

Autonomous Hockey Robot – Simulation Track

ARI3215 Robotics 2 – University of Malta
Simulation Track (ROS 2 Jazzy + Gazebo)

Project Overview

This project implements a fully simulated autonomous hockey-playing robot using ROS 2 Jazzy and Gazebo. The robot operates inside a virtual hockey arena and relies entirely on simulated sensors to perceive its environment and act autonomously.

Student 1 – Simulation & Robot Modelling

Student 1 is responsible for the simulation infrastructure, including robot modelling using URDF/Xacro, Gazebo world design, sensor integration (LiDAR and IMU), TF configuration, multi-robot namespaces, launch files, and ROS–Gazebo bridging.

System Requirements

- Ubuntu 22.04
- ROS 2 Jazzy Jalisco
- Gazebo (gz-sim)

Required Packages

```
sudo apt update && sudo apt install -y \
  ros-jazzy-desktop \
  ros-jazzy-ros-gz \
  ros-jazzy-xacro \
  ros-jazzy-robot-state-publisher \
  ros-jazzy-joint-state-publisher \
  ros-jazzy-tf2-tools \
  ros-jazzy-sensor-msgs \
  ros-jazzy-geometry-msgs \
  ros-jazzy-nav-msgs \
  python3-colcon-common-extensions \
  python3-rosdep \
  gz-sim

sudo rosdep init
rosdep update
```

How to Run the Simulation (Terminal-by-Terminal)

Terminal 1 – Build Workspace

```
cd ~/ros2_ws  
colcon build  
source install/setup.bash
```

Terminal 2 – Launch Gazebo and Spawn Robots

```
source ~/ros2_ws/install/setup.bash  
ros2 launch hockey_robot_description spawn_robots.launch.py
```

This terminal starts Gazebo, loads the hockey arena, spawns two robots (robot1 and robot2), and launches robot_state_publisher for both.

Terminal 3 – Start ROS ↔ Gazebo Bridges

```
source ~/ros2_ws/install/setup.bash  
ros2 launch hockey_robot_description bridge.launch.py
```

This enables communication for LiDAR, IMU, velocity commands, and puck pose debugging.

Terminal 4 – RViz Visualisation (Optional)

```
source ~/ros2_ws/install/setup.bash  
rviz2
```

In RViz, set the Fixed Frame to 'robot1/base_link' and add TF, LaserScan, and RobotModel displays.

Verification Commands

```
ros2 topic list  
ros2 run tf2_tools view_frames
```

Notes for Classmates

- Always source the workspace in every terminal
- Do not close the Gazebo terminal while running
- Robots run in separate namespaces (robot1, robot2)
- Behaviour and perception nodes should be launched after the simulation is running

AI Concepts Demonstrated

Perception (LiDAR, IMU), reasoning through state-based behaviour, and action via autonomous navigation using velocity commands.

Academic Integrity

All work is original and developed for ARI3215 Robotics 2. Standard ROS 2 and Gazebo libraries are used in accordance with their documentation.