
EE405A

Gazebo Simulator

(TA) Gyuree Kang
School of Electrical Engineering
KAIST

October 30, 2022

fingb20@kaist.ac.kr

Gazebo

What is Gazebo?

- Gazebo is an open source 3D robotics simulator with high-performance physics engines.

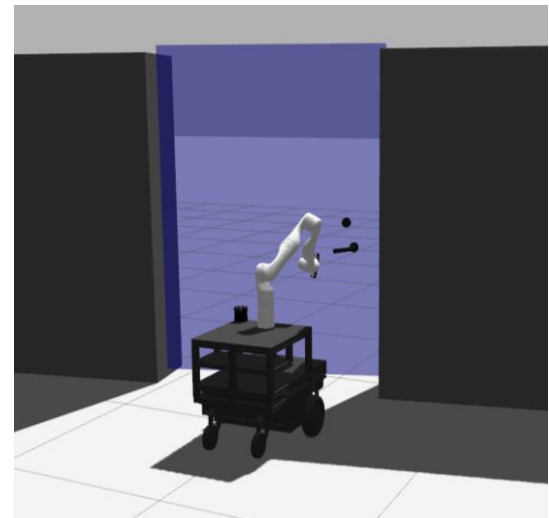
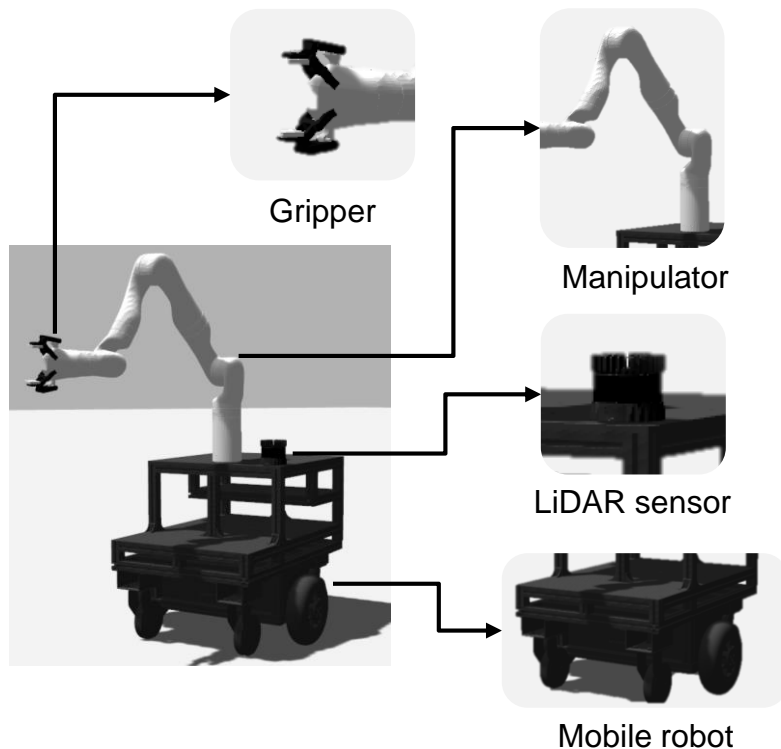


Tutorials: <https://classic.gazebosim.org/>

Gazebo

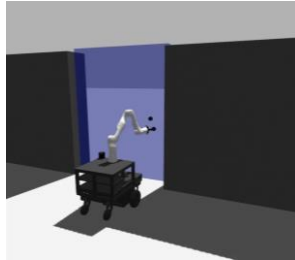
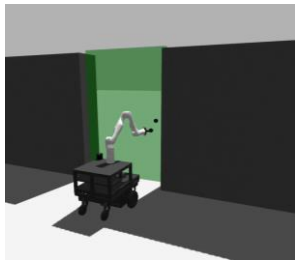
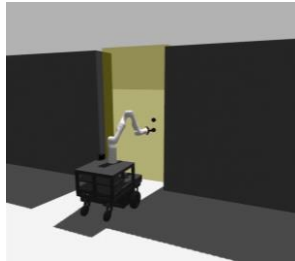
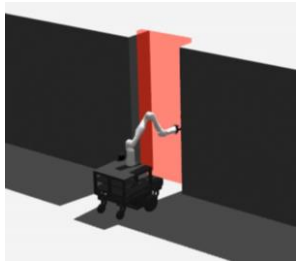
What can we do with Gazebo?

