SC-635 Advanced Topics in Mobile Robotics

Experiment Module : Range sensing and collision avoidance

January 31, 2020

Systems and Control Engineering

Indian Institute of Technology Bombay

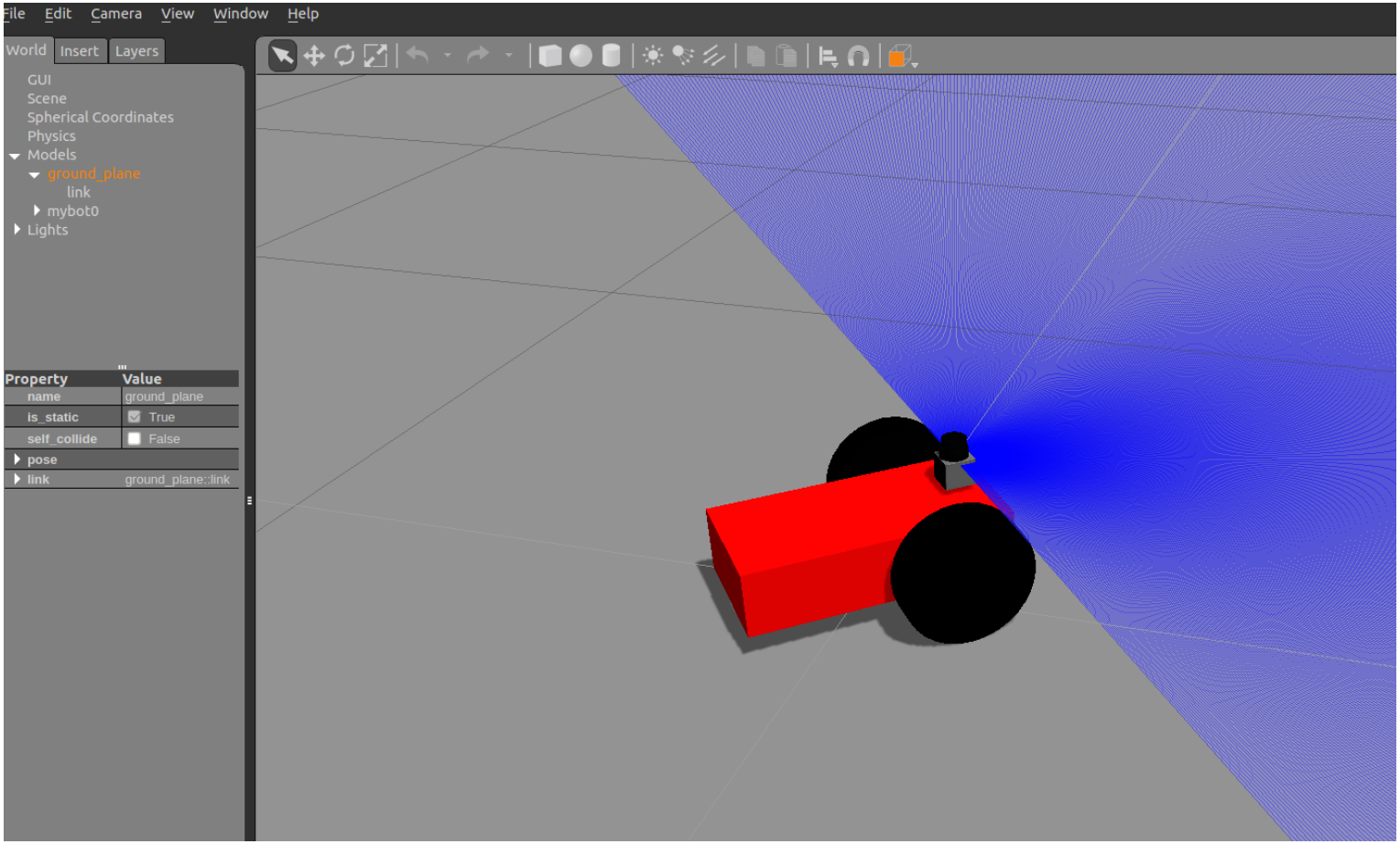
Overview

1. Sensing in gazebo

Recapitulation

- Topics
- Nodes
- Messages
- Launch File
- Passing arguments to nodes from launch file
- Gazebo simulation environment
- Using /odom topic
- Waypoint Navigation

Snapshot


File Edit Camera View Window Help
World Insert Layers
GUI
Scene
Spherical Coordinates
Physics
Models
ground_plane
link
mybot0
Lights
Property Value
name ground_plane
is_static True
self_collide False
pose
link ground_plane::link

Plugins

Gazebo drivers to extract information and publish over topics

- Model : Robot URDF
  - Links : Physical entity or solids
  - Joints : connection between two links
  - Collision properties : Bounding box etc
  - Sensors

