

Intelligent Robotics Assignments

Group 30

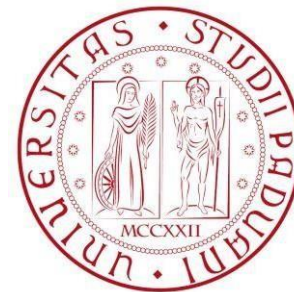
Manthara Cristy Jo

christyjo.manthara@studenti.unipd.it

Maulik Rupeshbhai Sompura

maulikrupeshbhai.sompura@studenti.unipd.it

Professor. Emanuele Menegatti



Assignment 1

Goal : Develop a navigation routine for the TIAGo robot to navigate autonomously in an indoor environment with movable obstacles.

Key Components :

- Action definitions
- Client-server architecture
- Object detection

Assignment 2

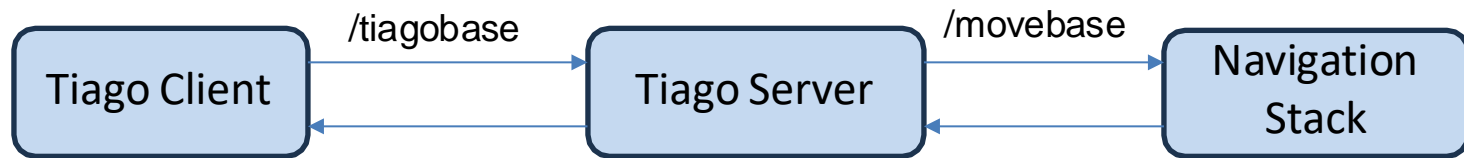
Goal : Implement a fetch and delivery behavior in TIAGo within a simulated environment.

Key Components :

- Use MoveIt! for pick and place routines.
- Detect objects with TIAGo's camera and AprilTag library.
- Generate collision-free trajectories.
- Implement a robust grasping routine.
- Attach objects virtually using Gazebo ros link attacher plugin.

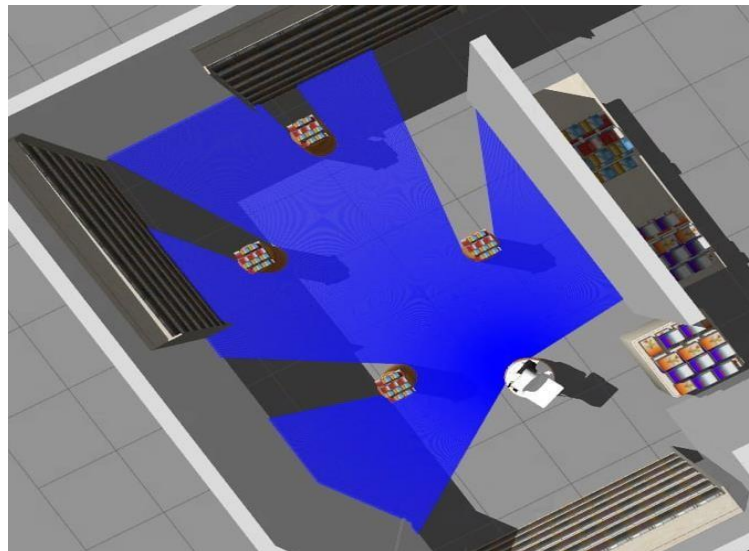
Assignment 1 – High level Design

SYSTEM ARCHITECTURE



DBSCAN Clustering (Density-Based Spatial Clustering of Applications with Noise)

- Apply DBSCAN to find clusters
- groups points into clusters based on their proximity.
- identifies dense regions separated by sparser areas, making it well-suited for detecting obstacles in the environment