- Build and control your own robot model with actuator controllers.
- Obtain sensor data with simple plugins (2D/3D LiDAR, camera, force-torque sensor, contact sensor, IMU, etc.).
- Visualize robot and the environment (world).



Robot and surrounding environment

Gazebo



TEAM SINGABOAT-VRX

 NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

| | | | |
|-------------------------------------|---|--------------------------------------|-------------------------------------|
| Tanmay Samak Team Captain | Chinmay Samak Team Vice-Captain | Chern Peng Lee Team Member | Dr. Ming Xie Team Advisor |
|-------------------------------------|---|--------------------------------------|-------------------------------------|

Team Email: singaboat.vrx@gmail.com

Team SINGABOAT-VRX @ NTU | Virtual RobotX (VRX) Competition 2022 | Task Descriptions

Gazebo

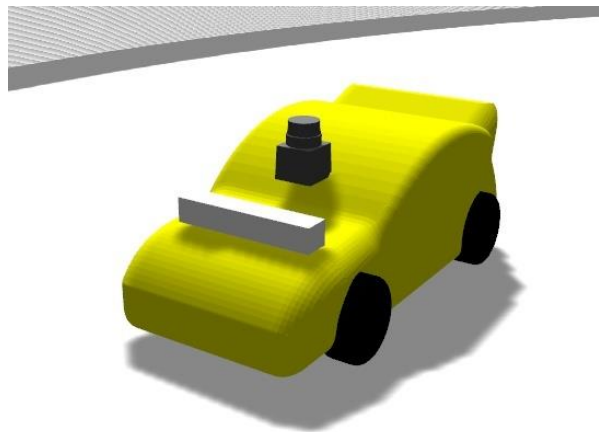
Goal

- Build a custom robot model.
- Make a simulation world to deploy the robot.
- Obtain sensor data from the simulator.
- Control the robot through external controller.

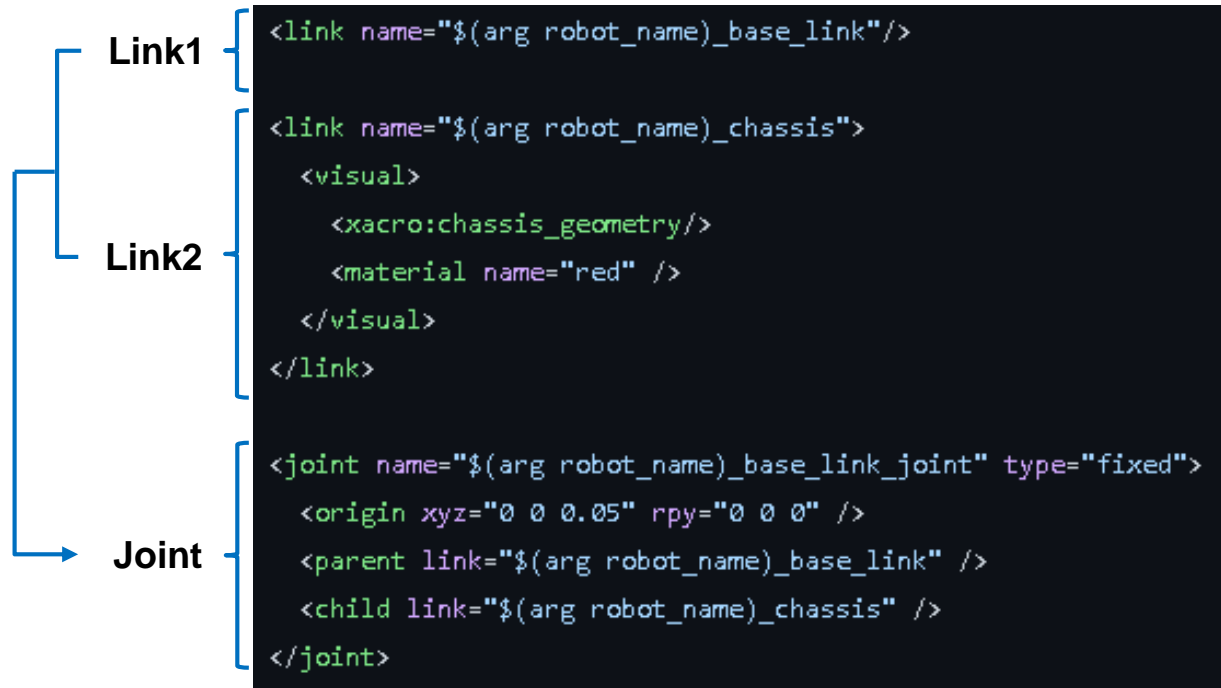
Download code: <https://github.com/Guri-cccc/EE405A-2022-F1-simulation.git>

