Projet Mindstorm

HEAFALA Natacha, LARREINEGABE Gerardo

Abstract

L'objectif de ce projet est de résoudre le problème de ramasseur de balles, le robot Mindstorm qui nous a été confié et que nous avons nommé Çucho". Le principe est de mettre le plus de palets possibles dans le camp adverse et dans les règles de la persycup.

Une source de code nous a été proposé pour nous permettre d'y ajouter et/ou modifier quelques fonctionnalités, ou uniquement, de s'en inspirer. Après avoir étudié, analysé et compris globalement les méthodes déjà implémentées et tout ce dont le robot était capable de faire (Qualités, défauts, problèmes, etc.), nous avons décidé de ne garder que l'implémentation des interfaces en relation avec le matériel. On a essayé de programmer par nous mêmes, les différents états du robot, le déroulement du jeu et les différentes actions et situations possibles.

Pour mieux comprendre ce qui a été, seront présentés dans ce rapport :

1. Le lancement du robot
2. Déroulement du jeu
3. Problèmes rencontrés et possibilité d'amélioration

1. Lancement du robot

A chaque lancement du robot, plusieurs paramètres sont à définir pour les différentes calibrations nécessaires.

1.1. Calibration des couleurs

La calibration des couleurs est sauvegardée à son premier lancement et peut être recalibrée à chaque lancement. Toutes les informations pour permettre cette calibration seront affichées à l'écran. Le robot doit être sur le terrain avec le capteur de couleur allumé sur la couleur indiquée.

Preprint submitted to Planification automatique et techniques d'intelligence artificielle28 de abril de 2017

1.2. Calibration des pinces

Contrairement à la calibration des couleurs, celle des pinces doit être faite à chaque lancement du robot. En effet, l'ouverture des pinces dépendra toujours de l'état final du lancement précédent, donc la calibration est utile. Elle permettra de définir la position des "bras"dans les cas où il contient un palet (fermé), et quand il n'en a pas (ouvert).

1.3. La position de départ du robot

Le robot peut être positionné dans un des emplacements autorisés, à conditions de bien paramètrer sa position au lancement. Tout sera indiqué à l'écran, à savoir :

- Si on est le seul robot sur le terrain ou non,
- Si la ligne noire est à droite du robot à son départ (pour éviter au mieux d'être en collision avec le mur),
- Si la première ligne qu'on risque de rencontrer est la ligne bleue ou la ligne verte (pour le déroulement des actions après une collision quelconque).

Pour un emplacement de départ sur la ligne noire, l'initialisation de la position du robot par rapport à cette ligne peut être initialisée à oui ou non.

2. Déroulement du jeu

2.1. firstMove, playStart

Calibration faite. Le premier état est "firstMove", il permet d'avancer jusqu'à trouver un palet pour passer dans l'état "playStart". Dans cet état, nous avons permis une rotation de quelques degrés (modifiables) permettant d'éviter les autres palets en place. Il dépose le palet et effectue un demi-tour vers le terrain pour en chercher d'autres.

2.2. Continuité du programme

Premier palet déposé. Nous avons utilisé tous les capteurs présents (vision, pression et couleurs) pour s'orienter, se déplacer, repérer un palet, le récupérer, etc. L'utilisation de la caméra, nous est utile, uniquement, pour vérifier la présence de palet sur le terrain.

2.3. Arrêt du robot

Le robot s'arrête automatiquement quand la caméra indique qu'il n'y a plus de palets sur le terrain. Pour l'arrêter manuellement, il est possible de le stopper en pressant le bouton du centre et du bas. Il lui arrive encore de ne plus s'y retrouver sur le terrain, dans ce cas là, on l'arrête avec ces boutons.

3. Problèmes rencontrés et possibilité d'améliorations

3.1. Caméra

Nous avons rencontrés des problèmes de localisation par rapport à la caméra. En effet, notre idée de départ était de savoir se situer sur le terrain, puis de s'orienter, et de se déplacer grâce aux coordonnées des palets que fournit la caméra. Or, le robot n'est pas différencié des palets ce qui nous a amené à notre configuration du déroulement du jeu actuelle. Il aurait été sûrement plus rapide et plus stratégique pour la recherche de palets, si nous avions fait en fonction des coordonnées. La localisation par caméra aurait tout aussi bien pu nous éviter de se tromper au niveau de la rotation (de quel côté se tourner) pour empêcher toute déviation ou désorientation causée par un mur ou un autre robot. On a jamais eu ce problème lors des tests, mais on n'oublie pas que cette situation est possible.

3.2. Vision

Les distances maximale et minimale de la vision d'un palet ne sont pas très précises. Notre programme permet plusieurs essais pour un palet. Mais, il serait préférable de calculer quel angle de vue peut atteindre le robot, pour qu'on puisse s'orienter vers un palet avec plus de précisions (à l'aide de la distance de celui-ci).

3.3. Bug connu

Quelques fois, le lancement du robot s'arrête. Il faut tout simplement le relancer. Cela est du aux plusieurs threads lancés en parallèle (Celui de la verification du mur, ne doit pas être le premier thread à être lancé).

3.4. Mur

Le mur est très difficile à prévoir ; On prend en compte quelques situations, dans lesquelles le robot réagit correctement. Mais si le robot parvient à trop dévier à cause du mur, l'orientation n'est plus efficace (Il a perdu le nord). Si on prend en compte la caméra, la réorientation (initialisation du nord) peut se faire.