

## Projet technique

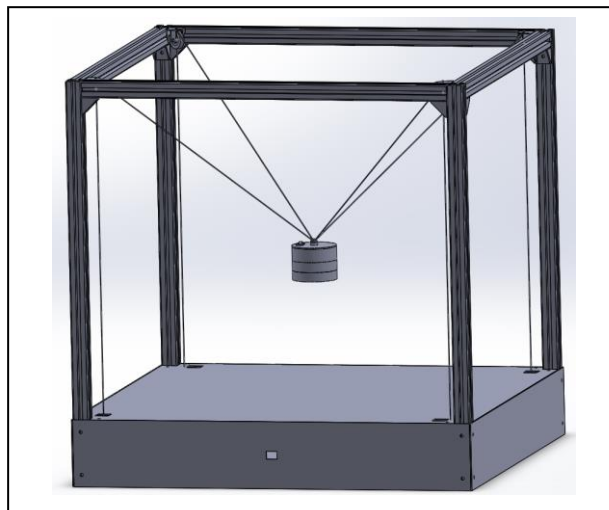
Présenté par :

(G. AUBERT – B. DECHÊNE – M. IBARRA – A. MOURIER – G. RICHARD)

# Robot parallèle à câbles

Supervisé par :

(A. GILLET)





---

## Résumé/Abstract

Le **Proof Of Concept (POC)** est une démonstration de faisabilité d'un projet qui porte sur une idée technologique, parmi tous les projets proposés en début HEI4, notre choix s'est porté sur la réalisation d'un robot parallèle à câbles. L'utilisation est purement ludique afin d'appréhender et comprendre les capacités que possède un robot parallèle à câbles dans le monde de l'industrie.

The **Proof Of Concept (POC)** is a feasibility demonstration of a project that relates to a technological idea, among all the projects proposed at the beginning of HEI4, our choice fell on the realization of a cable-driven parallel robot. The use is purely playful to apprehend and figure it out the capabilities of a cable-driven parallel robot in the world of industry.

## 1 Table des matières

Liste des figures .....	6
Introduction .....	8
<b>1. Projet.....</b>	<b>9</b>
1.1 Les robots parallèles à câbles .....	9
1.2 Notre concept.....	11
1.3 Concepteurs.....	12
<b>2 Organisation .....</b>	<b>12</b>
2.1 Client.....	12
2.2 Structure de l'équipe .....	13
<b>3 Analyse fonctionnelle.....</b>	<b>13</b>
3.1 Problématique .....	13
3.2 Diagramme bête à cornes .....	14
3.3 Diagramme pieuvre .....	14
<b>4 Objectifs .....</b>	<b>15</b>
4.1 Technique .....	15
4.2 Pédagogiques.....	16
4.3 Applications .....	16
<b>5 Planification.....</b>	<b>16</b>
<b>6 Etude technique.....</b>	<b>17</b>
6.1 Choix de la structure.....	17
6.2 Choix des composants .....	17
6.2.1 Moteurs.....	18
6.2.2 Drivers .....	18
6.2.3 Carte électronique.....	20
6.2.4 Electroaimant .....	22
6.2.5 Alimentation .....	22
6.3 Mécanique .....	23
6.3.1 Treuil .....	23
6.3.2 Equerre .....	26
6.3.3 Passage de câble .....	26
6.3.4 Câble .....	28
6.4 Développement de l'effecteur du robot .....	29

6.5	Mise en place de l'électronique .....	31
6.6	Assemblage.....	31
6.7	Modélisation du robot.....	32
6.8	Architecture logicielle.....	33
6.8.1	IHM (mode manuel) .....	33
6.8.2	IHM (application ludique) .....	35
6.8.3	Trames.....	36
6.8.4	Gestion interne du robot .....	37
6.8.5	Boucle fermée et remise à zéro .....	38
6.9	Performance du produit .....	41
6.9.1	Précision .....	41
6.9.2	Robustesse .....	41
6.9.3	Vitesse .....	41
<b>7</b>	<b>Coût final du démonstrateur .....</b>	<b>42</b>
<b>8</b>	<b>Perspectives d'évolution .....</b>	<b>42</b>
8.1	Applications .....	42
8.2	Modifications.....	43
8.3	Potentiel changement de concept .....	43
<b>9</b>	<b>Difficultés .....</b>	<b>44</b>
9.1	Anticipées .....	44
9.2	Rencontrées.....	44
<b>10</b>	<b>Conclusion .....</b>	<b>46</b>
10.1	Générale .....	46
10.2	Personnelles.....	46
	AUBERT Guillaume .....	46
	DECHENE Bastien.....	46
	IBARRA Mariela .....	47
	MOURIER Antoine .....	47
	RICHARD Gauthier .....	47
<b>11</b>	<b>Bibliographie .....</b>	<b>48</b>
<b>12</b>	<b>Annexes.....</b>	<b>49</b>

## Liste des figures

Figure 1: Robot parallèle à 8 câbles (4 en haut & 4 en bas).....	10
Figure 2: Robot parallèle à 8 câbles (8 en haut).....	10
Figure 3: Robot parallèle à 6 câbles (6 en haut).....	10
Figure 4: Robot parallèle à 8 câbles dans un repère mobile.....	10
Figure 5: Robot parallèle à câbles avec une structure particulière .....	10
Figure 6: Robot parallèle à câbles version simplifiée.....	11
Figure 7 : Tableau des différentes attributions des tâches du projet .....	13
Figure 8: Diagramme bête à cornes .....	14
Figure 9: Diagramme Pieuvre.....	15
Figure 10: Description diagramme Pieuvre.....	15
Figure 11 : Structure initiale.....	17
Figure 12 : Structure modifiée .....	17
Figure 13: Tableau caractéristique pour le choix du moteur .....	18
Figure 14 : Moteur Pas à Pas Nema 23.....	18
Figure 15: Détermination couple moteur.....	18
Figure 16: Câblage driver n°1 .....	19
Figure 17: Câblage driver n°2.....	19
Figure 18: Description carte Arduino Mega .....	21
Figure 19: Logiciel de développement Arduino.....	21
Figure 20: Électroaimant.....	22
Figure 21: Alimentation.....	22
Figure 22: Treuil .....	23
Figure 23: Support.....	23
Figure 24: Bobine .....	24
Figure 25: Bobine vue de côté .....	24
Figure 26: Blocage câble .....	24
Figure 27: Roulement .....	25
Figure 28: Fixation moteur.....	25
Figure 29: Arbre moteur forme D-Cut .....	25
Figure 30: Equerre.....	26
Figure 31: Équerres positionnées.....	26
Figure 32: Passage de câble 3D .....	26
Figure 33: Passage de câble réel .....	26
Figure 34: Simulation RPAC.....	27
Figure 35: Bobine de câble.....	28
Figure 36: Effecteur n°1 .....	29
Figure 37: Effecteur n°2.....	29
Figure 38 : Effecteur n°3.....	29
Figure 39 : Effecteur final.....	30
Figure 40: Assemblage final .....	31
Figure 41: Représentation .....	32
Figure 42: Architecture logicielle.....	33

# POC : ROBOT PARALLÈLE À CÂBLES

---

Figure 43: IHM mode manuel .....	33
Figure 44: Consignes de position .....	34
Figure 45: Contrôle de l'électroaimant .....	34
Figure 46: Contrôle des treuils .....	34
Figure 47 : Contrôle de connexion au mode manuel .....	35
Figure 48 : Interface du plateau de Tic-Tac-Toe .....	35
Figure 49 : Contrôle de connexion au mode de jeu .....	36
Figure 50 : Choix du mode de jeu .....	36
Figure 51 : Plateau de jeu .....	36
Figure 52 : Cheminement des trames envoyées .....	37
Figure 53: IHM de gestion du robot .....	39
Figure 54 : Servomoteur .....	39
Figure 55 : Schéma bloc du système pouvant être mis en place .....	40
Figure 56: Autre architecture de robot à câbles .....	44

### Introduction

Le **Proof Of Concept** est une démonstration de faisabilité, c'est une réalisation expérimentale concrète et préliminaire illustrant une certaine méthode permettant de démontrer la faisabilité. Celle-ci est réalisée tout au long de la deuxième année du cycle ingénieur en HEI4. Dans le cadre de la formation d'ingénieur généraliste, nous devons savoir et apprendre à prototyper la partie matérielle d'une étude de faisabilité et donc de réaliser un système fonctionnel afin d'évaluer la performance de l'étude.

Cette étude porte sur une idée technologique mise en avant lors de la rentrée en HEI4. Les professeurs ont proposé certains systèmes à réaliser. Parmi tout cela, nous avons porté notre choix sur la réalisation d'un robot parallèle à câbles.

Notre système fera l'objet de démonstration lors des portes ouvertes, salons ou forums. L'utilisation est purement ludique afin de comprendre et de se rendre compte des capacités que possède un robot parallèle à câbles dans le monde de l'industrie.

La première apparition de notre système doit se faire lors du Showroom, à la suite de notre présentation POC en mi-février 2021. Pour cette occasion, nous devons présenter des démonstrations permettant d'illustrer les capacités du robot.



## 1. Projet

### 1.1 Les robots parallèles à câbles

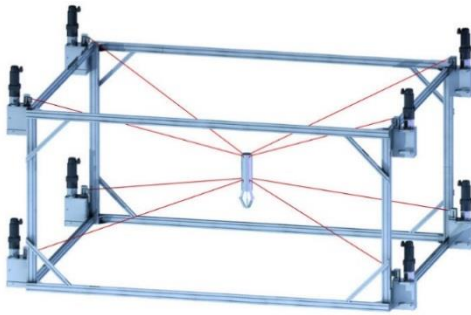
Dans les années 1980 et le début des années 1990, aux États-Unis et principalement au Massachusetts Institute of Technology (MIT), le concept et le développement des premiers robots parallèles à câbles sont introduits.

Les robots parallèles sont une catégorie de robots présentant des chaînes cinématiques fermées. Le champ des architectures cinématiques possibles est très vaste, ainsi que le nombre de domaines d'application qui s'étendent de la micromanipulation à la manipulation de charges élevées, en passant par d'autres applications. Les performances des robots parallèles sont complémentaires et généralement à l'opposé de celles des robots séries. Ils ouvrent donc de nouvelles pistes pour la robotisation de certaines tâches. Ils représentent aujourd'hui une faible part de marché pour les robots industriels commercialisés, mais ils s'imposent naturellement pour certaines applications.

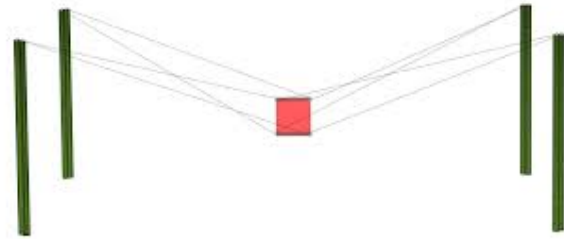
Les robots parallèles à câbles se présentent comme une nouvelle classe de robots parallèles. Cette technologie est dépendante des câbles, donc sujette à l'unilatéralité des efforts exercés par les câbles sur la plate-forme.

Ce type de robot est très performant étant donné sa vitesse, mais aussi, sa précision. En effet, moins l'effecteur présente d'inertie, plus il sera aisé de lui donner une très bonne dynamique, d'où l'intérêt de ce concept. Nous pouvons retrouver ses différentes utilisations dans l'industrie (déplacement de charges lourdes, peintures précises, Pick & Place...).

Le nombre de degrés de liberté est dépendant du nombre de câbles utilisé, de leur positionnement et de la structure du robot, il existe naturellement différentes sortes de robot parallèle à câbles. En voici quelques modèles :



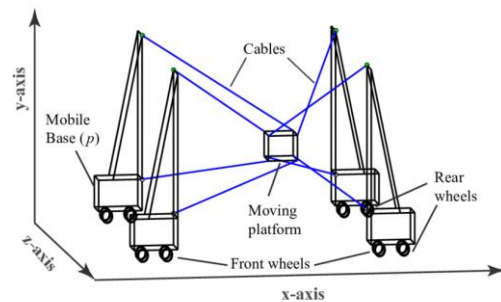
*Figure 1: Robot parallèle à 8 câbles (4 en haut & 4 en bas)*



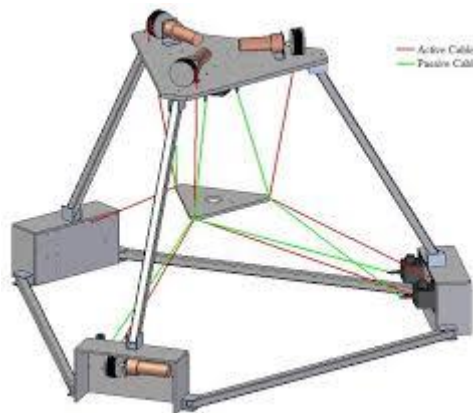
*Figure 2: Robot parallèle à 8 câbles (8 en haut)*



*Figure 3: Robot parallèle à 6 câbles (6 en haut)*



*Figure 4: Robot parallèle à 8 câbles dans un repère mobile*



*Figure 5: Robot parallèle à câbles avec une structure particulière*

Chacun de ces modèles possède leurs propres avantages et inconvénients en termes de degré de liberté et de dynamique.

Ces robots présentent un très grand intérêt pour l'industrie moderne grâce à leurs qualités. Certains de ces robots sont capables d'atteindre des vitesses vraiment impressionnantes (au niveau de l'effecteur du robot), ceux qui pourrait permettre d'optimiser des processus de Pick & Place par exemple.