Car model

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)



Car model



Link definitions and joint relationship

Bring up mesh

```
<xacro:macro name="chassis_geometry">
  <origin xyz="0.46 -0.135 -0.180" rpy="1.57 0 3.14" />
  <geometry>
    <mesh filename="package://f1tenth-sim/urdf/meshes/chassis.stl" scale="0.000990 0.0009375 0.0009375"/>
  </geometry>
</xacro:macro>
```

Link parameters

Car model (links)

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

Collision mesh

Visual mesh

```
<link name="$(arg robot_name)_laser_link">
  <inertial>
    <xacro:hokuyo_inertial_params />
  </inertial>
  <collision>
    <origin xyz="0 0 0" rpy="0 0 0"/>
    <geometry>
      <box size="0.1 0.1 0.1"/>
    </geometry>
  </collision>
  <visual>
    <origin xyz="0 0 0" rpy="0 0 0"/>
    <geometry>
      <mesh filename="package://f1tenth-sim/urdf/meshes/hokuyo.stl"/>
      <material name="grey" />
    </geometry>
  </visual>
</link>
```

Link definition

Car model (joints)

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

Joint types

- Fixed
- Revolute:
 - rotation with limit
- Continuous:
 - rotation without limit
- Prismatic
- Floating:
 - joint with 6 degrees of freedom
- Planar

```
<joint name="$(arg robot_name)_base_link_joint" type="fixed">  
  <origin xyz="0 0 0.05" rpy="0 0 0" />  
  <parent link="$(arg robot_name)_base_link" />  
  <child link="$(arg robot_name)_chassis" />  
</joint>
```

Fixed joint

```
<joint name="$(arg robot_name)_left_front_wheel_joint" type="continuous">  
  <origin xyz="0 0 0" rpy="1.5708 0 0" />  
  <parent link="$(arg robot_name)_left_steering_hinge" />  
  <child link="$(arg robot_name)_left_front_wheel" />  
  <axis xyz="0 0 -1" />  
  <limit effort="10" velocity="100" />  
</joint>
```

Continuous joint

```
<joint name="$(arg robot_name)_left_steering_hinge_joint" type="revolute">  
  <origin xyz="0.325 0.1 0" rpy="0 1.5708 0" />  
  <parent link="$(arg robot_name)_chassis" />  
  <child link="$(arg robot_name)_left_steering_hinge" />  
  <axis xyz="-1 0 0" />  
  <limit lower="-1.0" upper="1.0" effort="10" velocity="100" />  
</joint>
```

Revolute joint

Car model (joints)

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

```
<xacro:macro name="steering_hinge_transmission" params="name">
  <transmission name="${name}_transmission" type="SimpleTransmission">
    <type>transmission_interface/SimpleTransmission</type>
    <joint name="${name}_joint">
      <hardwareInterface>hardware_interface/EffortJointInterface</hardwareInterface>
    </joint>
    <actuator name="${name}_motor">
      <hardwareInterface>hardware_interface/EffortJointInterface</hardwareInterface>
      <mechanicalReduction>1</mechanicalReduction>
    </actuator>
  </transmission>
</xacro:macro>
```

