**Experiment 7**

**Create a Custom Gazebo World**

**Objective**

To create a custom world in Gazebo with both static and dynamic objects.

**Theory**

The following terminologies are significant for the implementation of a new world in Gazebo

1. World: The term used to describe a collection of robots and objects (such as buildings, tables, and lights), and global parameters including the sky, ambient light, and physics properties.
2. Static: Entities marked as static (those having the <static>true</static> element in SDF), are objects which only have collision geometry. All objects which are not meant to move should be marked as static, which is a performance enhancement
3. Dynamic: Entities marked as dynamic (either missing the <static> element or setting false in SDF), are objects which have both inertia and a collision geometry.\

**Procedure**

1. Create the workspace.

mkdir ~/simulation\_ws/src -p

1. Compile our empty workspace

source /opt/ros/noetic/setup.bash

source /usr/share/gazebo/setup.sh

cd ~/simulation\_ws/

catkin\_make

1. Create our ROS package: **my\_simulations**

Open new terminal

source ~/simulation\_ws/devel/setup.bash

cd ~/simulation\_ws/src

catkin\_create\_pkg my\_simulations

1. Create **launch**, **world** and **urdf** folders inside the my\_simulations package.

cd my\_simulations

mkdir launch world urdf

1. Inside the **launch** folder, create a file named **my\_world.launch**:

nano launch/my\_world.launch

1. In that file, let’s put the following content:

<?xml version="1.0" encoding="UTF-8" ?>

<launch>

<!-- overwriting these args -->

<arg name="debug" default="false" />

<arg name="gui" default="true" />

<arg name="pause" default="false" />

<arg name="world" default="$(find my\_simulations)/world/empty\_world.world" />

<param name="robot\_description" command="cat '$(find my\_simulations)/urdf/robo.urdf'" />

<arg name = "x" default = "0"/>

<arg name = "y" default = "0"/>

<arg name = "z" default = "0"/>

<!-- include gazebo\_ros launcher -->

<include file="$(find gazebo\_ros)/launch/empty\_world.launch">

<arg name="world\_name" value="$(arg world)" />

<arg name="debug" value="$(arg debug)" />

<arg name="gui" value="$(arg gui)" />

<arg name="paused" value="$(arg pause)" />

<arg name="use\_sim\_time" value="true" />

</include>

</launch>

1. Create a file named **empty\_world.world** in the **world** folder:

nano world/empty\_world.world

In that file, let’s add the following content:

<?xml version="1.0" ?>

<sdf version="1.5">

<world name="default">

<!-- A global light source -->

<include>

<uri>model://sun</uri>

</include>

<!-- A ground plane -->

<include>

<uri>model://ground\_plane</uri>

</include>

</world>

</sdf>

1. Open new terminal

source ~/simulation\_ws/devel/setup.bash

roslaunch my\_simulations my\_world.launch

1. Create New world in Gazebo using Building editor and inserting models from repository.
2. Save the above file as **test.world** in the **world** folder of the working package.
3. Update with new world file name in launch file already created.
4. Try step 8 again.

**Editing in Gazebo**

1. Start Gazebo with simple ground plane using the following command

$ gazebo

1. *Adding objects:* Gazebo provides two mechanisms for adding objects to Gazebo. The first is a set of simple shapes, located above the render window.

The second is via the model database, which is accessible by selecting the Insert tab in the upper left corner.

1. *Adding Simple Shapes*: Boxes, spheres, and cylinders may be added to the world by clicking on the appropriate icon above the render window. Each shape is of unit size:

Box: 1x1x1 meter

Sphere: 1 meter diameter

Cylinder: 1 meter diameter, 1 meter length

Select the box icon, and then move your mouse onto the render window. You will see a box that moves with your mouse. Left click when you are happy with the position of the box.

Repeat the same procedure for the sphere and cylinder.

1. Adding Model from the Model Database: Gazebo's model database is a repository of all types of models including robots, tables, and building. Select the Insert tab in the upper left hand corner to access the model database.
2. The list of models are divided into sections according to their current location. Each section is labeled with a path or URI. Selecting an object located on a remote server will cause the model to be downloaded and stored in ~/.gazebo/models.
3. *Position Models*:
4. Translation: The translate tool allows you to move the object along the x, y, and z axes. Select this tool now (or press t) and click on the object you want to move. A three axes visual marker will appear over the object, which allows you to move the object in x, y, and z directions. Click on the object itself and drag it to move on the x-y plane. The objective is to control which axis the object moves along by pressing and holding the x, y, or z key while dragging the object. Hold the Ctrl key to snap the movement to a 1 meter grid. If the object is not aligned with the world (for example after you use the rotate tool explained next), hold the Shift key so the visual markers show up aligned with the world, and can translate in world coordinates.
5. Rotation: The rotate tool allows you to orient a model around the x, y, and z axes. Select this tool (or press r) and click on the object you want to move. Three ring-shaped visual marker will appear over the object, which allows you to rotate the object around the x, y, and z axes. You can also just click on the object itself and hold the x, y, or z keys while dragging it to constrain the motion to one of these axes. You can hold the Ctrl key to snap the movement to 45 degree increments. If the object is not aligned with the world, you can hold the Shift key so the visual markers show up aligned with the world, and you can rotate about the world axes. Try rotating the objects into a different configuration.
6. Scale: The scale tool allows you to resize a model in the x, y, and z directions. Currently the scale tool only works with simple shapes, i.e. box, cylinder and sphere. Select this tool now (or press s) and click on a simple shape. A three axes visual marker will appear over the object, which allows you to scale the x, y, and z dimensions of the object. You can also just click on the object itself and hold the x, y, or z keys while dragging it to constrain the scaling to one of these axes. You can hold the Ctrl key to scale in 1 meter increments. Try scaling the simple shapes into different sizes.
7. *Delete Models*: Models may also be deleted by selecting them and the hitting the Delete key, or by right-clicking on a model and selecting Delete.
8. *Saving a World:* Once you are happy with a world it can be save through the File menu. Select the File menu now, and choose Save World As. A pop-up will appear asking you to enter a new filename. Enter my\_world.sdf and click okay.
9. Loading a World: A saved world may be loaded on the command line

gazebo my\_world.sdf