## 1.2 Notre concept

Dans le cadre de notre projet POC, compte tenu des contraintes de temps et de nos faibles connaissances en robotique (en particulier, dans le domaine des robots parallèles à câbles), nous avons décidé de concevoir le robot parallèle à câbles le plus simple possible. En voici une schématisation ci-dessous :



*Figure 6: Robot parallèle à câbles version simplifiée*

Il s'agit d'un effecteur simplement supporté par quatre câbles supérieurs, il possède trois degrés de liberté dans l'espace. Il ne sera pas capable d'atteindre des vitesses importantes en descente car sa descente est uniquement contrôlée par la pesanteur. Il sera néanmoins capable de supporter des charges importantes (relative à la puissance de nos moteurs). Ici, l'outil est une pince mais nous l'avons remplacé par un électroaimant afin de déplacer des pièces métalliques ferreuses.

## 1.3 Concepteurs

Pour réaliser ce POC, notre équipe est composée de cinq personnes :



## 2 Organisation

### 2.1 Client

Le client de notre projet est le directeur d'HEI Campus Centre, M. GILLET qui pour lors des portes ouvertes, showrooms, salons ou forums désirait posséder un démonstrateur de robot parallèle à câbles, afin de faire comprendre le fonctionnement ainsi que d'avoir une application concrète à montrer lors de ces événements.

## 2.2 Structure de l'équipe

Au niveau de l'équipe, nous nous sommes réparti les tâches de la façon suivante :

Étudiants	Tâches
DECHENE Bastien	<ul style="list-style-type: none"> <li>• Gestion de projet</li> <li>• Réalisation de la partie "Modélisation 3D sous SOLIDWORKS"</li> <li>• Réalisation de la partie "Montage"</li> <li>• Assistance pour la partie "Programmation"</li> </ul>
AUBERT Guillaume	<ul style="list-style-type: none"> <li>• Réalisation de la partie "Programmation"</li> <li>• Réalisation de la partie "Montage"</li> <li>• Assistance pour la partie "Calculs mathématiques"</li> </ul>
IBARRA Mariela	<ul style="list-style-type: none"> <li>• Réalisation de la partie "Electronique"</li> <li>• Réalisation de la partie "Vérification de conformité"</li> <li>• Assistance pour la partie "Modélisation 3D sous SOLIDWORKS"</li> </ul>
MOURIER Antoine	<ul style="list-style-type: none"> <li>• Réalisation de la partie "Programmation"</li> <li>• Réalisation de la partie "Vérification de conformité"</li> <li>• Assistance pour la partie "Calculs mathématiques"</li> </ul>
RICHARD Gauthier	<ul style="list-style-type: none"> <li>• Réalisation de la partie "Calculs mathématiques"</li> <li>• Réalisation de la partie "Vérification de conformité"</li> <li>• Assistance pour la partie "Electronique"</li> </ul>
Pour tous les étudiants, il sera nécessaire de participer à la rédaction du CDCF, du rapport et à la réalisation de la soutenance finale.	

*Figure 7 : Tableau des différentes attributions des tâches du projet*

Il faut noter que cette organisation de l'équipe n'a pas été parfaitement respectée et que nous avons chacun travaillé sur les parties des autres dans un souci d'entraide et d'assimilation de nouvelles compétences mais aussi dans l'optique de mener à terme le projet.

## 3 Analyse fonctionnelle

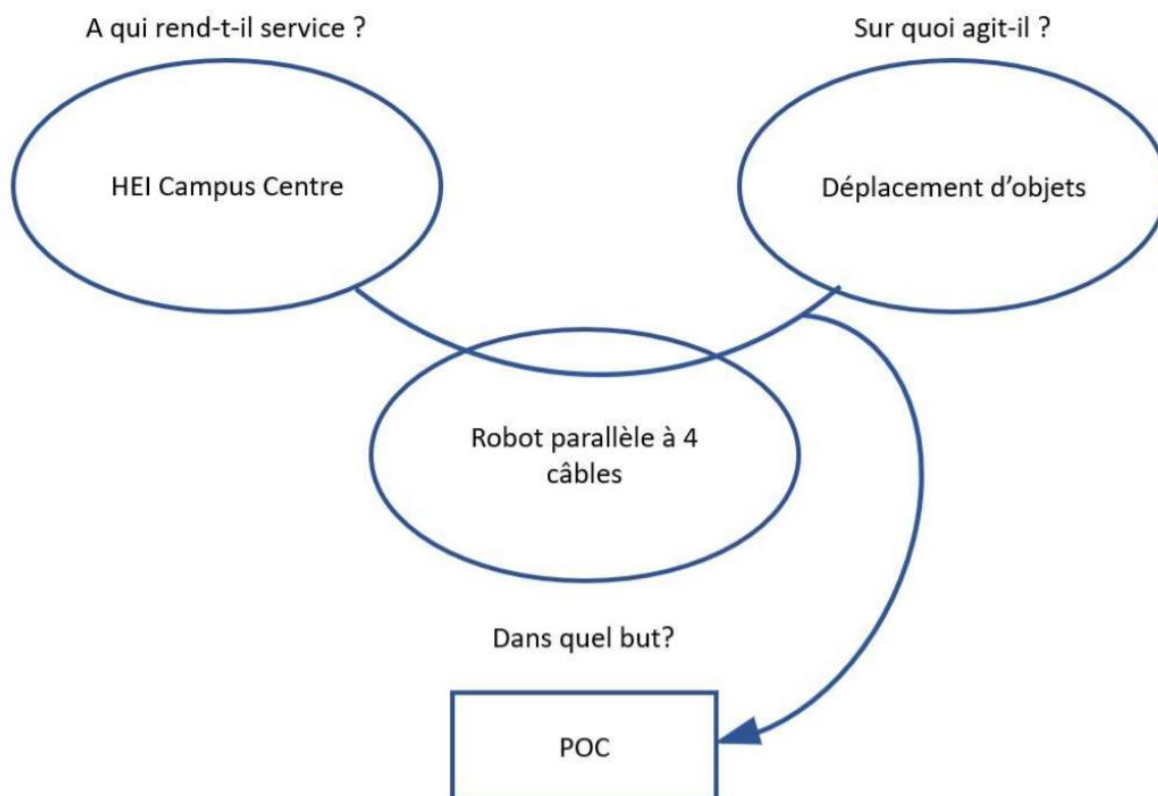
### 3.1 Problématique

Lors de la réalisation de ce projet, nous nous sommes basés sur la problématique suivante :

« Est-il possible de réaliser un démonstrateur de robot parallèle à câbles pour une application ludique ? »

### 3.2 Diagramme bête à cornes

Le diagramme bête à cornes est un outil d'analyse fonctionnelle du besoin. Cet outil est nécessaire à l'élaboration du besoin et permet de définir celui-ci sous forme de fonctions simples. Ce diagramme nous permet de nous renseigner sur la cible ou l'utilisateur, la matière d'œuvre sur laquelle nous allons agir et la fonction principale, son but, son intérêt. Nous avons alors réalisé un diagramme bête à cornes nous permettant de définir plus précisément notre besoin.



*Figure 8: Diagramme bête à cornes*

### 3.3 Diagramme pieuvre

Le diagramme pieuvre, ou graphe des interactions, est utilisé pour analyser les besoins et identifier les fonctions de services d'un produit. Le diagramme pieuvre met en évidence les relations entre les différents éléments du milieu environnant et le produit. Au centre du diagramme se trouve notre système, afin de répondre à notre problématique et autour de celui-ci, se trouvent des fonctions de service qui conduisent à la satisfaction du besoin. Elles se décomposent en deux catégories, d'une part, les fonctions principales et d'autre part les fonctions contraintes.



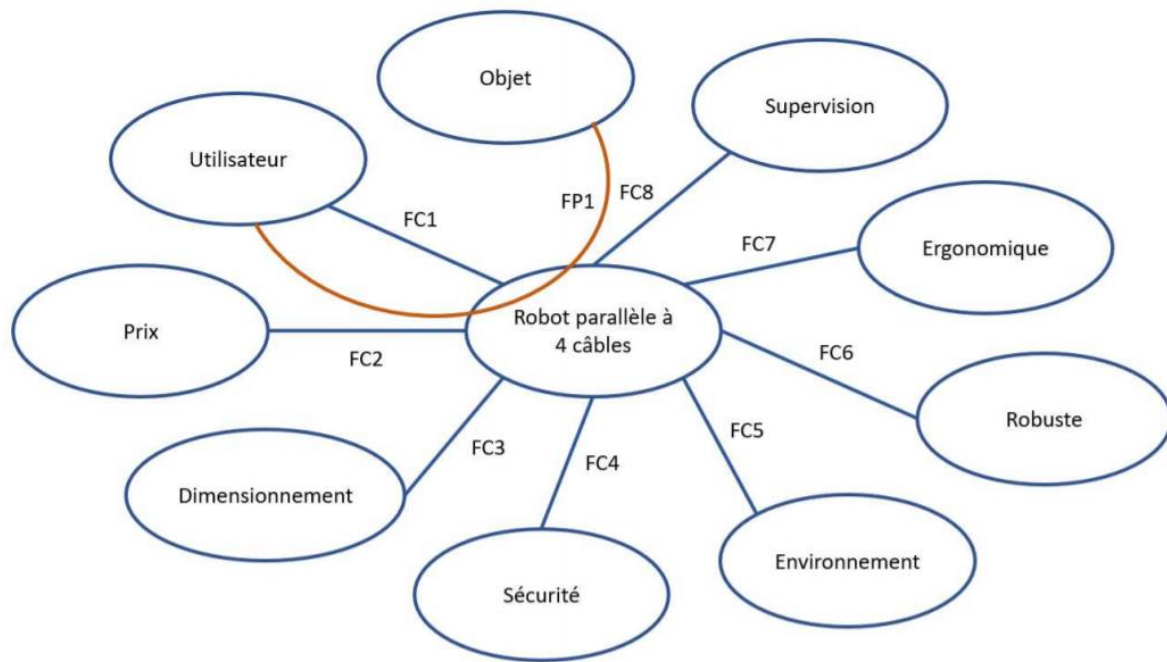


Figure 9: Diagramme Pieuvre

Désignation	Description
FP1	Déplacer un objet dans l'espace en respectant les axes mathématiques (x, y et z) via le contrôle de l'utilisateur.
FC1	Permettre à l'utilisateur de contrôler le robot.
FC2	Le prix du robot ne doit pas excéder 500 €.
FC3	Le robot doit occuper un volume de 1 m <sup>3</sup> maximum.
FC4	Le robot doit avoir un espace de travail sécurisé et protégé.
FC5	Le robot ne doit pas gêner l'espace environnant.
FC6	Le robot doit être résistant et pouvoir supporter le poids de l'objet.
FC7	Le robot doit être facile et adapté à son utilisation.
FC8	Permettre le contrôle du robot via une Interface Homme-Machine (IHM).

Figure 10: Description diagramme Pieuvre

## 4 Objectifs

### 4.1 Technique

L'objectif technique de ce projet a été de réaliser un déplacement de charge via l'utilisation d'un robot parallèle à quatre câbles. Ce déplacement s'effectue sur les trois axes, c'est-à-dire les axes X, Y et Z. Les déplacements sont contrôlés par un utilisateur via des interfaces logicielles.

## POC : ROBOT PARALLÈLE À CÂBLES

### 4.2 Pédagogiques

L'objectif pédagogique est la réalisation du projet technique de fin d'école d'ingénieur au sein de l'école HEI Campus Centre. De plus, grâce à cela, nous avons amélioré nos connaissances concernant les domaines suivants :

- Automatique ;
- Electrique ;
- Electronique ;
- Informatique ;
- Gestion de projet ;
- Mécanique ;
- Mécatronique ;
- Robotique.

### 4.3 Applications

Conformément au cahier des charges défini avec notre client, les objectifs sont simples. Premièrement, de réaliser la structure, de faire fonctionner le robot en mode manuel par exemple. Et en second, de réaliser une application ludique à travers le jeu du Tic-Tac Toe.

## 5 Planification

Pour gérer au mieux ce projet, un diagramme prévisionnel ont été mis en place afin de planifier les actions à réaliser.

Le Diagramme Gantt est l'un des outils les plus efficaces pour représenter visuellement l'état d'avancement des différentes tâches/activités qui constituent un projet. La colonne de gauche du diagramme énumère toutes les tâches à effectuer tandis que la ligne d'en tête représente les unités de temps les plus adaptées au projet (semaines, mois, etc...). Dans notre cas, le mois est l'unité de temps utilisé. Chaque tâche est matérialisée par une barre horizontale, dont la position et la longueur représentent la date de début, la durée et la date de fin.

Le diagramme GANTT prévu initialement ainsi que le diagramme réel sont disponibles en annexes.

On remarque que le planning réel est totalement différent du planning qui avait été prévu. Cette forte différence provient des différents reports causés par la crise sanitaire.



## 6 Etude technique

### 6.1 Choix de la structure

Il a été décidé, lors de la rédaction du cahier des charges, que la structure du robot devrait tenir dans un volume d'environ  $1\text{ m}^3$ . La structure étant essentiellement composée de profilés en aluminium, HEI nous a fourni une structure initialement prévue pour un autre projet afin de pouvoir réaliser notre robot. Voici la structure obtenue au début du projet :



*Figure 11 : Structure initiale*

Nous avons ensuite fait découper les profilés aux bonnes dimensions afin de pouvoir construire la structure répondant au cahier des charges :



*Figure 12 : Structure modifiée*

### 6.2 Choix des composants

Le choix des composants a été une étape cruciale dans la réalisation de ce projet. Choisir les bons composants permet de réaliser un robot répondant au cahier des charges tout en étant le plus performant possible.

