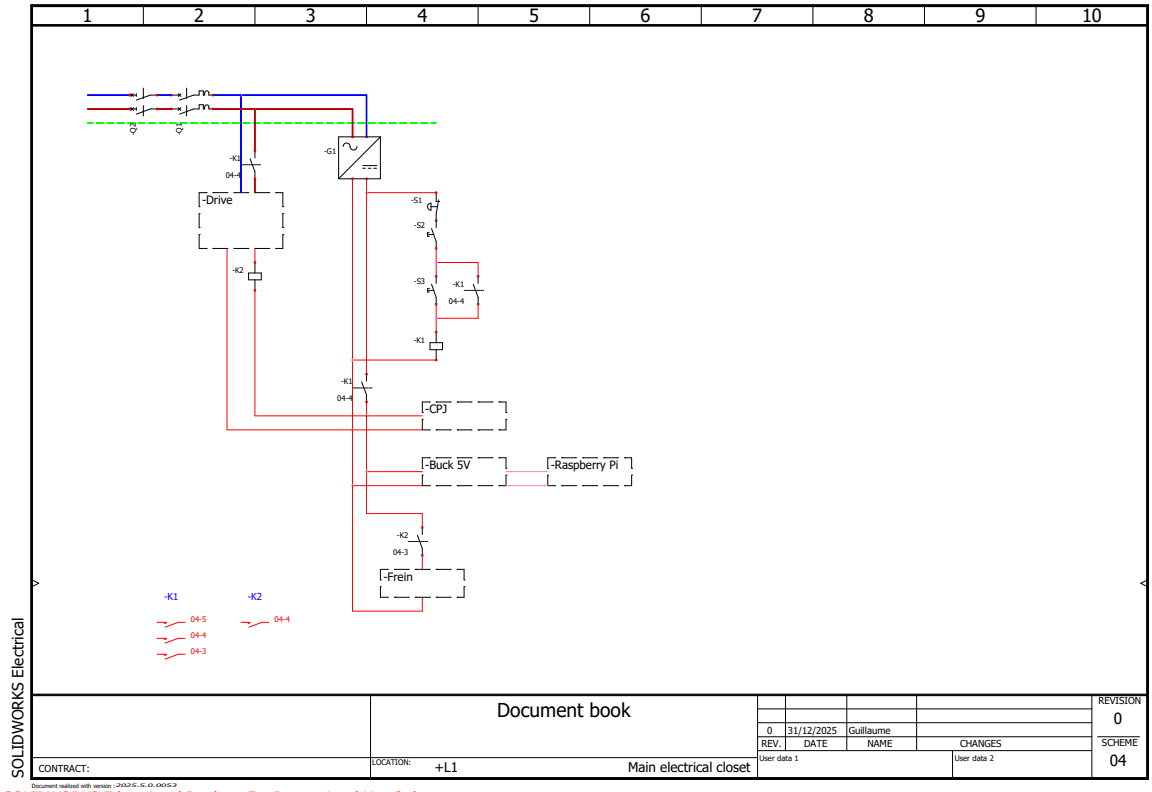
$$I = \frac{\pi d^4}{64} \tag{D.1}$$

Le montage étant bi-articulé, la longueur de flambement équivalente correspond à la longueur libre de la tige, soit $L_e = L = 300$ mm. Cette longueur comprend la tige filetée ainsi que le capteur. La section minimale du capteur étant supérieure à celle de la tige filetée, le calcul se base uniquement sur la seule géométrie de la tige. La charge critique de flambement P_{cr} s'évalue alors à l'aide de la formule d'Euler [32] :

$$P_{cr} = \frac{\pi^2 EI}{L_e^2} \tag{D.2}$$

L'application numérique donne une charge limite $P_{cr} \approx 10\,643$ N. Cette valeur étant strictement supérieure à la limite théorique d'effort admissible par le vérin dans la pire des configurations (9 000 N). Les essais peuvent donc être conduits en toute sécurité pour le matériel.

D.1 Schéma electrique du circuit de sécurité



SOLIDWORKS Educational Product. For Instructional Use Only.

Bibliographie

[1] Pouria Sarhadi and Samereh Yousefpour. State of the art : hardware in the loop modeling and simulation with its applications in design, development and implementation of system and control software. 3(4) :470–479.

[2] ROBOTRAN Team. *Modeling Multibody Systems with ROBOTRAN*. Institute of Mechanics, Materials and Civil Engineering (IMMC), UCLouvain, October 2025.

[3] P. Fisette and J. C. Samin. *ROBOTRAN : Symbolic Generation of Multi-Body System Dynamic Equations*, pages 373–378. Springer Netherlands, Dordrecht, 1993.

[4] Hai-Bo Yuan, Hong-Cheol Na, and Young-Bae Kim. System identification and robust position control for electro-hydraulic servo system using hybrid model predictive control. *J. Vib. Control*, 24(18) :4145–4159, September 2018.

