Lab Activity: Lidar test

Marc Font, Laia Raja and Alba Navarro

1. Changes in LIDAR angles

The Lidar specifications in the URDF file `rubot_mecanum_lidar.urdf` are in the *Laser Distance Sensor YDLIDAR X4 controller* section. Our lidar sensor is a RPLIDAR A1 with the following specifications:

- angle_min: -3.141593 (rad)
- angle_max: 3.141593 (rad)
- angle_increment: 0.008727 (rad) -> 720 laser beams

In this section we changed angle_min from 0 rad to $-\pi$ rad (-3.141593 rad) and angle_max from -2π rad to π rad (3.141593 rad). The angle_increment is defined by the number of laser beams which is set to 760 samples. We didn't change this parameter in the urdf file.

```xml
<!-- Laser Distance Sensor YDLIDAR X4 controller-->
<gazebo reference="base_scan">
  <sensor name="lidar" type="ray">
  <pose>0.215 0 0.215 0 0 0</pose>
  <always_on>true</always_on>
  <visualize>false</visualize>
  <update_rate>5</update_rate>
  <ray>
    <scan>
      <horizontal>
        <samples>720</samples> <!-- = laser beams -> angle_increment = 0.008727 rad -->
        <resolution>1.00000</resolution>
        <min_angle>-3.141593</min_angle> <!-- min angle changed (from 0 rad to -pi)-->
        <max_angle>3.141593</max_angle> <!-- max angle changed (from 2pi rad to pi)-->
      </horizontal>
    </scan>
    <range>
      <min>0.120000</min>
      <max>3.5</max>
      <resolution>0.015000</resolution>
    </range>
    <noise>
      <type>gaussian</type>
      <mean>0.0</mean>
      <stddev>0.01</stddev>
    </noise>
  </ray>
```

We launch the bringup with the new `rubot_mecanum_lidar.urdf` to see robot the Gazebo: ```shell ros2 launch my_robot_bringup my_robot_bringup_sw.launch.xml x0:=1.0 y0:=-0.5 yaw0:=0.0 rubot:=rubot/rubot_mecanum_lidar.urdf custom_world:=square3m_walls.world ``` Here is the Gazebo bringup where we can see the robot in x=1, y=-0.5 i yaw=0
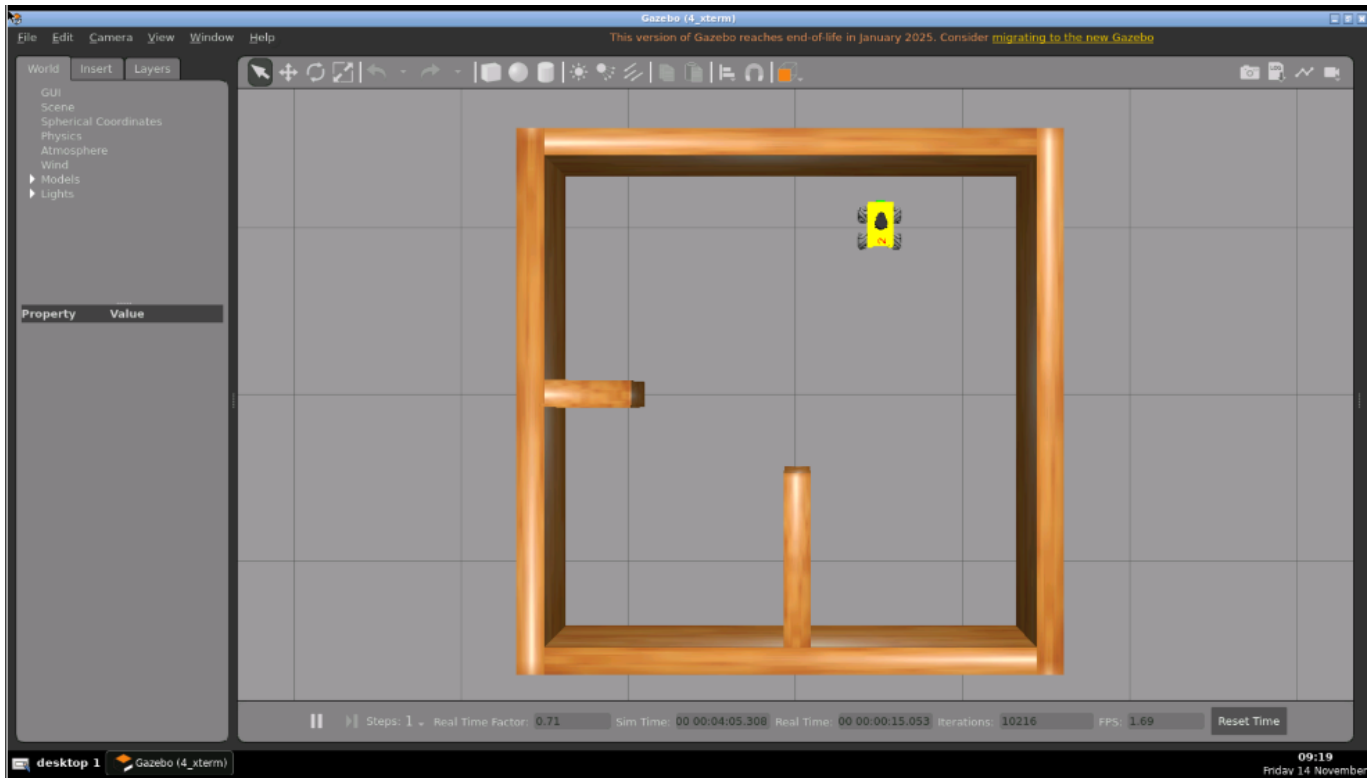
2. Detect the distance and orientation of the nearest object

We created the following new files to verify proper Lidar readings:

- New launch file name: `custom_rubot_lidar_test.launch.xml`
- New python file name: `custom_rubot_lidar_test.py`

The launch file is:

```xml
1    <?xml version="1.0"?>
2  ∨ <launch>
3        <!-- Custom LIDAR Test -->
4  ∨    <node
5          pkg="my_robot_control"
6          exec="custom_rubot_lidar_test_exec"
7          name="custom_lidar_test_node"
8          output="screen">
9        </node>
10   </launch>
```

We had to update the `setup.py` file in `my_robot_control` by adding: ```shell 'custom_rubot_lidar_test_exec = my_robot_control.custom_rubot_lidar_test:main' ```

The `custom_rubot_lidar_test.py` is the file `my_robot_lidar_test.py` with modifications to obtain the correct distances and angles from the lidar. The original version does not take into account that the lidar is oriented backwards (with a 180º rotation). To correct this in `custom_rubot_lidar_test.py`, we added a variable called `angle_offset_deg = 180.0`. We also added the `wrap_angle` function, which always returns angles between -180 and 180. This way, when we apply the wrap_angle function to the measured angle and

add the 180º offset, we obtain the degrees deviated from the front position of the robot (not the lidar). We created the `angle_to_index` function that relates the distance to the angle as the original code did manually. Every second, the programme measures the distance of the robot from the objects around it in relation to the front, right and left of the robot. The programme saves the smallest distance to identify the closest object (the wall it will hit first). The programme prints the distance of the closest object and its direction in relation to the front of the robot. Positive values are to the right and negative values are to the left.

When we launch `custom_rubot_lidar_test.launch.xml` as follows we can see these changes `shell ros2 launch my_robot_control custom_rubot_lidar_test.launch.xml`