Automous and Mobile Robotics

***Supervisors**:*

Gianluca Palli,

Alessio Caporali

***Presented by**:*

Silvia Cagnolati,

Daniele Crivellari,

Niccolò Mazzocchi

Project goals

Map generation

Arm & head positioning

RPP controller

SLAM

Navigation

Robot Localization

Obstacle avoidance

Goal pose

Pick and Place

ArUco marker identification

Transportation of each ArUco marker to its destination

State Machine

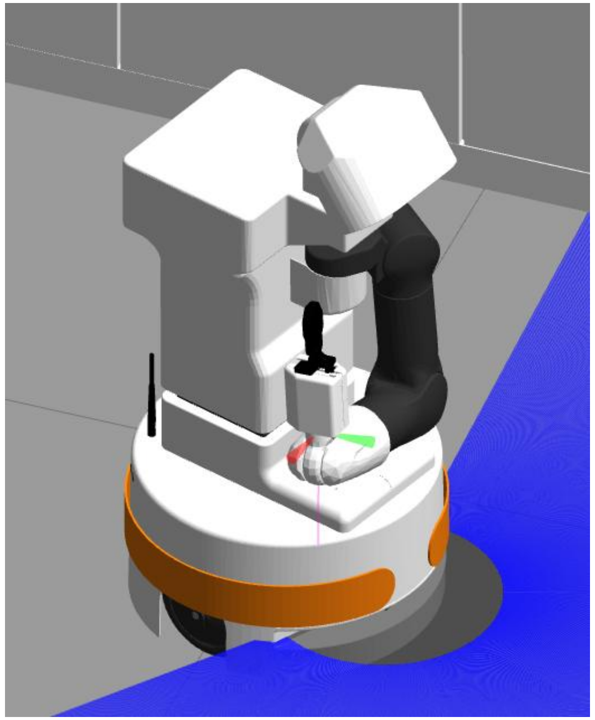
Task synchronization and execution

Task 1 – Map Generation

Arm and head initial positioning

- In order not to collide against the obstacles during the navigation, we set the arm in a safe position.
- Leaning forward the head to better visualize the ArUco markers for the following task.

Parameter	Value
arm_1_joint	153 °
arm_2_joint	- 86 °
arm_3_joint	38 °
arm_4_joint	110 °
arm_5_joint	94 °
arm_6_joint	- 80 °
arm_7_joint	0 °
head_1_joint	0 rad
head_2_joint	- 0.95 rad



Task 1 – Map Generation

Regulated Pure Pursuit controller

- Simpler, robust and more reliable controller for real robots.
- Smooth handling of the final pose with rotation in place if needed.
- Allows the robot to accurately follow the planned trajectory.

RPP controller

Parameter	Value
use_velocity_scaled_lookhaed_dist	False
desired_linear_vel	0.5
min_lookahead_dist	0.0
max_lookahead_dist	0.9
lookahead_time	1.5
rotate_to_heading_angular_vel	0.8
use_rotate_to_heading	True
allow_reversing	False
transform_tolerance	0.2
max_angular_accel	3.2
max_robot_pose_search_dist	10.0

