Project : Giraff robot for question handling in conferences

Course: Introduction to Robotics
Program: Masters in Artificial Intelligence Systems
Submitted by: Fariha Al Ferdous
ID# 238888

Introduction

This is a Giraff robot which is a robot with multiple links with 5 dof and 1 unused dof.

Background Research

Forward Kinematics is a domain where the End-effector position can be gained and from Differential Kinematics is used to gain the Jacobian to get the velocity of the end-effector in the task space.

Inverse Kinematics can be computed to gain joint configurations from end-effector position which works like a backward process of forward kinematics. Inverse Dynamics is related to the motion control of a robot which includes PD Controller which has been used in the project.

Software and APIs

- ROS2
- Python
- RVIZ2
- GAZEBO
- OS-Ubuntu
- Oracle VM - VirtualBox

Project Specifications

Project Overview

This is to model and simulate a robot, a giraffe robot with 5 dofs where it has a redundancy of 1 dof. The task asked for is 4D so there is a null space for the 1 dof.

1) At first URDF model has been designed for the whole robot. Forward Kinematics has been computed and direct differential kinematics have been computed in code utilizing the specifications given in the project brief.
2) It was assumed at first the Pinochhio library cannot be used in ROS2 since I have done the project in ROS2 not ROS1, so I did it all without using the Pinochhio library. Although if required, the pinochhio task can be done in future.
3) For the trajectory planning, the RRT* algorithm has been hard coded. Since there was no obstacle in the workspace as mentioned in the project brief, a very basic collision detection algorithm has been used which checks if the edges are in a distance from any assumed obstacle. For the cost of the nodes in the path, distance measurements between the nodes has been used. The achieved configurations along the path has been checked with a for loop if they are within the joint limits. If yes, only then they are chosen. Singularity checking has been implemented and also Null space task has been implemented using the last q configuration.

4) An Inverse dynamics control system has been coded with PD gain Feedback controller has been designed which takes care of the problem number 4 and 5 of the project brief.
5) PD Gain Controller has been designed with values of Kp and Kd to have a settling time of 7secs as instructed in the project brief.
6) The 1 dof has been used in minimizing the distance using a step size subtracting from the actual goal into a desired goal of minimized distance.
7) For the simulation, RVIZ2 has been used to view the model structure and GAZEBO has been used for the last task of the project brief which is moving from initial configuration to a destination position.

User Interface

Videos and screenshot attached with email

Conclusion and Future Work

I have not done the project using Pinochhio APIs since in the beginning I had a doubt whether that will work with ROS2. I wanted to use the updated version of ROS so used ROS2 iron distribution. I was able to visualize the robot on rviz2 but on Gazebo it took a weird shape. The parameters show fine but because of some pending computation tasks, only the last link is showing. And also the computations came out incomplete which I would work on further to completion. Currently the project is stuck at numpy computations which I am into to complete. Also, I would like to implement Pinochhio apis with ROS2 if it allows. Project package link is in email.

References

1) ROS2 Documentations
2) Course PDFs
3) Numpy and other required library websites
4) Coding media platforms on the web for debugging