[5] Linmot - moteur linéaire p10. `https://linmot.com/fr/produits/moteurs-lineaire/moteur-lineaire-p10/`. Consulté le : 26/05/2026.

[6] Festo E.U.R.L. *Electric cylinders ESBF, with spindle drive - Manual.*

[7] Stepperonline. *A6-EC Servo Drive - User Guide.*

[8] François Louf. Capteur – jauges de déformation, oct 2020. Consulté le 15 mai 2026.

[9] Thomas Kleckers. Spoilt for choice : piezoelectric or strain gauge based force transducers ? Technical article, HBM Test and Measurement, Darmstadt, Allemagne.

[10] Interface, Inc. *Load Cell Field Guide : Force Measurement Solutions.* Interface, Inc., Scottsdale, Arizona, USA, 2022.

[11] Measurement Computing Corporation. *MCC 128 : 16-bit Voltage Measurement DAQ HAT for Raspberry Pi*, December 2020. Rev 1, Doc Ref : DS-MCC-128.

[12] Hottinger Brüel & Kjaer GmbH. *Notice de montage S9M.* Hottinger Brüel & Kjaer GmbH, Darmstadt, Allemagne, avril 2025. Réf. Mat. : 7-2001.3019.

[13] Open Robotics. Ros 2 documentation : About topics, services, and actions (jazzy jalisco), 2024. En ligne ; consulté en 2026.

[14] Lepl 1504 - livrable 2 analyse dynamique du vtt complet et morphologie du vélo chargé.

[15] SCAIME SAS. *Fiche Technique / Data Sheet : CPJ / CPJ-2S Conditionneur analogique - Entrée pont de jauges / Analog conditioner - Strain gages input.* SCAIME SAS, Juvigny, France, December 2018. Réf : FT-CPJ_CPJ2S-FE-1218.

[16] Penny+Giles Controls Ltd. *SLS130 Linear Displacement Sensor.* Curtiss-Wright Controls Integrated Sensing, Christchurch, Dorset, United Kingdom, jan 2012. Doc Ref : SLS MLS 01/12.

[17] Debroux Bastien. Démonstrateur électromécanique pour de la simulation hardware-in-the-loop : Conception mécanique. 2026.

[18] M Kähler, Roman Rachholz, Sven Herrmann, János Zierath, Robert Souffrant, Daniel Kluess, R Bader, and Christoph Woernle. Development of a biomechanical multibody model for the hardware-in-the-loop simulation of total hip endoprostheses. 05 2011.

[19] P.Righettini, H.Giberti, S.Chatterton, and A.Oldani. Test bench design for motorcycle suspensions hil test. 2006.

[20] Nicola DE LELLIS and Quentin MATERNE. Conception et réalisation d'un simulateur à retour haptique pour l'étude de l'interaction homme/véhicule. Master's thesis, École polytechnique de Louvain, 2026.

[21] Siemens. *SIMATIC Connector for ROS - ROXSIE*.

[22] Beckhoff Automation GmbH. Ethercat : The ethernet fieldbus. `https://www.ethercat.org`, 2003–2024.

[23] Thomas Kleckers. Force sensors based on strain gages and piezoelectric crystal-based force transducers in mechatronic systems : a comparison. SENSOR+TEST Conferences 2011 SENSOR Proceedings, 2011.

[24] Hottinger Brüel & Kjær GmbH. *U2B : Force transducer*. HBK (Hottinger Brüel & Kjær), Darmstadt, Germany, 8 2025. Ref. B00482 05 E00 03.

[25] SCAIME SAS. *MTRZ : Capteurs de Force - Traction/Compression (0 ... 1 kN / 0 ... 100 kN)*. SCAIME SAS, Juvigny, France, 3 2021. Réf. FT-FE-MTRZ-0321.

[26] Interface, Inc. Understanding load cell frequency, may 2025. Blog post.

[27] Measurement Computing Corporation. *MCC DAQ HATS Library Library Documentation*. Measurement Computing Corp., 2020. Consulté : 2026-05-25.

[28] Open Source Robotics Foundation. Ros 2 jazzy jalisco (lts), 2024. ROS 2 Documentation.

[29] ethercat_driver_ros2. `https://github.com/ICube-Robotics/ethercat_driver_ros2`. Consulté le : 26/05/2026.

[30] Robotran Team. *Rosbotran with ROS2*, 2022. Robotran Tutorial Documentation.

[31] Steven W. Smith. *Digital Signal Processing : A Practical Guide for Engineers and Scientists*, chapter 15, pages 277–284. Newnes, Burlington, MA, 2003.

[32] Robert C. Juvinall and Kurt M. Marshek. *Fundamentals of Machine Component Design*. John Wiley & Sons, Hoboken, NJ, 7th edition, 2020.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**