### 6.2.1 Moteurs

Afin de réaliser la motorisation du robot, nous avons décidé d'utiliser des moteurs pas-à-pas bipolaires car ils étaient adaptés à notre utilisation. Ils permettent des rotations continues (contrairement à la plupart des servomoteurs), ils permettent des déplacements de précision (contrairement à des moteurs à courant continu seuls) et ils sont bien moins coûteux que d'autres types de moteurs (comme les moteurs brushless).

Nous avons décidé de choisir des moteurs pas-à-pas bipolaire **NEMA 23** présentant ces caractéristiques :

Tension nominale	3,2 V
Courant maximum consommé par phase	2,8 A
Couple maximum du moteur	1,9 Nm

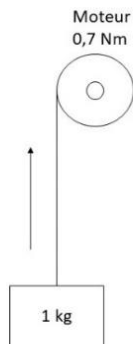
*Figure 13: Tableau caractéristique pour le choix du moteur*

La tension nominale ainsi que le courant maximum consommé par phase permettent de dimensionner les drivers.

Nous avons déterminé que le couple de ce moteur était suffisant car nous avons démontré expérimentalement qu'un moteur seul de 0,7 Nm est capable de soulever une masse de 1 kg verticalement sur un câble.



*Figure 14 : Moteur Pas à Pas Nema 23*



*Figure 15: Détermination couple moteur*

Donc, nous pouvons en déduire que quatre moteurs possédant un couple de 1,9 Nm seront suffisants pour soulever une masse de seulement quelques centaines de grammes. Des moteurs de cette taille seront parfaits dans le cas où le robot servirait lors d'une application plus lourde, les années suivantes par exemple.

Les caractéristiques mécaniques du moteur (dimension, forme de l'axe, etc.) ont été utilisés pour la réalisation de la CAO des treuils du robot.

### 6.2.2 Drivers

Lors du processus de dimensionnement des drivers, nous avons testé deux types de drivers de moteurs pas-à-pas.

Le premier était totalement sous-dimensionné par rapport à la puissance des moteurs et résulte d'une grave erreur de dimensionnement. Le premier driver était de la référence **A4988**. Il se mettait en place de cette manière :

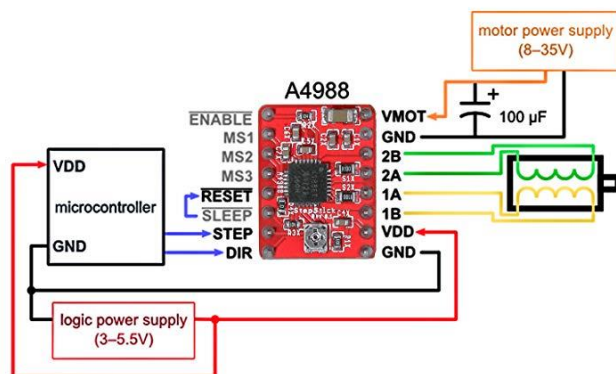


Figure 16: Câblage driver n°1

Ce fut un échec car les drivers souffraient d'un échauffement grave et cela aurait amené à la destruction du matériel donc nous avons dû changer de référence de driver. L'erreur de notre part provient du fait que cette référence de driver est utilisée sur la majorité des imprimantes 3D de classe amateur, nous avons donc présumé beaucoup trop rapidement qu'ils seraient suffisants pour contrôler les moteurs, sans regarder réellement les caractéristiques électriques et donc leurs compatibilités. Le problème d'échauffement provenait du fait que le moteur appelait trop de courant par rapport à la capacité du driver et que celui-ci alimentait le moteur au-dessus de sa tension nominale de fonctionnement (d'où le courant si élevé). Il ne pouvait pas non plus supporter le pic de courant au démarrage du moteur.

Le second était évidemment plus robuste et plus adapté, nous avons pris des drivers **L298N**. Il s'agit d'un driver normalement utilisé pour piloter deux moteurs à courant continu mais il est possible de s'en servir pour piloter des moteurs pas-à-pas bipolaires. Ce driver est basé sur un double pont en H piloté en PWM. On utilise un microcontrôleur pour le commander tel que celui d'une carte Arduino.

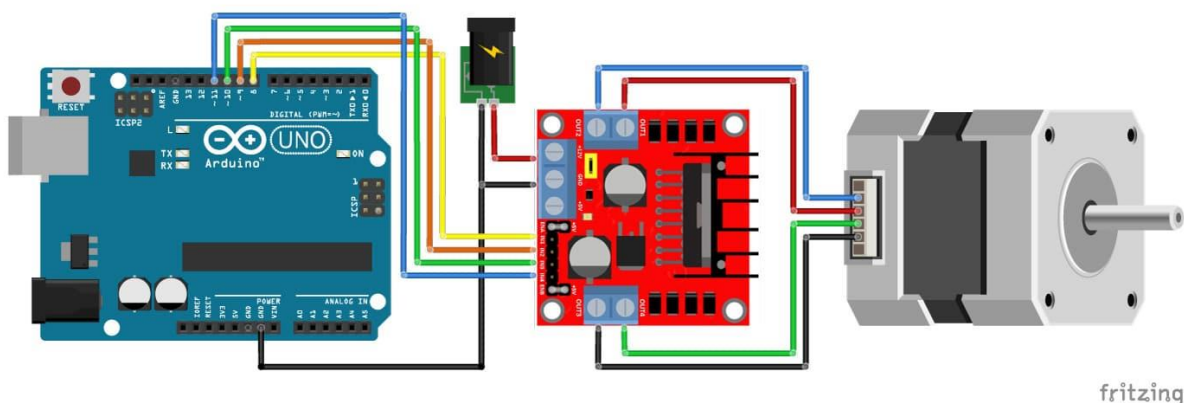


Figure 17: Câblage driver n°2

## POC : ROBOT PARALLÈLE À CÂBLES

Ils permettent d'ailleurs d'alimenter les moteurs proches de leurs tensions nominales et de débiter le courant nécessaire au bon fonctionnement même lors de la phase de démarrage.

Après de nombreux essais, les drivers ne présentent aucun échauffement durant le fonctionnement, même lors de cycles avec des démarrages fréquents. Il faut noter que lors des démarrages, un moteur appelle un courant de démarrage très fort (5 à 7 fois le courant nominal) et c'est à ce moment que le driver chauffe le plus. Donc, le nouveau driver est valide pour notre utilisation.

Il faut noter qu'à cause de notre alimentation stabilisée d'ordinateur, nous ne pouvons alimenter les drivers qu'en 12 V ou en 5V si nous voulons une puissance suffisante.

Par conséquent, nous avons choisi d'alimenter le pont en H du driver avec 5V car cette tension est proche de la tension nominale du moteur et donc permet de limiter l'échauffement du moteur et du driver.

### 6.2.3 Carte électronique

Initialement, nous voulions utiliser une carte Arduino Uno pour gérer l'ensemble du robot mais à cause du changement de driver nous avons dû passer sur une carte électronique de taille supérieure (en matière d'entrée/sortie) car l'Arduino Uno ne possédait pas assez de sorties (notamment en sortie PWM) pour gérer les quatre nouveaux drivers.

Nous avons finalement utilisé une carte Arduino Mega 2560 pour piloter notre robot.

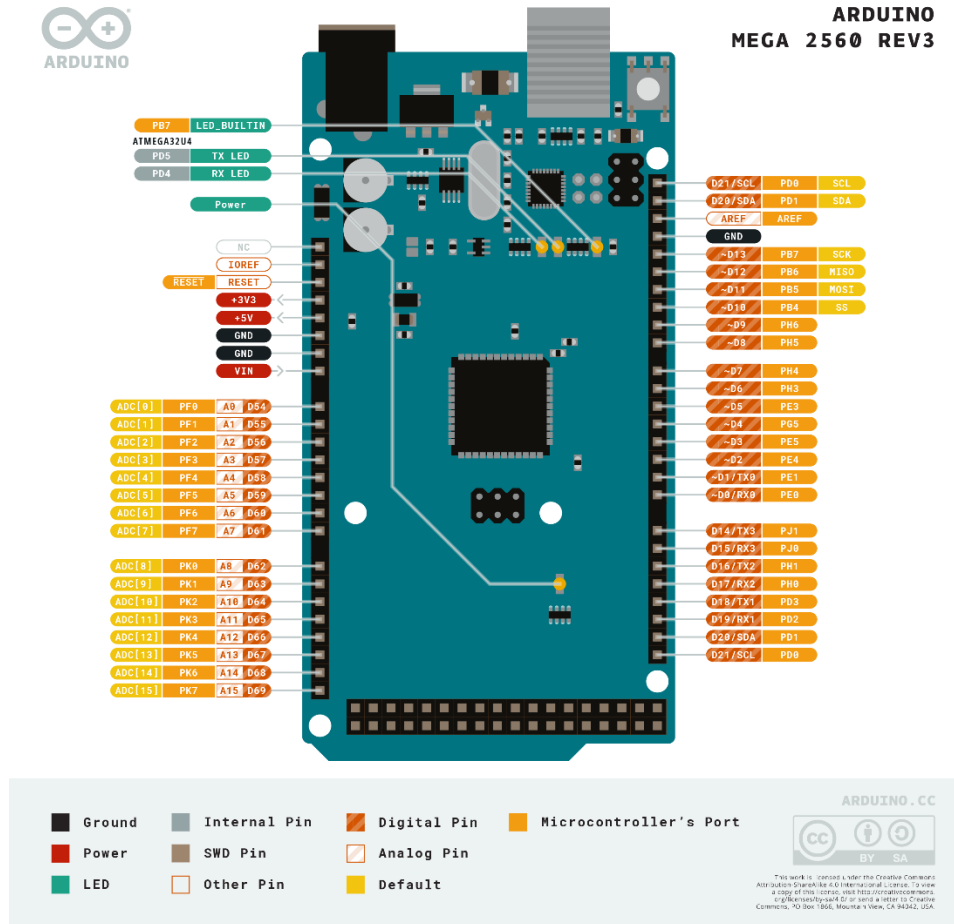


Figure 18: Description carte Arduino Mega

L'Arduino Mega 2560 est une carte microcontrôleur basée sur l'ATmega2560. Il dispose de 54 broches d'entrées / sorties numériques (dont 15 peuvent être utilisées comme sorties PWM), 16 entrées analogiques, 4 UART (ports série matériels), un oscillateur à cristal de 16 MHz, une connexion USB, une prise d'alimentation, un entête ICSP, et un bouton de réinitialisation. D'après ces spécifications, elle est parfaitement adaptée pour notre utilisation.

Le micrologiciel a été développé sous l'environnement de développement Arduino IDE.



Figure 19: Logiciel de développement Arduino

### 6.2.4 Electroaimant

Afin de permettre à l'effecteur du robot de soulever des pièces métalliques ferreuses, nous l'avons équipé de cet électroaimant :



*Figure 20: Électroaimant*

C'est un électroaimant de faible dimension permettant de soulever des charges jusqu'à un kilogramme, ce qui est suffisant pour notre application.

Il est piloté en direct par un microcontrôleur et ne nécessite pas d'alimentation supplémentaire.

Cependant, en pratique, il ne soulève pas réellement un kilogramme. Il soulève beaucoup moins mais cela suffira pour notre application. Une amélioration de cet élément serait intéressante par la suite.

### 6.2.5 Alimentation

Afin d'alimenter notre robot, nous avons besoin d'une alimentation de grosse puissance car les moteurs sont de forts consommateurs d'énergie.

Au vu des prix élevés des alimentations stabilisées dans le commerce, nous avons choisi de récupérer et d'utiliser une alimentation d'ordinateur fixe de 250 W.



*Figure 21: Alimentation*

Pour utiliser ce genre d'alimentation comme une alimentation stabilisée de laboratoire, il est nécessaire de réaliser quelques modifications de câblage afin de la faire démarrer. Une sécurité doit être supprimée pour réaliser le démarrage (la sécurité a été shunté). Après modification, cette alimentation nous assure une puissance nécessaire pour faire fonctionner le robot.

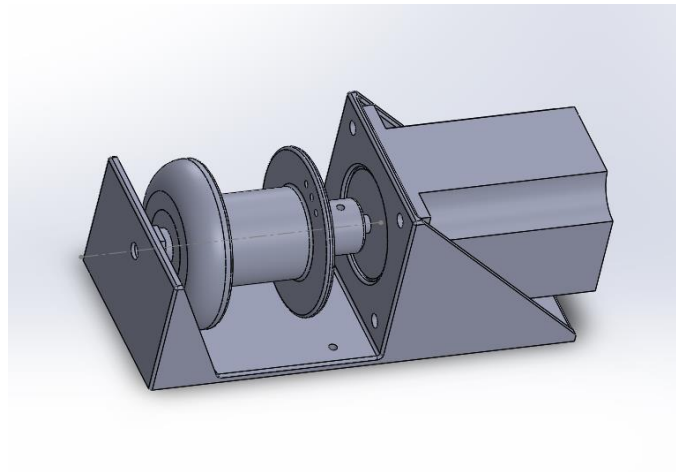


## 6.3 Mécanique

Différentes pièces mécaniques ont été développées dans le cadre du projet, elles ont toutes des fonctions précises et ont été réalisées en impression 3D (en PLA) soit sur l'imprimante 3D de HEI, soit sur nos imprimantes personnelles. Aussi, nous ferons ici la description de pièces mécaniques importantes utilisées dans notre projet et non réalisées en impression 3D.

