

Assistance électrique d'un chariot de manutention



1

Étudiants :

Julia EL ZEINE
Raphaël MURAINÉ
Harry NGUYEN
Mathilde ORAIN
Célia ROCHE

Enseignant-responsable du projet :

Didier LEMOSSE

1. Chariot de manutention semblable à celui de l'INSA.

Source de l'image : <http://www.bruneau.fr/services-generaux/manutention/chariot/chariot-manutention/chariot-manutention-plateau-standard-0F14142.htm?ovalue=Liste>

Date de remise du rapport : 15/06/2020

Référence du projet : STPI/P6/2020 – 011

Intitulé du projet : Assistance électrique d'un chariot de manutention

Type de projet : Projet de mécanique, conception

Objectifs du projet :

- Concevoir un prototype du chariot de manutention intégrant l'assistance électrique.
- Réaliser un modèle CAO du chariot.
- Permettre au chariot d'avancer sans que l'opérateur n'exerce un effort trop grand.
- Concevoir un moyen, simple et rapide à installer, permettant de mesurer la force appliquée par l'opérateur sur un prototype du chariot. Mesurer les efforts pour faire avancer et reculer le chariot.

Mots-clés du projet :

Assistance électrique

Conception

Simulation

Déformation

Table des matières

Notations et Acronymes	4
Remerciements	5
Introduction	6
Organisation du travail	7
1 Présentation	8
1.1 Présentation du chariot	8
1.2 Équations [3]	8
1.2.1 Bilan des forces – PFD	8
1.2.2 Estimation de la force exercée par l'utilisateur	8
1.2.3 Calcul de $f_{ass\ max}$	9
2 Conception, CAO	10
2.1 Méthodes et réalisation	10
2.1.1 Le réel-présentiel	10
2.1.2 La représentation sous SolidWorks	10
2.2 Problèmes rencontrés et moyens mis en place pour les résoudre	13
2.2.1 Travailler dans un contexte de crise sanitaire	13
2.2.2 Problèmes rencontrés et méthodes pour les résoudre	14
3 Simulation, asservissement	15
3.1 Résolution numérique d'une EDO [4]	15
3.1.1 Discrétisation	15
3.1.2 Approximation des fonctions connues	15
3.1.3 Approximation des inconnues	15
3.1.4 Algorithmique	16
3.2 Simulations de la théorie	16
3.2.1 Premiers essais avec un tableur	16
3.2.2 Synthèse des calculs avec Matlab	17
3.3 Étude en pseudo-code	18
4 Mesure de déformation	19
4.1 Étude théorique	19
4.1.1 Analyse de la situation	19
4.1.2 Dimensionnement du pont et de la tige	20
4.2 Mise en pratique : installation du système de mesure de déformation	21
4.2.1 Connexions et branchements	21
4.2.2 Programme Arduino	21
4.3 Pistes d'amélioration	21
Conclusion et perspectives	23
Conclusions	23
Perspectives pour la poursuite de ce projet	23
Bibliographie	24

Annexes	25
Code source Matlab	25
Fonction de calcul du frottement	25
Programme principal	25
Essai de pseudo-code	27
Mesure de déformation – Compléments	28

Notations et Acronymes

PFD : Principe Fondamental de la Dynamique

CAO : Conception Assistée par Ordinateur

Impression 3D : Impression en trois dimensions

2D / 3D : deux dimensions / trois dimensions

Pseudo-code : Le pseudo-code permet de fournir une solution au problème à l'aide d'un langage formel, lisible et indépendant de tout langage informatique.

Remerciements

Nous tenons en premier lieu à remercier notre enseignant, Monsieur Didier LEMOSSE, pour son investissement et son aide dans ce projet. En effet, la période de crise sanitaire due au COVID-19 a nécessité une adaptation rapide, par un apprentissage à distance parfois fastidieux, tant sur le plan matériel que psychologique. Malgré tout, nous sommes parvenus à maintenir un esprit d'équipe en nous encourageant mutuellement, afin de réussir au mieux ce projet. Nos remerciements se tournent également vers chaque membre de ce groupe, sans qui il n'y aurait pas eu cet aboutissement.

Introduction

Le monde a toujours été témoin de révolutions techniques et industrielles. Parmi ces révolutions figure l'invention de l'assistance électrique.

L'assistance est un dispositif apportant une énergie extérieure dans le but d'amplifier, de réguler et de répartir l'effort exercé par l'utilisateur d'une machine, d'une automobile (direction, freinage)...

L'assistance électrique est de plus en plus utilisée ces dernières années. En effet, de nombreux vélos en sont dotés afin d'apporter une aide complémentaire au cycliste. Cependant, elle n'est pas réservée exclusivement aux vélos : elle s'adapte également à des chariots de manutention pour réduire la charge réellement transportée par l'utilisateur.

Celle-ci se compose d'une motorisation et d'électronique.

La motorisation est mise en place à l'aide d'un moteur et d'une batterie. Il existe plusieurs types de motorisations : sur la roue avant, sur la roue arrière, moteur pédalier.

L'électronique, quant à elle, est dotée d'un contrôleur et d'un logiciel intégré. Le contrôleur est comme la cartemère du chariot, sa position est déterminée en fonction du type de moteur.

Un autre composant essentiel de l'assistance est le capteur. En effet, il est l'élément qui permettra à l'assistance de se déclencher et de s'adapter aux efforts de l'opérateur. On recense différents types de capteurs (de couple, de cadence, de vitesse, d'effort).

L'objectif principal de ce projet était de produire un prototype simple du chariot de manutention, intégrant l'assistance électrique. Les tests effectués sur celui-ci auraient permis de conclure quant à nos travaux et ainsi de savoir s'ils auraient pu être adaptés au vrai chariot par la suite.

Malheureusement, nous n'avons pas eu l'occasion de créer ce prototype dans le contexte de crise sanitaire, due au COVID-19.

Organisation du travail

Pour réaliser ce projet, nous avons décidé de décomposer le problème en trois parties :

- Conception / CAO
- Simulation / Contrôle
- Mesure de déformation

La première partie, réalisée par Mathilde et Raphaël, a pour but de modéliser sur ordinateur, à l'aide du logiciel SolidWorks, le chariot de manutention.

La seconde, menée par Julia et Célia, permet de simuler un scénario et de le comprendre, pour pouvoir adapter l'assistance en fonction de la force exercée par l'opérateur sur le chariot.

Enfin, la dernière partie, celle de Harry, est une étude de la déformation de la barre du chariot, suite à l'effort fourni par l'utilisateur.

La répartition du travail est représentée ci-dessous :

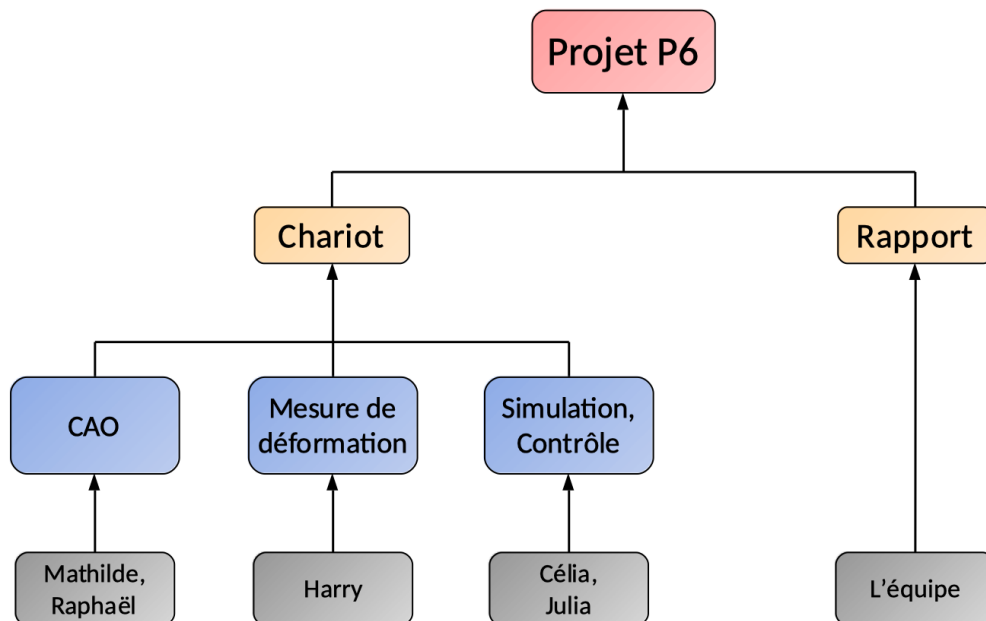


FIGURE 1 – Organigramme de la répartition du travail

Chapitre 1

Présentation

1.1 Présentation du chariot

Le chariot de manutention, fourni par l'INSA dans le cadre de ce projet, est un simple chariot permettant de transporter des charges plus ou moins lourdes.

Il est constitué d'un plateau sur lequel sont déposées les charges, de quatre roues et d'une barre, reliée au plateau, sur laquelle l'opérateur peut pousser pour faire avancer le chariot.

Adapter une assistance électrique sur ce chariot nécessite de réfléchir à l'installation des divers éléments qui la composent. En effet, intégrer un moteur sur le chariot prend de la place, et il ne faudrait pas utiliser celle réservée aux charges sur le plateau. Cependant, une installation sous ce dernier est aussi délicate car exige de protéger les équipements.

Ces remarques sont évidentes mais à ne pas négliger lors de la conception du projet.

1.2 Équations [3]

1.2.1 Bilan des forces – PFD

Cette étude est menée de manière scalaire et non vectorielle.

Tout d'abord, les forces présentes sont les suivantes :

- Force de l'utilisateur f_{op}
- Assistance du moteur f_{ass}
- Poids dû à la masse transportée P
- 4 fois (car 4 roues présentes) la réaction R , avec $R = R_n + R_t$ la somme des réactions normale et tangentielle

Ensuite, on obtient le PFD (en version scalaire) suivant :

$$f_{op}(t) + f_{ass}(t) \pm mg \sin(\alpha) = m \times a(t)$$

On note α le coefficient de frottements de Coulomb et m la masse totale telle que $m = m_{chariot} + m_{charge}$. Le signe de $mg \sin(\alpha)$ est opposé à celui du déplacement x car c'est une force de frottement.

