README.md

Mobile Manipulation Capstone Project

Submitted by: Yuchen Wang

Northwestern University

Introduction

In this capstone project, I wrote a software that plans a trajectory for the end-effector of the youBot mobile manipulator (a mobile base with four mecanum wheels and a 5R robot arm), performs odometry as the chassis moves, and performs feedback control to drive the youBot to pick up a block at a specified location, carry it to a desired location, and put it down.

Implementation

Reference Trajectory Generation

Given eight configurations indicating the relationship between end-effector, cube and world frame under different conditions, generate a reference trajectory for the gripper on frame **{e}**.

The output is written to a cvs file containing 13 attributes: **r11, r12, r13, r21, r22, r23, r31, r32, r33, px, py, pz, gripper state**

$$T_{se} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

Kinematics Simulator for youBot

Given the current configuration of youBot (*Chassis phi, Chassis x, Chassis y, J1, J2, J3, J4, J5, W1, W2, W3, W4, Gripper*), joints speed and wheel speed, return the next configuration of the robot after a short time dt(*default as 0.01s*).

Forward Control

The feedback control of the the mobile manipulator is given by kinematic task-space feedforward plus feedback control law:

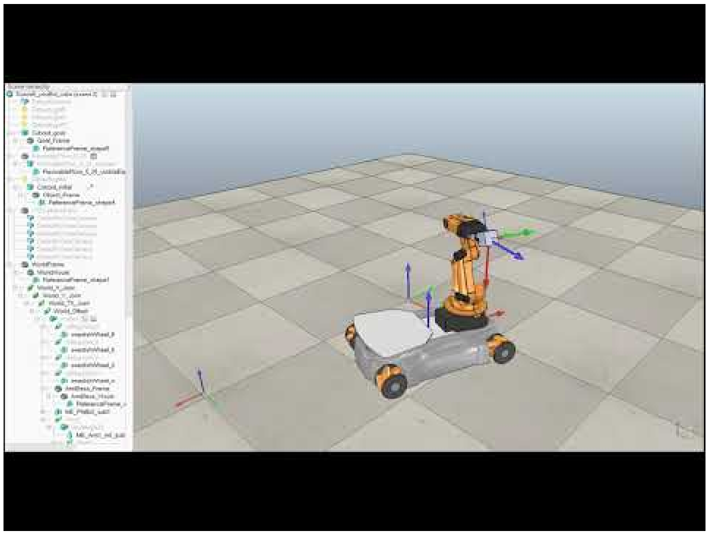
$$\mathcal{V}(t) = [\mathrm{Ad}_{X^{-1}X_d}]\mathcal{V}_d(t) + K_p X_{\mathrm{err}}(t) + K_i \int_0^t X_{\mathrm{err}}(\mathrm{t})d\mathrm{t}.$$

- Given the current, next and actual end-effector configurations, PI controller gains, return the commanded end-effector twist ***V*** and the error list of each joint.
- Given the joint angles, Body Jacobians and several other configurations, return the Jacobian of robot arm and base.

Result

Working V-REP Simulation

With the robot configuration return by the script, V-REP could successfully simulate the youBot moving to the cube, grabbing it and put it to another location and the movement is smooth enough.



Error Plot

By plotting out all six errors, it is clear that the the motion is smooth is smooth without any overshoot, also the settling time is short.