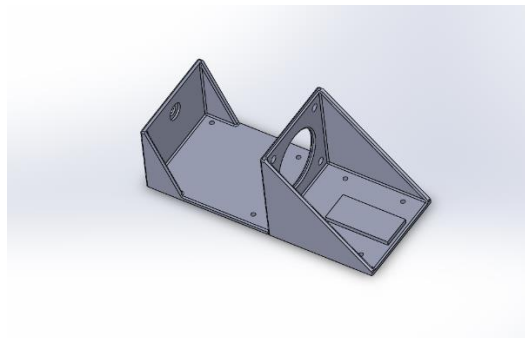
### 6.3.1 Treuil

Nous allons ici expliquer la conception mécanique des treuils. Ils sont au nombre de quatre et possèdent de nombreux éléments.



*Figure 22: Treuil*

- Le support

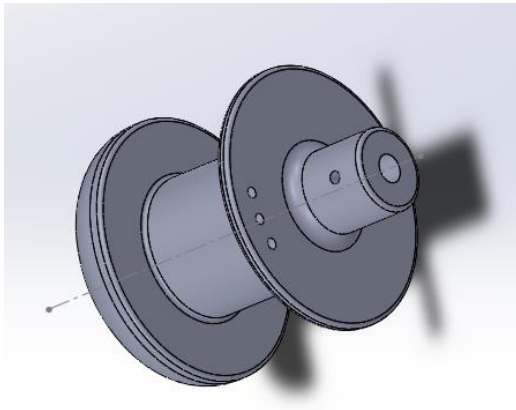


*Figure 23: Support*

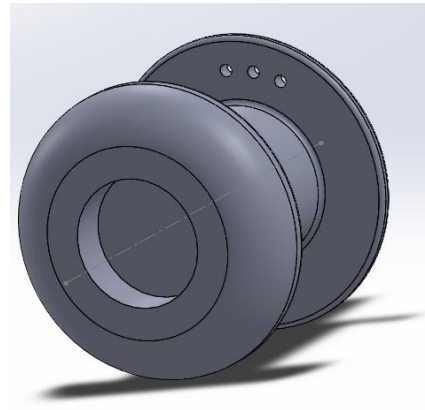
Il permet de supporter le moteur ainsi que la bobine et il est fixé au châssis du robot grâce à 8 vis M4.

Initialement, il était prévu pour être une pièce monolithique pour améliorer la rigidité en solidarissant l'ensemble mais nous n'avons pas trouvé d'imprimante 3D possédant une surface de travail assez conséquente et donc capable de réaliser ce travail donc nous avons scindé la pièce en deux parties.

- La bobine



*Figure 24: Bobine*



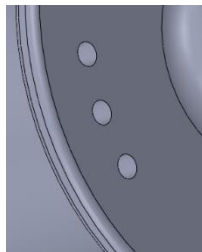
*Figure 25: Bobine vue de côté*

La bobine correspond à la pièce où s'enroule le câble, il y a de nombreuses choses à dire sur elle. Elle a été conçue pour qu'un tour de moteur enroule 10 cm de câble.

$$P = 2 \cdot \pi \cdot R \text{ donc } R = \frac{P}{2 \cdot \pi} = \frac{10}{2 \cdot \pi} \approx 1,6 \text{ cm}$$

Donc, le diamètre intérieur est de 1,6 cm afin d'enrouler 10 cm à chaque rotation de moteur.

On remarque trois perçages dans l'une des parois.

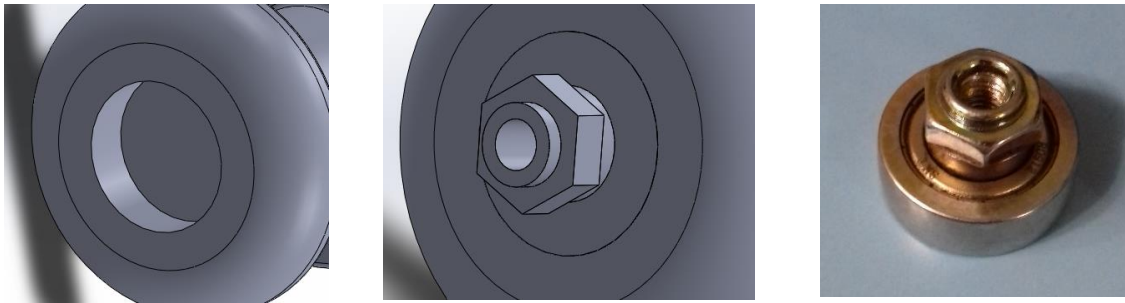


*Figure 26: Blocage câble*

Ils servent à bloquer le câble. Le câble se glisse en serpentín dans les trois perçages.

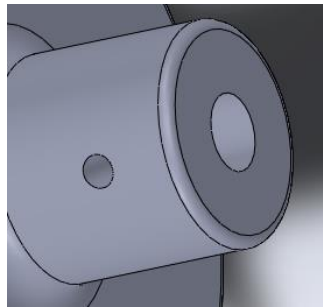
On remarque aussi un espace à l'arrière de la bobine. Celui-ci sert à insérer un roulement (de récupération) afin de faciliter la rotation du treuil et à gagner en précision.





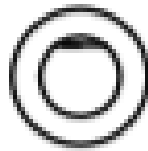
*Figure 27: Roulement*

Finalement, le dernier point à aborder porte sur la fixation du moteur.



*Figure 28: Fixation moteur*

Il permet de solidariser l'axe moteur avec la bobine. L'arbre moteur étant de forme D-Cut, la vis permet donc de bloquer l'axe.



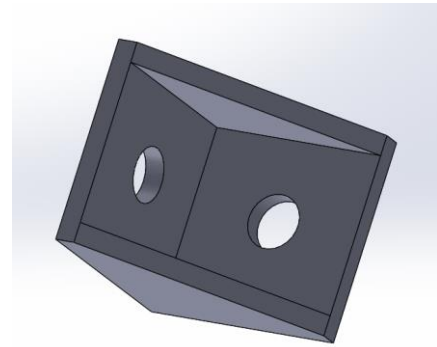
*Figure 29: Arbre moteur forme D-Cut*

Le perçage a été taraudé grâce à un taraud machine.

Un système anti-retour n'a pas été mis en place car le moteur du treuil présente un couple suffisant pour que la charge ne puisse pas dérouler le câble mis sur le tambour.

### 6.3.2 Équerre

Compte tenu du prix exorbitant des équerres de montage pour les profilés en aluminium, nous avons décidé d'en imprimer quelques-unes en PLA avec une imprimante 3D. Nous avons opté pour cette solution car le prix de revient est beaucoup moins important et leurs résistances seront suffisantes aux vues des faibles contraintes imposées à l'endroit de leur positionnement. Voici la CAO de l'une de ces équerres :



*Figure 30: Équerre*

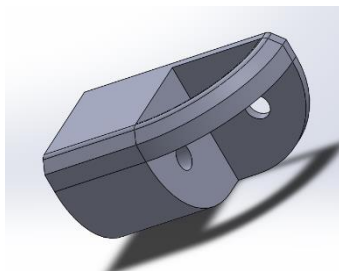
Ces équerres sont au nombre de huit et servent à soutenir le plan sur lequel travaille le robot.



*Figure 31: Équerres positionnées*

### 6.3.3 Passage de câble

Il s'agit d'un moyen simple et efficace que nous avons mis en place afin de réaliser le retour du fil au sein de l'espace de travail de notre robot. Voici l'un des quatre passeurs de fil mis en place sur le robot :



*Figure 32: Passage de câble 3D*



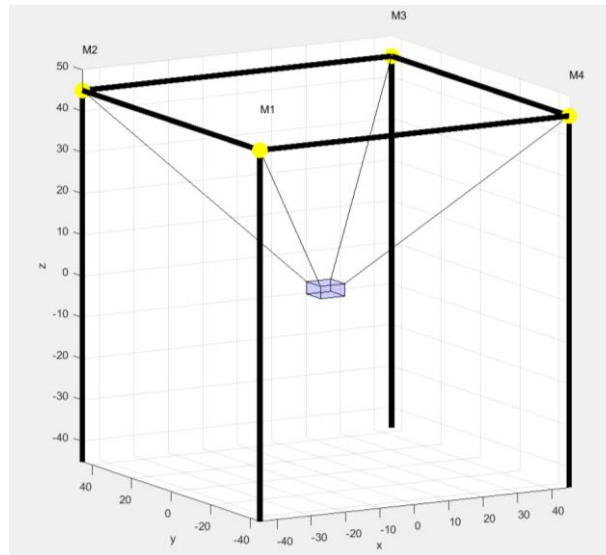
*Figure 33: Passage de câble réel*

Il est basé sur le principe d'un scion de canne à pêche dans lequel le fil en nylon est guidé par un simple anneau en carbone lisse. Ici, nous faisons passer le fil dans cet anneau en PLA et nous avons limité l'effet abrasif par polissage et par application d'une dose de produit lubrifiant.

## POC : ROBOT PARALLÈLE À CÂBLES

On remarque une certaine surépaisseur qui permet d'aligner le passeur de fil avec la bobine du treuil. Après différents essais en conditions réelles, nous avons dû réduire celle-ci, afin de s'accorder avec la simulation.

Sur la simulation ci-dessous, ces passages de fils sont représentés par les points jaunes. On remarque évidemment qu'ils sont en partie responsables de la taille de la zone de travail du robot



*Figure 34: Simulation RPAC*

Par la suite, dans l'optique d'améliorer la précision du système ainsi que son côté professionnel, nous avons décidé d'implémenter des poulies rotatives.



Ce montage est intégré dans le système final. Elles ont l'avantage d'être directionnelles et donc d'accompagner le mouvement de l'effecteur jusqu'à sa position désirée.

#### 6.3.4 Câble

Le câble utilisé dans notre projet est une bobine standard comme celles utilisées pour le parachutisme ou la survie. Elle possède un diamètre de 2,5 mm et est une simple tresse en nylon.

Cette tresse ne rentre dans sa zone élastique qu'à partir de 80 kg et en rupture à 200 kg, donc cette donnée nous assure que les câbles ne varieront pas en longueur, durant l'utilisation, à cause de la charge. Ils ne céderont pas non plus sous la charge.

Nous avons préféré utiliser une tresse en nylon plutôt qu'un filin en acier car le prix est bien moins cher, on évite les difficultés d'enroulement des câbles sur les treuils (ces câbles sont relativement rigides et difficiles à manipuler sans charge) et on évite une abrasion intense des pièces en PLA étant contact avec le câble.



*Figure 35: Bobine de câble*

Cependant, à cause du mauvais design du treuil, nous avons remarqué que l'enroulement n'était pas linéaire. En effet, à chaque tour de treuil, le câble est enroulé de 8 à 12 cm au lieu des 10 cm nominaux. Le problème vient du fait qu'aucun guide n'a été mis en place pour enrouler proprement le câble de manière connue et il s'enroule donc de façon aléatoire.

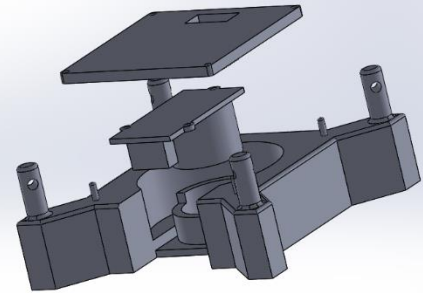
Cette différence empêche alors le robot d'atteindre avec précision la position voulue. Pour pallier cela, deux solutions sont apparues, changer la conception mécanique du treuil ou bien trouver un câble de diamètre inférieur pour limiter l'erreur. Nous avons choisi la seconde solution dans un souci de simplicité de mise en place.

Nous avons alors sélectionné un fil en nylon de type fil de pêche d'un diamètre 0.5 mm. Cela sera la solution finalement gardée pour le projet.

## 6.4 Développement de l'effecteur du robot

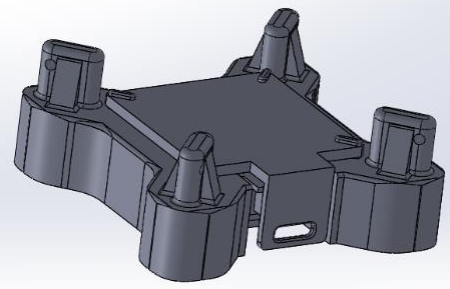
L'effecteur correspond à la partie du robot qui sera en mouvement dans l'espace de travail. Il a été modélisé puis imprimé en 3D.

L'image ci-contre correspond à l'assemblage du premier effecteur du robot. On remarque quatre points d'accroche pour les câbles et un emplacement pour l'électroaimant vu précédemment. L'effecteur a été modélisé avec des dimensions répondant à la simulation MATLAB afin de ne pas fausser le positionnement du robot.



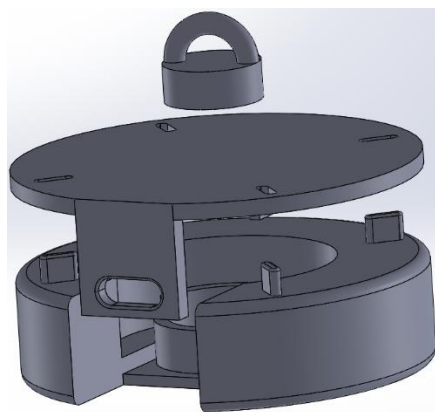
*Figure 36: Effecteur n°1*

Lors de nos tests, nous avons cassé l'effecteur, ce qui nous a permis de retravailler le design de celui-ci afin de le rendre plus robuste et donc avoir une meilleure tenue sur le long terme. Il s'agit du second effecteur du robot.



*Figure 37: Effecteur n°2*

A cause d'un problème d'équilibrage, nous n'arrivions pas à atteindre le parallélisme de l'effecteur avec le sol. Nous avons donc décidé de réaliser un rework de l'effecteur.