Assignment 1

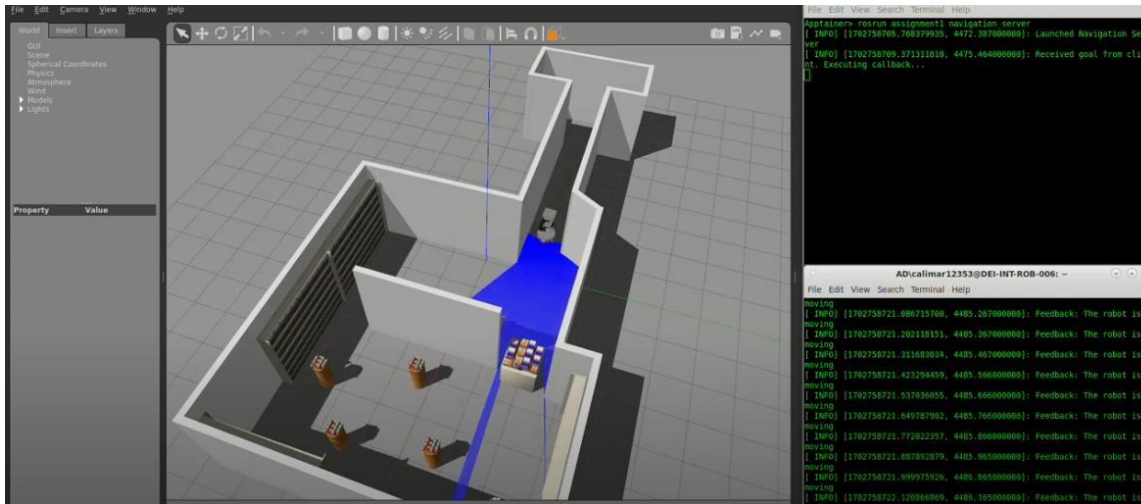
Obstacle Detection

Circle Fitting through the Hyper-Fit Algorithm

- Run circle fitting algorithm to each cluster(distinguishing cylindrical obstacles)
- optimization technique that minimizes the geometric error between the observed data points and the model of a circle.