Task 1 – Map Generation

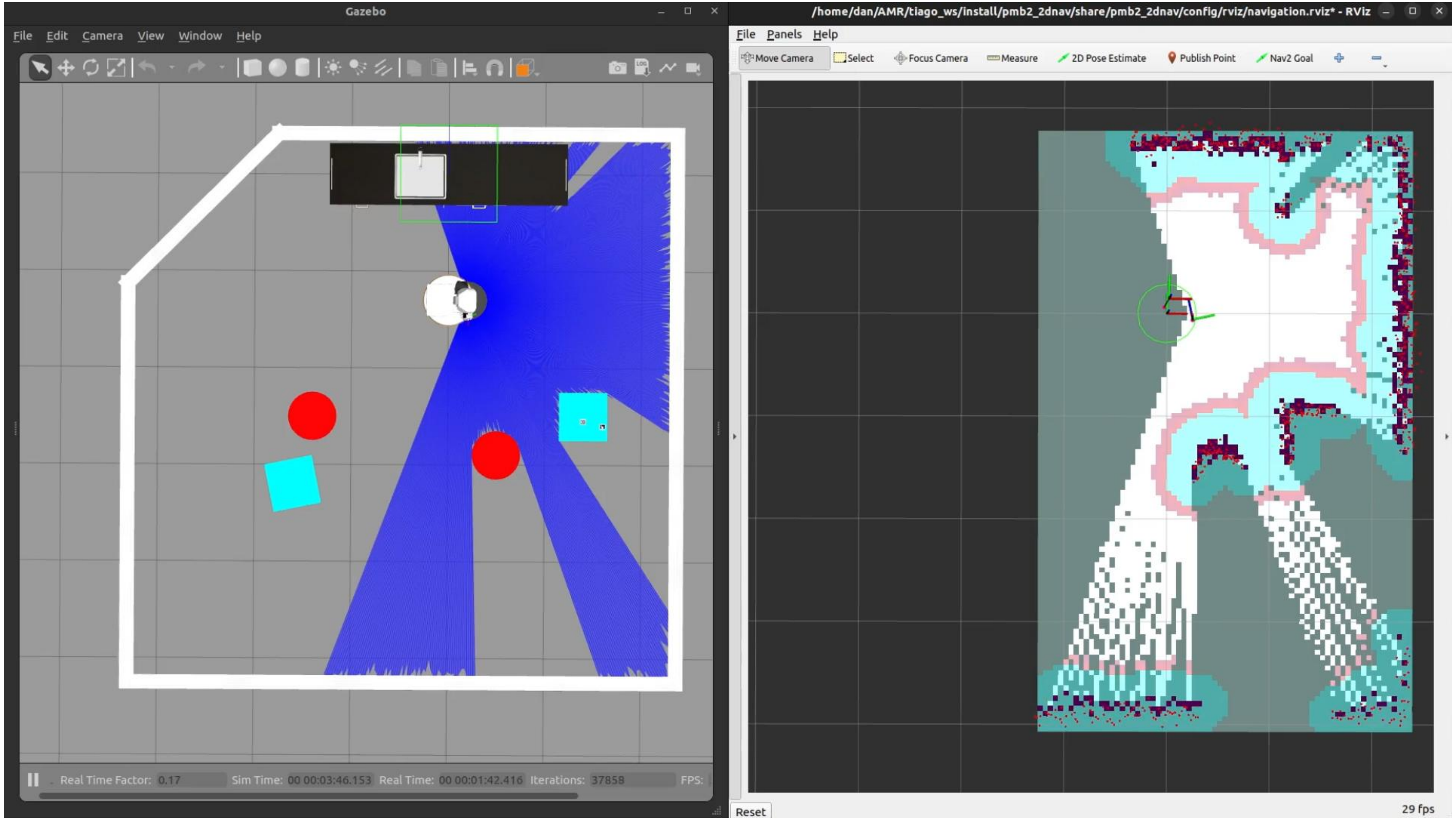
SLAM: Simultaneous Localization And Mapping

- Technique that enable the robot to construct a consistent map of an unknown environment.
- Estimates its own pose within that map.
- Essential for autonomous navigation, dynamic obstacle avoidance, and path planning.

Category	Parameter	Previous value	Recent value
amcl	max_particles	2000	4000
amcl	min_particles	500	2000
tiago parameter	min_lidar_distance	0.05	0.20

Category	Parameter	Previous value	Recent value
global cost map	inflation_radius	0.55	0.40
controller server	xy_goal_tolerance	0.25	0.20
controller server	yaw_goal_tolerance	0.25	0.10

Task 1 - Map Generation



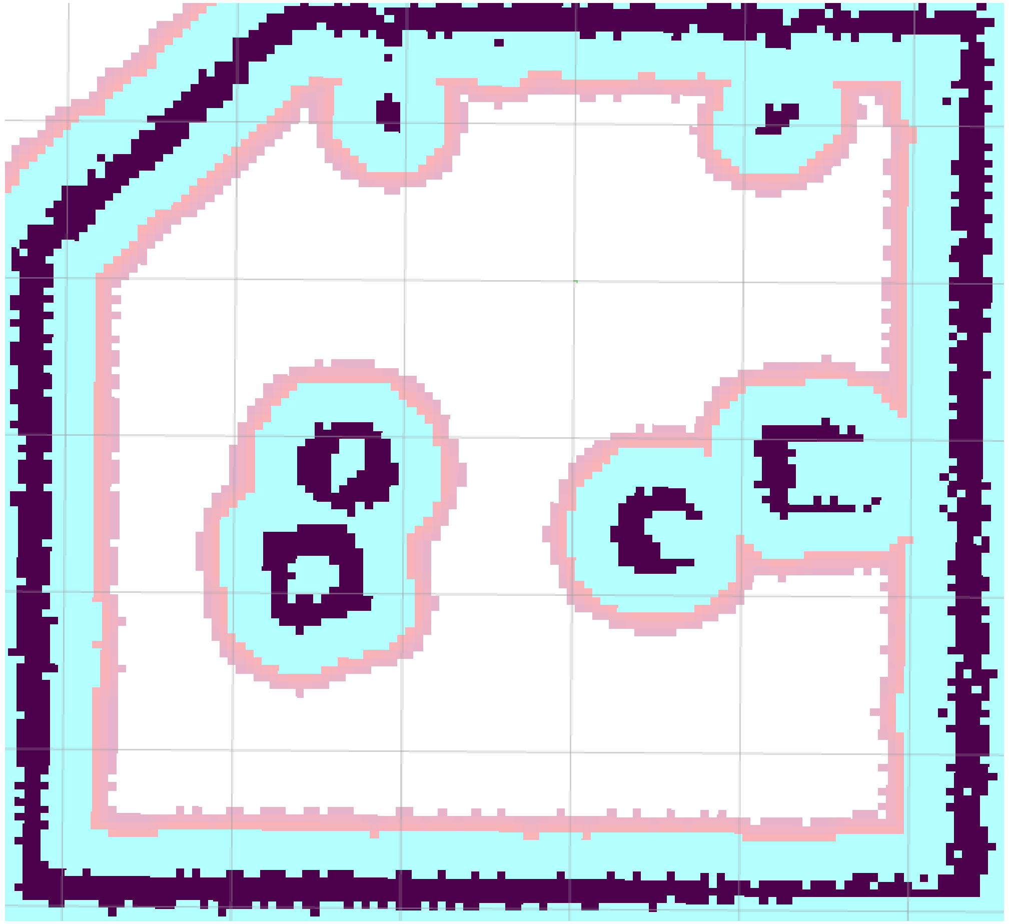
/home/dan/AMR/tiago_ws/install/pmb2_2dnav/share/pmb2_2dnav/config/rviz/navigation.rviz* - RViz