Revolute joint transmission

```
<xacro:macro name="wheel_transmission" params="name">
  <transmission name="${name}_transmission" type="SimpleTransmission">
    <type>transmission_interface/SimpleTransmission</type>
    <joint name="${name}_joint">
      <hardwareInterface>hardware_interface/EffortJointInterface</hardwareInterface>
    </joint>
    <actuator name="${name}_motor">
      <hardwareInterface>hardware_interface/EffortJointInterface</hardwareInterface>
      <mechanicalReduction>1</mechanicalReduction>
    </actuator>
  </transmission>
</xacro:macro>
```

Continuous joint transmission

EE 405A

Sensor plugins (camera)

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

Define sensor link → `<gazebo reference="$(arg robot_name)_camera_lense_link">`

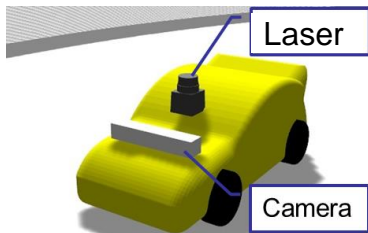
Define sensor type and name → `<sensor type="camera" name="$(arg robot_name)_left_sensor">`

Define update rate (hz) → `<update_rate>30.0</update_rate>`

Define sensor properties

Add noise to sensor data

```
<xacro:unless value="$(optenv DISABLE_GAZEBO_CAMERA false)">
  <gazebo reference="$(arg robot_name)_camera_lense_link">
  <sensor type="camera" name="$(arg robot_name)_left_sensor">
    <update_rate>30.0</update_rate>
    <!-- math.atan(320 / 687.8065795898438) * 2 -->
    <camera name="front_camera">
      <!-- horizontal_fov>0.8709216071359963</horizontal_fov -->
      <horizontal_fov>1.74</horizontal_fov>
      <image>
        <width>1920</width>
        <height>1080</height>
        <format>B8G8R8</format>
      </image>
      <clip>
        <near>0.02</near>
        <far>300</far>
      </clip>
      <noise>
        <type>gaussian</type>
        <mean>0.0</mean>
        <stddev>0.007</stddev>
      </noise>
    </camera>
  </sensor>
</unless>
```



Sensor configuration

Camera properties

Sensor plugins (camera)

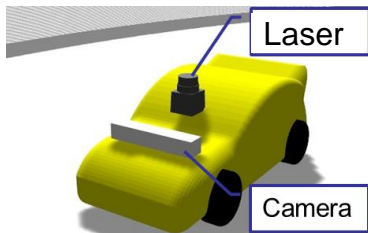
File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

Camera topic update rate(hz)

Define camera topic names

```
<plugin name="camera_controller" filename="libgazebo_ros_camera.so">
  <alwaysOn>true</alwaysOn>
  <updateRate>30.0</updateRate>
  <cameraName>/$(arg robot_name)/camera</cameraName>
  <imageTopicName>image</imageTopicName>
  <cameraInfoTopicName>rgb/camera_info</cameraInfoTopicName>
  <frameName>$(arg robot_name)_camera_lense_link</frameName>
  <hackBaseline>0</hackBaseline><!-- set this to 0.12 for the second camera -->
  <distortionK1>0.0</distortionK1>
  <distortionK2>0.0</distortionK2>
  <distortionK3>0.0</distortionK3>
  <distortionT1>0.0</distortionT1>
  <distortionT2>0.0</distortionT2>
</plugin>
</sensor>
</gazebo>
</xacro:unless>
```

Camera plugin



Sensor configuration

Sensor plugins (laser)

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

Define sensor link

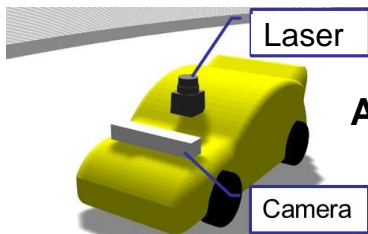
Define sensor type and name

Define update rate (hz)