Obstacle Filtering

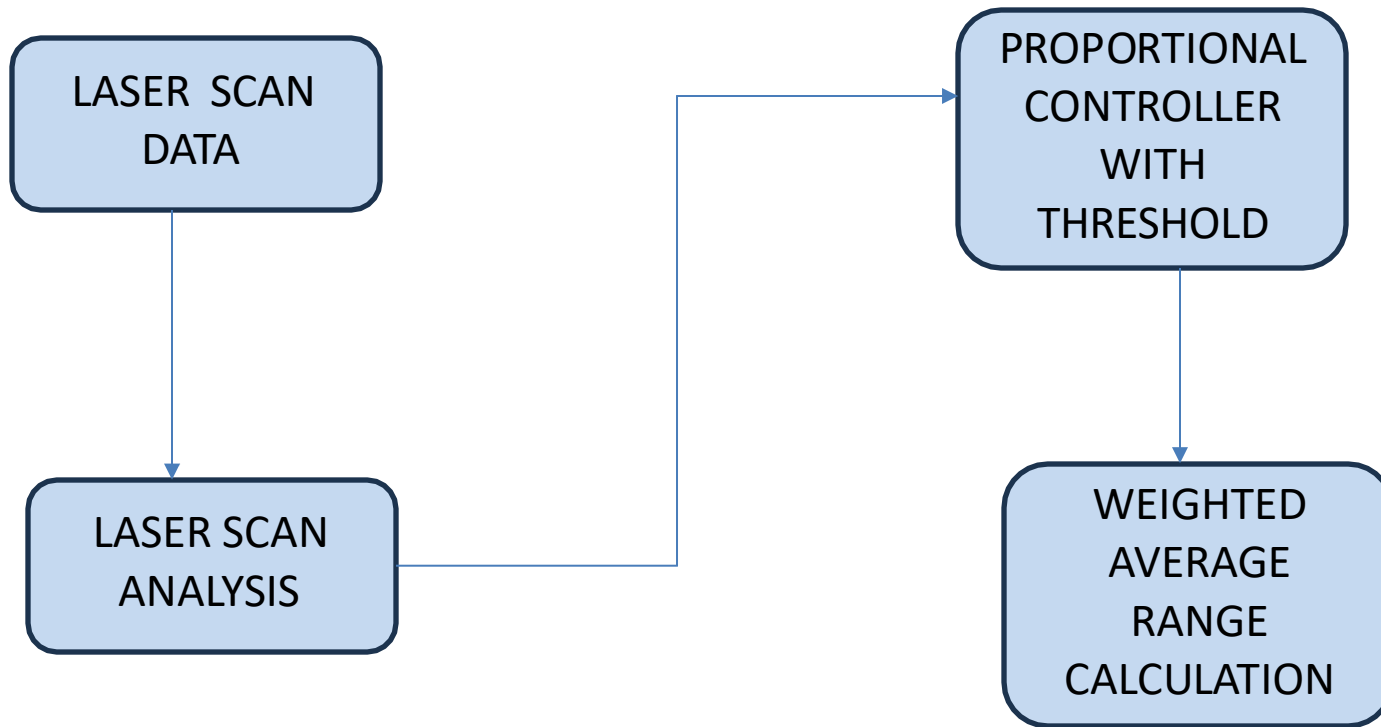
To obtain refined list of obstacles



Our Idea(Corridor Navigation)

Assignment 1

Corridor

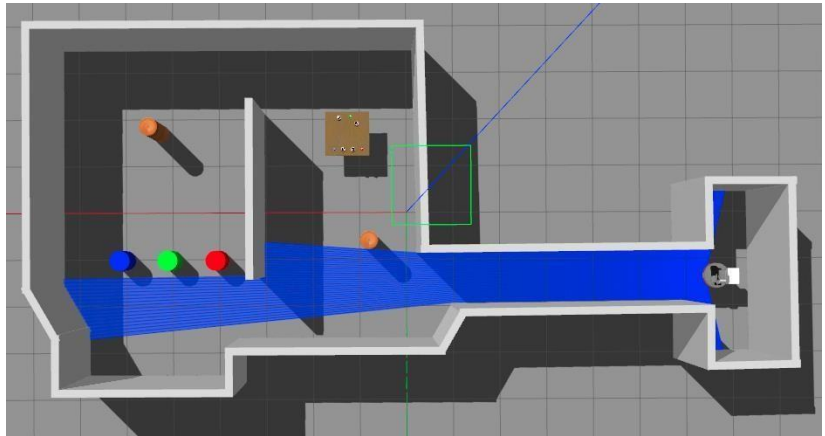


Challenges:.

Assignment 2

Tasks

- Navigating through a narrow corridor and reach to given position
- Detecting objects on the table by using:
 - The robot's camera
 - AprilTag
- Executing the pick-and- place routine while avoiding obstacles



Assignment 2

Implementation

Object Detection (apriltagdetected.cpp): It takes actions such as move head for detection, detect apriltag markers and calculate object poses relative to the robot.

Arm Manipulation (arm manipulator.cpp): Responsible for collision detection and pick & place operations.

Node Communication (tiago client.cpp and tiago server.cpp): Client server architecture responsible for the communication to pick and place operations.

