

1 Introduction

L'objectif du TP est de faire naviguer un robot mobile dans le plan pour qu'il atteigne une configuration finale q_{goal} en évitant les obstacles ponctuels de l'environnement W . La configuration du robot mobile ponctuel est définie par $q_r = (x_r, y_r)^T$.

Nous utiliserons la méthode des champs de potentiels pour faire naviguer le robot mobile. Dans cette méthode, le robot est assimilé à une particule se déplaçant suivant les lignes de courant d'un potentiel créé en fonction de l'environnement perçu par le robot. Ce potentiel traduit différents objectifs tels que l'évitement d'obstacles ou une direction de déplacement préférée. Il est calculé par sommation de différentes primitives de potentiels traduisant chacun de ces objectifs. Ces différents potentiels peuvent avoir une étendue spatiale limitée ou non (par exemple, n'avoir une influence que près des obstacles) et leur intensité peut dépendre ou non de la distance. Les primitives de potentiel qui seront considérées sont détaillées ci-dessous.

1.1 Potentiel attractif

Le potentiel attractif peut être défini sous forme parabolique :

$$U_{att_para}(q) = \frac{1}{2} \cdot K_{att} \cdot \|q - q_{goal}\|^2$$

Ou sous forme conique :

$$U_{att_con}(q) = K_{att} \cdot \|q - q_{goal}\|$$

1.2 Potentiel répulsif

Le potentiel répulsif peut être défini sous forme exponentielle :

$$U_{rep_exp}(q) = U_{max} \cdot \exp(-\|q - q_{obst}\| / K_{rep})$$

Ou sous forme hyperbolique :

$$U_{rep_hyp}(q) = \frac{1}{2} \cdot K_{rep} \cdot \left(\frac{1}{\|q - q_{obst}\|} - \frac{1}{d_0} \right)$$

1.3 Mouvement

Le robot mobile est dans un premier temps considéré ponctuel et n'est soumis à aucune contrainte de mouvement.

Son modèle cinématique est du type :

$$\dot{q} = \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} v_x \\ v_y \end{pmatrix}$$

Le potentiel final auquel est soumis le robot est $U = U_{att} + U_{rep}$. A chaque itération la force artificielle $F(q) = -\nabla U(q)$ indique la direction à suivre. On prendra pour simplifier $v(q) = F(q)$. Enfin, il est indispensable de programmer une fin de boucle pour arrêter le processus en cas de puits de potentiel.

Dans un second temps, le contrôleur développé doit être adapté à un robot *turtlebot3*, non-holonyme et non-ponctuel. Plusieurs modifications devront être faites, notamment la conversion de la commande de vitesse en une commande applicable à un robot de type *differential-drive*. Dans ROS, ce robot peut recevoir une commande telle que :

$$\dot{q} = \begin{pmatrix} v \\ \omega \end{pmatrix}$$

Avec v , la vitesse linéaire d'avance du robot, et ω sa vitesse de rotation. Cette conversion est déjà implémentée au travers de la fonction `compute_speed`.

2 Setup et introduction à ROS

Compiler et initialiser le workspace à l'aide des commandes suivantes :

```
$ source /opt/ros/melodic/setup.bash
$ cd turtlebot_ws
$ catkin build
$ source devel/setup.bash
```

Lancer et prenez en main le robot *turtlebot* dans le simulateur *Gazebo* :

```
$ export TURTLEBOT3_MODEL=burger
$ roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

Et dans un autre terminal :

```
$ roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

3 Travail à effectuer

L'ensemble du travail consiste à compléter le fichier *PF_TP_vide.py* qui se trouve dans le package ROS *PF_TP* (*turtlebot_ws/src/PF_TP/src*). La fonction `create_randomMap` permet de créer une carte d'environnement avec obstacles. La création génère des obstacles aléatoirement. Pour faire l'étude de paramètres sur une même carte, la sauvegarder et la recharger.

Question 1.

Calculer deux potentiels attractifs de forme différente $U_{att}(q)$ ayant un seul minimum global à la configuration but $q_{goal} = (x_{goal}, y_{goal})^T$ et $U_{att}(q) \geq 0 \quad \forall q$. Les visualiser avec Python. Quels sont les avantages et inconvénients de chaque forme ?

Question 2.

Calculer deux potentiels répulsifs de forme exponentielle $U_{rep}(q)$ garantissant la non-collision avec les obstacles. Les visualiser avec Python. Quels sont les avantages et inconvénients de chaque forme ?

Question 3.

Calculer plusieurs potentiels globaux $U(q)$. Les visualiser avec Python.

Question 4.

Utiliser les fonctions précédentes pour faire naviguer le robot mobile. Compléter le contrôleur *PF_control_holonom*.

Question 5.

On se propose maintenant de remédier aux inconvénients des formes de potentiel attractif en utilisant leur complémentarité. On définit alors une forme de potentiel attractif hybride et une distance $d_{att_lim} = 2m$ qui caractérise la frontière entre les deux formes primitives. Il est important de vérifier des contraintes de continuité et de dérivabilité à la jonction des deux formes, à savoir :

$$U_{att_para}(\|q - q_{goal}\| = d_{att_lim}) = U_{att_con}(\|q - q_{goal}\| = d_{att_lim})$$

$$\nabla U_{att_para}(\|q - q_{goal}\| = d_{att_lim}) = \nabla U_{att_con}(\|q - q_{goal}\| = d_{att_lim})$$

Pour que ces conditions puissent être vérifiées, il faut donc modifier légèrement la forme $U_{att_con}(q)$ en

$$U_{att_con}(q) = K_{att} \cdot \|q - q_{goal}\| + \delta$$

Donner les relations qui en découlent, puis créer et visualiser ce potentiel.

Question 6.

Analyser puis compléter le code du contrôleur *PF_control_turtlebot* permettant d'appliquer cette stratégie de commande à un robot non-holonyme et non-ponctuel.

Utiliser la fonction *create_gazeboMap* pour générer un modèle de l'environnement *Gazebo*.

Question 7.

Faire naviguer le robot sur la carte d'environnement *map_puits.npy* fournie. Quelle est la caractéristique de cette carte ? Étudier l'influence des gains attractif et répulsif pour une même trajectoire. Conclure. Proposer et tester une solution permettant de sortir des puits de potentiel.

Question 8.

Analyser et expliquer la fonction *compute_speed*.

Question 9.

Nous supposons maintenant que la position du robot est obtenue par un capteur de localisation globale. Sa position est donc entachée d'une erreur aléatoire. Modifier votre programme pour prendre en compte cette erreur de localisation. Conclure quant à la robustesse de la méthode.