Plugins

Gazebo drivers to extract information and publish over topics

- Model : Robot URDF
  - Links : Physical entity or solids
  - Joints : connection between two links
  - Collision properties : Bounding box etc
  - Sensors
- Sensor : IMU, Range sensors, Camera, Kinect

Plugins

Gazebo drivers to extract information and publish over topics

- Model : Robot URDF
  - Links : Physical entity or solids
  - Joints : connection between two links
  - Collision properties : Bounding box etc
  - Sensors
- Sensor : IMU, Range sensors, Camera, Kinect
- Visual : To load visual elements such as video / markers

Plugins

Gazebo drivers to extract information and publish over topics

- Model : Robot URDF
  - Links : Physical entity or solids
  - Joints : connection between two links
  - Collision properties : Bounding box etc
  - Sensors
- Sensor : IMU, Range sensors, Camera, Kinect
- Visual : To load visual elements such as video / markers

find more information at

http://gazebosim.org/tutorials?tut=ros_gzplugins

Model : Differential Drive robot

URDF model in xacro

```
<?xml version='1.0'?>
<robot name="myrobot" xmlns:xacro="http://www.ros.org/wiki/xacro">

  <xacro:include filename="$(find mybotlaser_description)/urdf/mybot.gazebo" />

  <link name='chassis'>
    <inertial>
      ...
    </inertial>
    <collision name='collision'>
      ...
    </collision>
    <visual name='chassis_visual'>
      ...
    </visual>
    <collision name='caster_collision'>
      ...
    </collision>

    <visual name='caster_visual'>
      ...
    </visual>
  </link>

  <link name="left_wheel">
    <inertial>
      ...
    </inertial>
    <collision name="collision">
      ...
    </collision>
```

Model : Differential Drive robot (continued)

```
32      <visual name="left_wheel_visual">
33        ...
34      </visual>
35    </link>
36
37    <link name="right_wheel">
38      ...
39    </link>
40    <link name="hokuyo">
41      ...
42    </link>
43    <joint type="continuous" name="left_wheel_hinge">
44      ...
45    </joint>
46    <joint type="continuous" name="right_wheel_hinge">
47      ...
48    </joint>
49    <joint name="hokuyo_joint" type="fixed">
50      ...
51    </joint>
52
53  </robot>
```

2D range sensor

Hokuyo 2D range scanner : Properties

```
<gazebo reference="hokuyo">
    <sensor type="ray" name="head_hokuyo_sensor">
        <pose>0 0 0 0 0 0</pose>
        <visualize>true</visualize>
        <update_rate>40</update_rate>
        <ray>
            <scan>
                <horizontal>
                    <samples>720</samples>
                    <resolution>1</resolution>
                    <min_angle>-1.570796</min_angle>
                    <max_angle>1.570796</max_angle>
                </horizontal>
            </scan>
          <range>
              <min>0.1</min>
              <max>30.0</max>
              <resolution>0.01</resolution>
          </range>
          <noise>
              <type>gaussian</type>
              <mean>0.0</mean>
              <stddev>0.01</stddev>
          </noise>
        </ray>
        <plugin name="gazebo_ros_head_hokuyo_controller" filename="
      libgazebo_ros_laser.so">
            <topicName>laser/scan</topicName>
            <frameName>hokuyo</frameName>
        </plugin>
    </sensor>
</gazebo>
```

