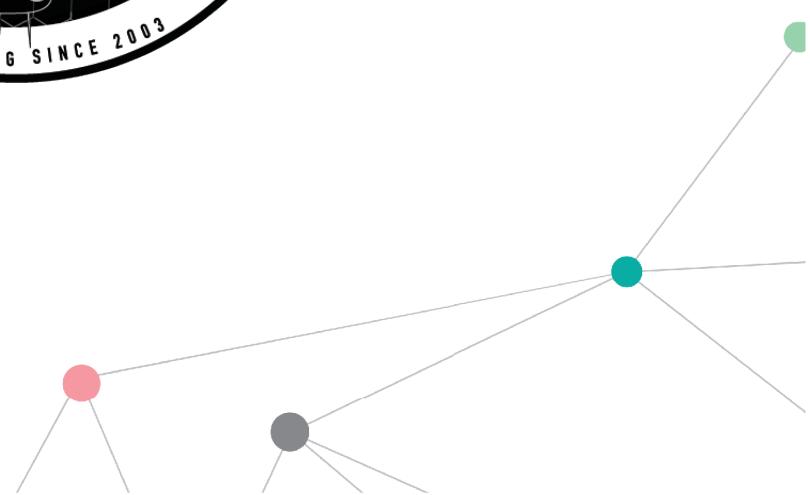


Eirbot  
Association de robotique de l'ENSEIRB-MATMECA

---

## Présentation des projets 2020

---



## Notre projet : coupe de France de robotique édition 2020

Cette année, Eirbot prend le large! Nos robots doivent partir voguer à travers le monde. Nous allons devoir maîtriser la navigation pour arriver à bon port. C'est après une tempête qu'il faudra reconstruire le chenal, réactiver le phare et redresser les manches à air pour espérer gagner. Nos robots évolueront sur la table suivante



FIGURE 1 – Schéma de la table de jeu

En plus de l'environnement de jeu nous devons suivre des règles précises disponibles sur ce [lien](#). Cette année Eirbot mise sur la qualité plutôt que la quantité : le robot autonome aura pour objectif de faire avec précision les actions qu'il entreprendra. L'objectif sera alors de construire et réactiver le phare, de redresser les manches à air et de retourner à bon port après avoir décodé la boussole.

## Plans du robot : des bases solides pour un bon départ

Les plans de notre robot ont été réalisés. Ils ont pu devenir concrets grâce à la découpe laser du FabLab de l'ENSEIRB MATMECA et à notre imprimante 3D. En addition des profilés ont été utilisés pour monter la structure et créer des étages faciles à monter et démonter.

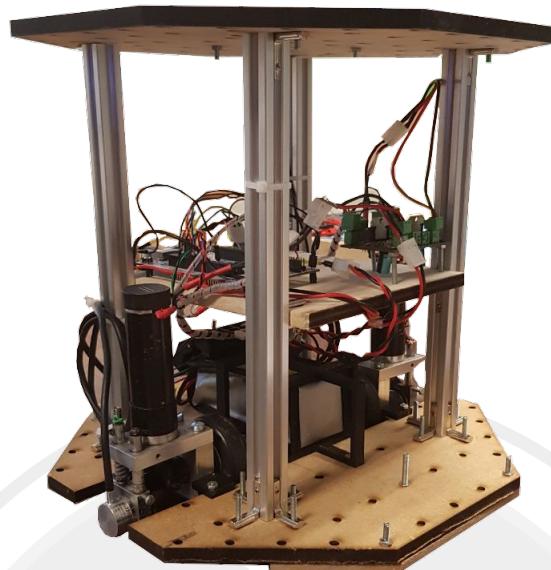


FIGURE 2 – Etat actuel de notre robot

Concernant notre phare, il doit se déployer et réaliser un balayage lumineux (il doit faire moins de 30 cm avant l'activation et plus de 70 cm après). Nous avons choisi l'impression d'un bras robotique pour notre phare, le système d'éclairage du phare sera similaire à celui d'un vrai phare. Voici l'avancement actuel

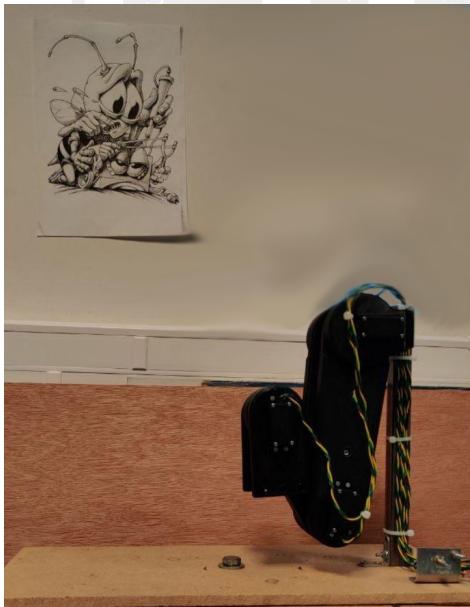


FIGURE 3 – Etat actuel de notre phare replié



FIGURE 4 – Etat actuel de notre phare déplié

## Fonctionnement général du robot : un contrôle précis pour une stratégie avancée

Comme dit précédemment Eirbot mise sur la qualité de nos actions plutôt que la quantité. Le travail informatique a donc été divisé en deux parties. D'une part l'asservissement qui est implémenté sur une Nucléo-F429ZI. D'autre part, la stratégie qui est implantée sur une Raspberry pi 3b+. En addition un protocole de communication entre les deux cartes a été créé.

