Robotique : modélisation et commande

GMC714 - Devoir no7
Introduction à la commande optimale

Préparé par

Pr. Alexandre Girard

Instructions :

Vous pouvez faire les calculs à la main ou avec un script Matlab ou Python.
Vous pouvez consulter vos collègues pour vous entraider, mais chacun doit individuellement effectuer une résolution et produire un devoir.
La remise doit être un seul *pdf* qui contient tous vos résultats et calculs.

Évaluation selon une échelle descriptive globale :

A : L'étudiant arrive à toute les solutions, avec seulement des erreurs mineures, et démontre qu'il maîtrise les notions abordées dans le devoir.
B : L'étudiant n'arrive pas à obtenir toutes les solutions, mais démontre qu'il a en bonne partie assimilé les notions abordées dans le devoir du à un effort soutenu de résoudre chacun des numéros.
C : L'étudiant n'arrive pas à obtenir la majorité des solutions, ne démontre pas qu'il a assimilé les notions abordées dans le devoir et travaillé sérieusement sur chacun des numéros.
E : L'étudiant ne présente aucune démarche sérieuse.

Tutorial sur la commande optimale

Suivre et comprendre le notebook suivant (optionnel pas de livrable) :

Exercice de code
Optimal control overview
`https://colab.research.google.com/drive/1wXmlIqNGC2LrJkmyj56Y109b5ZDHVboq?usp=sharing`

1 Fonction de coût pour un pendule

Compétences à développer :

— Compréhension des paramètres d'une fonction de coût quadratique
— Compréhension de la forme générique de coût additif $J = \int_0^{t_f} g(x,u,t)dt + h(x_f, t_f)$

Pour ce numéro du devoir, vous devrez utilisez le code disponible au lien ici :

Exercice de code
Commande optimale d'un pendule simple
`https://colab.research.google.com/drive/1BzMh7bBNgflchsbgWkDFkgP2s-5HNqW9?usp=sharing`

Vous pouvez copier le *notebook* et travailler en-ligne directement dans colab, mais si vous préférez travaillez directement sur votre ordinateur, vous pouvez aussi télécharger la librairie disponible ici : `https://github.com/SherbyRobotics/pyro` et travailler localement sur votre machine.

Pour chacune des situations suivantes :
a) Situation de référence : exécuter le code avec la fonction coût quadratique par défaut.
b) Ajuster les valeurs dans la matrice Q pour pénaliser plus l'erreur en position du pendule.
c) Modifiez la fonction $g(x,u,t)$ pour obtenir une solution qui correspond au temps minimal.
d) [Optionnel] Testez et explorez d'autres variantes de fonction coût.
analysez et interprétez les résultats pour :
1) la figure de coût-à-venir J^* calculée pour tout les états (qui correspond au coût minimal qui va être encouru à partir de cet état si les actions optimales sont prises).
2) la loi de commande générée (couple en fonction de l'angle et la vitesse).
3) la trajectoire pour le système lorsque le pendule débute à partir de la position en bas.

Note : Pour plusieurs raisons l'algorithme peut avoir de la difficulté à converger pour certaines fonctions de coût. Essayez des changements plus mineurs si c'est le cas. Le but ici n'est pas de vous faire travailler pour ajuster les paramètres de convergence (un sujet pas encore abordé).

---

2 Navigation optimale dans un graphe

Compétences à développer :

— Programmation dynamique pour un problème avec des états et actions discrètes.
— Algorithmes pour déterminer un chemin le plus court dans un graphe

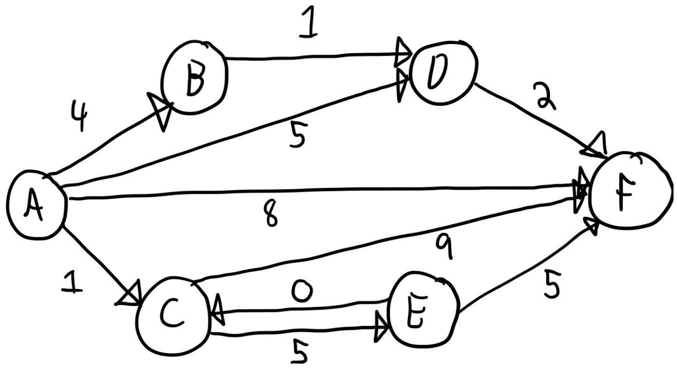

FIGURE 1 – Graphique qui représente des chemins possibles pour aller vers la position F

Appliquez l'algorithme de programmation dynamique exacte :

$$J_k^*(\underline{x}_k) = \min_{\underline{u}_k} \left[g_k(\underline{x}_k, \underline{u}_k) + J_{k+1}^*(\underbrace{f_k(\underline{x}_k, \underline{u}_k)}_{\underline{x}_{k+1}}) \right] \tag{1}$$

$$u_k^*(\underline{x}_k) = \underset{\underline{u}_k}{\operatorname{argmin}} \left[g_k(\underline{x}_k, \underline{u}_k) + J_{k+1}^*(\underbrace{f_k(\underline{x}_k, \underline{u}_k)}_{\underline{x}_{k+1}}) \right] \tag{2}$$

pour résoudre les actions optimales (choix de l'arc à suivre) pour se rendre à l'état cible (Noeud F) à partir de l'état actuel (les Noeuds $x_k \in [A, B, C, D, E]$) et de l'index de temps actuel k. Le coût de chaque option de chemin est représenté par le chiffre indiqué pour chaque arc sur le graphique ci-dessus. Considérez une fonction de coût sur un horizon de 5 pas de temps (N=5) et calculez les actions optimales pour les index de temps $k = [0, 1, 2, 3, 4]$. Considérez que le coût final est infini si on ne termine pas sur le Noeud F à $k = 5$. Le tableau suivant peut vous aider pour synthétiser les résultats :

$k =$	0	1	2	3	4	N = 5
$J^*(A) =$						
$u^*(A) =$						
$J^*(B) =$						
$u^*(B) =$						
$J^*(C) =$						
$u^*(C) =$						
$J^*(D) =$						
$u^*(D) =$						
$J^*(E) =$						
$u^*(E) =$						
$J^*(F) =$						