1.2.2 Estimation de la force exercée par l'utilisateur

On a

$$\begin{aligned} f_{op}(t) + f_{ass}(t) \pm mg \sin(\alpha) &= m \times a(t) \\ f_{op}(t) &= m(a(t) \pm g \sin(\alpha)) - f_{ass}(t) \end{aligned} \quad (1.1)$$

Or à $t = 0$, l'accélération a est nulle et l'assistance du moteur n'est pas encore enclenchée. On obtient donc $a(0) = 0$ et $f_{ass}(0) = 0$.

Ainsi, à $t = 0$, par l'équation 1.1, $f_{op}(0) = \pm mg \sin(\alpha)$.

Il ne reste plus qu'à fixer α et déterminer m , la masse maximale que l'utilisateur doit pousser.

On prend $\alpha = 0,05$ et on fixe m à $m = 300 \text{ kg}$.

Ainsi, on obtient à $t = 0$,

$$\begin{aligned}\|f_{op}(0)\| &= \|mg \sin(\alpha)\| \\ &\simeq 300 \times \sin(0,05) \times 9,81 \\ &\simeq 147,09 \text{ N}\end{aligned}$$

Finalement, on prendra $\|f_{op}(t)\| = 150 \text{ N}$.

1.2.3 Calcul de $f_{ass \max}$

Différents calculs sont nécessaires avant de pouvoir déterminer la valeur maximale de l'assistance électrique. Voici la démarche :

1. Force de roulement :

$$F_{roulement} = CRR_{constructeur} \times (m_{chariot} + m_{charge}) \times g = 0,0173 \times (12 + 150) \times 9,81 \simeq 27,5 \text{ N}$$

On note $CRR_{constructeur}$ le coefficient de roulement des roues fourni par le constructeur.

2. Les forces aérodynamique et pente sont négligées.
3. Force déduite de l'accélération (connue) :

$$F_a = (m_{chariot} + m_{charge}) \times a_{chariot} = (12 + 150) \times 0,5 = 81 \text{ N}$$

4. Donc on a la force totale :

$$F_{tot} = F_{roulement} + F_a \simeq 27,5 + 81 = 108,5 \text{ N}$$

Ainsi, on prendra, pour t quelconque, $f_{ass \max}(t) = 108,5 \text{ N}$.

Chapitre 2

Conception, CAO

Dans le cadre du module Projet P6, nous avons eu l'opportunité de nous occuper de la partie CAO sous SolidWorks. Le choix de cette sous-thématique fut motivé en partie par notre initiation au logiciel à travers le module CTI3. De plus, nous trouvions que cela complétait assez bien les deux autres parties dont se sont occupés nos camarades de projet.

Le projet se nommant *Assistance électrique d'un chariot de manutention*, nous nous sommes occupés de la représentation fidèle d'un chariot de l'INSA. Dans un premier temps, nous allons vous parler des différentes méthodes utilisées, ensuite, vous introduire les problèmes rencontrés et enfin, la manière dont nous avons pu les résoudre. Nous avons adapté notre approche au contexte particulier de COVID-19 dans lequel nous nous sommes trouvés pendant l'élaboration de ce projet.

2.1 Méthodes et réalisation

2.1.1 Le réel-présentiel

Ayant choisi la représentation du chariot comme réaliste, afin de rester le plus fidèle possible au modèle étudié, nous avons consacré nos premières séances au dimensionnement de celui-ci. À l'aide d'un mètre, nous avons relevé chaque mesure du chariot : sa hauteur, sa longueur, sa largeur, ainsi que les dimensions de chacun des éléments le composant.

La dernière partie de notre dimensionnement a concerné l'échelle à laquelle nous avons réalisé nos pièces. En effet, nous avions comme projet l'impression 3D de notre chariot, mais ses dimensions n'étant pas adaptées à la capacité de l'imprimante 3D de l'INSA, nous avons décidé de prendre une échelle au tiers.

2.1.2 La représentation sous SolidWorks

2.1.2.1 Réalisation indépendante des éléments

Nous avons commencé par réaliser les différentes pièces composant le chariot et avons rapidement remarqué trois grandes classes distinctes que nous pourrions assembler à la fin : le plateau, le cadre et les roulettes. Tout d'abord, nous nous sommes occupés du plateau.

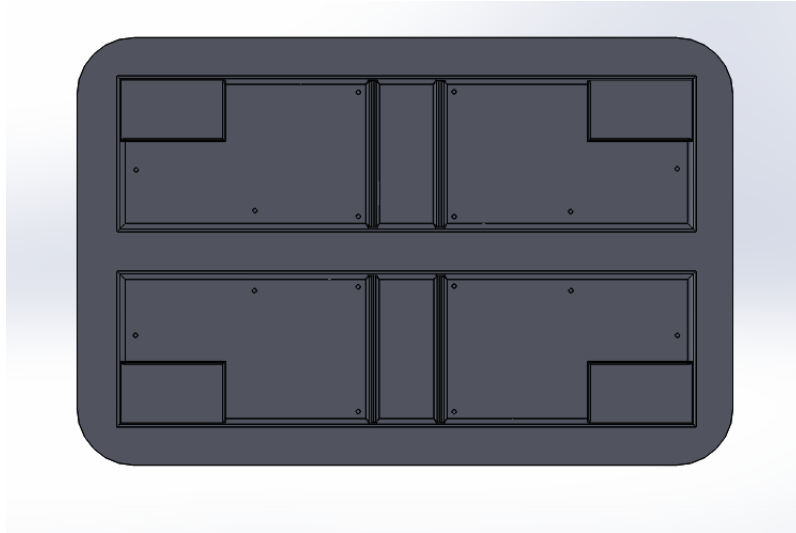


FIGURE 2.1 – Plateau

La principale difficulté était de reproduire le dessous. En effet, en s'appuyant uniquement sur des photos, nous devons créer des renforcements et ensuite y appliquer des congés (voir ci-dessous).

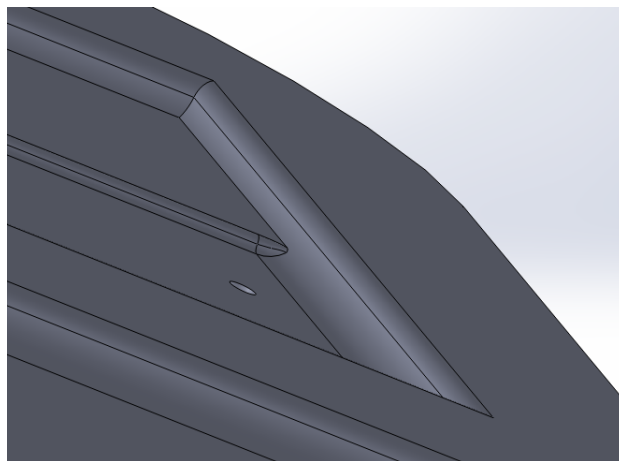


FIGURE 2.2 – Renforcements et congés

Cependant, les reliefs ne sont pas facilement reconnaissables sur une image 2D, il fallait donc aussi faire appel à nos souvenirs concernant le chariot.

Après nous être occupés du plateau, nous nous sommes ensuite intéressés au cadre. Nous avons extrudé sur environ 18 cm deux cercles de diamètre 0,5 cm pour obtenir deux cylindres parallèles. Nous avons courbé les extrémités de chaque cylindre avec un angle de 90° et les avons ensuite reliés.

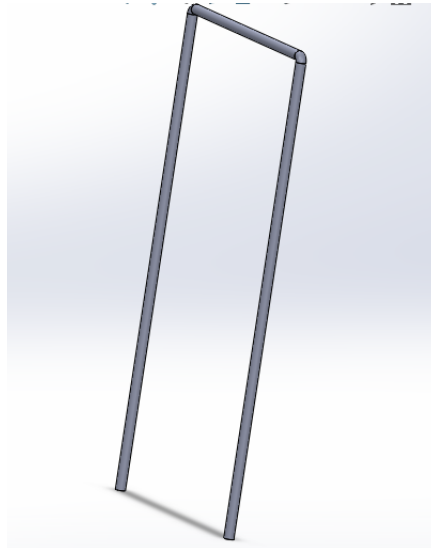


FIGURE 2.3 – Cadre

Le cadre obtenu s'éloigne un peu du modèle initial car il n'est pas incliné. En effet, il n'y a pas la possibilité de transmettre des efforts sur la jauge de déformation en y effectuant une pression. Grâce aux simplifications effectuées, nous avons pu passer assez rapidement sur cette pièce.

Ensuite, nous avons abordé la conception des roues. Beaucoup de temps a été nécessaire car les photos du chariot avaient été prises selon un même angle. Nous avons donc été obligés de nous appuyer sur d'autres modèles de roues. La difficulté était sur le support de la roue permettant de la relier au plateau.

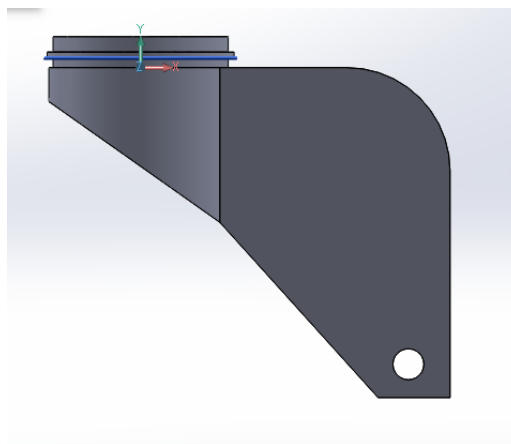


FIGURE 2.4 – Support des roues

Il fallait à la fois qu'il soit plus grand qu'elle, pour ne pas qu'elle puisse toucher le plateau, et en même temps qu'il permette une liaison pivot avec le plateau le long de l'axe Y . Nous avons également dû réaliser des épaulements pour pouvoir ajouter une contrainte de coïncidence.

2.1.2.2 Assemblage des éléments

Avant de nous lancer sur l'assemblage final, nous avons monté les différentes pièces de l'ensemble roulettes. Nous avons laissé un trou dans la pièce en contact avec le plateau (pièce 1), de sorte à y insérer notre deuxième pièce à l'aide d'une contrainte concentrique et d'un épaulement. Notre deuxième pièce (pièce 2) étant libre en rotation, nous pouvions alors y ajouter la dernière pièce : la roue. Dans un premier temps, un cylindre a été créé autour duquel la roue peut tourner librement, puis le tout a été ajusté à travers des trous prévus sur la pièce 2.

