Robot Operating System (ROS)

Lab 6: URDF and Rviz

Haitham El-Hussieny, PhD

November 7, 2022

Department of Mechatronics and Robotics Engineering
Egypt-Japan University of Science and Technology (E-JUST)
Alexandria, Egypt.

Outline

ROS Visualization (Rviz)

ROS Visualization (Rviz)

Rviz

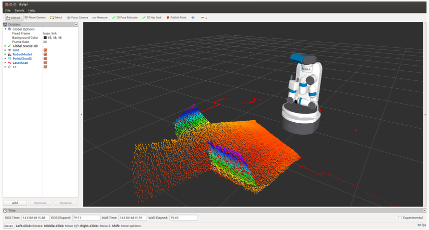
Rviz is a powerful 3D visualization tool for ROS. It allows the user to:

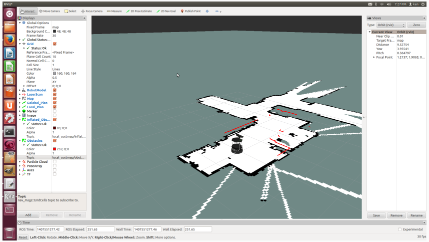
- view the simulated robot model.
- log sensor information.
- Replay the logged sensor information. (**rosbag**).

To launch Rviz:

```
$ roscore
```

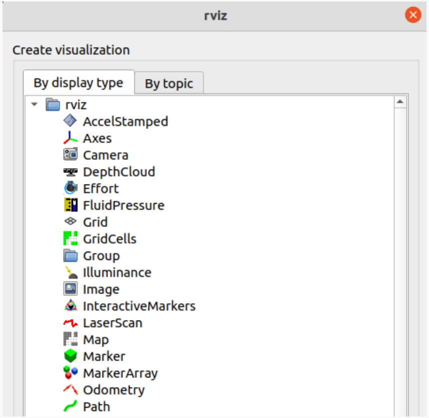
```
$ rviz
```





ROS Visualization (Rviz)

- On the left panel of the rviz main screen is the Displays panel, where the user can **add**, **remove**, or rename the visualization elements in the 3D environment.
- For further details on the display types, go to official ROS website.



Introduction to URDF

Introduction to URDF

URDF file

Unified Robotics Description Format, URDF, is an XML specification used in academia and industry to model multibody systems such as robotic manipulator arms or mobile robot.

Introduction to URDF

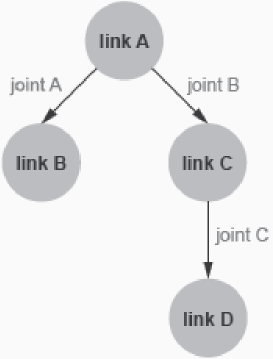
URDF file

Unified Robotics Description Format, URDF, is an XML specification used in academia and industry to model multibody systems such as robotic manipulator arms or mobile robot.

URDF Elements and Attributes

URDF files comprise various XML elements, such as <robot>, <link>, <joint>, nested in a tree.

```
<robot>
        <link>
        ...
        </link>
        <joint>
        ...
        </joint>
</robot>
```

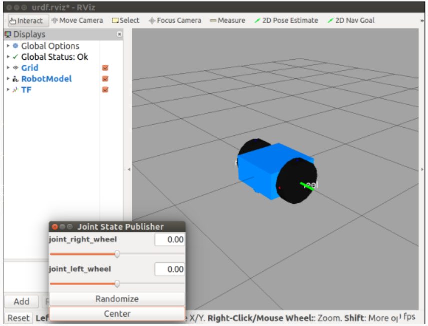


Building a differential drive robot URDF

Building a differential drive robot URDF

Objective

For our first robot model, we will build a URDF file for a two-wheeled differential drive robot. The model will be created incrementally, and we will view the results at each step in rviz. When our simple two-wheeled robot is complete, we will add Gazebo formatting and view the model in Gazebo.



