

Cours ISC - 2A SLE

Implantation des systèmes de contrôle

Projet Robot LEGO : Cahier des charges

Thao Dang et Ahmad Hably

CNRS-VERIMAG et GIPSA

Attention : Le cahier de charges peut évoluer selon le déroulement du projet. Il faut consulter le lien suivant pour la version à jour.
<http://www-verimag.imag.fr/~tdang/DocumentsCours/2014ProjetCahierCharges.pdf>

1 Contexte et Objectifs

Ce projet s'inscrit dans le contexte du développement de systèmes de commande embarquée basé sur des modèles. Par l'utilisation d'une chaîne d'outils pour l'implantation de systèmes de commande embarquée, nous illustrons des aspects importants de l'implantation sûre et de la génération automatique de logiciel de contrôle. Le but de ce projet est de concevoir et implanter un contrôleur pour faire suivre au robot une ligne noire sur le sol, en utilisant deux capteurs de lumière. La réalisation de ce projet se fait parallèlement à la partie théorique

2 Conception de contrôleur

Le robot possède deux capteurs de lumière (à gauche) et à droite qui sont positionnés sur le devant et pointent vers le sol afin de détecter une ligne noire. Le robot possède deux moteurs connectés à ses deux roues dont les vitesses sont v_G et v_D . La solution utilisée dans ce projet est comme suit. Le contrôleur peut être décomposé en deux blocs.

- Bloc de calcul d'écart. Ce bloc calcule les erreurs d'orientation et de vitesse. Les capteurs sont utilisés pour mesurer l'écart de la trajectoire du robot par rapport

à la ligne noire, à partir des données c_G et c_D fournies par les capteurs à gauche et à droite. La vitesse moyenne du robot dépend de l'erreur d'orientation. Quand l'erreur d'orientation à corriger est grande, l'erreur de vitesse est donc réduite afin que le robot ne rate pas la ligne.

- Bloc de correcteurs. Les erreurs d'orientation et de vitesse sont les entrées du bloc de correcteurs. L'orientation est corrigée par un correcteur de type PI, et l'erreur de vitesse par un correcteur de type P.

Un modèle Simulink est fourni (voir la page du cours)

3 Implantation de contrôleur

Cette partie concerne le sujet principal de ce cours. Dans la suite nous résumons les tâches à faire et les questions à adresser dans le rapport final (à rendre le jour de soutenance).

Tâches :

1. "Preprocessing"

- Échantillonnage : discrétisation des contrôleurs, choix de période d'échantillonnage, ajout d'un filtre d'anti-alias (si besoin).

Question : expliquer le choix de période d'échantillonnage et de la méthode de discrétisation, expliquer le calcul du filtre (si un tel filtre a été nécessaire).

- Calibrage des entrées venant des capteurs : le modèle Simulink fourni a été conçu sous l'hypothèse que l'intervalle de valeurs données par des capteurs de lumière est de 0 (noir) à 100 (blanc). Il faut trouver cet intervalle en réalité pour chaque capteur et effectuer un calibrage (si nécessaire). Comme la lumière d'ambiance influence le fonctionnement des capteurs de lumière, il faut effectuer un calibrage chaque fois avant de lancer le programme de commande, par exemple on pose un capteur sur la ligne et la valeur donnée par le capteur sera considérée comme celle pour le "noir" dans la condition actuelle. On fait similairement afin de déterminer la valeur pour le "blanc". Tout ceci peut se faire par une tâche appelée une fois

avant la boucle de la fonction "step" du contrôleur dans le programme NXC.

Question : expliquer le calibrage.

- Conversion entre les types de données : dans notre approche, le modèle de contrôleurs en SIMULINK est compilé vers le code NXC. A la différence de SIMULINK, NXC ne permet que des flottant et des entiers, et la conversion peut faire perdre la précision. Il est aussi important de prendre en compte les bornes des flottants et des entiers representables en NXC.

Question : expliquer quand la conversion peut causer des problèmes de précision. Si c'est le cas pour votre contrôleur, quelle est la solution ?

2. Traduction du contrôleur en SIMULINK vers LUSTRE. Il faut noter que l'on traduit seulement le contrôleur (et pas le modèle Simulink entier qui contient aussi des blocs décrivant la dynamique du robot et l'émulateur de capteurs).
3. Compilation de LUSTRE vers NXC : ceci se fait en utilisant le compilateur lus2c. Il faut ensuite compléter le code de glue en NXC (pour spécifier les entrées, sorties, etc).

Question : expliquer le code de glue.

4. Compilation du code NXC et chargement du code dans le robot (en utilisant l'environnement Brixcc)
5. Expérimentation sur le robot et débogage si besoin.

Question : expliquer et interpréter la procédure d'expérimentation et les résultats.

4 Evaluation : rapport et soutenance

Le projet sera présenté et démontré sur les robots lors d'une soutenance d'environ 10-15mn. A rendre le jour des soutenances :

1. Un rapport qui adresse toutes les questions mentionnées dans la section "Implantation de contrôleurs".

2. Tous les programmes (Simulink, Lustre, NXC) et tous les autres fichiers nécessaires pour que l'enseignants puissent tester le contrôleur et reproduire les résultats sans aucun ajout de fichiers ou ajustement de paramètres.