

CS 6341 Robotics Homework 1

Professor Yu Xiang

September 3, 2025

In this homework, write down your solutions for problem 1 and finish the coding problem 2. Upload your solutions and code to eLearning. Our TA will check your solutions and run your scripts to verify them.

Problem 1

(3 points)

Grübler's Formula. Exercise 2.7 in Lynch and Park, Modern Robotics.

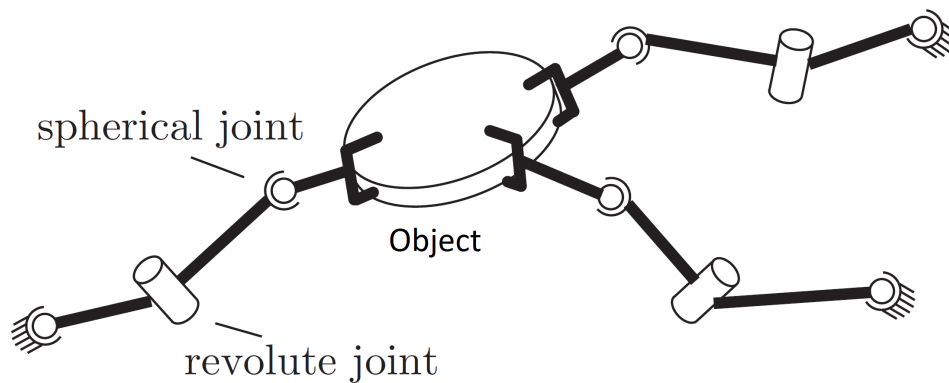


Figure 1: Three cooperating SRS arms grasping a common object.

Three identical SRS open-chain arms are grasping a common object, as shown in Figure 1. Find the number of degrees of freedom of this system using Grübler's formula. Write down how you apply Grübler's formula to this system.

Problem 2

(7 points)

ROS2 programming and Rviz.

In this problem, you will learn how to build a ros workspace and visualize a Fetch mobile manipulator robot in Rviz. **You can directly use Ubuntu, or Docker or virtual machine to install ROS2 according to your own set up.** Refer to the ROS2 Jazzy page if needed:

<https://docs.ros.org/en/jazzy/>.

(2.1) **(For Docker users)** First, build a docker image with ros2 following the instructions in the lecture. Test your ros2 installation by running the rviz2 command.

Before we start on building the ROS workspace, Let's learn how to mount a host folder into Docker. Usually, your code will be stored in your host folder. You can use the "-v" option of the Docker run command to mount a folder into the Docker environment. For example, the following command will mount a folder in Windows "C:\data" as a folder "/data" in Docker:

```
docker run -it -v C:\data:/data ubuntu:ros
```

In this way, you can save all your code in the host machine and use them in the Docker environment.

(2.2) Creating a ROS workspace. A ROS workspace is a place to store your own ROS packages. Following the link here to create a ROS workspace:

<https://docs.ros.org/en/jazzy/Tutorials/Beginner-Client-Libraries/Creating-A-Workspace/Creating-A-Workspace.html>

You should create the ROS workspace in the mounted folder from the host machine.

(2.3) Adding package to your ROS workspace. We are going to use two existing ROS packages to visualize the Fetch mobile manipulator. Download them from the following links to add to your workspace, i.e., add the packages to the src folder with git clone:

- fetch_description:

```
git clone https://github.com/IRVLUTD/fetch_ros2_IRVL.git
```

- urdf_tutorial:

```
git clone https://github.com/ros/urdf_tutorial -b ros2
```

You can read this page about ROS packages: <https://docs.ros.org/en/jazzy/Tutorials/Beginner-Client-Libraries/Creating-Your-First-ROS2-Package.html#>.

(2.4) Building and testing your ROS workspace. You can build the packages in the ROS workspace by running

```
colcon build
```

If the workspace has been built successfully, run the following command to test the urdf_tutorial package:

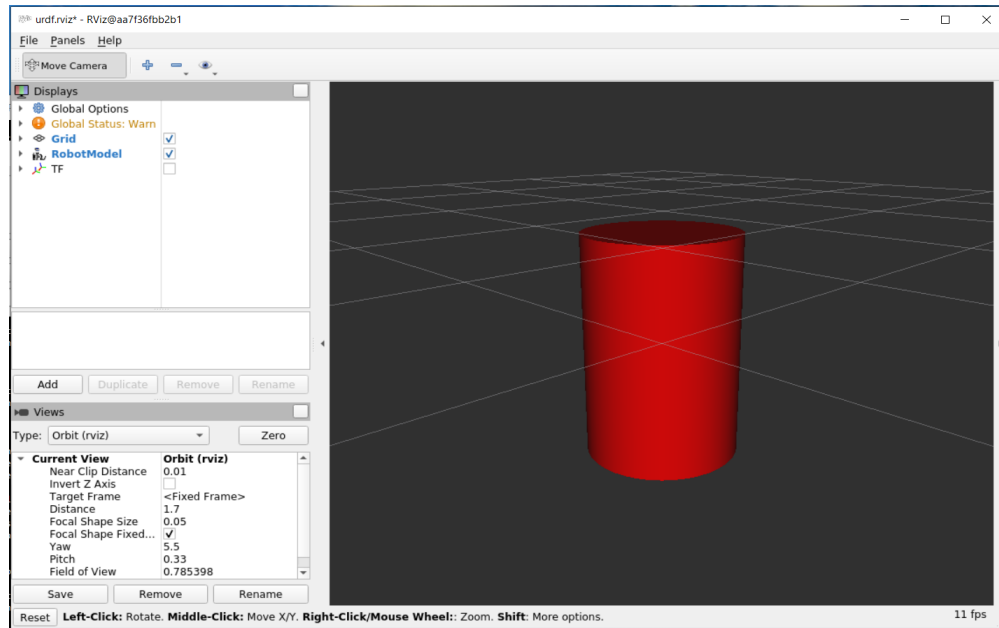


Figure 2: Rviz interface by running “roslaunch urdf_tutorial display.launch”.

- `source install/setup.bash` #source your workspace
- `ros2 launch urdf_tutorial display.launch.py` (you may need to install the urdf-launch library: `apt install ros-jazzy-urdf-launch`)

The second command will launch the `display.launch` file in the `urdf_tutorial` package in your workspace. You should see the Rviz interface as shown in Figure 2.

(2.5) Visualizing Fetch robot model. Create a new launch file named `display_fetch.launch.py` in `urdf_tutorial/launch` by copying the `display.launch.py` file.

Modify the content of the new launch file to display the Fetch robot. The definition of the robot is the urdf file under `fetch_description/robots/fetch.urdf`. Run the following command for testing:

- `ros2 launch urdf_tutorial display_fetch.launch.py`

You should see the visualization as in Figure 3. You need to rebuild your ROS workspace after adding the new launch file.

(2.6) Tuning the robot joints. Use the `/joint_state_publisher` window to set the following joints of the robot arm to the corresponding values:

- `shoulder_pan_joint`: -1.32
- `shoulder_lift_joint`: 0.7
- `upperarm_roll_joint`: 0.0
- `elbow_flex_joint`: -2.01
- `forearm_roll_joint`: 0.0

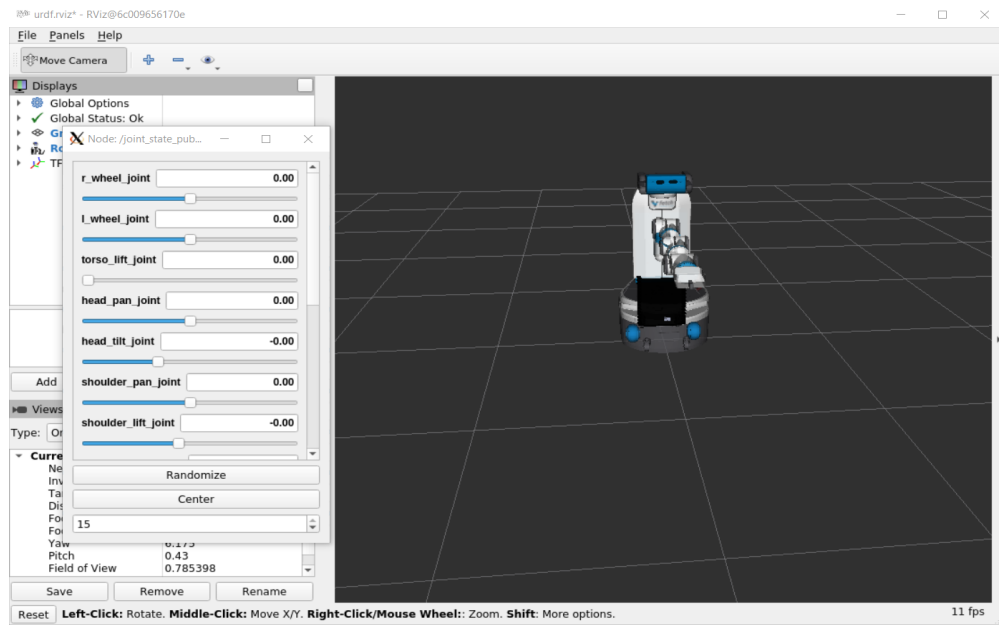


Figure 3: Rviz visualization of a Fetch mobile manipulator.

- wrist_flex_joint: -0.56
- wrist_roll_joint: 0.0

You can keep the other joint values as the default ones. Capture a screenshot of the robot in Rviz for this homework submission.

Submission guideline: Zip your src folder in your workspace and submit the zip file to eLearning. Make sure you have a screen capture included for (2.6). Our TA will build your workspace and run commands to verify it.