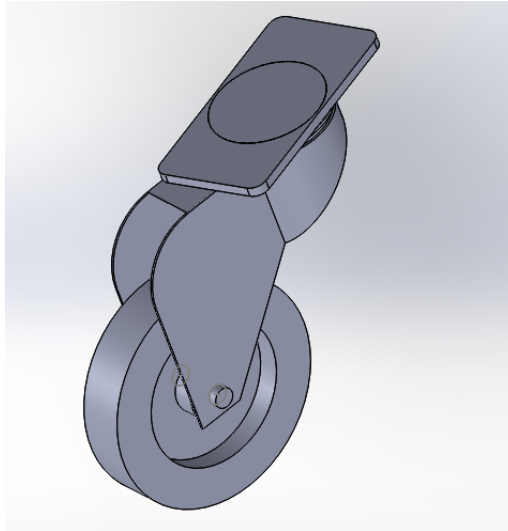


FIGURE 2.5 – Assemblage roulettes

L'assemblage final comprend le plateau, le cadre et l'assemblage roulettes évoqué ci-dessus. Tout d'abord, nous avons décidé de fixer le plateau dans l'espace puis d'y ajouter le cadre et les roulettes, car il est en interaction avec chacun d'eux. Pour les assemblages roulettes, nous avons prévu des surfaces sur le plateau à leur effet. Nous avons fait coïncider les surfaces pour chaque roulette, puis, afin de réaliser la liaison encastrement, des trous ont été créés à travers les deux surfaces dans lesquels nous avons inséré des boulons (modélisés au besoin) pour bloquer le tout.

Pour l'ajout du cadre à l'assemblage, nous avons réalisé une simple contrainte de coïncidence, bloquée en translation.



FIGURE 2.6 – Chariot

2.2 Problèmes rencontrés et moyens mis en place pour les résoudre

2.2.1 Travailler dans un contexte de crise sanitaire

Au début du projet, nous avons été confinés en raison de la pandémie de COVID-19 et n'avons pas pu refaire de cours en présentiel depuis. Par chance, nous avons, avant ce confinement, relevé toutes les mesures et dimensions nécessaires pour réaliser graphiquement le chariot de manutention. De plus, étant deux élèves en classe de CTI3, nous pouvions chacun nous entraîner et travailler sur le projet de notre côté. Cependant, nous

avons décidé d'œuvrer ensemble via des conférences Zoom, car nous avons pensé que travailler indépendamment aurait pu créer des problèmes de compréhension et de coordination, eux-mêmes pouvant résulter de problèmes dans l'assemblage tout entier.

Étant donné que ce projet nécessite de la précision et de la rigueur, nous avons opté pour des réunions le week-end afin d'avancer sur le projet, en plus des heures de cours les vendredis matin.

2.2.2 Problèmes rencontrés et méthodes pour les résoudre

Au-delà des problèmes techniques, bugs, problèmes d'ordinateur ou de connexion, d'autres difficultés ont été rencontrées. En effet, à de nombreuses reprises, nous avons reçu des messages d'erreurs jamais eus auparavant (par exemple sur les congés ou l'assemblage). Le centre d'aide SolidWorks en ligne a cependant pu nous aider à régler ces complications.

Nous avons néanmoins rencontré des difficultés lors de l'ajout des roues sur le plateau dans l'assemblage final. En effet, il était question de problèmes d'hyperstatisme lors de l'insertion de boulons. Ce problème a été résolu en associant directement les roues dans l'assemblage principal, permettant ainsi à chacune de tourner indépendamment des autres.

Nous avons également essayé d'insérer un moteur électrique (disponible sur GRABCAD) dans notre assemblage mais le format étant trop lourd, le logiciel ne l'a pas supporté. Il a donc fallu recommencer l'assemblage et faire sans.

Chapitre 3

Simulation, asservissement

La partie suivante expose les différentes étapes de notre travail. L'asservissement est un problème complexe dont le principe est de s'adapter au scénario rencontré. Nous avons choisi le scénario le plus simple pour débiter dans ce projet, à savoir : le chariot est à l'arrêt, avance en ligne droite sur un sol plat non incliné et s'arrête.

3.1 Résolution numérique d'une EDO [4]

La projection horizontale de l'équation (valable en mouvement seulement) que vous avez à résoudre est

$$m \cdot \ddot{x}(t) = -m \cdot g \cdot \sin(\alpha) + f_{op}(t) + f_{ass}(t)$$

avec $f_{op}(t)$ la force de l'opérateur et $f_{ass}(t)$ celle de l'assistance moteur.

Le principe de la résolution numérique s'appuie sur deux concepts :

1. la discrétisation de la variable de description, ici le temps t
2. une approximation des fonctions sur cette discrétisation

3.1.1 Discrétisation

L'intervalle que l'on va étudier est discrétisé. Ainsi à la place d'avoir $t \in [0, t_{max}]$ on a $t_i \in [t_1, \dots, t_N]$ tel que $t_1 = 0$ et $t_N = t_{max}$.

On introduit généralement le pas de temps Δt tel que $t_{i+1} = t_i + \Delta t$.

Remarque : il est classique de choisir un Δt constant pendant toute l'étude, mais rien n'interdit de changer la valeur du pas sur l'intervalle étudié.

3.1.2 Approximation des fonctions connues

Pour les fonctions variables supposées connues, comme $f_{op}(t)$ et $f_{ass}(t)$, on doit fournir les valeurs $f_{op}(t_i)$ et $f_{ass}(t_i)$.

Si on ne connaît pas les valeurs des fonctions exactement aux instants t_i , on doit les approximer à partir de valeurs connues, par interpolation polynomiale par exemple.

Cette étape est considérée comme un pré-traitement de la résolution numérique, c'est-à-dire une opération réalisée en amont par l'opérateur. Même si dans notre cas l'asservissement est corrélé au mouvement, son évaluation est considérée comme un pré-traitement de la résolution.

3.1.3 Approximation des inconnues

On va faire la même chose pour les inconnues

$$x(t) \mapsto x_i = x(t_i)$$

$$\dot{x}(t) \mapsto \dot{x}_i = \dot{x}(t_i)$$

$$\ddot{x}(t) \mapsto \ddot{x}_i = \ddot{x}(t_i)$$

Par contre, on doit ajouter une relation entre ces variables, on parle de schéma, il en existe plusieurs.

Par exemple, le « schéma aux différences finies centrées d'ordre deux » (l'un des plus simples et souvent suffisant), permet d'écrire les relations suivantes :

$$\dot{x}_i \approx \frac{x_{i+1} - x_{i-1}}{2 \cdot \Delta t} \text{ et } \ddot{x}_i \approx \frac{x_{i+1} - 2 \cdot x_i + x_{i-1}}{\Delta t^2}$$

3.1.4 Algorithmique

Pour obtenir une approximation de la solution de l'EDO, on commence par évaluer l'état aux deux premiers pas de temps.

Par exemple à l'équilibre sans sollicitation, on a $x_1 = x_2 = 0$. (choisi par l'opérateur)

On peut ensuite écrire à chaque instant t_i :

$$m \cdot \frac{x_{i+1} - 2 \cdot x_i + x_{i-1}}{\Delta t^2} = -m \cdot g \cdot \sin(\alpha) + f_{op}(t_i) + f_{ass}(t_i)$$

et donc obtenir la récurrence

$$x_{i+1} = 2 \cdot x_i - x_{i-1} - \Delta t^2 \cdot g \cdot \sin(\alpha) + \frac{\Delta t^2}{m} \cdot f_{op}(t_i) + \frac{\Delta t^2}{m} \cdot f_{ass}(t_i) \tag{3.1}$$

On peut ainsi évaluer la valeur de $x(t)$ aux instants t_i par récurrence à partir des deux premières valeurs.

3.2 Simulations de la théorie

3.2.1 Premiers essais avec un tableur

Les premiers essais de simulations ont été faits avec un tableur.

Sur la base des équations fournies en 3.1, nous avons essayé de modéliser le déplacement, la vitesse et l'accélération en fonction du temps, après avoir choisi les forces $f_{op}(t)$ et $f_{ass}(t)$ arbitrairement.

Pour ce faire, nous avons décidé de travailler sur une période de 40 secondes, soit $t \in [0; 40]$, avec un pas de 1 seconde ($\Delta t = 1 s$). La force maximale de l'opérateur a été fixée à 25 N et celle de l'assistance calculée à 108,5 N (voir 1.2.3).

Nous avons opté, dans un premier temps, pour des forces linéaires, ne dépassant pas la valeur maximale, avec un écart de 5 s entre ces deux forces. Voici le diagramme les représentant en fonction du temps :

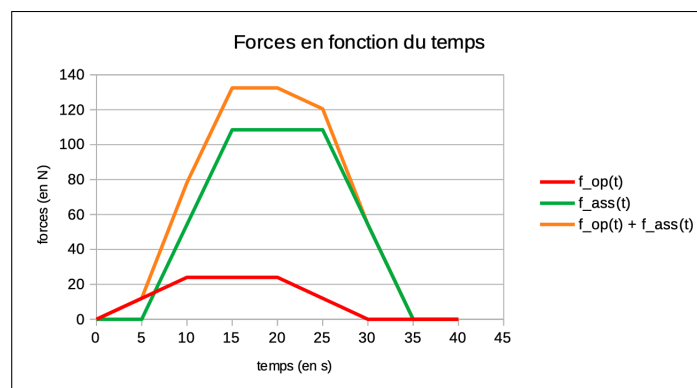


FIGURE 3.1 – Forces f_{op} et f_{ass} en fonction du temps t

Ensuite, nous avons calculé le déplacement à l'aide de l'équation 3.1, en fixant $x_0 = 0$ et $x_1 = 0$. Nous avons immédiatement remarqué un problème, pas dans la théorie mais pour l'application dans la réalité : le déplacement avait une forme « exponentielle » et les premières valeurs étaient négatives.