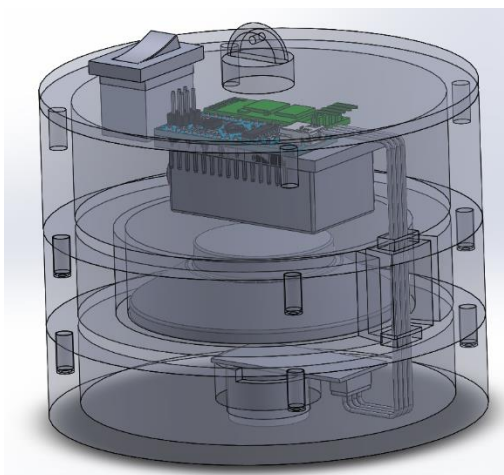
*Figure 38 : Effecteur n°3*

Ce nouvel effecteur a l'avantage d'avoir ses points de fixation au centre de sa structure. Il s'équilibre donc par gravité.

Cependant, même si ce nouvel effecteur est plus équilibré que les précédents, il reste trop léger pour tendre parfaitement les câbles et possède un autre gros défaut : un câblage filaire. Ce câblage filaire est très lourd et n'avait pas été pris en compte dans la modélisation. Il change donc totalement la dynamique du robot. Nous avons alors réalisé un effecteur final répondant à ces contraintes :

- Fixations des câbles en un seul point ;
- Centre de gravité assez bas pour obtenir une bonne stabilité ;
- Masse suffisante pour tendre correctement les câbles sans pouvoir dérouler le câble sur les treuils ou empêcher l'enroulement ;
- Bon équilibrage grâce au placement correct des composants ;
- Communication sans-fil ;
- Autonome en énergie ;
- Modulable pour modifier l'effecteur en cas de problème lors du développement.

Voici notre effecteur final qui a été testé et validé :



*Figure 39 : Effecteur final*

Il présente 3 étages et 1 couvercle. Sa constitution est rudimentaire mais l'effecteur est parfaitement fonctionnel.

Le premier étage contient l'électroaimant, le deuxième correspond à l'étage de lestage. L'étage de lestage contient actuellement un poids métallique de 300g et un compartiment vide qui pourra être rempli si nécessaire pour alourdir le système. Le dernier étage contient l'électronique avec :

- Une carte Arduino Nano ;
- Un module Bluetooth HC-05 ;
- Une batterie 9V ;
- Des LEDs d'états.

Le système est connecté comme esclave au robot via Bluetooth. Il reçoit les ordres et les effectue.



Après quelques essais, nous avons remarqué que cet effecteur répondait à toutes nos attentes.

## 6.5 Mise en place de l'électronique

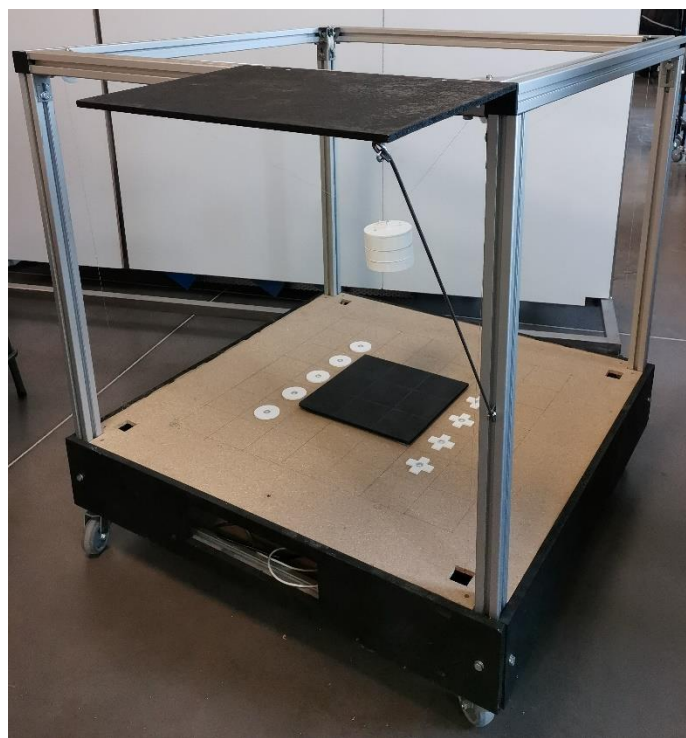
Le schéma de câblage mis en place pour notre robot est disponible en annexe.

Du côté robot, les quatre drivers de moteur ont été câblés à la carte Arduino Mega 2560. Les moteurs sont branchés sur les drivers et l'ensemble est alimenté en 5 V. La carte Arduino Mega 2560 est aussi relié au module Bluetooth maître.

Du côté effecteur, l'électroaimant est câblé à la carte Arduino Nano gérant ce bloc. La carte Arduino Nano est branchée au module Bluetooth esclave et à des LEDs d'état.

## 6.6 Assemblage

Nous avons assemblé le robot en suivant les plans fournis par la CAO et nous obtenons donc ce résultat final :



*Figure 40: Assemblage final*

## 6.7 Modélisation du robot

D'après nos recherches, nous avons reconnu que le robot est assez complexe à modéliser avec une méthode classique de robotique tel que la convention de Denavit-Hartenberg modifiée. Nous avons donc décidé d'utiliser une approche plus simple basée sur des calculs automatiques de distance dans l'espace (distance euclidienne). Cependant, cette méthode ne nous permet d'obtenir directement que le modèle géométrique et non le cinématique ni le dynamique. Le modèle géométrique permet seulement de calculer les longueurs des câbles et les nombres de rotations des moteurs.

Nous avons posé ce schéma pour l'un des quatre câbles :

Pour calculer la longueur d'un câble pour atteindre une position voulue, il suffit seulement de faire la différence entre les coordonnées du passage de câble et les coordonnées du point de fixation de l'effecteur. On considère le câble comme rigide et indéformable. Par exemple :

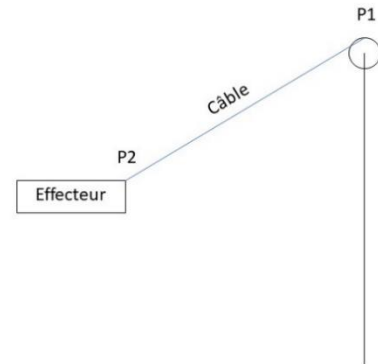


Figure 41: Représentation

$$L_{c1} = \sqrt{(x_{p1} - x_{p2})^2 + (y_{p1} - y_{p2})^2 + (z_{p1} - z_{p2})^2}$$

Il suffit d'utiliser ce principe simple sur les quatre câbles et nous pouvons obtenir les quatre longueurs de câble pour une position voulue.

Ensuite, pour déterminer la différence de longueur d'un câble d'une position à une autre, il suffit de faire la différence entre la position finale et la position initiale :

$$\Delta_{c1} = L_{c1_f} - L_{c1_i}$$

Finalement, pour déterminer le nombre de rotation moteur à effectuer, il faut prendre en compte le design du treuil. Comme vu précédemment dans ce rapport, un tour de bobinage de treuil correspond à enrouler 10 cm de câble. Donc trivialement, on obtient cette formule :

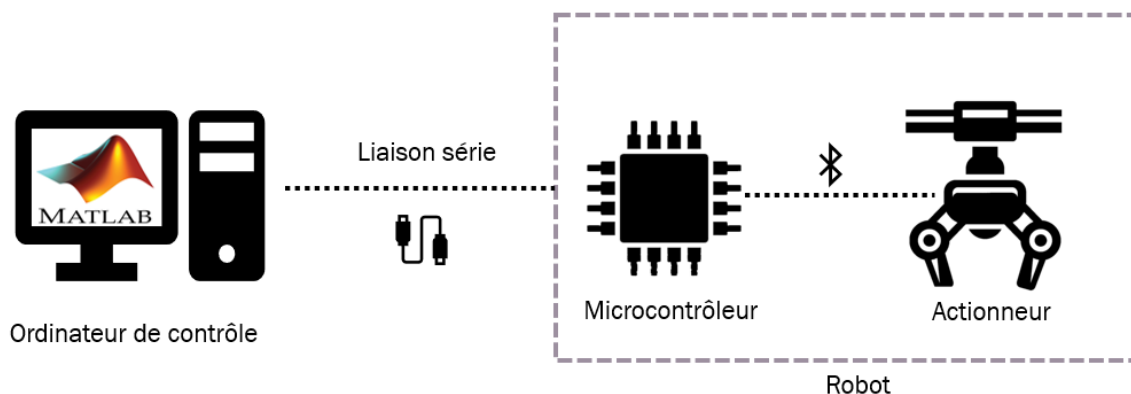
$$R_{m1} = \frac{\Delta_{c1}}{10}$$

L'application de ces quelques formules permet de commander le robot. Il suffira juste d'envoyer des trames de commande au robot avec les bonnes informations pour faire déplacer correctement l'organe terminal.



## 6.8 Architecture logicielle

Afin de commander le robot parallèle à câbles, nous avons mis en place une certaine architecture logicielle regroupant des éléments au sein d'une chaîne. Voici comment se présente cette chaîne dans sa globalité :

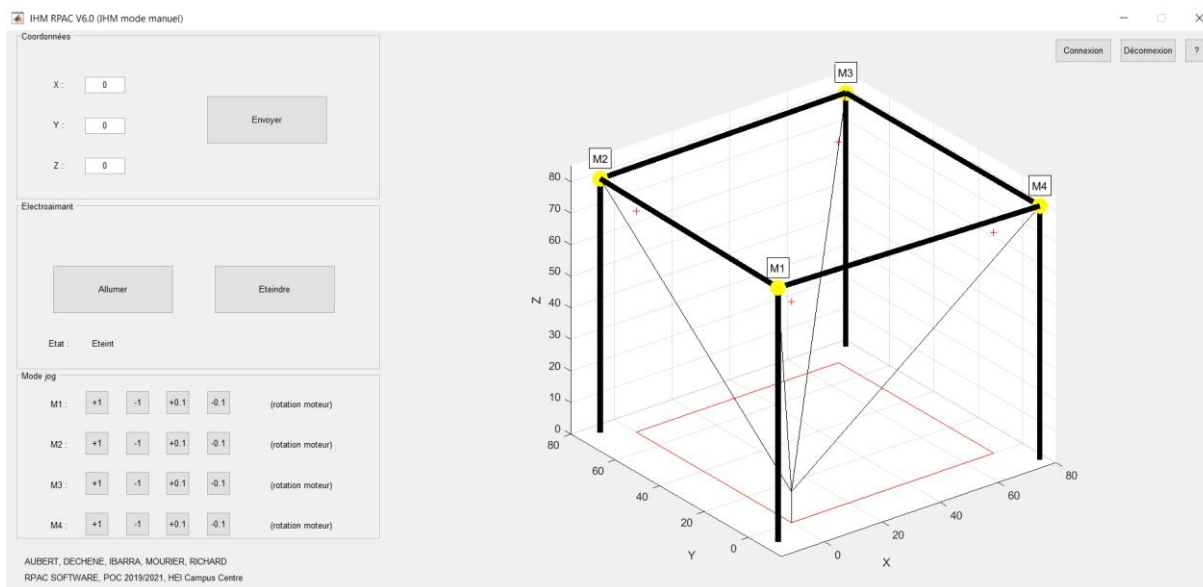


*Figure 42: Architecture logicielle*

Nous allons maintenant détailler les éléments de cette chaîne un par un.

### 6.8.1 IHM (mode manuel)

Une IHM a été réalisée sous MATLAB afin de réaliser le contrôle du robot en mode manuel :



*Figure 43: IHM mode manuel*

Cette interface possède plusieurs fonctionnalités. Elle permet de contrôler le robot en mode manuel.

Le premier panneau permet de sélectionner la consigne de position du robot.

*Figure 44: Consignes de position*

Il faut rentrer les coordonnées en X, Y et Z et appuyer sur le bouton « Démarrer », l'interface envoie alors une trame au robot et il se place ensuite à la position voulue dans l'espace. En parallèle, une animation 3D, représentant le déplacement réel du robot, se réalise.

Le deuxième panneau permet de sélectionner l'état de l'électroaimant : allumé ou éteint.

*Figure 45: Contrôle de l'électroaimant*

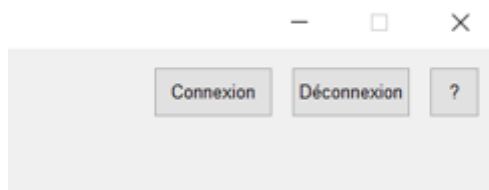
Il suffit d'appuyer sur l'un des deux boutons et l'interface enverra une trame au robot permettant de piloter l'électroaimant.

Le dernier panneau permet de contrôler manuellement les treuils du robot (mode « jog »). Cette partie est utile pour la remise à zéro. Elle a aussi été utile lors du développement.

*Figure 46: Contrôle des treuils*

Un simple appui sur un bouton et une trame sera envoyée au robot. Le robot réalisera l'action demandée.