Nous pouvons commencer par décrire succinctement l'asservissement que nous avons élaboré. L'étape d'asservissement est cruciale. L'idée est que l'on puisse demander au robot d'avancer d'une certaine distance et qu'il donne les bonnes consignes aux moteurs pour réaliser cette distance avec le moins d'erreur possible. Sans cela les stratégies qui seront mises en place par le suite ne pourront pas aboutir.

Avec un asservissement précis nous pouvons réaliser une stratégie plus complexe. Ainsi la stratégie dépend de deux facteurs principaux, la capacité à naviguer sur la table et la capacité à détecter les adversaires et à réagir lorsque l'un d'eux se présente. Pour la navigation un algorithme A\* a été implanté. C'est un algorithme de recherche de chemin connu dérivé de l'algorithme de Dijkstra. Le principe de ce dernier est résumé sur cette [animation](#)<sup>1</sup>.

Les détails techniques de l'implémentation sont disponibles sur le [Github](#)<sup>2</sup> de l'association. A ce stade, l'algorithme est opérationnel et il nous permet d'un point  $x, y$  donné rejoindre n'importe quel point  $x', y'$  en évitant les obstacles fixes. Nous allons maintenant présenter notre stratégie concernant les obstacles mobiles c'est-à-dire les adversaires.

Un système infrarouge (GP2) est utilisé pour la détection, ces derniers sont placés juste au dessus des gobelets disposés sur la table (les gobelets sont considérés comme des obstacles fixes, nous ne voulons pas les détecter avec le système infrarouge). Lorsque un robot adverse est détecté, nous récupérons l'information sur la distance fournie par le système infrarouge, puis nous évitons l'obstacle. A ce stade nous avons un système de détection nous permettant d'éviter les robots adverses en les contournant.

Nous avons présenté nos principales stratégies mais nous n'avons pas décrit tous nos codes. Les autres codes sont disponibles sur notre dépôt [Github](#)<sup>3</sup>.

En choisissant de séparer les codes sur deux cartes différentes, nous avons une difficulté supplémentaire qui apparaît. Il faut réaliser un protocole de communication permettant d'envoyer des ordres de la carte maître (Raspberry Pi) vers la carte esclave (Nucléo). Cette dernière répond aux commandes par une action (déplacement, activation des actionneurs, ...) ou par des informations sur l'état du robot (position, capteurs de distance, ...). Nous avons choisi d'établir la communication via un port série et nous avons créé un protocole spécifique aux besoins de notre projet. Le protocole implémente un système de confirmation des commandes. Prenons un exemple simple, lorsque nous lançons la Navigation sur la Raspberry pi, l'A\* va calculer les positions à rejoindre. Au fur et à mesure que l'algorithme trouve les positions il envoie une requête de déplacement via le port série, attend la réponse du protocole et s'il n'y a pas de soucis envoie la prochaine instruction.

1. [https://fr.wikipedia.org/wiki/Algorithme\\_A\\*](https://fr.wikipedia.org/wiki/Algorithme_A*)

2. <https://github.com/eirbot/eirbot2020-1A/blob/master/code/rasp/src/navigation.cpp>

3. <https://github.com/eirbot/eirbot2020-1A>

## Electronique du robot : une architecture sur mesure

Pour que un robot fonctionne correctement, il a besoin d'être alimenté. La conception d'une carte d'alimentation est une étape fondamentale lors de la mise en oeuvre d'un projet de robotique. Cette dernière assure l'alimentation du circuit à une tension souhaitée. La stabilité du signal doit être également maintenue malgré les perturbations liées au circuit. Pour le robot, nous avons opté pour une carte comportant 3 rails d'alimentation.

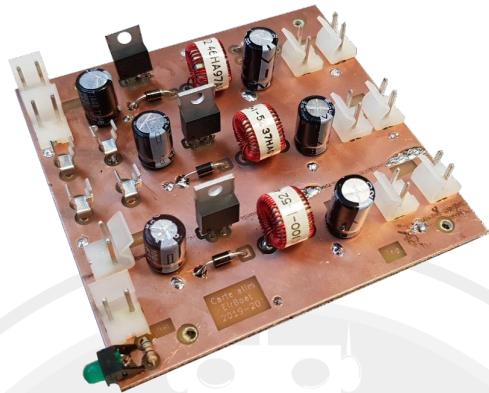


FIGURE 5 – Etat actuel de notre carte d'alimentation

Deux de ces rails fournissent une alimentation de 5V, le premier alimente les microcontrôleurs, le suivant les actionneurs. Le dernier est consacré aux moteurs, alimenté en 12V.

Mis à part les composants, l'intégralité de la carte a été conçue par nos soins . La conception est effectuée grâce au logiciel Kickad et la réalisation est possible dans l'enceinte de l'ENSEIRB-MATMECA. Le perçage des trous et la soudure des éléments se réalise avec le matériel de l'association.

En plus de l'alimentation du robot ce dernier utilise deux moteurs pour se déplacer. Il faut designer une carte de puissance afin de pouvoir les contrôler. Cette carte fait la liaison entre l'alimentation, la carte Nucleo et les moteurs eux-mêmes. Les cartes de puissances sont actuellement en cours de fabrication.

## Nos projets antérieurs

Notre association possède une certaine ancienté et en plus de participer à la coupe de France nous réalisons plusieurs projets voici quelques photos des projets et des robots des années précédentes



