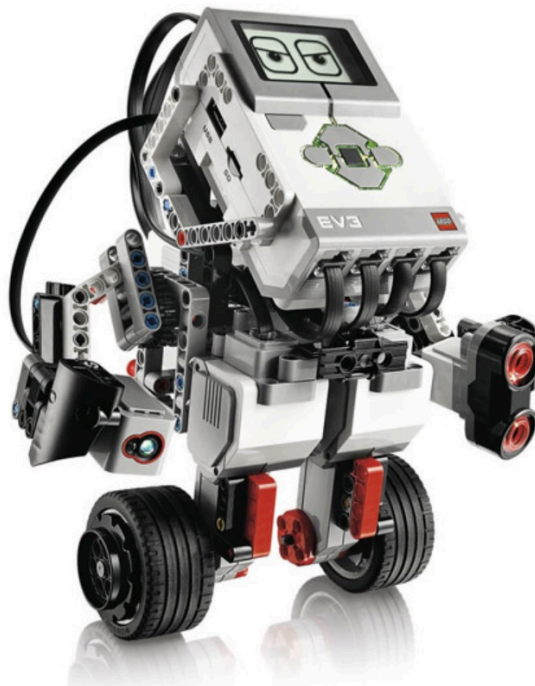


Robot Ramasseur

RAPPORT FINAL



L3 MIASHS
Intelligence Artificielle

AIT EL HADJ Anas
AMACHAT Yousra
HALILY Youssef
NGUIRANE Marième
TAMSOURI Mohammed

Sommaire

1. Introduction	3
2. Stratégies pour la réalisation du programme	3
2.2. Gestion des Collisions	4
3. Analyse des forces et faiblesses de Notre Équipe et Robot	5
3.1. Les points forts	5
3.2. Les points faibles	5
4. Analyse des succès et des échecs de notre robot	6
4.1. Succès	6
4.2. Échecs	6
5. Amélioration	6
6. Références	8

1. Introduction

Au cours des 12 dernières semaines, notre équipe a travaillé sur la réalisation d'un programme permettant à un robot lejo mindstorm de récupérer un maximum de palets en un minimum de temps. Ce document a pour but de faire un bilan de ce projet en revenant sur la stratégie adoptée, les forces, les faiblesses ainsi que les réussites et les échecs à l'issue de celui-ci. À la suite de ce bilan, quelques améliorations possibles seront proposées afin de permettre un meilleur résultat à l'avenir.

2. Stratégies pour la réalisation du programme

Afin que le ramassage se fasse le plus efficacement possible, nous avons distingué le ramassage des trois premiers palets de celui des autres. En effet, la position de ceux-ci est plus ou moins connue étant en début de partie tandis que les autres sont placés plus aléatoirement : ils peuvent potentiellement être déplacés par les robots au cours de la partie. Nous avons ainsi mis en place une méthode pour le ramassage du premier palet, une autre commune pour le deuxième et le troisième palet, et une dernière pour tous les autres.

Pour les autres palets, nous avons mis en œuvre une méthode de recherche de palets. Le robot commence par chercher l'objet le plus proche parmi ceux situés une distance supérieure à 34 cm. Il se rapproche de l'objet repéré tout en vérifiant, s'il s'agit d'un palet. Une fois qu'il est identifié comme un palet, il s'avance donc jusqu'à ce que le capteur tactile soit activé, le ramasse et le dépose dans le champ adverse. Toutefois, s'il n'y a pas activation du capteur touche, le robot recule de 40 cm, et recommence sa recherche. Si l'objet repéré n'est pas un palet ou si aucun objet intéressant n'est détecté, le robot tourne de 180 degrés, avance de 30 cm, puis reprend le même processus jusqu'à ce qu'un nouveau palet soit identifié et ramassé.

Cette méthode est également utilisée dans le cas où le robot ne trouve pas de palets avec la méthode du deuxième palet ou troisième palet afin que le programme puisse continuer son exécution. Celle-ci ne s'arrête que lorsque le robot a déposé six palets dans le champ adverse.

2.2. Gestion des Collisions

Dans le cadre de la compétition, le robot peut rencontrer divers objets, notamment le robot adverse et les murs, ce qui peut l'endommager ou le ralentir. Des mesures ont été prises pour y pallier. Grâce au capteur ultrasonique, on mesure la distance séparant le robot de l'objet situé dans sa trajectoire, lorsque celle-ci est inférieure à 10 cm, ce qui implique que l'objet en question n'est pas un palet, ceux-ci n'étant pas détectables à moins de 32,6 cm. Le robot tourne de 90 degrés, puis avance sur une distance de 25 cm, puis effectue une autre rotation de 90 degrés contournant ainsi l'obstacle. Pour s'assurer d'avoir bien contourné l'objet, il avance ensuite d'une distance égale à celle détectée par le capteur, augmentée de 10 cm. Cette méthode est appliquée chaque fois que le robot se déplace sur le plateau.

2.3. Organisation du travail au sein du groupe

Au début de projet, les différentes tâches et temps qui leur sont allouées ont été définis grâce au cahier des charges et à l'échéancier.

Le travail a été réalisé en se basant sur ceux-ci et chaque étape, l'équipe a été divisée en 3 et chaque sous-groupe travaillait de son côté. Une réunion est ensuite organisée avec le groupe pour que chacun présente ce qu'il a fait. Des réunions au sein des sous-groupes ont également été faites au besoin.