Une gestion de la connexion au robot a aussi été incorporée :



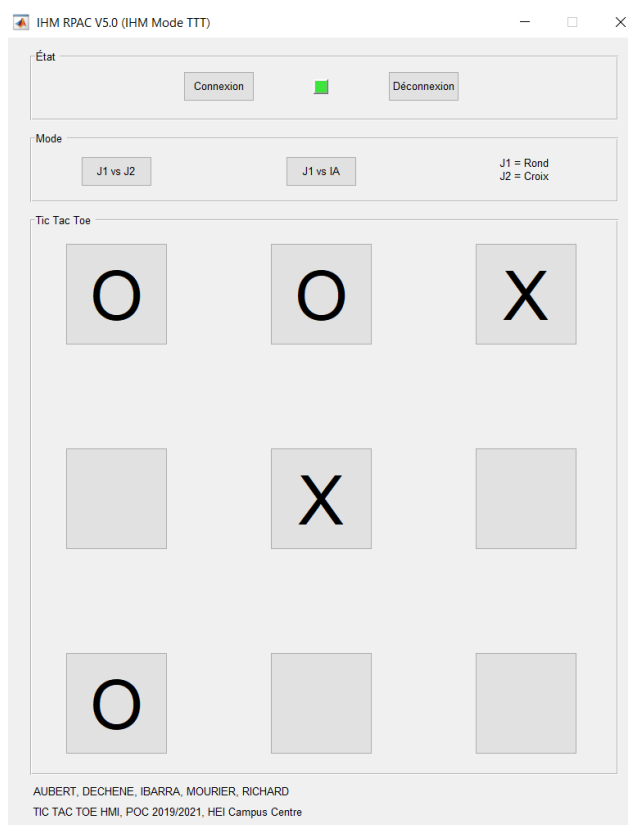
*Figure 47 : Contrôle de connexion au mode manuel*

De plus, elle nous a permis de calculer les trames de données à implémenter dans le second logiciel, celui gérant l'application ludique.

## 6.8.2 IHM (application ludique)

La seconde partie du projet consistait à réaliser une application ludique afin de montrer les performances de notre robot. Conformément à notre cahier des charges, nous avons réalisé un Tic-Tac Toe.

Une interface de commande a été réalisée :

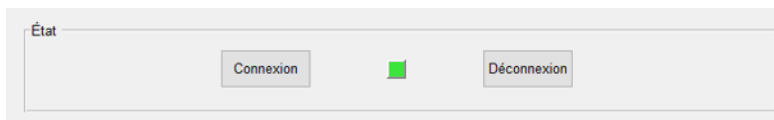


*Figure 48 : Interface du plateau de Tic-Tac-Toe*

Cette interface présente deux modes, l'un correspond à un mode joueur contre joueur et l'autre à un joueur contre IA.

L'interface présente différents boutons et fonctionnalités.

Le premier panneau permet la connexion ou la déconnexion du robot.



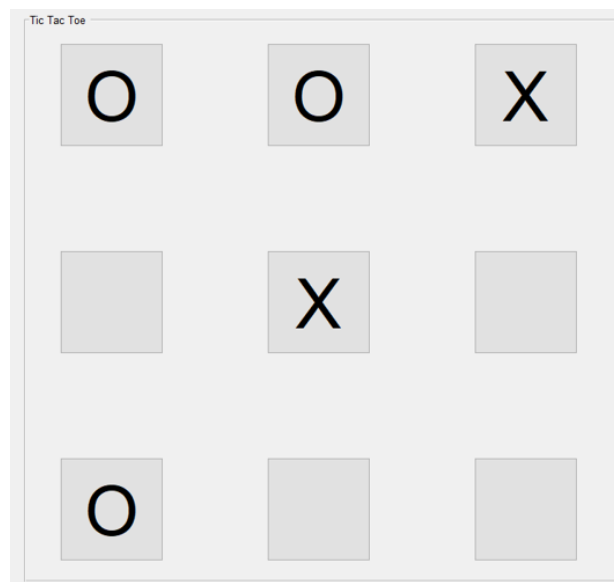
*Figure 49 : Contrôle de connexion au mode de jeu*

Le deuxième panneau permet le choix du mode de jeu et affiche différentes informations importantes par rapport au jeu (mode, tour d'un joueur, gagnant, perdant...).



*Figure 50 : Choix du mode de jeu*

Le dernier correspond à l'état actuel du plateau :



*Figure 51 : Plateau de jeu*

De plus, des déplacements (sous forme de trames) ont été mémorisés afin de réaliser les déplacements nécessaires au bon déroulement du jeu.

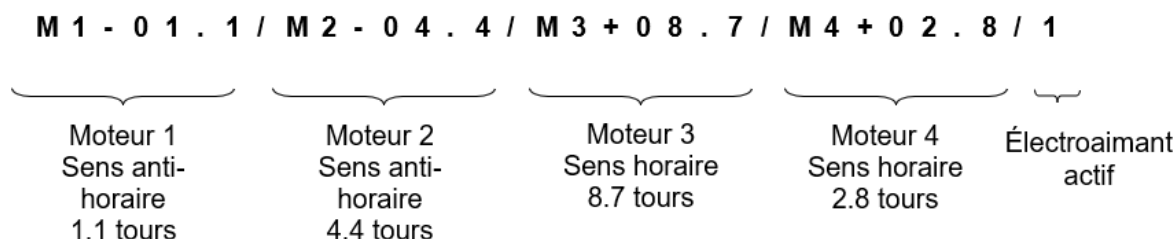
## 6.8.3 Trames

Pour transférer les informations du logiciel vers le robot, nous avons utilisé des trames de communication. Celles-ci contiennent les identifiants pour chaque moteur, le sens de rotation, le nombre de tours à effectuer, ainsi que la commande pour le changement d'état de l'électroaimant.

Sous MATLAB, les trames sont codées grâce à un script. En fonction du moteur et du nombre de rotations associé pour atteindre une position, le logiciel peut composer la trame. Le logiciel renvoie aussi l'état actuel de l'électroaimant en fin de trame.

D'un autre côté, pour la gestion de l'électroaimant, le même protocole de communication des trames est employé. Cependant, cette fois nous ne donnons pas de tours à effectuer aux moteurs, il y a seulement l'état de l'électroaimant qui change.

Il a fallu normaliser les trames afin de faciliter le codage et le décodage. Elles contiennent 33 caractères et prennent la forme de cet exemple.



*Figure 52 : Cheminement des trames envoyées*

#### 6.8.4 Gestion interne du robot

Une fois la trame reçue dans la carte Arduino gérant le robot, celui-ci calcule des vitesses différentes pour chaque moteur selon la distance à parcourir par chacun. Cette distance est mesurée en pas. Nos moteurs ont besoin de réaliser 200 pas pour pouvoir effectuer un tour complet de 360°.

Partant du fait que la trame reçue contient les tours à réaliser par chaque moteur, nous réalisons un simple calcul pour obtenir le nombre de pas nécessaires pour effectuer la consigne :

$$Nb_{pas} = 200 \cdot Nb_{tour}$$

Donc, si nous avons une consigne de 8,5 tours, le moteur réalisera 1700 pas.

Une particularité de notre programme a été la difficulté rencontrée lors du lancement des rotations des moteurs. Celle-ci doit être effectuée en même temps, les arrêts doivent aussi fonctionner comme tel. Pour répondre à cette problématique, nous devons déterminer quel moteur parmi les quatre prenait le plus de temps pour parcourir la distance souhaitée.

Pour ce faire, nous employons la formule pour le calcul de la vitesse en isolant la variable temps et en employant la vitesse maximale déclarée dans le code (pour nos moteurs, nous utilisons une vitesse maximale de 200 pas/sec).

$$t_{max} = \frac{d_{moteur\_X}}{v_{max}}$$

Une fois le moteur le plus lent trouvé, on recalcule une nouvelle vitesse pour chaque moteur en prenant ce temps maximal comme référence :

$$v_{moteur\_X} = \frac{d_{moteur\_X}}{t_{max}}$$

Où,

$v_{moteur\_X}$  : Vitesse à atteindre pour chaque moteur (Moteur 1, Moteur 2, Moteur 3 ou Moteur 4)

$d_{moteur\_X}$  : Distance à réaliser par chaque moteur

$t_{max}$  : Temps maximum nécessaire pour qu'un moteur réalise son parcours (parmi les 4).

La nouvelle vitesse, calculée grâce aux nombres de tours de tous les moteurs, permet à ceux-ci de les arrêter en même temps.

Grâce aux drivers, nous arrivons à réaliser la commande des quatre moteurs avec une seule carte Arduino et donc un déplacement propre de l'effecteur du robot.

### 6.8.5 Boucle fermée et remise à zéro

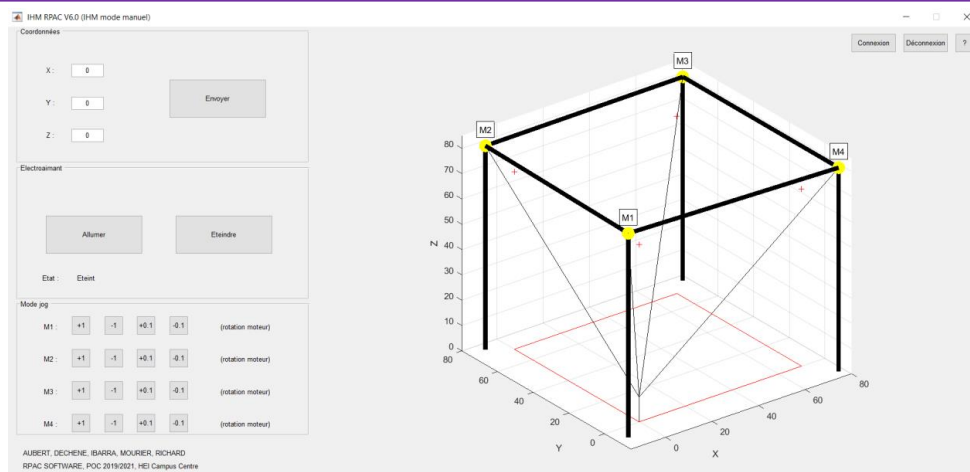
La remise à zéro automatique lors de l'initialisation et la correction du phénomène de perte de pas sont des problèmes épineux que nous avons eu du mal à résoudre.

#### 6.8.5.1 Remise à zéro

Le premier problème consistait à la remise à zéro du robot. L'ensemble des robots préhenseurs ou des lignes de production présentent des cycles d'initialisation (remise à zéro), notre robot ne déroge pas à cette règle. Le robot doit être capable de se remettre en position initiale avant d'effectuer une tâche.

N'ayant pas pu réaliser un mode automatique de remise en position initiale, nous avons mis en place un protocole de remise à zéro dit « manuel ».

Grâce à l'interface ci-dessous permettant de gérer l'ensemble des fonctionnalités du robot manuellement, nous pouvons emmener le robot à sa position initiale avec des déplacements successifs sur les moteurs, arrimer l'effecteur à sa position initiale (support métallique agrippé par l'électroaimant) et retendre manuellement les câbles avec l'IHM.



*Figure 53: IHM de gestion du robot*

Une fois cette opération réalisée, le robot est à sa position initiale et nous pouvons le faire travailler.

## 6.8.5.2 Boucle fermée

Lors du développement, un problème important a été décelé. En cas de perte de pas (phénomène touchant les moteurs pas à pas en cas de couple résistant trop important ou de vitesse trop élevée), le résultat peut être dramatique. Si nous perdons un pas, sachant que nos moteurs possèdent 200 pas par tours, nous perdons  $1,8^\circ$ . On en déduit donc un décalage linéaire de 0,05 cm en sachant qu'un tour complet enroule 10 cm de câble. Un pas serait négligeable pour notre application mais si cela arrive trop souvent, nous pourrions retrouver un robot présentant une tension trop importante ou pas assez importante dans les câbles engendrant ainsi des décalages ou des casses.

Ce problème vient du fait que notre robot travaille avec des moteurs en boucle ouverte et non en mode asservi.

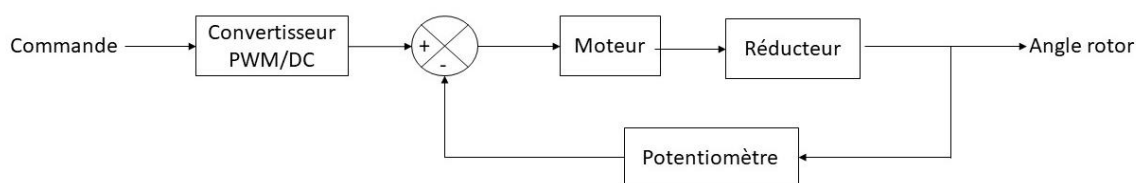
Malheureusement, ce problème a été trouvé trop tard lors du développement et nous n'avons pas pu le corriger. La solution actuelle consiste à limiter la vitesse et la charge du robot pour éviter les décalages.

Cependant, nous avons tout de même réfléchi à une solution qui pourra être potentiellement mise en place par la suite. Afin de résoudre les problèmes de perte de pas, nous devrions mettre en place un capteur par moteur ainsi qu'un asservissement. Ce capteur doit être un capteur de position. Nous avons posé l'idée d'utiliser un potentiomètre à la manière de ceux utilisés dans un servomoteur classique de modélisme.



*Figure 54 : Servomoteur*

Voici le schéma bloc de l'asservissement typique de ce genre de servomoteur :



*Figure 55 : Schéma bloc du système pouvant être mis en place*

Un contrôleur (PI, PID, etc.) peut être ajouté dans la branche principale.

Les capteurs de positions les plus simples sont probablement les potentiomètres. Ce type de capteur est simplement formé d'un curseur mobile qui se déplace le long d'un matériau résistif et reflète ainsi une distance ou un angle. Les potentiomètres sont des composants électroniques passifs, il faut donc les alimenter avec une source externe pour pouvoir effectuer la mesure.

La variation entre la position du curseur (rotation ou translation) et la résistance entre une des extrémités du composant et le curseur mobile est à peu près linéaire donc nous pourrions facilement obtenir la position angulaire.

Les plages de mesures angulaires des potentiomètres sont en générales faibles (typiquement un demi à trois quarts de tour) à l'exception des potentiomètres dits multi-tours où le curseur mobile se déplace sur une surface résistive en forme d'hélice. Les potentiomètres sont des capteurs de position simples et peu chers mais la linéarité de la mesure est souvent leur point faible et mieux vaut les calibrer soigneusement.