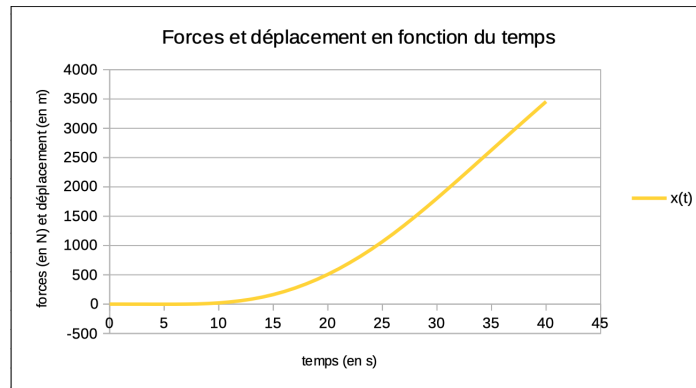


FIGURE 3.2 – Déplacement x en fonction du temps t

Suite aux conseils donnés par notre enseignant, nous avons modifié la façon de calculer le déplacement. En effet, pour éviter cette croissance « exponentielle », il fallait borner l'effort fourni ($effort = f_{op} + f_{ass} - m\ddot{x}$). On obtenait donc :

$$\begin{cases} x_{i+1} = \text{équation (3.1)} & \text{si } effort < mg \sin(\alpha) \text{ et } \dot{x}_i = 0 \\ x_{i+1} = mg \sin(\alpha) & \text{sinon} \end{cases}$$

En effet, quand l'effort dépasse la valeur de $mg \sin(\alpha)$, le chariot de manutention accélère de façon modérée, jusqu'à saturer l'accélération.

Cependant, après ces remarques, le déplacement était constant, égal à $mg \sin(\alpha)$, sur quasiment tout l'intervalle de temps.

De nombreux tests ont été effectués, parfois incohérents (déplacement négatif...), mais nous avons toujours cherché à comprendre d'où venaient les erreurs et comment y remédier. De plus, tous les calculs étaient liés, c'est-à-dire que le calcul d'une variable appelait celui d'une autre, elle-même nécessitant celui de la première variable ; les calculs avec le tableur devenaient rapidement illisibles et incompréhensibles.

3.2.2 Synthèse des calculs avec Matlab

La synthèse de tous ces calculs faits avec un tableur nous a été proposée par notre enseignant. Le code source est disponible en annexes (4.3).

Une fonction permettant de calculer le frottement a été implémentée à part, dans un souci de lisibilité et de décomposition du code (pour la réutiliser ultérieurement si besoin).

Les calculs en Matlab ont donc permis de déterminer les valeurs des différentes variables, x , \dot{x} , \ddot{x} , f_{op} , f_{ass} , à chaque tour de boucle.

Les résultats obtenus sont cohérents avec l'interprétation que nous pouvons leur donner :

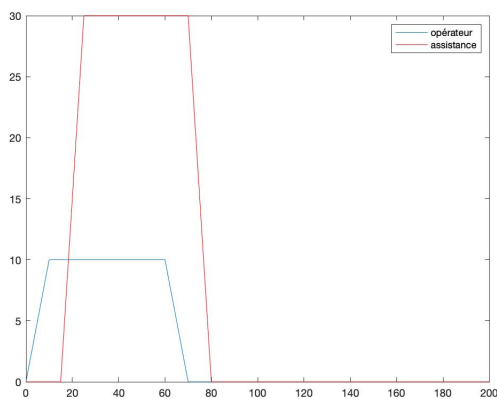


FIGURE 3.3 – Forces f_{op} (bleu) et f_{ass} (rouge) en fonction de t

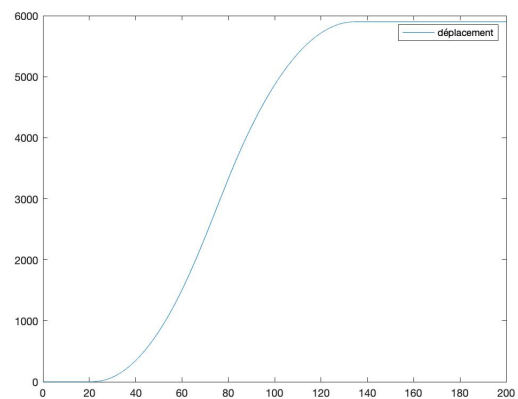


FIGURE 3.4 – Déplacement en fonction de t

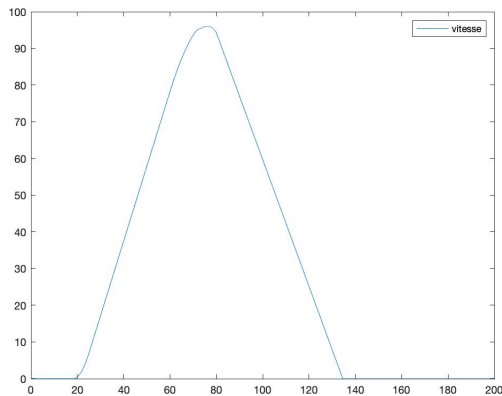


FIGURE 3.5 – Vitesse en fonction de t

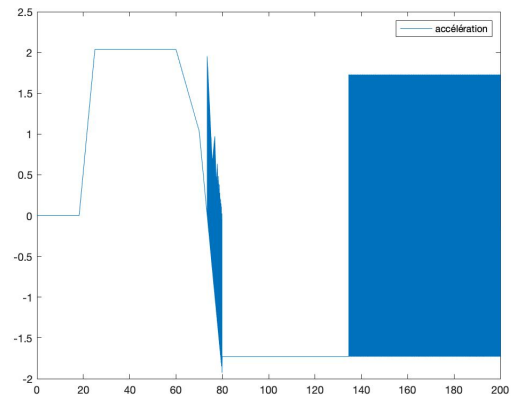


FIGURE 3.6 – Accélération en fonction de t

- Les forces ont bien un comportement linéaire comme choisi lors des essais avec le tableur.
- On retrouve le comportement « exponentiel » du déplacement mais avec une constance dès que $t \simeq 135 s$.
- La vitesse est nulle, croît, atteint un maximum puis diminue et redevient nulle. Lorsque celle-ci est nulle, le déplacement est bien constant.
- L'accélération varie et est bien cohérente avec le graphe de la vitesse. Cependant, nous remarquons sur ce dernier graphe une zone très bleutée ; ceci est le résultat d'une variation de l'accélération autour de sa valeur finale 0. En effet, elle n'atteint pas exactement 0 mais oscille autour de cette valeur.

3.3 Étude en pseudo-code

Le code fourni en annexes a pour but de fixer un scénario et de déduire la force totale ($f_{op} + f_{ass}$) en faisant une itération conditionnée par la vitesse.

Nous avons choisi le scénario le plus simple : un chariot en mouvement rectiligne sur une ligne droite, plane, et qui ne recule pas.

Ainsi, le fonctionnement du chariot, à travers le scénario, est tel que :

- Pour le temps $t \in [0; 15 s]$, le chariot accélère progressivement jusqu'à atteindre une accélération maximale et enfin la vitesse maximale.
Cette accélération est calculée, à travers une condition, par une boucle « répéter ».
- Pour $t \in [15; 90 s]$, le chariot avance avec une vitesse constante, fixée à $2 m/s$. L'itération dans cette phase joue le rôle d'un régulateur dont le but est de maintenir cette vitesse.
- Pour $t \in [90; 115 s]$, le chariot commence à décélérer jusqu'à s'arrêter. Ceci est de même conditionné par une itération sur la force ainsi que par la vitesse et l'accélération.
- Enfin, le chariot reste dans sa position finale fixe pour $t \in [115; 120 s]$.

Chapitre 4

Mesure de déformation

Dans cette partie, nous étudierons tout ce qui concerne la partie ACQUÉRIR de la chaîne d'information, consistant en la mesure de la déformation de la barre du chariot et la transmission du signal reçu en grandeur électrique pour la carte de commande. Je tenterai de détailler au mieux la mise en place de la mesure sur le prototype pour les futurs élèves amenés à travailler sur ce projet.

(Les figures dont je fais référence dans cette partie sont disponibles en annexes.)

4.1 Étude théorique

4.1.1 Analyse de la situation

Les connaissances et compétences acquises par les membres du groupe ayant choisi le module *P9-1 : Résistance des Matériaux* avec Matthieu TOURBIER seront utiles.

Le système étudié est la barre du prototype. On peut modéliser ce système comme une poutre droite E en liaison encastrement avec le plateau 0. L'utilisateur exerce une force \vec{F} au bout de la poutre E . La poutre est de section circulaire de grand diamètre D et de longueur l connus et de petit diamètre d inconnu. Le but est de déterminer l'épaisseur de la tige creuse (donc d) à utiliser pour le prototype expérimental. Voir le schéma [Figure 4.1](#).

Actions mécaniques sur la poutre E :

$$\begin{aligned} \text{— Liaison encastrement en O : } \{T_{E/0}\} &= \begin{Bmatrix} X_O & 0 \\ Y_O & 0 \\ 0 & N_O \end{Bmatrix}_O \\ \text{— Force de l'utilisateur en A : } \{T_{ext/E}\} &= \begin{Bmatrix} 0 & 0 \\ -F & 0 \\ 0 & 0 \end{Bmatrix}_A \end{aligned}$$

Déterminons le torseur de cohésion de la poutre E . Voir [Figure 4.2](#).

$$\{T_{coh}\} = \{T_{\bar{E}/E_2}\} = -\{T_{\bar{E}/E_1}\} = \{T_{ext/E}\}$$

Il faut déplacer le torseur $\{T_{ext/E}\}$ au point G qui est le centre de gravité de la section S .

$$\{T_{ext/E}\} = \left\{ \begin{array}{c} -F\vec{y} \\ \vec{M}_G(T_{ext/E}) = \vec{M}_A(T_{ext/E}) + \vec{GA} \wedge -F\vec{y} \end{array} \right\}$$

$$\boxed{\{T_{coh}\} = \begin{Bmatrix} 0 & 0 \\ -F & 0 \\ 0 & -F(l-x) \end{Bmatrix}_G}$$

On remarque que la poutre est sollicitée en flexion plane autour de l'axe (G, \vec{z}) .

La contrainte σ exercée sur une section de normale \vec{x} est donnée par la relation suivante : $\sigma = -\frac{M_{fz}}{I_{Gz}} y$ avec :

— M_{fz} le moment de flexion d'axe (G, \vec{z})

- I_{Gz} le moment quadratique par rapport à l'axe (G, \vec{z})
- y la coordonnée du vecteur \overrightarrow{GM} dans la direction \vec{y}

On a alors :

$$\sigma(x, y) = \frac{F(l-x)}{I_{Gz}}y$$

Dans ce contexte où la mesure de la déformation de la barre du prototype de chariot est déterminante pour le bon fonctionnement de l'assistance électrique, nous devons placer notre capteur à un endroit stratégique, c'est-à-dire là où la déformation de la barre est la plus importante pour permettre un relevé de mesures fiable et correct. La déformation est la plus importante là où la contrainte sur une section de la poutre est la plus forte, à savoir à la surface de la barre, au niveau de la liaison avec le plateau. Donc le point M où la contrainte est la plus forte a pour coordonnées $(x_{max}; y_{max}) = (0; \frac{D}{2})$. On a $\sigma_{max} = \frac{FDl}{2I_{Gz}}$. Désormais on choisira $y = \frac{D}{2}$

et $x = 0$.