Robot Model

Building a differential drive robot URDF

Step 1: Create a new ROS package.

- Create a new package in your own ROS workspace/src folder:

```
$ catkin_create_pgk diff_robot_gazebo
```

- Build the packages in the workspace:

```
$ cd ~/catkin_ws
$ catkin_make
```

- make a **urdf** folder inside the new created package:

```
$ mkdir urdf
```

Building a differential drive robot URDF

Step 2: Create a robot chassis.

- The robot URDF file is a set of **links** and **joints**.
- The link describes a rigid body and its physical properties:
  - dimensions.
  - position of its origin.
  - color.
- The joint describes the kinematic and dynamic properties:
  - links connected.
  - types of joint.
  - axis of rotation.
  - amount of friction and damping.

Building a differential drive robot URDF

Step 2: Create a robot chassis.

In the **urdf** folder, create a new file called diff_robot.urdf

```xml
<?xml version='1.0'?>
    <robot name="dd_robot">
    <!-- Base Link -->
        <link name="base_link">
            <visual>
                <origin xyz="0 0 0" rpy="0 0 0" />
                <geometry>
                    <box size="0.5 0.5 0.25"/>
                </geometry>
            </visual>
        </link>
    </robot>
```

Building a differential drive robot URDF

Step 2: Create a robot chassis.

```xml
<?xml version='1.0'?>
    <robot name="dd_robot">
    <!-- Base Link -->
        <link name="base_link">
            <visual>
                <origin xyz="0 0 0" rpy="0 0 0" />
                <geometry>
                    <box size="0.5 0.5 0.25"/>
                </geometry>
            </visual>
        </link>
    </robot>
```

Description:

- Robot name is dd_robot.
- The robot consists of one link called base_link.
- The link is a chassis box of width=0.5, depth=0.5 and height=0.25.
- The link is located at the origin without any orientation.

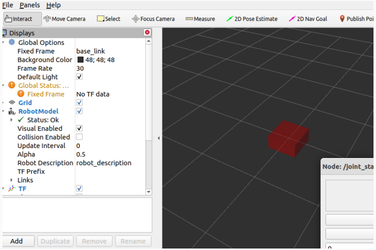
Building a differential drive robot URDF

Step 3: Display the created URDF file into Rviz.

```
$ roslaunch urdf_tutorial display.launch model:=`rospack find diff_robot_gazebo`/urdf/diff_robot.urdf gui:=true
```

roslaunch: launch many nodes at the same time.

rospack: return the path of a ROS package.



Building a differential drive robot URDF

Step 4: Adding wheels.

- Joints used to describe the relationship between the links.
- In URDF joint types: **Fixed, Revolute, Prismatic, Float, Planar, Continuous**.

```
<!-- Right Wheel -->
<link name="right_wheel">
    <visual>
        <origin xyz="0 0 0" rpy="1.570795 0 0" />
        <geometry>
            <cylinder length="0.1" radius="0.2" />
        </geometry>
    </visual>
</link>
<joint name="joint_right_wheel" type="continuous">
    <parent link="base_link"/>
    <child link="right_wheel"/>
    <origin xyz="0 -0.30 0" rpy="0 0 0" />
    <axis xyz="0 1 0" />
</joint>
```

- A continuous joint for the wheel is added to allow for continuous rotation.
- The wheel is a cylinder rotated $\pi/2$ radians around x.
- The joint has the base_link as a parent and the right_wheel as a child and 0.3 m away from it's parent on x-axis.

Building a differential drive robot URDF

Step 4: Adding wheels.