Project with robot URDF

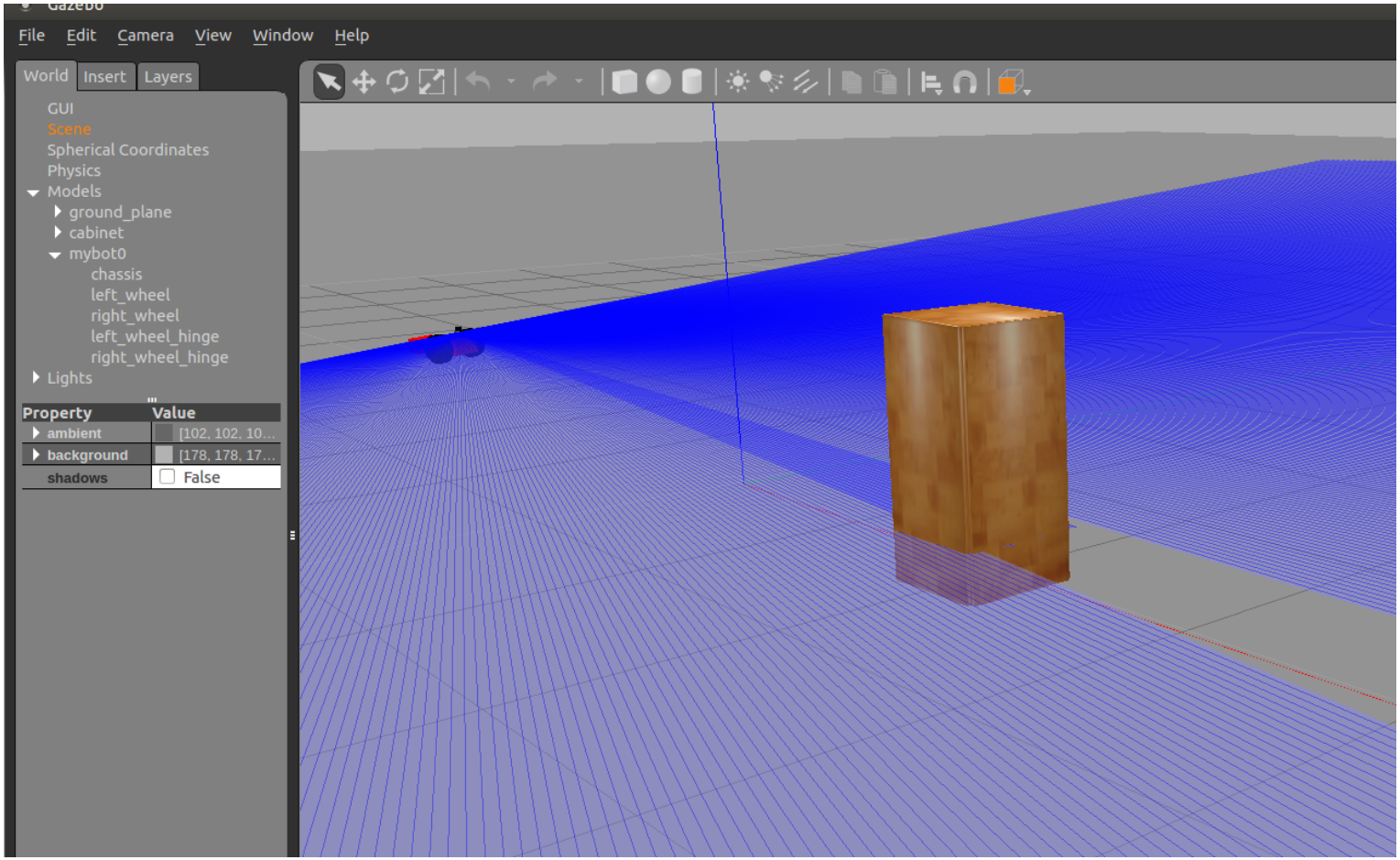
To launch the project

```
$ roslaunch template_a3 bot_in_world.launch
```

Download the project at :
http://bit.ly/2tdpTemplateA3

This project contains two separate projects named bot_description and template_a3

Snapshot


File
Edit
Camera
View
Window
Help
World
Insert
Layers
GUI
Scene
Spherical Coordinates
Physics
Models
ground_plane
cabinet
mybot0
chassis
left_wheel
right_wheel
left_wheel_hinge
right_wheel_hinge
Lights
Property
Value
ambient
[102, 102, 10...
background
[178, 178, 17...
shadows
False