D'après la loi de Hooke, la contrainte selon \vec{x} est proportionnelle à la déformation longitudinale de la poutre, ce qui se traduit par la relation suivante : $\sigma_x = E\varepsilon_x$, avec E le module de Young du matériau utilisé pour la tige. On a alors $\varepsilon_x = \frac{\sigma_{max}}{E} = \frac{FDl}{2EI_{Gz}}$. On remarque que ε est proportionnel à la force F de facteur $\frac{Dl}{2EI_{Gz}}$.

Le capteur que nous allons utiliser pour mesurer la déformation est une jauge de déformation (aussi appelée jauge de contrainte ou extensomètre à fils résistants). C'est une feuille mince très petite constituée d'un fil fin qui se colle sur l'objet dont on va mesurer la déformation. S'agissant d'une mesure fine, on doit passer par la conception d'un pont de Wheatstone. Pour comprendre son fonctionnement, voir [Figure 4.3](#).

4.1.2 Dimensionnement du pont et de la tige

Voici les caractéristiques de la tige que nous nous sommes fixées pour le prototype :

- D et l sont des données fixées par Mathilde et Raphaël qui se chargent de la modélisation du prototype du chariot. $D = 5 \text{ mm}$ et $l = 239 \text{ mm}$.
- La tige sera en ABS, un polymère thermoplastique. Son module de Young vaut $E = 3000 \text{ MPa}$.
- $I_{Gz} = \frac{\pi(D^4 - d^4)}{64}$ pour un tube circulaire creux .
- $F_{max} = 2,5 \text{ N}$ est la force maximale que peut exercer l'opérateur sur une des deux barres du prototype. D , d et l sont exprimées en mm , E en MPa , F en N et I_{Gz} en mm^4 .

Il faut maximiser la déformation de la barre pour obtenir les plus grandes valeurs de tension possibles, celles-ci étant relativement petites. On a : $\varepsilon = \frac{FDl}{2EI_{Gz}} = \frac{64FDl}{2E(D^4 - d^4)\pi}$. Il faut donc que l'épaisseur soit la plus petite possible. Nous avons prévu de réaliser le tube par imprimante 3D qui permet une finesse d'épaisseur pouvant aller jusqu'à 1 mm .

Il faut également adapter le diamètre intérieur d de la barre du prototype, de sorte qu'on obtienne une déformation inférieure ou égale à une certaine valeur avec F_{max} . En sachant que les jauges mesurent en général des déformations allant jusqu'à 5%, alors $\varepsilon \leq 0,05 \iff d \leq \left(D^4 - \frac{64FDl}{0,1E\pi}\right)^{\frac{1}{4}}$.

Donc finalement, $d \leq 4,53 \text{ mm}$. Pour s'assurer une certaine marge de sécurité on prendra $d = 4 \text{ mm}$, soit 1 mm d'épaisseur.

Concernant le pont de Wheatstone :

- Le facteur de jauge vaut pour la grande majorité des jauges de déformation 2. Donc on prendra $k = 2$.
- Le pont sera alimenté par la carte Arduino, branchée sur 5V.
- Pour choisir correctement les résistances, j'ai effectué des simulations du montage en pont de Wheatstone sur le logiciel Tinkercad. Voir [Figure 4.4](#). Les jauges sont proposées avec des résistances¹ de 120, 350, 700 ou 1000Ω. J'ai donc, de manière expérimentale, mesuré la tension e pour ces différentes valeurs de R et pour divers couples de résistances $(R; R_4)$ et $(R_2; R_3)$. J'ai constaté les observations suivantes :
 - Plus l'écart des valeurs entre des couples $(R; R_4)$ et $(R_2; R_3)$ est grande, plus la tension e sera faible.
 - Si on prend $(R; R_4) = (R_3; R_4)$ alors la valeur de la tension mesurée augmentera si on diminue la valeur de résistance choisie.

1. <https://www.hbm.com/fileadmin/mediapool/hbmdoc/temporary-shop-files/b4709.pdf>

Après ces observations et d'après les différentes résistances proposées pour la jauge, on prendra donc une valeur commune pour toutes les résistances du circuit, à savoir $R = R_2 = R_3 = R_4 = 120\ \Omega$.

Il faudra brancher les jauges sur un amplificateur 5 V car la tension en sortie du pont est trop faible.

4.2 Mise en pratique : installation du système de mesure de déformation

4.2.1 Connexions et branchements

Nous allons utiliser une jauge de contrainte linéaire à une grille de mesure de résistance nominale $120\ \Omega$, voir [Figure 4.5](#).

La jauge sera placée selon les coordonnées $(x; y)$ déterminées précédemment et dans la longueur de la barre.

Il faut connecter la jauge au pont de Wheatstone, préalablement monté sur une platine d'essai, en soudant les fils aux deux pattes de connexion et en branchant les deux autres extrémités sur la platine, voir [Figure 4.6](#).

Une fois la jauge connectée au pont de Wheatstone, il faut relier le tout à un amplificateur de contrainte. Nous allons utiliser le module HX711, particulièrement bien adapté au pont de Wheatstone et peu cher. Voir [Figure 4.7](#).

Il faut à présent relier le pont de Wheatstone au HX711. Pour cela, il faut brancher les points de la diagonale alimentée AB (voir [Figure 4.6](#)) aux pattes RED et BLK ou YLW . Concernant la diagonale de sortie où l'on mesure la tension e , le point C est connecté à GRN et le point D à WHT . Les connexions peuvent être réalisées soit par soudure, soit par connecteurs et câbles Dupont. Voir [Figure 4.8](#).

Les branchements entre l'amplificateur HX711 et la carte Arduino sont explicités [Figure 4.9](#).

4.2.2 Programme Arduino

La librairie Arduino concernant le HX711 et le dossier comportant le programme de mesure ("*SPARKFUN HX711 LOAD CELL AMPLIFIER EXAMPLES (ZIP)*") à télécharger sont disponibles à cette adresse², section *Installing the HX711 Arduino Library and Examples*.

Nous effectuons d'abord une calibration pour obtenir une mesure juste de l'effort de l'opérateur. Pour cela, le prototype doit être correctement maintenu à la verticale, grâce à un étau par exemple (voir [Figure 4.10](#)). Sur le logiciel de programmation *Arduino*, il faut ouvrir le fichier *SparkFun_HX711_Calibration*. Le processus de calibration est détaillé à la [Figure 4.11](#).

Une fois la calibration réalisée, le fichier *SparkFun_HX711_Example* est ouvert et la valeur du facteur de calibration est remplacée par celle obtenue précédemment. Si on souhaite afficher la masse en *kg*, il faudra modifier le code en créant une variable *float* qui récupère la valeur donnée après mesure et la diviser par 10. Voir [Figure 4.12](#).

À présent, nous avons en place le système de mesure de déformation qui permettra de commander le moteur par la suite.

4.3 Pistes d'amélioration

Les jauges de déformation sont très sensibles aux variations de température. Pour pallier ce problème, on peut utiliser la configuration en demi-pont de Wheatstone, constituée de deux jauges identiques. Voir [Figure 4.13](#).

Pour gérer la marche arrière, on pourra coller une jauge sur l'autre barre verticale du prototype, au niveau de la zone de traction de la barre quand l'opérateur décide de reculer le chariot, dans le but de maximiser l'allongement de la jauge et éviter des problèmes de confusion entre variables dans l'asservissement du moteur. On branchera cette seconde jauge à un deuxième amplificateur HX711 et à la carte Arduino. Voir [Figure 4.14](#) pour les branchements.

². https://learn.sparkfun.com/tutorials/load-cell-amplifier-hx711-breakout-hookup-guide?_ga=2.106773060.65950924.1591578527-1370434676.1591488749#installing-the-hx711-arduino-library-and-examples

Il faudra répéter l'étape de calibration pour la seconde jauge. Cette fois-ci, on inversera le sens de maintien du prototype.

Ainsi, si on souhaite pousser le chariot vers l'avant, la première jauge mesurera une déformation plus grande et donc un effort F_1 plus conséquent que celui de la seconde jauge, F_2 . Le programme asservissant les moteurs prendra toujours la valeur d'effort la plus grande entre les deux jauges. Par exemple si $F_2 > F_1$ et que $F_2 \geq F_{2(min)}$, qui est le seuil d'effort minimal pour activer l'assistance électrique, alors le programme détectera que c'est une manoeuvre de marche arrière du chariot et donc le moteur tournera dans le sens inverse, et inversement. Cependant, n'ayant pas pu mettre en place le système, cette hypothèse reste à vérifier et à améliorer.

Conclusion et perspectives

Conclusions

Travailler sur la partie CAO du projet *Assistance électrique d'un chariot de manutention* nous a permis d'améliorer nos compétences sur SolidWorks. Cela nous a aidé à mieux comprendre comment était conçu le chariot de manutention, puisque nous l'avons construit nous-mêmes pièce par pièce, et également à voir les contraintes qui y sont exercées.

Concernant la partie simulation, il était très intéressant d'essayer de rendre le problème plus visuel avec des graphiques. Nous n'avons pas toujours eu les résultats souhaités ou même terminé cette étude, mais nous avons cherché par nous-mêmes, tenté d'expliquer les observations obtenues, qu'elles soient cohérentes ou non. Au premier abord, on pourrait croire que ce projet n'est pas compliqué, mais quand on y travaille, on se rend compte qu'il y a de nombreux paramètres à prendre en compte. L'asservissement est une étude complexe à mener mais tout aussi intrigante.

La mesure de la déformation n'est pas compliquée à mettre en place et concerne de nombreux objets de la vie quotidienne (par exemple les balances électroniques). J'ai trouvé cela très intéressant de comprendre comment fonctionnent les jauges de déformation et comment les intégrer à un circuit électronique pour mesurer des efforts. J'aurais vraiment voulu le réaliser et le tester car il me reste beaucoup de questions actuellement en suspens et dont les réponses se trouvent dans l'expérimentation du prototype.