Define sensor properties

Add noise to sensor data

```
<xacro:unless value="$(optenv DISABLE_GAZEBO_LASER false)">
  <gazebo reference="$(arg robot_name)_laser_link">
    <material>Gazebo/Grey</material>
  </gazebo>
  <sensor type="ray" name="$(arg robot_name)_hokuyo_sensor">
    <pose>0 0 0.0124 0 0 0</pose>
    <visualize>>false</visualize>
    <update_rate>40</update_rate>
    <ray>
      <scan>
        <horizontal>
          <samples>1081</samples>
          <resolution>1</resolution>
          <min_angle>-2.3561944902</min_angle>
          <max_angle>2.3561944902</max_angle>
        </horizontal>
      </scan>
      <range>
        <min>0.1</min>
        <max>30.0</max>
        <resolution>0.01</resolution>
      </range>
      <noise>
        <mean>0.0</mean>
        <stddev>0.01</stddev>
      </noise>
    </ray>
  </sensor>
</xacro:unless>
```



Sensor configuration

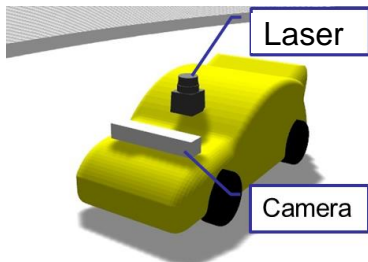
Sensor plugins (laser)

File path: [EE405A-2022-F1-simulation/f1tenth-sim/urdf/macros.xacro](#)

Define camera topic names

```
<plugin name="gazebo_ros_hokuyo_controller" filename="libgazebo_ros_laser.so">
  <topicName>/$(arg robot_name)/scan</topicName>
  <frameName>$(arg robot_name)_laser_link</frameName>
</plugin>
</sensor>
</gazebo>
</xacro:unless>
```

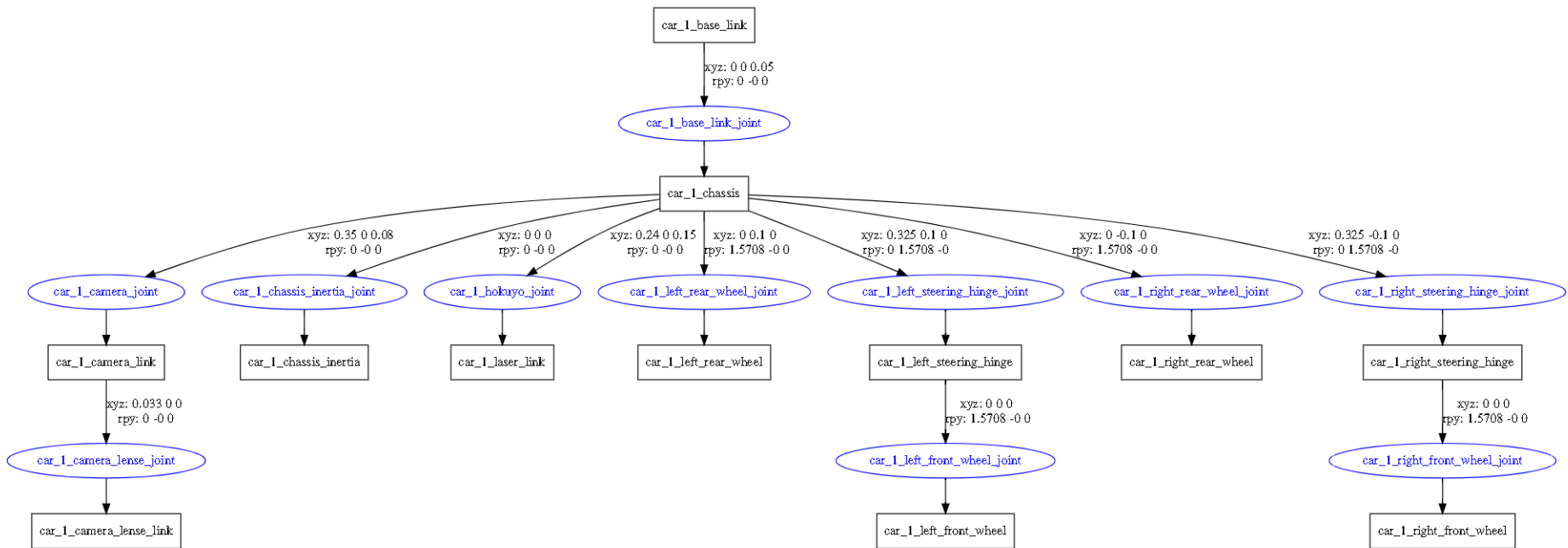
Laser plugin



Sensor configuration

Car model

- Convert xacro to urdf:
roslaunch xacro xacro.py macros.xacro > macros.urdf
- Draw graphviz diagram of urdf:
urdf_to_graphviz macros.urdf