Beaucoup de travail a été effectué en dehors des heures de cours, notamment au FabLab pour tester directement le code sur la plateforme.

Nous avons employé différents moyens pour structurer et réaliser le projet qui sont

- **Google Drive** : stockage et suivi des documents à chaque étape
- **Google Docs** : rédaction des documents
- **Eclipse** : programmation en Java
- **GitHub** : hébergement du code et des documents
- **WhatsApp, Google Meet** : communication et coordination entre les membres.

Par ailleurs, des réunions régulières ont été organisées au sein de l'équipe, ainsi que des échanges fréquents avec notre encadrant, M. Damien Pellier, ont joué un rôle essentiel dans le suivi et l'orientation du projet.

3. Analyse des forces et faiblesses de Notre Équipe et Robot

3.1. Les points forts

L'un de nos points forts a été notre détermination et notre capacité d'adaptation. En effet, malgré les obstacles rencontrés, notre équipe a su aller au-delà des erreurs, en tirer des leçons et explorer de nouvelles pistes, sans hésiter à réaliser de nombreux tests pour perfectionner le code et apporter des améliorations progressives au projet.

De plus, la répartition des tâches a été efficace. Bien que la communication ait parfois posé un problème, nous pouvons souligner que les tâches ont bien été découpées et réparties. Chaque membre connaissait précisément les tâches qui lui avaient été confiées, ce qui a permis d'accélérer l'avancement du travail.

3.2. Les points faibles

La communication entre les membres a été une des principales faiblesses de notre groupe. Les malentendus et les modifications répétées de notre code étaient souvent causés par cela, car les idées et les ajustements n'étaient pas clairement partagés ou validés collectivement. De plus, cette absence de coordination a provoqué des instabilités dans le déroulement du projet, nous contraignant à répéter certaines étapes déjà réalisées. En outre, cette instabilité a joué un rôle dans notre positionnement parmi les derniers, comme nous avons consacré un temps précieux à rectifier des erreurs qui auraient pu être évitées grâce à une planification plus efficace et une communication plus organisée au sein de l'équipe.

4. Analyse des succès et des échecs de notre robot

4.1. Succès

Notre projet a eu quelques succès. En effet, le robot a réussi à récupérer les deux premiers palets. Par la suite, il a également pu naviguer sur le plateau tout en évitant les obstacles, repérer la ligne blanche, une étape essentielle pour pouvoir déposer les palets. Ces résultats démontrent que les parties du programme, relatives aux pinces, à la détection des palets et des autres objets, au calibrage des couleurs, fonctionnent bien.

4.2. Échecs

Une des principales difficultés rencontrées dans notre projet sur le robot était due à un souci de déviation aussi bien lorsqu'il tournait que lorsqu'il avançait et en observant le mouvement des roues, nous avons remarqué une désynchronisation qui entraînait un désalignement et rendait les déplacements du robot imprécis. Malgré nos multiples tentatives de résoudre ce problème en modifiant les paramètres de contrôle et en ajustant les roues, le problème persistait et avait un impact sur les performances du robot. Les nombreuses modifications ont sur le long terme entraîné un dysfonctionnement du robot qui ne répondait plus du tout au programme en fin de compétition.

5. Amélioration

- **Accélération/décélération progressive** : Dans notre code, le robot se déplace à une vitesse constante, ce qui peut être optimisé pour plus de fluidité, notamment lors de la prise ou du dépôt des palets, surtout pour assurer le premier palet avant les autres.
 - **Solution** : Implémenter un contrôle de la vitesse, avec une accélération progressive au début du mouvement et une décélération progressive avant de s'arrêter ou d'effectuer une manœuvre précise (comme attraper un palet ou s'arrêter au dépôt)
- **Réduction du temps de rotation** : Le robot tourne beaucoup pour s'orienter vers les palets ou le camp adverse. On pourrait optimiser la méthode de rotation en ajustant la vitesse ou en utilisant des manœuvres plus efficaces.

- **Solution** : Plutôt que de faire une rotation à 360° ou de grandes rotations, le robot pourrait effectuer des **petites rotations ajustées** pour garder le contrôle tout en économisant du temps et de l'énergie.
- **Optimisation de la méthode esquive** : Pour éviter un objet, le robot tourne plusieurs fois de 90 degrés, or, la rotation étant déjà assez lente, le robot perd donc assez de temps à chaque fois que cette méthode est effectuée, or le temps est très limité lors de la compétition
 - **Solution** : Plutôt que de faire une rotation à 90° ou de grandes rotations, le robot pourrait effectuer une rotation d'un plus petit angle, avancer un peu, puis faire une rotation du même angle, mais dans le sens inverse pour retrouver sa direction initiale. Il esquivera ainsi l'objet en un temps minimal.

6. Références

- Site documentation Lejos:
[Overview \(leJOS EV3 API documentation\) \(sourceforge.io\)](#)
- Cahier des charges :
<https://github.com/AnasAEA/Groupe-IA-Miashs-LEJOS/blob/master/Cahier%20de%20charge.pdf>
- Plan de Développement :
<https://github.com/AnasAEA/Groupe-IA-Miashs-LEJOS/blob/master/Plan%20de%20developement%20corrig%C3%A9.pdf>
- Plan de Tests :
<https://github.com/AnasAEA/Groupe-IA-Miashs-LEJOS/blob/master/Plan%20de%20Tests.pdf>
- Code Source :
<https://github.com/AnasAEA/Groupe-IA-Miashs-LEJOS/tree/master/src>