- Joints used to describe the relationship between the links.
- In URDF joint types: **Fixed, Revolute, Prismatic, Float, Planar, Continuous**.

```xml
<!-- Left Wheel -->
<link name="left_wheel">
    <visual>
        <origin xyz="0 0 0" rpy="1.570795 0 0" />
        <geometry>
            <cylinder length="0.1" radius="0.2" />
        </geometry>
    </visual>
</link>
<joint name="joint_left_wheel" type="continuous">
    <parent link="base_link"/>
    <child link="left_wheel"/>
    <origin xyz="0 0.30 0" rpy="0 0 0" />
    <axis xyz="0 1 0" />
</joint>
```

- A continuous joint for the wheel is added to allow for continuous rotation.
- The wheel is a cylinder rotated $\pi/2$ radians around x.
- The joint has the base_link as a parent and the left_wheel as a child and 0.3 m away from it's parent on x-axis.

BUILDING A DIFFERENTIAL DRIVE ROBOT URDF

Step 5: Adding a castor.

- The castor will only be a visual element added to the chassis and not a joint.
- The caster will slide along the ground plane as the robot's wheels move.

```xml
    <!-- Base Link -->
        <link name="base_link">
            <visual>
                <origin xyz="0 0 0" rpy="0 0 0" />
                <geometry>
                    <box size="0.5 0.5 0.25"/>
                </geometry>
            </visual>
            <!-- Caster -->
            <visual name="caster">
                <origin xyz="0.2 0 -0.125" rpy="0 0 0" />
                <geometry>
                    <sphere radius="0.05" />
                </geometry>
            </visual>
        </link>

        <!-- Right Wheel -->
```

Modified base_link

Building a differential drive robot URDF

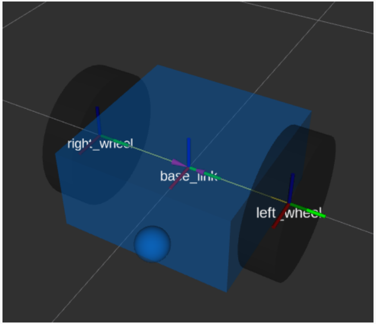
Step 6: Adding color.

Color is added with the **material** tag:

```
    <!-- Base Link -->
        <link name="base_link">
            <visual>

                <origin xyz="0 0 0" rpy="0 0 0" />
                <material name="blue">
                    <color rgba="0 0.5 1 1"/>
                </material>
                <geometry>
                    <box size="0.5 0.5 0.25"/>
                </geometry>
            </visual>
            <!-- Caster -->
```

Modified base_link



r: red, **g:** green, **b:** blue

a: alpha: transparency [0-1]

BUILDING A DIFFERENTIAL DRIVE ROBOT URDF

Step 7: Adding collision.

- A <collision> is added to each <link> to identify the boundaries of the object.
- If an object has complex visual shape, a simplified collision property should be defined.

```xml
    <!-- Base Link -->
        <link name="base_link">
            <visual>
                <origin xyz="0 0 0" rpy="0 0 0" />
                <material name="blue">
                    <color rgba="0 0.5 1 1"/>
                </material>
                <geometry>
                    <box size="0.5 0.5 0.25"/>
                </geometry>
            </visual>
            <!-- Base collision -->
            <collision>
                <origin xyz="0 0 0" rpy="0 0 0" />
                <geometry>
                    <box size="0.5 0.5 0.25"/>
                </geometry>
            </collision>

            <!-- Caster -->
```

Building a differential drive robot URDF

Step 7: Adding collision. (Castor)

```
            <!-- Caster -->
            <visual name="caster">
                <origin xyz="0.2 0 -0.125" rpy="0 0 0" />
                <geometry>
                    <sphere radius="0.05" />
                </geometry>
            </visual>
            <!-- Caster collision -->
            <collision>
                <origin xyz="0.2 0 -0.125" rpy="0 0 0" />
                <geometry>
                    <sphere radius="0.05" />
                </geometry>
            </collision>

        </link>