Graphviz diagram of the car model

World

File path: [EE405A-2022-F1-simulation/f1tenth-sim/world/oval_track.world](#)

Model center pose

Model collision mesh

```
<model name='oval_track'> <!-- race_track -->
  <pose frame=''>-39.47 -22.7 0 0 -0 0</pose> <!-- -39.47
  <static>1</static>
  <link name='body'>
    <collision name='collision'>
      <geometry>
        <mesh>
          <uri>model://oval_track/oval_track.stl</uri>
          <scale>0.003 0.0025 0.001</scale>
        </mesh>
      </geometry>
      <max_contacts>10</max_contacts>
      <surface>
        <contact>
          <ode/>
        </contact>
        <bounce/>
        <friction>
          <torsional>
            <ode/>
          </torsional>
          <ode/>
        </friction>
      </surface>
    </collision>
```

Model collision definition

World

File path: [EE405A-2022-F1-simulation/f1tenth-sim/world/oval_track.world](#)

Model visual mesh

```
<visual name='visual'>
  <geometry>
    <mesh>
      <uri>model://oval_track/oval_track.stl</uri>
      <scale>0.003 0.0025 0.001</scale>
    </mesh>
  </geometry>
</visual>
<self_collide>0</self_collide>
<enable_wind>0</enable_wind>
<kinematic>0</kinematic>
</link>
</model>
```

Model visual definition

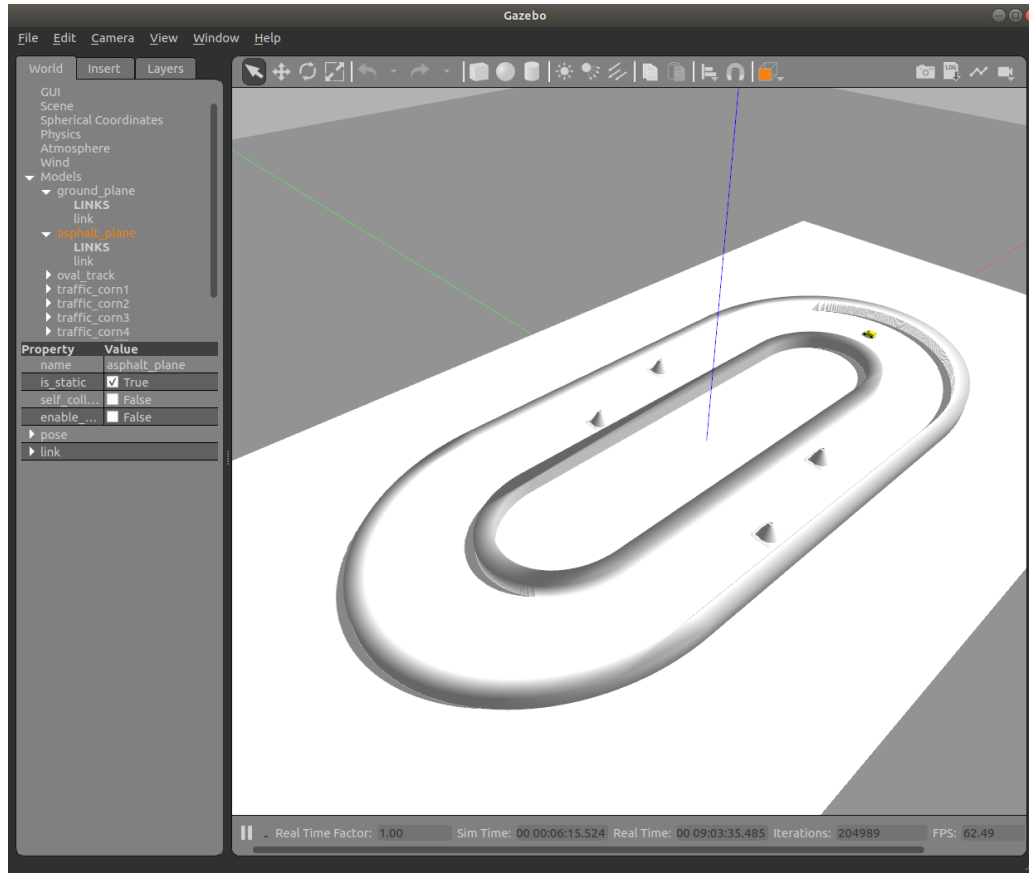
Model pose in the world

```
<model name='oval_track'> <!-- race_track -->
  <pose frame=''>0 0 0 0 -0 0</pose> <!-- -28.0315 -24.
  <scale>1 1 1</scale>
  <link name='body'>
    <pose frame=''>0 0 0 0 -0 0</pose> <!-- -28.0315 -2
    <velocity>0 0 0 0 -0 0</velocity>
    <acceleration>0 0 0 0 -0 0</acceleration>
    <wrench>0 0 0 0 -0 0</wrench>
  </link>
</model>
```