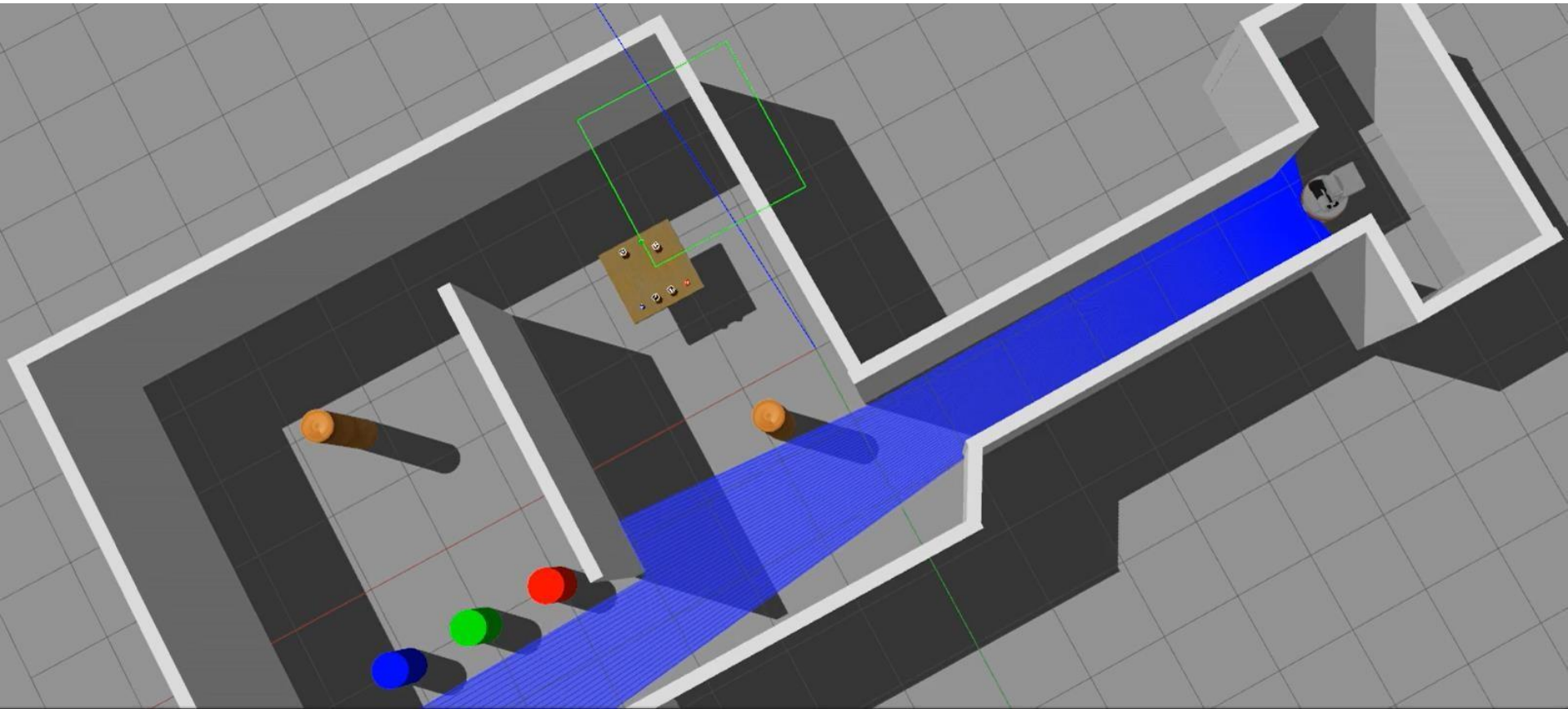
Barrelscanner (barrelscanner.cpp): Add collision to an object and handles grasping and placing sequences.

Services (Aprildetect.srv and barrelscan.srv): AprilDetect.srv: Manages AprilTag detection requests and responses.
BarrelScan.srv: Manages barrel scanning requests and responses.