Notons que les potentiomètres ne sont généralement pas utilisés en régime dynamique, c'est-à-dire lorsque le curseur mobile se déplace rapidement. En effet, le curseur frotte généralement de façon notable sur la surface résistive et cette charge, ajoutée au système dont on veut connaître la position, en modifie notablement le comportement. De plus, des mouvements rapides vont finir par user significativement la résistance sur laquelle le curseur mobile se déplace et cela va perturber la précision de la mesure. Cependant, vu notre faible vitesse de déplacement, nous pouvons nous permettre d'utiliser ce système.

Nous voudrions utiliser quatre potentiomètres multi-tours de 15 tours. Il faut savoir que nous choisirions cela car avoir quinze rotations de moteur est le pire cas en termes de nombre de rotations. Il faudra aussi réaliser une interface mécanique sur-mesure pour coupler le potentiomètre à l'arrière de nos moteurs.

Electroniquement, il faudrait utiliser quatre entrées analogiques sur la carte Arduino Mega pour récupérer les données de position des moteurs. Le signal serait conditionné (pont de mesure et filtrage) avant de passer dans l'ADC de la carte.

Par conséquent :



- Le potentiomètre permettrait de convertir la position du rotor en tension analogique ce qui permet de réaliser l'asservissement (comparaison de la position avec la consigne) ;
- Il s'agit d'un système peu cher (environ 5€) et facile à mettre en place par rapport aux codeurs optiques ;
- La précision est certes grossière mais le système reste efficace et mieux que rien (pas de décalage de plus de quelques degrés).

## 6.9 Performance du produit

### 6.9.1 Précision

Avec notre gestion des moteurs actuelle et l'informatique en lien avec eux, nous sommes capables d'assurer une précision à moins de cinq centimètres près. Cette précision devra évidemment être améliorée pour réaliser des applications plus précises.

### 6.9.2 Robustesse

Notre système n'est malheureusement pas des plus robustes. Il travaille en boucle ouverte au niveau de la motorisation et le risque de perte de pas est toujours présent. Cependant, nous avons tout fait pour ne pas pousser le robot dans ses limites (demander une vitesse de l'effecteur trop rapide ou lui demander trop de couple) et donc limiter au possible les erreurs de positions dues à ce phénomène.

### 6.9.3 Vitesse

Par notre modélisation du robot, nous ne possédons actuellement que le modèle géométrique du robot. Nous ne pouvons donc pas apporter d'information sur ses performances en vitesse, tout comme pour son accélération ou son jerk.

## 7 Coût final du démonstrateur

Finalement, nous pouvons poser ce tableau pour résumer les coûts du robot :

Elément	Prix	Quantité	Total
<b>Moteur</b>	32,52 €	4	130,08 €
<b>Câble</b>	6 €	1	6 €
<b>Electroaimant</b>	9,83 €	1	9,83 €
<b>Câblage</b>	~5 €	-	5 €
<b>PLA</b>	20 €	1	20 €
<b>Planche de medium</b>	Récupération	-	-
<b>Profilé aluminium</b>	Récupération	-	-
<b>Poulies</b>	2,5 €	4	10 €
<b>Spiro-bande</b>	10 €	1	10 €
<b>Quincaillerie</b>	~40 €	-	40 €
<b>Arduino Mega</b>	40 €	1	40 €
<b>Drivers</b>	2 €	4	8 €
<b>Alimentation</b>	Récupération	-	-
<b>Peinture</b>	7 €	2	14 €

Le coût final est donc de **292 €**.

Ce coût résume le prix du matériel installé sur le démonstrateur et non le prix du matériel utilisé lors du développement.

Si nous comptons le prix total du développement, nous arriverions dans les **450 €** pour environ **750 h** de travail effectifs au total.

## 8 Perspectives d'évolution

### 8.1 Applications

Compte tenu de notre design, il est compliqué de changer d'application.

L'effecteur n'étant monté qu'avec des câbles supérieurs, donc ne pouvant pas se déplacer avec une grande vitesse car les déplacements du haut vers le bas ne se réalisent que grâce à la pesanteur terrestre, il faudrait le monter avec quatre câbles inférieurs pour atteindre des hautes vitesses de déplacement.

Nous ne pouvons pas non plus réaliser de déplacements angulaires importants du porteur avec aussi peu de câble car le nombre de degrés de liberté de ce type de robot est lié au nombre de câbles et à leurs emplacements d'accroche sur l'effecteur.

Sans modifier le design, nous ne pouvons donc que réaliser des déplacements parallèles au sol avec une vitesse limitée dans les phases descendantes du robot.

Il est cependant toujours possible de changer le type de préhenseur en remplaçant par exemple l'électroaimant par une pince ou une ventouse afin de supporter d'autres types d'objets.

## 8.2 Modifications

Au fur et à mesure du projet, nous sommes tombés sur de nombreux imprévus que nous avons dû résoudre comme nous le pouvions. D'un autre côté, le budget ainsi que le temps ont été plutôt limités et cela a freiné notre développement. Durant ce projet, nous avons alors noté et appris de nos erreurs.

A notre sens, voici quelques améliorations à appliquer si le projet devait être repris :

- Changer le design des treuils (guide pour le câble, mise en place d'un capteur de position pour l'asservissement) ;
- Mettre en place l'asservissement en position des moteurs (robustesse) ;
- Augmenter la précision du robot (modifications logicielles principalement) ;
- Réalisation de la modélisation et du contrôle cinématique voir dynamique ;
- Soigner la réalisation avec des matériaux de meilleure qualité ;
- Mettre en place des protections de type grille ou enceinte en plexiglas pour protéger le robot ;
- Changer l'effecteur (l'électroaimant n'est pas de très bonne qualité et un travail d'intégration serait à refaire pour optimiser l'effecteur). En effet, un effecteur plus optimisé, plus intégré et miniaturisé serait intéressant (avec une philosophie plus mécatronique que l'effecteur actuel).

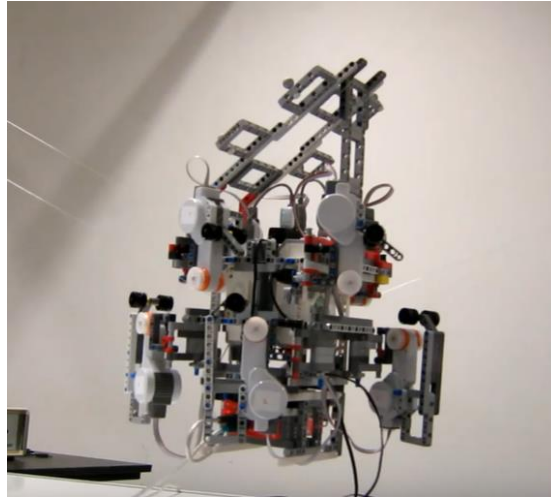
## 8.3 Potentiel changement de concept

Au fur et à mesure du projet, nous avons continué la phase de veille technologique.

Au cours de nos recherches, nous avons trouvé d'autres concepts de robots parallèles à câbles présentant une implémentation de composants différente de notre projet.

Le plus intéressant à notre sens est le concept utilisé par un laboratoire de recherche de l'école d'ingénieurs Télécom Physique Strasbourg où l'un des enseignants chercheurs est spécialisé dans le développement de robots parallèles à câbles.

Leur méthode consiste à concentrer l'ensemble des composants du robot (cartes électroniques, actionneurs, capteurs, etc.) au sein de l'effecteur du robot. L'effecteur du robot se présente alors comme un produit complet, un bloc mécatronique compact et miniaturisé au possible. Ces recherches ont mis au point plusieurs robots comme celui ci-dessous par exemple :



*Figure 56: Autre architecture de robot à câbles*

Leurs robots sont asservis et commandés grâce à des modèles complexes sur MATLAB-Simulink et sont conçus pour se stabiliser très rapidement lors d'un choc grâce à différentes méthodes (en couplant des unités de mesures inertielles, des asservissements visuels et différentes technologies d'actionneurs).

On peut remarquer que ces robots sont bien moins contraignants à mettre en place que le nôtre car il lui suffit seulement de quatre points d'ancrage pour être opérationnel, contrairement au nôtre qui nécessite une structure extérieure plus complexe.

Seulement, vu la masse suspendue par les câbles, ce robot ne peut pas atteindre une grande vitesse.

Donc, nous pouvons dire que passer sur ce type de concept pourrait être une amélioration par la suite si nous voulons rendre le système plus compact.

## 9 Difficultés

### 9.1 Anticipées

Nous pensions que la réalisation de la modélisation du robot serait la phase la plus complexe de ce projet mais finalement cela ne fut pas réellement un problème grâce à la simplification que nous avons mis en place sur MATLAB et grâce au temps important alloué sur cette tâche au début du projet.

### 9.2 Rencontrées

Lors de la réalisation de ce projet, nous avons rencontré de très nombreux problèmes. Ils ont pu être résolus mais non sans difficultés.

Les premières difficultés ont été les problèmes d'alimentation des moteurs. Des erreurs d'électronique ont fait que nos drivers et nos moteurs n'étaient pas compatibles, ce qui nous a obligé à changer de driver et de carte de gestion. Ce

## POC : ROBOT PARALLÈLE À CÂBLES

---

problème fut inquiétant et une perte de temps mais ne fut pas dramatique pour la réalisation du projet.

Une deuxième difficulté a été l'obtention du matériel liée à la structure du robot. Certains éléments comme les profilés en aluminium et certaines vis ont été complexes à obtenir.

Nous pouvons aussi citer les temps d'acheminement des composants électroniques qui ont été assez long (plusieurs mois), ce qui a causé du retard sur le développement. Dans le même genre, les temps d'impression 3D ont été plus long que prévu, ce qui a repoussé de quelques semaines le montage de la structure du robot.

Le dernier problème, qui est le plus important, a été de mettre en place un système permettant de réaliser la remise à zéro du robot.

D'autres problèmes annexes ont aussi été rencontrés :

- Difficulté pour recaler la simulation avec le robot physique ;
- Casse de l'effecteur pendant un essai qui a obligé la conception et la réalisation d'une nouvelle pièce ;
- Problème d'enroulement non linéaire ;
- Obligation de remplacer les passages de fils par des poulies rotatives ;
- Conception itérative de l'effecteur du robot.

Finalement, le plus complexe a été réellement de régler le fonctionnement du robot.

## 10 Conclusion

### 10.1 Générale

Pour conclure, ce **Proof Of Concept (POC)** qui portait sur une idée technologique d'un système fonctionnel, pour notre part un robot parallèle à câbles fut un réel succès. Le prototypage de notre solution fut convainquant. Nous avons pu répondre à la problématique de départ qui était "Est-il possible de réaliser un démonstrateur de robot parallèle à câbles pour une application ludique ?"

En effet, nous avons réalisé une structure permettant l'implémentation d'un jeu du Tic-Tac Toe ainsi que d'un mode manuel.

Ce projet fut très enrichissant et nous a apporté énormément au niveau connaissances. Cependant, il fut aussi très complexe à développer et nous avons dû revenir sur nos pas de nombreuses fois, le processus fut très itératif.

De plus, par la crise sanitaire de la COVID-19, le projet fut reporté 2 fois et donc ralenti. Cela a augmenté la difficulté du projet.

Pour finir, malgré les difficultés, nous souhaitons rester sur une note positive car nous avons appris énormément et nous avons réussi à réaliser un prototype fonctionnel. Mais aussi, nous avons réussi à réaliser un robot fonctionnel, novateur et très complexe avec un budget très limité, peu de personnel dessus et en peu de temps, malgré les difficultés de la situation.

### 10.2 Personnelles

#### AUBERT Guillaume

Etant passionné de robotique et de mécatronique, j'ai été tout de suite attiré par la réalisation d'un tel projet. Ce projet a été enrichissant d'un point de vue technique car il s'agit de la première fois pour moi que je réalise un robot de A à Z, et surtout un robot aussi complexe. Ce projet m'a permis de mettre en pratique les nombreuses connaissances que j'ai acquises à HEI Campus Centre et en entreprise mais aussi auparavant. J'ai pu mettre à profit de ce projet mes compétences d'électronicien et de développeur logiciel.

Seul bémol, la situation sanitaire a rendu ce projet très frustrant et fatigant à cause des nombreux reports.

#### DECHENE Bastien

N'ayant jamais eu de responsabilité de chef de projet auparavant, j'ai été attiré par ce poste au sein du groupe. Le sujet du projet m'a beaucoup attiré puisqu'il touche des domaines que j'affectionne particulièrement tels que la robotique, le dimensionnement 3D et la programmation. Lors de la réalisation de ce POC, j'ai dû déléguer les travaux

à réaliser aux membres de mon groupe et je me suis attribué la plupart de la gestion de la modélisation 3D du système. Cela a permis de dimensionner les éléments du système comme les vis ou encore les profilés utilisés. Nous avons ensuite pu commander ces composants pour construire l'ossature du robot. J'ai notamment été confronté à des problèmes de communication au sein du groupe, des problèmes de gestion que j'ai réussi à gérer grâce aux expériences passées de mes collègues. Originellement, j'ai mis en place un planning prévisionnel, celui-ci a été totalement chamboulé par la crise sanitaire COVID-19. Cette gestion de crise est passée par une période d'adaptation de plusieurs mois, rendant la réalisation du POC impossible. Finalement, nous nous sommes entr aidés sur nos parties et dans les dernières semaines, nous nous sommes intéressés à la précision du système ainsi qu'au fonctionnement des programmes.