Model pose and scale definition

World

roslaunch f1tenth_simulator simulator.launch

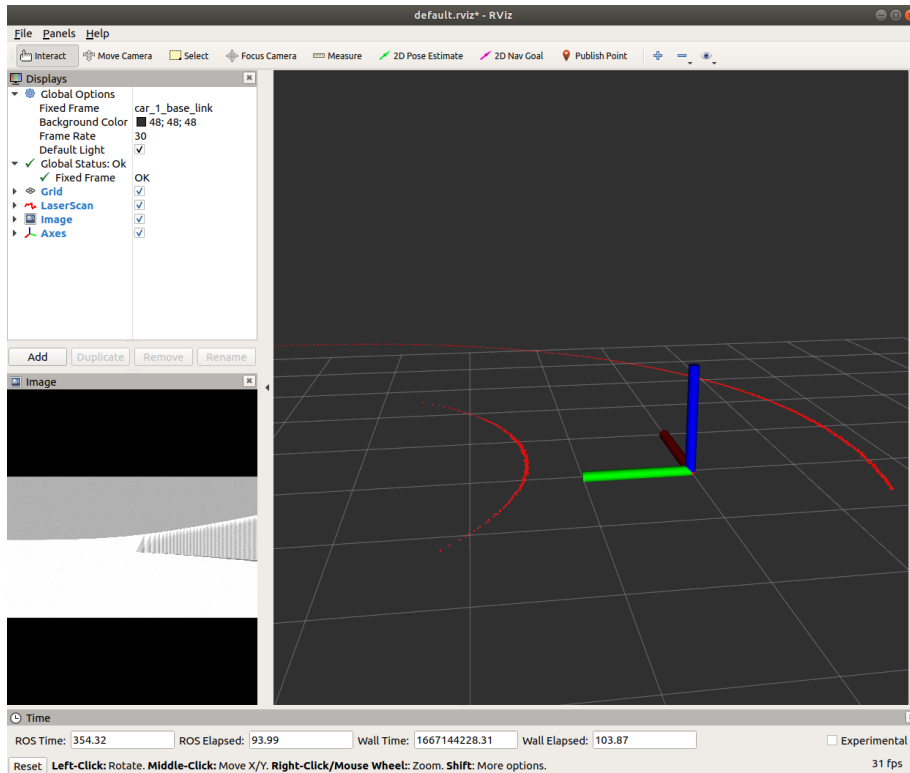


Gazebo gui

```
/car_1/base/footprint
/car_1/base/odom
/car_1/camera/image
/car_1/camera/image/compressed
/car_1/camera/image/compressed/parameter_descriptions
/car_1/camera/image/compressed/parameter_updates
/car_1/camera/image/compressedDepth
/car_1/camera/image/compressedDepth/parameter_descriptions
/car_1/camera/image/compressedDepth/parameter_updates
/car_1/camera/image/theora
/car_1/camera/image/theora/parameter_descriptions
/car_1/camera/image/theora/parameter_updates
/car_1/camera/parameter_descriptions
/car_1/camera/parameter_updates
/car_1/camera/rgb/camera_info
/car_1/command
/car_1/ground_truth
/car_1/joint_states
/car_1/left_front_wheel_velocity_controller/command
/car_1/left_front_wheel_velocity_controller/pid/parameter_descriptions
/car_1/left_front_wheel_velocity_controller/pid/parameter_updates
/car_1/left_front_wheel_velocity_controller/state
/car_1/left_rear_wheel_velocity_controller/command
/car_1/left_rear_wheel_velocity_controller/pid/parameter_descriptions
/car_1/left_rear_wheel_velocity_controller/pid/parameter_updates
/car_1/left_rear_wheel_velocity_controller/state
/car_1/left_steering_hinge_position_controller/command