```

Building a differential drive robot URDF

Step 7: Adding collision. (Left wheel)

```
<!-- Left Wheel -->
<link name="left_wheel">
    <visual>
        <origin xyz="0 0 0" rpy="1.570795 0 0" />
        <material name="black"/>

        <geometry>
            <cylinder length="0.1" radius="0.2" />
        </geometry>
    </visual>
</link>
```

Building a differential drive robot URDF

Step 7: Adding collision. (Right wheel)

```
        <!-- Right Wheel -->
        <link name="right_wheel">
            <visual>
                <origin xyz="0 0 0" rpy="1.570795 0 0" />
                <material name="black">
                    <color rgba="0.05 0.05 0.05 1"/>
                </material>
                <geometry>
                    <cylinder length="0.1" radius="0.2" />
                </geometry>
            </visual>
            <!-- Right Wheel collision -->
            <collision>
                <origin xyz="0 0 0" rpy="1.570795 0 0" />
                <geometry>
                    <cylinder length="0.1" radius="0.2" />
                </geometry>
            </collision>
```

Building a differential drive robot URDF

Step 8: Adding Physical properties (mass and inertia).

The <inertial> tag could be used to add:

- **Mass:** This is the weight defined in kilograms.
- **inertia** : This frame is a 3×3 rotational inertia matrix.

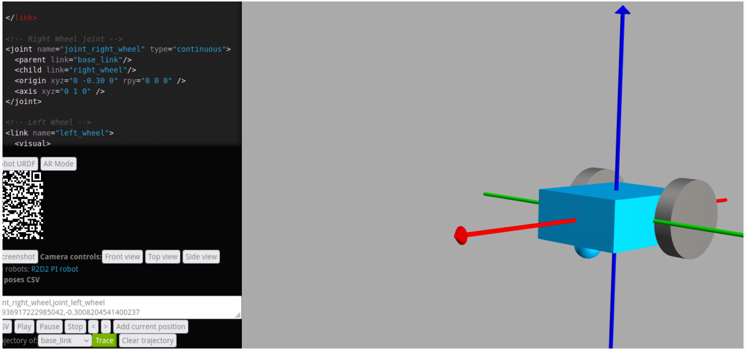
ixx	ixy	ixz
ixy	iyy	iyz
ixz	iyz	izz

```xml
<!-- Base Link -->
<link name="base_link">
  <visual>
    <origin xyz="0 0 0" rpy="0 0 0" />
    <geometry>
        <box size="0.5 0.5 0.25"/>
    </geometry>
    <material name="blue">
      <color rgba="0 0.5 1 1"/>
    </material>
  </visual>
  <!-- Base collision, mass and inertia -->
  <collision>
    <origin xyz="0 0 0" rpy="0 0 0" />
    <geometry>
        <box size="0.5 0.5 0.25"/>
    </geometry>
  </collision>
  <inertial>
    <mass value="5"/>
    <inertia ixx="0.13" ixy="0.0" ixz="0.0" iyy="0.21" iyz="0.0" izz="0.13"/>
  </inertial>
```

Building a differential drive robot URDF

Step 9: Visualization of the URDF file

- You can visulaize the URDF file using the **mymodelrobot** website.
- To import the URDF file into CoppeliaSim:
  1. Press **Modules**.
  2. Choose **Importers**.
  3. Press **URDF importer**.



Check the website here!

Building a differential drive robot URDF

Finally: Try this code in CoopeliaSim script

```
function sysCall_init()
    -- do some initialization here
        -- do some initialization here
    robot = sim.getObjectHandle('/my_robot')
    leftMotor = sim.getObjectHandle('/my_robot/joint_left_wheel')
    rightMotor = sim.getObjectHandle('/my_robot/joint_right_wheel')
end

function sysCall_actuation()
    -- put your actuation code here

    sim.setJointTargetVelocity(leftMotor,1)
    sim.setJointTargetVelocity(rightMotor,1)
end

function sysCall_sensing()
    -- put your sensing code here
end
```

End of Lecture

haitham.elhussieny@ejust.edu.eg