Launch files: RosnavigatePnP.launch, scannerandclient.launch

Object Detection and Arm manipulation

Assignment 2

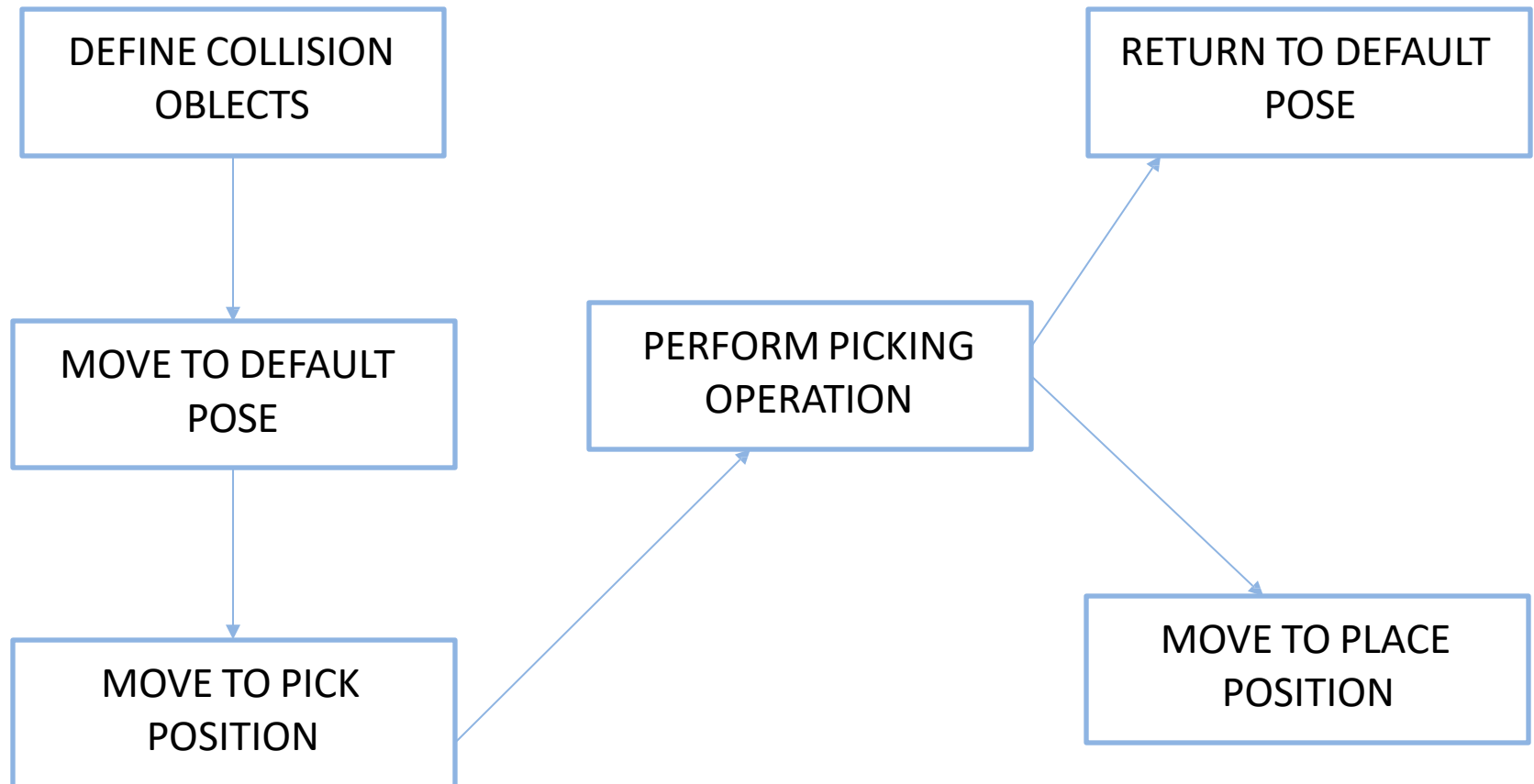


Node Communication



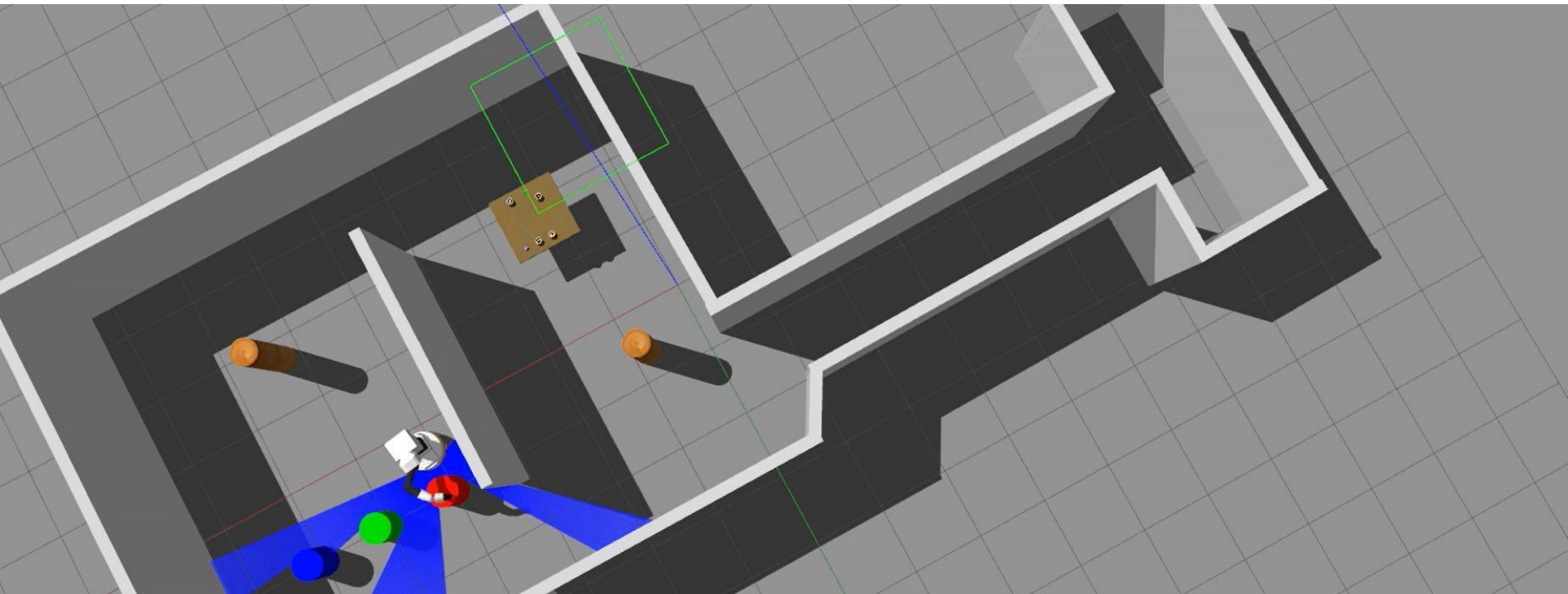
DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Pick Place

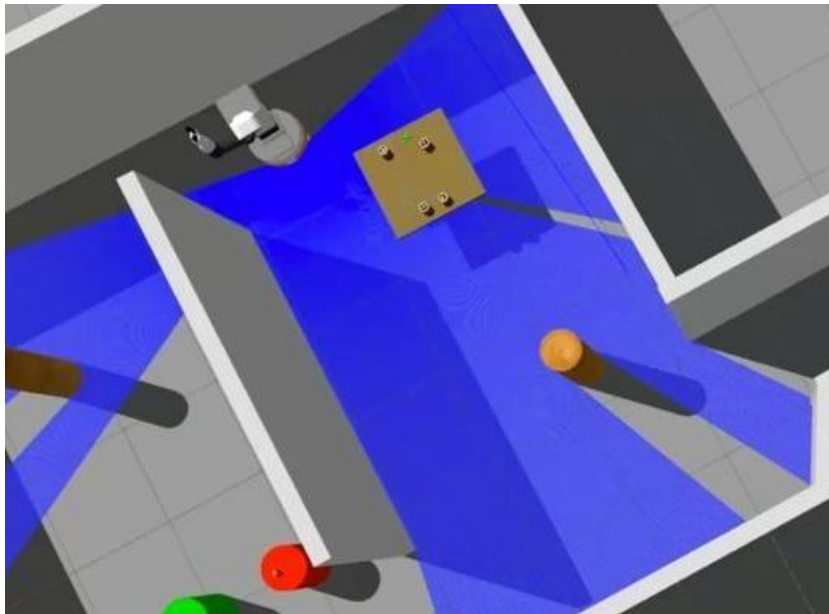


Assignment 2

Navigation



Obstacle Extractor : identifies the locations of obstacles within the environment by analysing and interpreting clusters of points. These clusters are processed to form segments and, ultimately, circles. The process involves detecting and analysing groups of points, with the ability to merge them using the Split And Merge algorithm and filtering them according to predefined criteria.



Barrel Scanner

Challenges and Solutions

- Object Slipping: Adjust gripper values and arm movement velocity.
- Collision Avoidance: Accurate collision objects for table and obstacles.
- Node Execution Order: Add checks and logging to ensure proper sequence.
- Service Requests: Manage parameters and use ROS tools to ensure smooth operation.

Thanks for your attention.