/car_1/left_steering_hinge_position_controller/pid/parameter_descriptions
/car_1/left_steering_hinge_position_controller/pid/parameter_updates
/car_1/left_steering_hinge_position_controller/state
/car_1/right_front_wheel_velocity_controller/command
/car_1/right_front_wheel_velocity_controller/pid/parameter_descriptions
/car_1/right_front_wheel_velocity_controller/pid/parameter_updates
/car_1/right_front_wheel_velocity_controller/state
/car_1/right_rear_wheel_velocity_controller/command
/car_1/right_rear_wheel_velocity_controller/pid/parameter_descriptions
/car_1/right_rear_wheel_velocity_controller/pid/parameter_updates
/car_1/right_rear_wheel_velocity_controller/state
/car_1/right_steering_hinge_position_controller/command
/car_1/right_steering_hinge_position_controller/pid/parameter_descriptions
/car_1/right_steering_hinge_position_controller/pid/parameter_updates
/car_1/right_steering_hinge_position_controller/state
/car_1/scan
/clock
/gazebo/link_states
/gazebo/model_states
/gazebo/parameter_descriptions
/gazebo/parameter_updates
/gazebo/set_link_state
/gazebo/set_model_state
/rosout
/rosout_agg
/tf
```

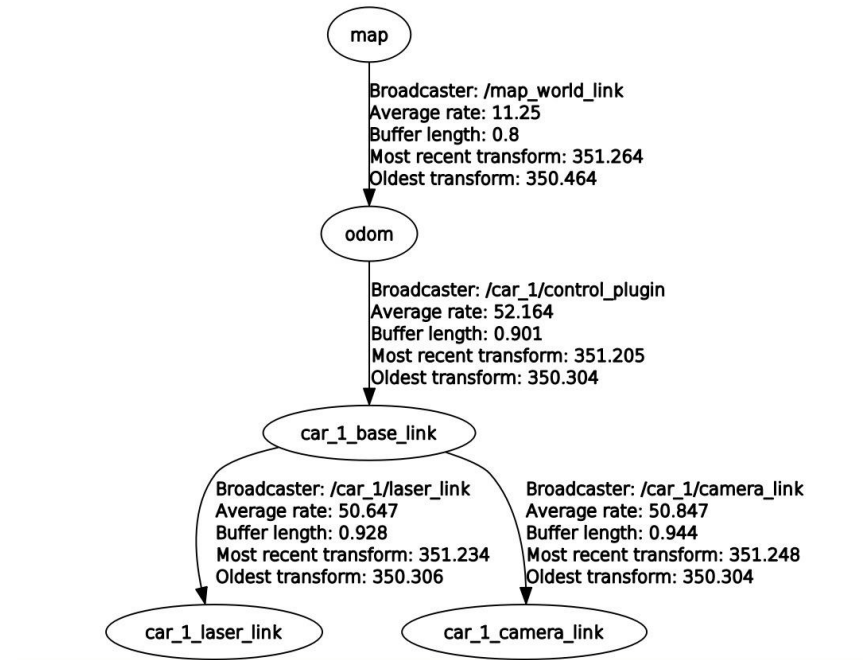
World

Rviz



Rviz sensor outputs

`roslaunch rqt_tf_tree rqt_tf_tree`



TF

Q & A

Email : fingb20@kaist.ac.kr