Ce projet nous a permis de découvrir de nouveaux sujets non approfondis en cours : à partir d'une problématique, nous cherchons des solutions afin de progresser sur l'étude. L'esprit de curiosité, le travail d'équipe, ceux de communication et d'argumentation sont autant de valeurs développées à travers cette expérience, permettant des échanges enrichissants sur les divers points de vue abordés. Ces qualités sont d'ailleurs essentielles à une future carrière d'ingénieur.

Perspectives pour la poursuite de ce projet

- Approfondir l'étude de l'asservissement.

- Prendre en compte les virages dans la mesure des forces.

- Mettre en place les travaux réalisés sur le prototype afin de faire des tests ; s'ils sont concluants, les adapter au vrai chariot de manutention.

- Joindre une console digitale qui permet à l'opérateur de mieux maîtriser le chariot.

Bibliographie

- [1] La Rédac Cyclable. 11 septembre 2018. *Vélo électrique : grandes notions sur l'assistance électrique* [En ligne]. <https://www.cyclable.com/blog/2018/09/11/velo-electrique-grandes-notions/> (consulté le 05/03/2020)
- [2] Anne-Sophie STAMANE. *Vélo électrique - Bien choisir son vélo électrique* [En ligne]. <https://www.quechoisir.org/guide-d-achat-velo-electrique-n8147/#les-criteres-pour-bien-choisir-son-velo-electrique> (consulté le 05/03/2020)
- [3] BERNAGE Grégory, GAMBIER Cyprien, GUILLEMETTE Loïc, HEILBRONNER Philippine, HUTEAU Benjamin, LAÏMOUCHE Farès, PEUZIAT Lucas. Juin 2019. Rapport de projet : *Principe de l'assistance électrique*, 26 pages. (document), 1.2
- [4] LEMOSSE Didier. Mars 2020. Aide fournie par l'enseignant : *Résolution numérique d'une EDO*. (document), 3.1
- [5] Sarah AL-MUTLAQ, Alex The Giant. *Load Cell Amplifier HX711 Breakout Hookup Guide* [En ligne]. <https://learn.sparkfun.com/tutorials/load-cell-amplifier-hx711-breakout-hookup-guide/introduction> (consulté le 11/06/2020)
- [6] Gilles THEBAULT. 3 mars 2020. *Arduino & Amplificateur HX711 SPARKFUN pour capteur d'effort (avec pont de Wheatstone)* [En ligne]. <http://gilles.thebault.free.fr/spip.php?article59> (consulté le 11/06/2020)
- [7] Guy Techlab. 19 septembre 2017. *Jauge de contrainte, HX 711 et Arduino* [Vidéo]. <https://www.youtube.com/watch?v=5LzKFFdCazo&list=WL&index=46&t=982s> (consulté le 11/06/2020)
- [8] Electronoobs. 26 janvier 2020. *How STRAIN GAUGE Works | Precision SCALE with Arduino* [Vidéo]. https://www.youtube.com/watch?v=LRd3W_p8PJ4&list=WL&index=46 (consulté le 11/06/2020)
- [9] Electronoobs. 3 mars 2019. *Basic configurations #1 - Wheatstone bridge* [Vidéo]. https://www.youtube.com/watch?v=ZqAM_wQ35ow&list=WL&index=47 (consulté le 09/06/2020)
- [10] GreatScott!. 3 décembre 2017. *Electronic Basics #33 : Strain Gauge/Load Cell and how to use them to measure weight* [Vidéo]. https://www.youtube.com/watch?v=1WFiKMSB_4M&list=WL&index=48 (consulté le 11/06/2020)
- [11] LEMOSSE Didier. Février 2020. *La mesure des déformations par les jauges de contraintes* [Document papier]. (consulté le 11/06/2020)

Annexes

Code source Matlab

Fonction de calcul du frottement

```

1  function ffirt = frottement(xp,fopt,fasst,m,g,sina,xpp)
2      if (xp(end) == 0)
3          if (abs(fopt+fasst) < (m*g*sina))
4              ffirt = fopt + fasst ;
5          else
6              sens = (fopt + fasst) / abs(fopt + fasst) ;
7              ffirt = m*g*sina*sens ;
8          end
9      else
10         sens = xp(end) / abs(xp(end));
11         if ( xp(end) / xp(end-1) > 1 )
12             ffirt = m*g*sina*sens;
13         else
14             ffirt = abs(m*xpp(end)) * sens;
15         end
16         % pause
17     end
18 end

```

Programme principal

```

1  % simulation mouvement chariot %
2  %% problem data
3
4  m = 10;
5  sina = 0.2;
6  g = 9.81;
7  fmax = m*g*sina;
8  tfinal = 200;
9  dt = 0.01;
10
11  fop = [
12      0 0
13      10 10
14      60 10
15      70 0
16      tfinal 0
17  ] ;
18
19  fass = [

```

```

20     0 0
21     15 0
22     25 30
23     70 30
24     80 0
25     tfinal 0
26     ];
27
28     x = (0);
29     xp = (0);
30     xpp = (0);
31
32     hist = 0;
33     t = 0;
34     while (t < tfinal)
35         t = t + dt;
36         fopt = interp1(fop(:,1),fop(:,2),t);
37         fasst = interp1(fass(:,1),fass(:,2),t);
38
39         ffrt = frottement(xp,fopt,fasst,m,g,sina,xpp);
40
41         %ffrt
42         %pause
43
44         xppt = (1/m) * (fopt+fasst-ffrt);
45         xpt = xp(end) + dt/2 * (xppt+xpp(end));
46         xt = x(end) + dt*xp(end) + dt^2/4 * (xppt+xpp(end));
47
48         xpp = [ xpp ; xppt ];
49         xp = [ xp ; xpt ];
50         x = [ x ; xt ];
51         hist = [hist;t];
52     end
53
54     close all
55
56     figure(1)
57     plot(hist,xpp)
58     legend('accélération')
59
60     figure(2)
61     plot(hist,xp)
62     legend('vitesse')
63
64     figure(3)
65     plot(hist,x)
66     legend('déplacement')
67
68     figure(4)
69     plot(fop(:,1),fop(:,2))
70     hold on
71     plot(fass(:,1),fass(:,2),'r')
72     legend('opérateur','assistance')

```

Essai de pseudo-code

Algorithme 4.1 Pseudo-code asservissement

Déclarations

$x_p, x_{pp}, x_{p_{max}}, x_{pp_{max}}, t, f_{tot}$: Réels
 pas : Entier

Début

$x_{p_{max}} \leftarrow 2$
 $x_{pp_{max}} \leftarrow 1$
 $f_{tot} \leftarrow 5$
 $pas \leftarrow 1$

Si $t = 0$ **Alors**

$x_p \leftarrow 0$
 $x_{pp} \leftarrow 0$

FinSi

Pour $t \leftarrow 0$ à 120 **Faire**

Si $t > 0$ et $t \leq 15$ **Alors**

Si $x_p < x_{p_{max}}$ **Alors**

Si $x_{pp} < x_{pp_{max}}$ **Alors**

Répéter

$f_{tot} \leftarrow f_{tot} + pas$
 $x_{pp} \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$
 $x_p \leftarrow x_{p_{max}} \times \frac{t}{15}$ // Pour être sûr de sortir de la boucle « répéter ».

Jusqu'à ce que $x_{pp} = x_{pp_{max}}$ et $x_p = x_{p_{max}}$

FinSi

Si $x_{pp} > x_{pp_{max}}$ **Alors**

Répéter

$f_{tot} \leftarrow f_{tot} - pas$
 $x_{pp} \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$
 $x_p \leftarrow x_{p_{max}} \times \frac{t}{15}$

Jusqu'à ce que $x_{pp} = x_{pp_{max}}$ et $x_p = x_{p_{max}}$

FinSi

FinSi

FinSi

Si $t > 15$ et $t \leq 90$ **Alors**

$x_{pp} \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$

Tant Que $x_{pp} > 0,05$ et $x_p \neq x_{p_{max}}$ **Faire** // 0,05 est dû à l'incertitude sur l'accélération,
// en effet, elle n'atteint pas 0 mais oscille autour.

$f_{tot} \leftarrow f_{tot} - pas$

$x_{pp} \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$

$x_p \leftarrow 2 \times x_{p_{max}} - \frac{t}{90} \times x_{p_{max}}$ // Pour être sûr de sortir de la boucle « tant que ».

FinTantQue

FinSi

Si $t > 90$ et $t \leq 115$ **Alors**

$x_{pp} \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$

Tant Que $x_{pp} \geq 0$ et $x_p \neq 0$ **Faire**

$f_{tot} \leftarrow f_{tot} - pas$

$x_{pp} \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$

$x_p \leftarrow x_{p_{max}} - \frac{t}{115} \times x_{p_{max}}$

FinTantQue

FinSi

...

...

```

Si  $t > 115$  et  $t \leq 120$  Alors
   $xpp \leftarrow \frac{1}{m} \times f_{tot} - g \sin(\alpha)$ 
  Tant Que  $f_{tot} \neq 0$  Faire
     $f_{tot} \leftarrow f_{tot} - pas$ 
  FinTantQue
FinSi
  
