

# ENGR 4421: Robotics II

Mapping

03/01/2022



# Outline

- Simulate Sensors
- Sensor Fusion
- SLAM Toolbox

# **Mapping Tutorial**

[https://navigation.ros.org/setup\\_guides/sensors/setup\\_sensors.html](https://navigation.ros.org/setup_guides/sensors/setup_sensors.html)

# Create Lidar Link

```
<joint name="lidar_joint" type="fixed">
  <parent link="base_link" />
  <child link="lidar_link" />
  <origin xyz="0 0 ${base_height+lidar_height}/2+lidar_zoff" rpy="0 0 0" />
</joint>
<link name="lidar_link">
  <visual>
    <origin xyz="0 0 0" rpy="0 0 0" />
    <geometry>
      <cylinder radius="${lidar_radius}" length="${lidar_height}" />
    </geometry>
    <material name="black" />
  </visual>
  <collision>
    <origin xyz="0 0 0" rpy="0 0 0" />
    <geometry>
      <cylinder radius="${lidar_radius}" length="${lidar_height}" />
    </geometry>
  </collision>
  <xacro:cylinder_inertia m="${lidar_mass}" r="${lidar_radius}" h="${lidar_height}" />
</link>
```

# Create IMU Link

```
<joint name="imu_joint" type="fixed">
  <parent link="base_link"/>
  <child link="imu_link"/>
  <origin xyz="${base_width}/4 0 ${base_height+imu_height}/2"/>
</joint>
<link name="imu_link">
  <visual>
    <geometry>
      <box size="${imu_width} ${imu_depth} ${imu_height}"/>
    </geometry>
    <material name="purple" />
  </visual>
  <collision>
    <geometry>
      <box size="${imu_width} ${imu_depth} ${imu_height}"/>
    </geometry>
  </collision>
  <xacro:box_inertia m="${imu_mass}" w="${imu_width}" d="${imu_depth}" h="${imu_height}" />
</link>
```

# Add Gazebo Plugins

- Plugin for Lidar: [https://github.com/ros-simulation/gazebo\\_ros\\_pkgs/wiki/ROS-2-Migration:-Ray-sensors](https://github.com/ros-simulation/gazebo_ros_pkgs/wiki/ROS-2-Migration:-Ray-sensors)
- Plugin for IMU: [https://github.com/ros-simulation/gazebo\\_ros\\_pkgs/wiki/ROS-2-Migration:-IMU-Sensors](https://github.com/ros-simulation/gazebo_ros_pkgs/wiki/ROS-2-Migration:-IMU-Sensors)
- Plugin for (depth) camera: [https://github.com/ros-simulation/gazebo\\_ros\\_pkgs/wiki/ROS-2-Migration:-Camera](https://github.com/ros-simulation/gazebo_ros_pkgs/wiki/ROS-2-Migration:-Camera)

# Create a Package

```
cd ~/<ros workspace>/src # go to `src/` in your ros workspace
ros2 pkg create --build-type ament_python <mapping package dir> # create a package
cd <mapping package dir> # go to your package
mkdir launch config # create useful directories
code .
# bring up VSCode and open the <mapping package directory> \
```

# Edit Package

- Download ekf configuration file

```
cd <mapping package dir>/config
```

```
wget https://raw.githubusercontent.com/linzhangUCA/mapping_demo/master/configs/ekf.yaml
```

- Create launch file

```
cd <mapping package dir>/launch
```

```
wget https://raw.githubusercontent.com/linzhangUCA/mapping_demo/master/launch/create_map.launch.py
```



# Extras in Launch

```
robot_localization_node = Node(  
    package="robot_localization",  
    executable="ekf_node",  
    name="ekf_filter_node",  
    output="screen",  
    parameters=[  
        str(mapping_package_path / "configs/ekf.yaml"),  
        {"use_sim_time": LaunchConfiguration("use_sim_time")},  
    ],  
)  
  
launch_slam = IncludeLaunchDescription(  
    PythonLaunchDescriptionSource(  
        str(slam_toolbox_package_path / "launch/online_async_launch.py")  
    )  
)
```

# Update Package setup.py

```
import os
from glob import glob
from setuptools import setup

package_name = '<urdf package dir>'

setup(
    ...
    data_files=[
        ...
        ('share/' + package_name, ['package.xml']),
        (os.path.join('share', package_name, 'launch'), glob(os.path.join('launch', '*.launch.py'))),
        (os.path.join('share', package_name, 'config'), glob(os.path.join('config', '*.yaml'))),
    ],
    ...
)
```

# Launch Mapping

- Build package and launch.

```
cd ~/<ros workspace> # e.g. cd ~/tutorial_ws
rosdep install -i --from-path src --rosdistro galactic -y
colcon build
source install/setup.bash
ros2 launch <mapping package dir> create_map.launch.py
```