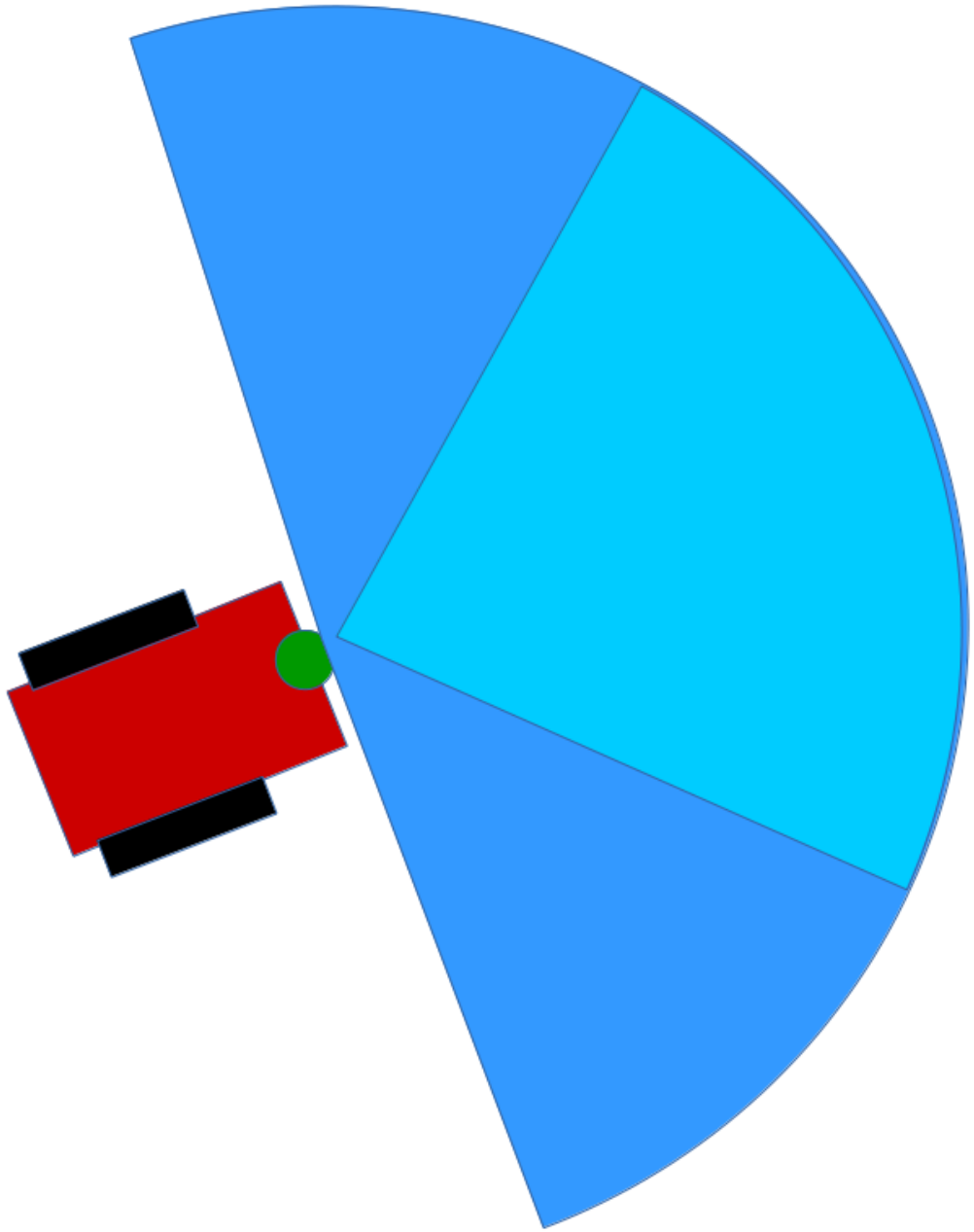
Topics

- */bot_0/odom*
- */bot_0/cmd_vel*
- */bot_0/laser/scan*

laser scan data

```
---
header:
  seq: 388
  stamp:
    secs: 1452
    nsecs:  78000000
  frame_id: "bot_0/hokuyo"
angle_min: -1.57079994678
angle_max: 1.57079994678
angle_increment: 0.00436940183863
time_increment: 0.0
scan_time: 0.0
range_min: 0.10000000149
range_max: 30.0
ranges: [inf, inf, ... inf, inf, 6.745013236999512, 6.7742509841918945,
      6.858757495880127, 6.950172424316406, 7.034490585327148, 7.132570743560791,
       7.150918960571289, 7.137244701385498, 7.134756565093994,
      7.109353542327881, 7.093964099884033, 7.113735198974609, 7.081282138824463,
       7.077101230621338, 7.062750816345215, 7.056343078613281,
      6.588943004608154, 6.707895755767822, 6.8123393058776855,
      6.905351161956787, inf, inf, inf, ... inf, inf, inf]
intensities: [0.0, 0.0, 0.0, ... 0.0, 0.0, 1.6619910980572116e-37,
      1.6619910980572116e-37, 1.6619910980572116e-37, 1.6619910980572116e-37,
      1.6619910980572116e-37, 1.6619910980572116e-37, 1.661822493825984e-37,
      1.661822493825984e-37, 1.661822493825984e-37, 1.661822493825984e-37,
      1.661822493825984e-37, 1.661822493825984e-37, 1.661822493825984e-37,
      1.661822493825984e-37, 1.661822493825984e-37, 1.661822493825984e-37,
      1.8521688029792693e+28, 1.8521688029792693e+28, 1.8521688029792693e+28,
      1.8521688029792693e+28, 0.0, 0.0, ... 0.0, 0.0, 0.0]
```

Simplified obstacle data



Callback for simplified laser data

```
1  def laser_callback(msg):
2      global range_data
3      data=msg.ranges
4      range_data=[min(data[480:719]),min(data[240:479]),
       min(data[0:239])]
```

Assignment

Use the provided template project and implement following

The robot loads at (-5, 0). An obstacle (a cabinet) is placed at (2,0) in Gazebo world. Your objective is to drive the robot towards goal at (5,0) using following algorithms:

- Bug 0
- Bug 1
- Bug 2

The template project is located at :

http://bit.ly/2tdpTemplateA3

Thank you