```

FinPour

Fin

Mesure de déformation – Compléments

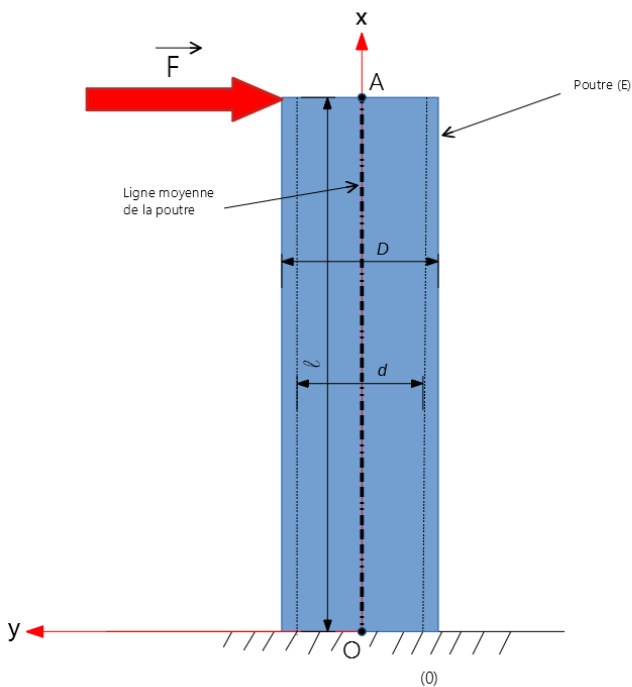


FIGURE 4.1 – Schéma simplifié du plateau et barre de chariot

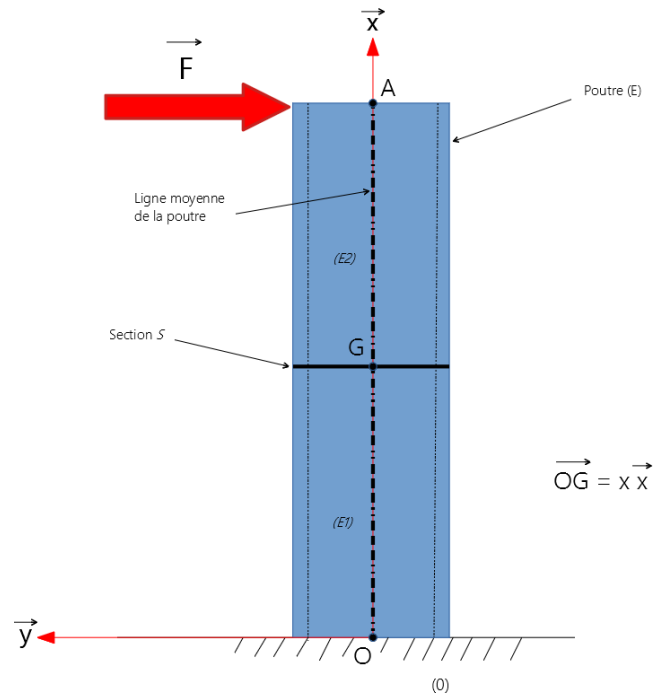
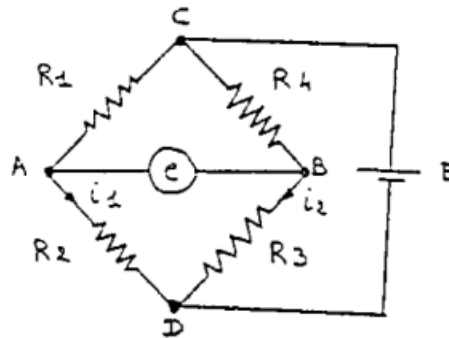


FIGURE 4.2 – Le torseur de cohésion est le torseur associé aux actions mécaniques (forces et moments) qui s'exercent sur la section S du tronçon $(E1)$

FIGURE 4.3 – Fonctionnement du pont de Wheatstone

Le pont de Wheatstone est un circuit électrique comportant 4 résistances.



Son principe est le suivant : une diagonale du pont (ici CD) est alimentée par une tension continue E . Une autre diagonale (AB) est traversée par une tension e . Cette tension sera mesurée et servira à la chaîne d'information.

Initialement, la tension e mesurée est nulle, le pont est dit équilibré. Si $e = 0$ alors les potentiels aux points A et B sont identiques. Donc d'après la loi d'Ohm : $R_1 i_1 = R_4 i_2$ et $R_2 i_1 = R_3 i_2 \iff R_1 R_3 = R_2 R_4$.

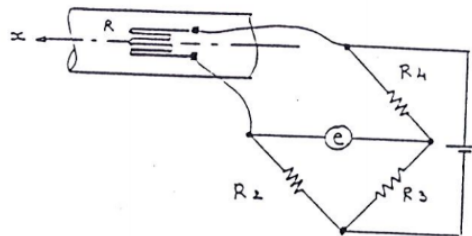
On choisira par exemple $R_1 = R_4$ et $R_2 = R_3$.

Si depuis cet équilibre il y a une variation des valeurs de résistance de ΔR , alors il apparaît une tension e mesurable. On a alors la relation suivante :

$$e = C \times E \left(\frac{\Delta R_1}{R_1} - \frac{\Delta R_2}{R_2} + \frac{\Delta R_3}{R_3} - \frac{\Delta R_4}{R_4} \right)$$

avec $C = \frac{1}{4} = \frac{R_1 R_4}{(R_1 + R_4)^2}$.

Nous allons faire varier uniquement la résistance R_1 qui sera celle de la jauge de déformation.



Par le phénomène de piézorésistance, la jauge de déformation voit, sous l'effet d'une contrainte, sa résistance varier et donc l'apparition d'un courant qui la traverse. La déformation étant très petite, sa variation de résistance est donc difficilement mesurable avec un Ohmmètre. C'est pour cela qu'on utilise le pont de Wheatstone.

La tension e est proportionnelle à la variation de résistance ΔR de la jauge, les autres résistances étant invariantes. On a alors $e = \frac{E}{4} \times \frac{\Delta R}{R}$. Or un capteur comme cette jauge peut mesurer l'allongement unitaire d'une pièce dans la direction de la jauge. On a donc $\frac{\Delta l}{l} = \varepsilon = \frac{1}{k} \times \frac{\Delta R}{R} \iff \frac{\Delta R}{R} = \varepsilon k$.

On a alors $e = \frac{E}{4} \times \varepsilon k$ avec k le facteur de jauge. On remarque que la tension e mesurée aux bornes de la résistance R_1 de la jauge est proportionnelle à sa déformation longitudinale ε .

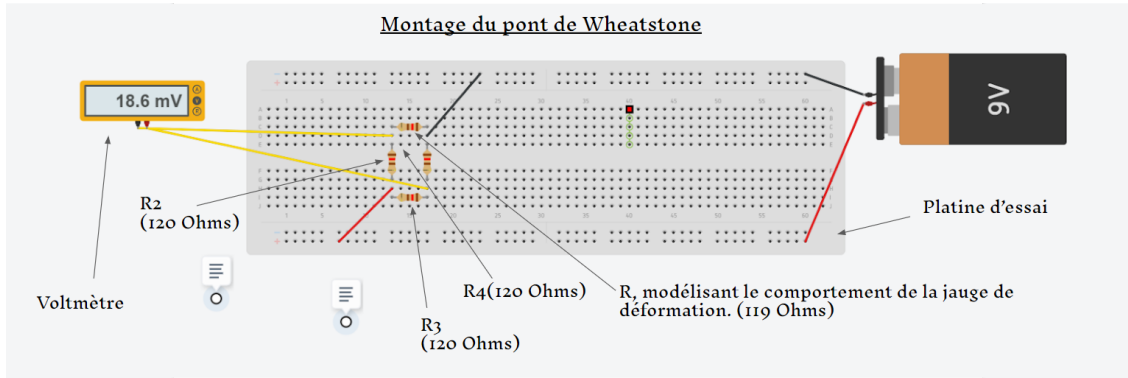
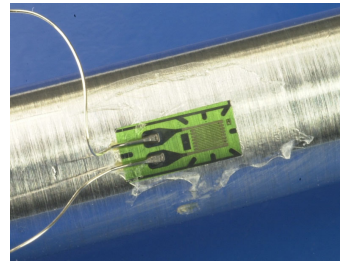
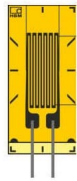


FIGURE 4.4 – Montage du pont de Wheatstone

Le but est d'obtenir une mesure de la tension la plus élevée possible pour une variation de résistance la plus petite possible. Sur Tinkercad, la plus petite variation de résistance ΔR que j'ai pu obtenir est de 1Ω . En-dessous de cette valeur, le voltmètre n'est pas assez précis pour détecter la variation de résistance et donc afficher la variation de tension associée.



(a) On remarque les pattes de connexion en bas de la jauge

(b) Collage de la jauge sur un tube

FIGURE 4.5 – Jauge de contrainte

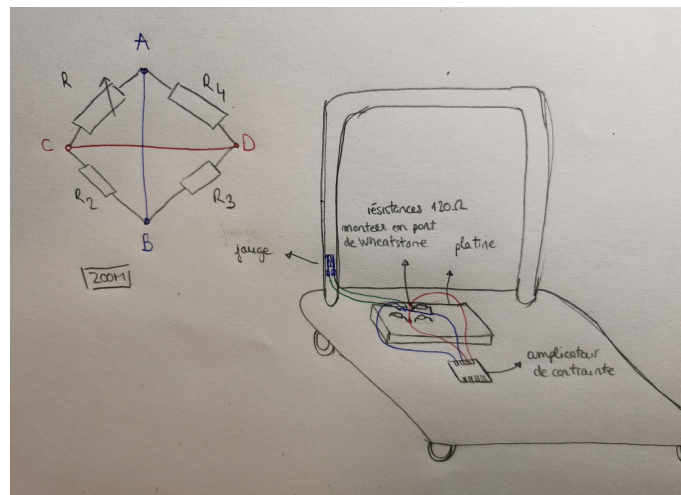
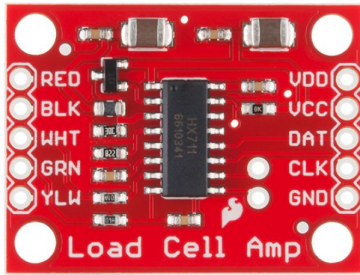
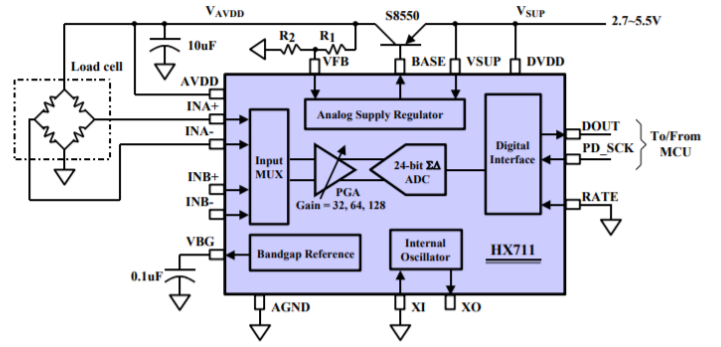


FIGURE 4.6 – Croquis du montage

Pour des soucis d'illustration, la jauge est collée sur la face avant de la barre. Il vaut mieux la coller de l'autre côté, là où la barre subit un effort de traction pour ainsi maximiser la déformation de la jauge.