### **IBARRA Mariela**

Tout au long du projet, nous avons pu partager nos connaissances et nos compétences ce qui m'a permis d'améliorer les compétences que j'avais acquises et en même temps d'en apprendre d'autres. J'ai pu retrouver la même sensation qu'un projet en entreprise avec ce projet. Pour moi, l'essentiel du projet était de pouvoir être autonome avec mon équipe afin que chacun puisse avoir un rôle et une importance pour qu'il puisse se sentir investi tout au long de ces deux années. Je remercie toute l'équipe ainsi que le chef de projet pour son investissement. Je peux dire que j'ai apprécié cette équipe de travail car il y avait une bonne ambiance que ce soit dans pendant ou hors des heures du travail.

### **MOURIER Antoine**

Tout au long du projet, diverses problématiques sont survenues et m'ont permis de mener à bien mes réflexions en termes de conception mécanique et d'en apprendre d'avantages sur les robots parallèles. J'ai apprécié travailler sur ce projet dont je n'avais aucune connaissance.

En revanche, je pense que s'il y a une chose à optimiser concernant le robot parallèle à câbles, c'est l'aspect visuel de notre prototype, le rendre plus attractif et agréable à voir.

### **RICHARD Gauthier**

J'ai beaucoup apprécié travailler sur ce projet car il m'a permis de progresser dans tous les domaines que ce soient les améliorations de compétences techniques, humaines ou encore transversales. J'ai eu l'occasion de travailler sur tous les aspects du projet que ce soit du soutien en électronique, au niveau de la mécanique, de la programmation ou de la CAO. Je pense que tout ce que j'ai acquis me servira durant la dernière période d'entreprise mais aussi pour la conception de mon CV.

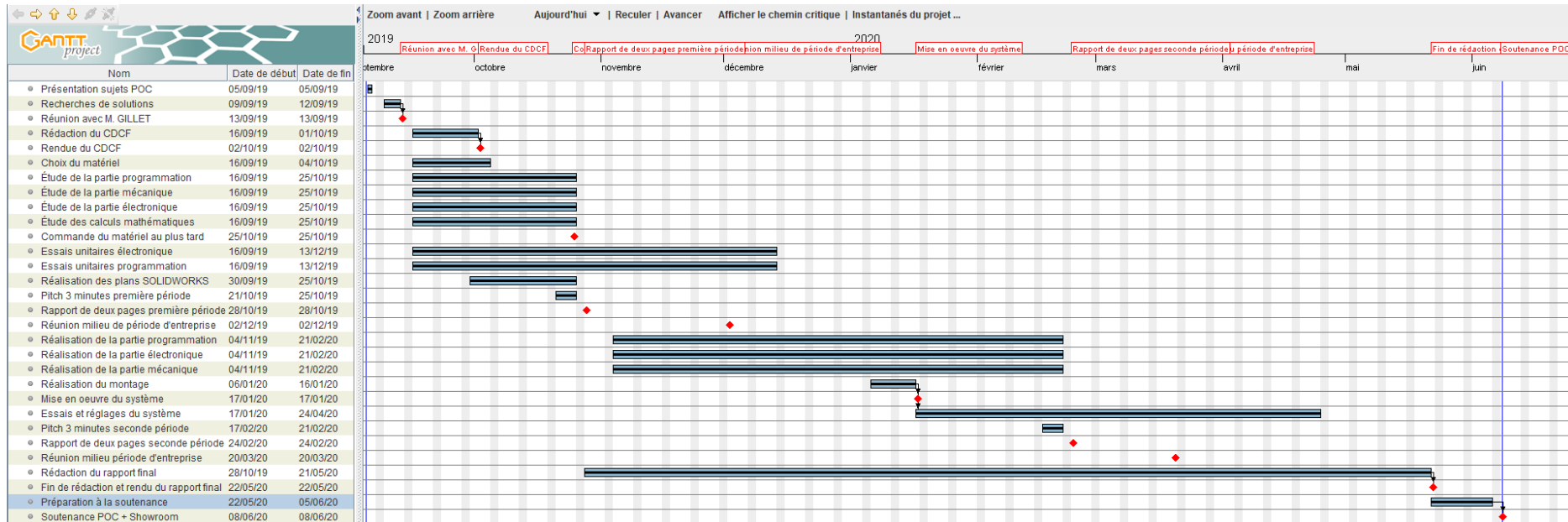
## 11 Bibliographie

- [1] Vidéo « JV Techno #2 Robotique parallèle à câbles ». Disponible sur : [https://www.youtube.com/watch?v=52pOPf\\_IVDU](https://www.youtube.com/watch?v=52pOPf_IVDU).
- [2] Vidéo « Robotique parallèle à câbles : cas d'application assemblage ». Disponible sur : <https://www.youtube.com/watch?v=J3R01lqwUi8>.
- [3] Vidéo « TBot: a high-speed cable-driven parallel robot ». Disponible sur : <https://www.youtube.com/watch?v=9BN3TZJDsms>.
- [4] Vidéo « Extreme Fast Cable-Driven Parallel Robot ». Disponible sur : [https://www.youtube.com/watch?v=0LWx\\_Y9DKPc](https://www.youtube.com/watch?v=0LWx_Y9DKPc).
- [5] Thèse « Commande modale de robots parallèles à câbles flexibles ». Disponible sur : <https://tel.archives-ouvertes.fr/tel-01544522/document>.
- [6] Mémoire de Nathaniel ZOSO « Modélisation, simulation et commande d'un robot parallèle plan à câbles sous-actionné ». Disponible sur : [https://robot.gmc.ulaval.ca/fileadmin/documents/Memoires/nathaniel\\_zoso.pdf?fbclid=IwAR2FjukGZNLFEh7CKW4SGy3S\\_Gx3bIQ4pvt61\\_dLTxpW70nxSwww3oozmKU](https://robot.gmc.ulaval.ca/fileadmin/documents/Memoires/nathaniel_zoso.pdf?fbclid=IwAR2FjukGZNLFEh7CKW4SGy3S_Gx3bIQ4pvt61_dLTxpW70nxSwww3oozmKU).
- [7] Thèse de Samuel BOUCHARD « Géométrie des robots parallèles entraînés par des câbles ». Disponible sur : [https://robot.gmc.ulaval.ca/fileadmin/documents/Theses/samuel\\_bouchard.pdf?fbclid=IwAR0fQLfIGTaHVPMzPq0EWb\\_ok5hZUlearOpJ\\_IWQNJbqMmQH1o2Jxh1c4Cc](https://robot.gmc.ulaval.ca/fileadmin/documents/Theses/samuel_bouchard.pdf?fbclid=IwAR0fQLfIGTaHVPMzPq0EWb_ok5hZUlearOpJ_IWQNJbqMmQH1o2Jxh1c4Cc).
- [8] Page internet « Modéliser les robots à câbles ». Disponible sur : <https://interstices.info/modeliser-les-robots-a-cables/>.



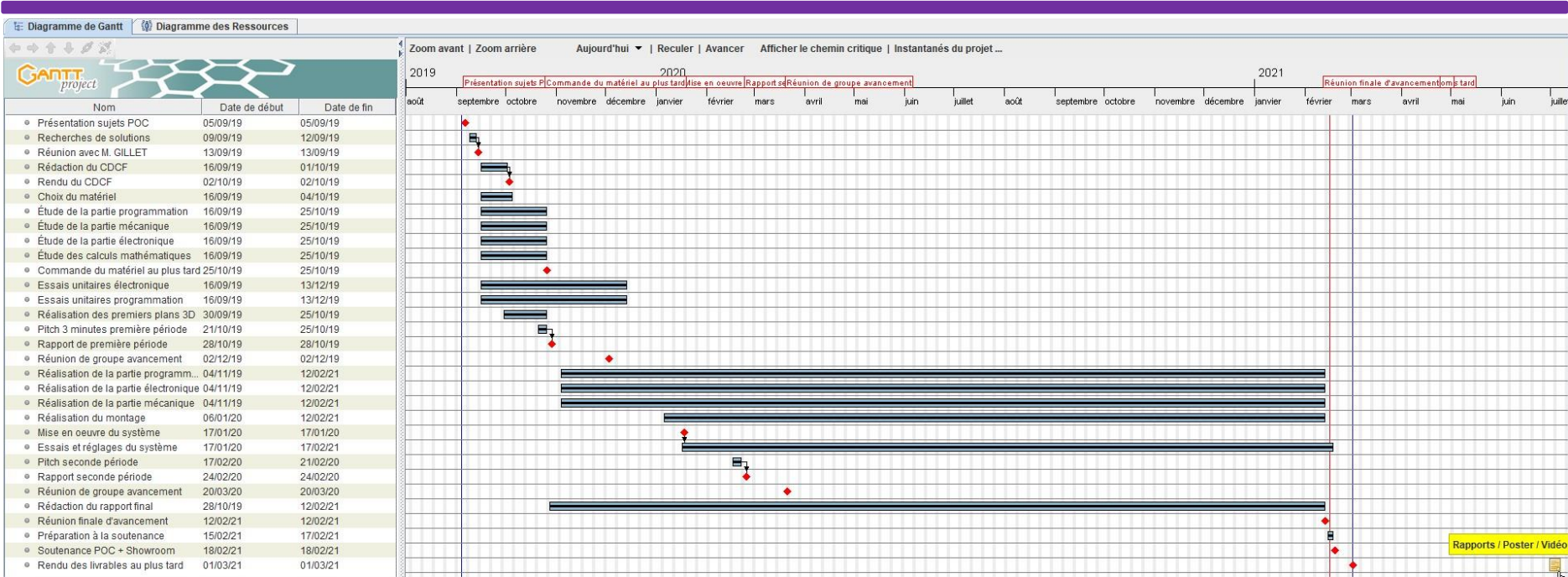
# POC : ROBOT PARALLÈLE À CÂBLES

## 12 Annexes



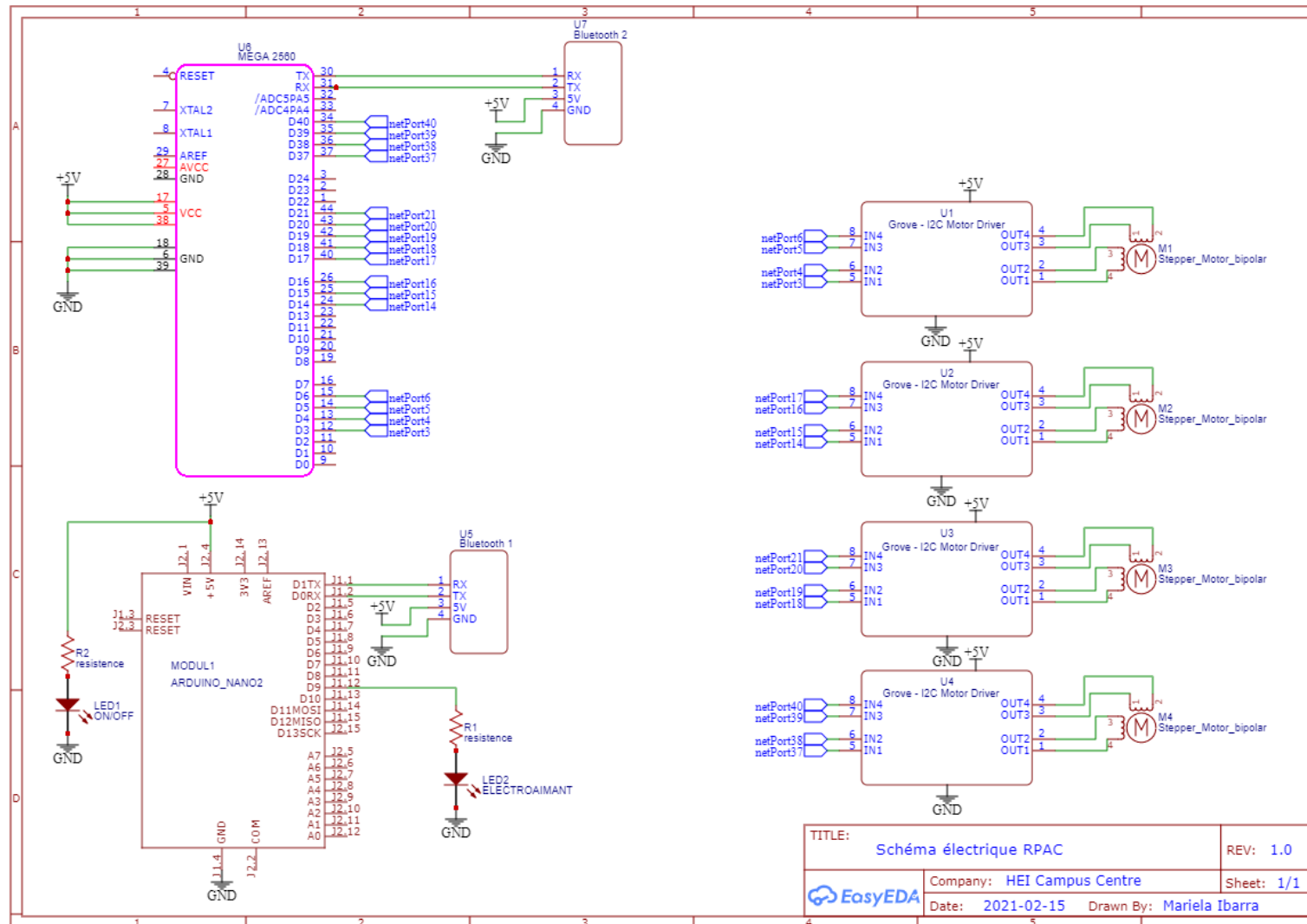
Annexe 1 : Diagramme de Gantt prévisionnel

POC : ROBOT PARALLÈLE À CÂBLES



Annexe 2 : Diagramme de Gantt réalisé

## POC : ROBOT PARALLÈLE À CÂBLES



Annexe 3 : Schéma de câblage du robot