(a) Image du composant



(b) Schéma-bloc du HX711

FIGURE 4.7 – HX711

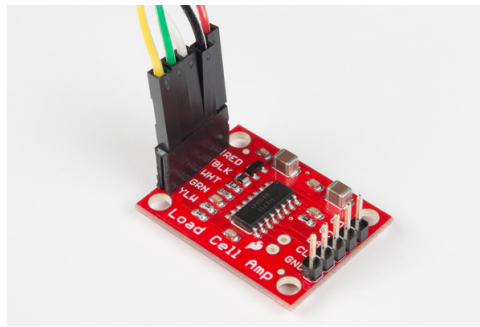
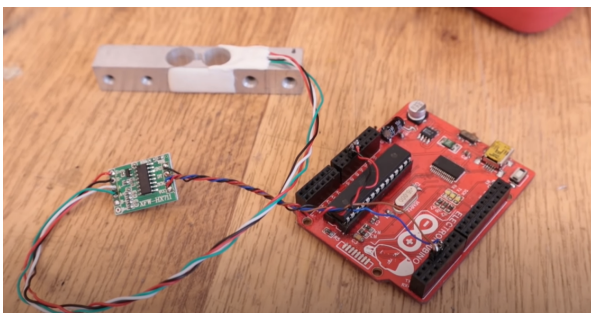
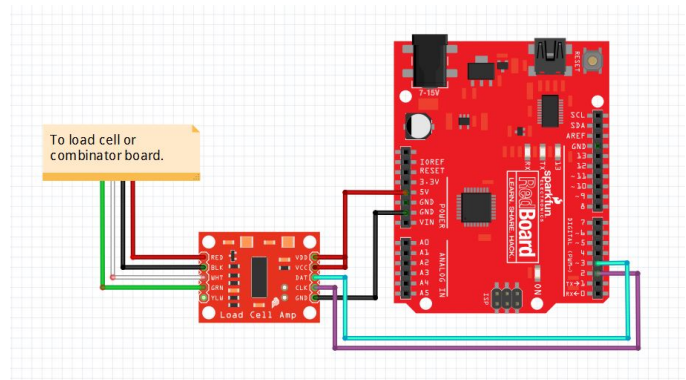


FIGURE 4.8 – Connexion au HX711 par câble Dupont



(a) Exemple de photographie



(b) Schéma des branchements

FIGURE 4.9 – Montage final

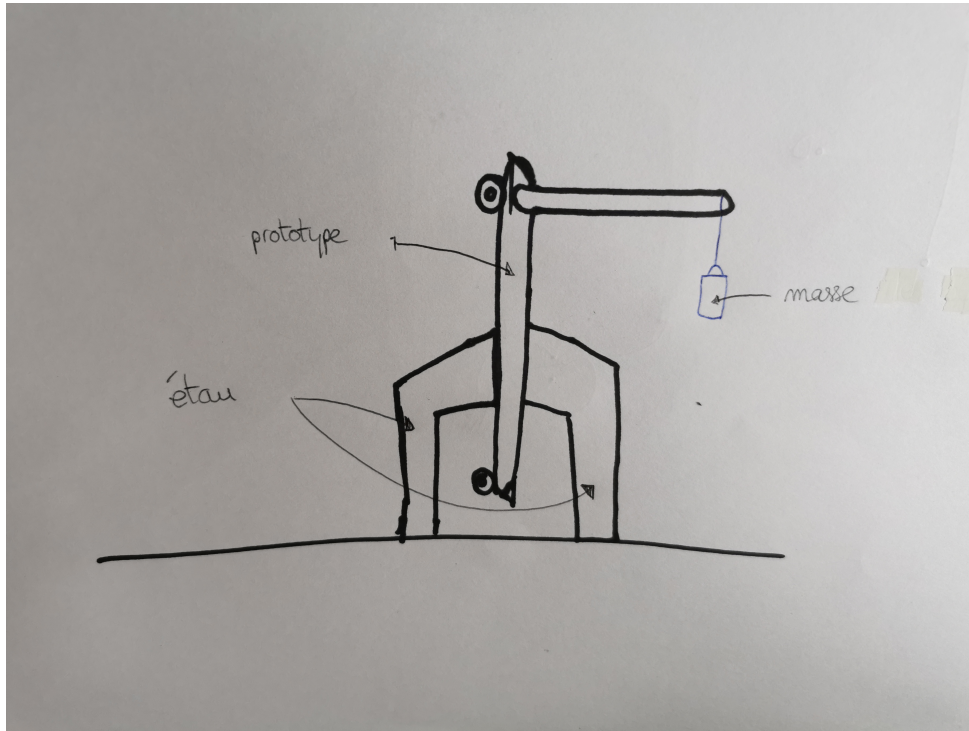


FIGURE 4.10 – Maintien du chariot prototype pour la calibration (les composants électroniques ne sont pas représentés mais ils sont bel et bien présents)

```

SparkFun_HX711_Calibration

#include "HX711.h" //This library can be obtained here http://librarymanager/All#Avia\_HX711

#define LOADCELL_DOUT_PIN 3
#define LOADCELL_SCK_PIN 2

HX711 scale;

float calibration_factor = -7050; // -7050 worked for my 440lb max scale setup

void setup() {
  Serial.begin(9600);
  Serial.println("HX711 calibration sketch");
  Serial.println("Remove all weight from scale");
  Serial.println("After readings begin, place known weight on scale");
  Serial.println("Press + or a to increase calibration factor");
  Serial.println("Press - or z to decrease calibration factor");

  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
  scale.set_scale();
  
```

FIGURE 4.11 – Calibration de la jauge

Après avoir branché en USB la carte Arduino à l’ordinateur, on téléverse le programme et ouvre le moniteur série (situé en haut à droite dont le symbole est une loupe).

On laisse le programme fonctionner quelques secondes à vide puis on accroche une masse, dont on connaît la valeur, au milieu de la barre centrale du prototype. Le but va être d’obtenir sur le moniteur série la bonne valeur de la masse. Pour cela, on va agir sur le facteur de calibration en envoyant à la carte Arduino les commandes “+” ou “-” pour changer de 10 la valeur de ce facteur. On répète l’opération jusqu’à ce que le programme affiche la valeur correcte de la masse.

La jauge étant collée sur seulement une des deux barres verticales, on divisera par 2 la valeur de la masse connue qu’on souhaite voir affichée à l’écran.

Attention aux unités : ici la masse est affichée en livres (*lb*), il faudra donc procéder à des conversions entre *kg* et *lb*.

```
#include "HX711.h" //This library can be obtained here http://librarymanager/All#Avia\_HX711

#define calibration_factor -7050.0 //This value is obtained using the SparkFun_HX711_Calibration sketch

#define LOADCELL_DOUT_PIN 3
#define LOADCELL_SCK_PIN 2

HX711 scale;

void setup() {
  Serial.begin(9600);
  Serial.println("HX711 scale demo");

  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
  scale.set_scale(calibration_factor); //This value is obtained by using the SparkFun_HX711_Calibration sketch
  scale.tare(); //Assuming there is no weight on the scale at start up, reset the scale to 0

  Serial.println("Readings:");
}

void loop() {
  Serial.print("Reading: ");
  Serial.print(scale.get_units(), 1); //scale.get_units() returns a float
  float masse = (scale.get_units());
  masse = masse/10;
  Serial.print(" kg "); //You can change this to kg but you'll need to refactor the calibration_factor
  Serial.println();
}
```

FIGURE 4.12 – Programme Arduino permettant de mesurer la force appliquée par l’opérateur sur la barre du prototype.

Ici les variables *LOADCELL_DOUTPIN_PIN* et *LOADCELL_SCK_PIN* sont les données transmises par le HX711 donc proviennent directement de la déformation de la jauge.

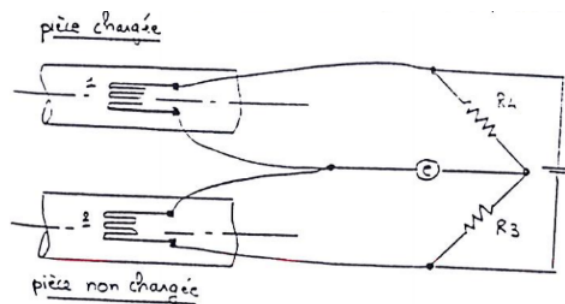


FIGURE 4.13 – Fonctionnement d’un pont de Wheatstone à deux jauges

La première jauge sera comme d’habitude collée à la barre. La deuxième sera collée sur une pièce constituée du même matériau (l’ABS en l’occurrence) qui ne subira aucune force. La résistance de la première jauge variera de ΔR en fonction de la charge subie et de $\Delta R'$ en fonction de la dilatation de la pièce due à la variation de température. La résistance de la seconde jauge ne variera que de $\Delta R'$ car elle ne subit aucune charge.

Dans un pont de Wheatstone, les variations de résistances adjacentes se retranchent tandis que celles opposées s’additionnent. Donc si on place les deux jauges de manière adjacente, la variation de résistance $\Delta R'$ due à la température s’annule et donc on ne garde que la variation de résistance ΔR due à la charge appliquée à la barre. On supprime ainsi l’influence de la température dans la mesure de la déformation.

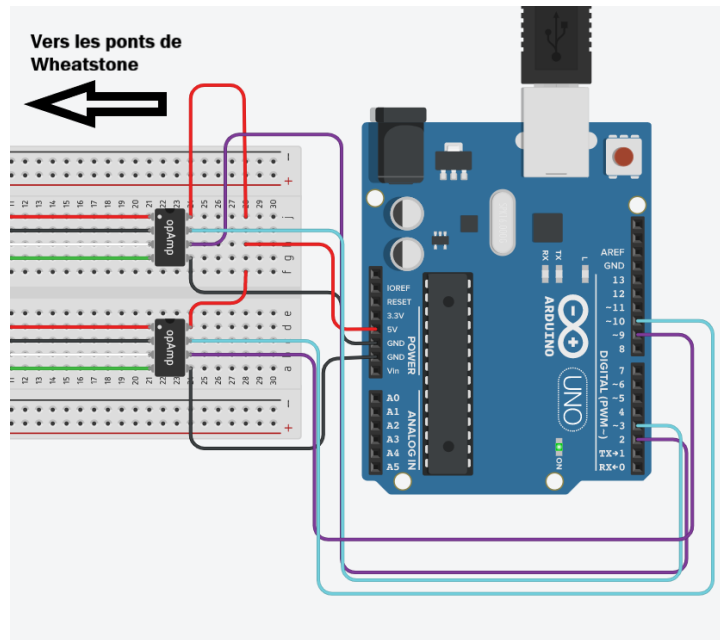


FIGURE 4.14 – Branchements pour deux modules HX711 à la carte Arduino

Le HX711 n'étant pas disponible sur Tinkercad, j'ai pris un amplificateur opérationnel simple à l'apparence similaire. On le considèrera donc comme des modules HX711. Les couleurs de fils sont les mêmes que celles référencées dans la [Figure 4.9b](#).