Real Time Factor: 0.17 Sim Time: 00 00:03:46.153 Real Time: 00 00:01:42.416 Iterations: 37858 FPS:

Task 1 – Map Generation

RViz Map

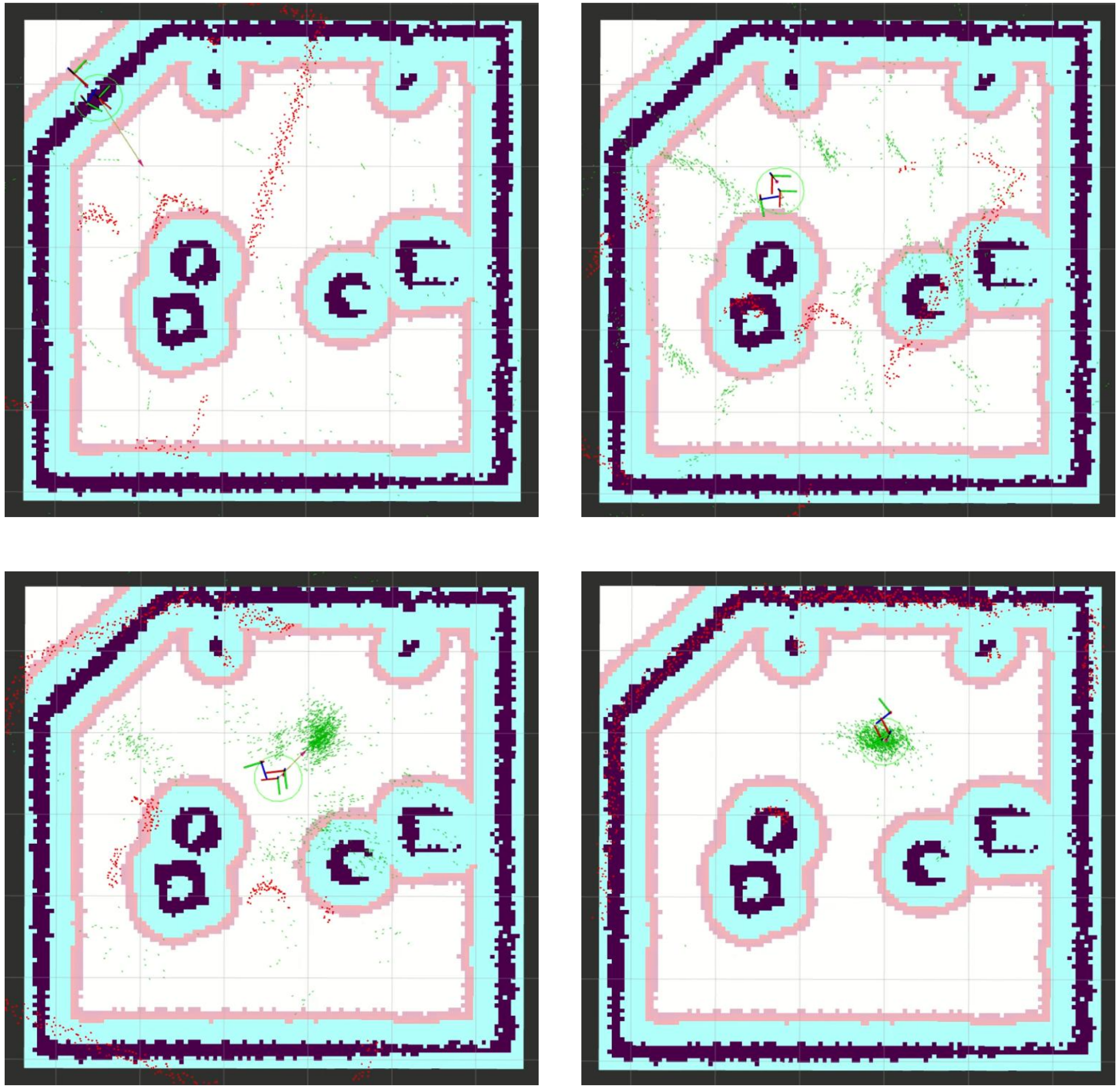
- At the end, we get the following map, in which the robot will navigate.
- The robot will use this map for all the next tasks.
- Ros2 autonomously generates a global cost map for navigation purposes.



Task 2 - Navigation

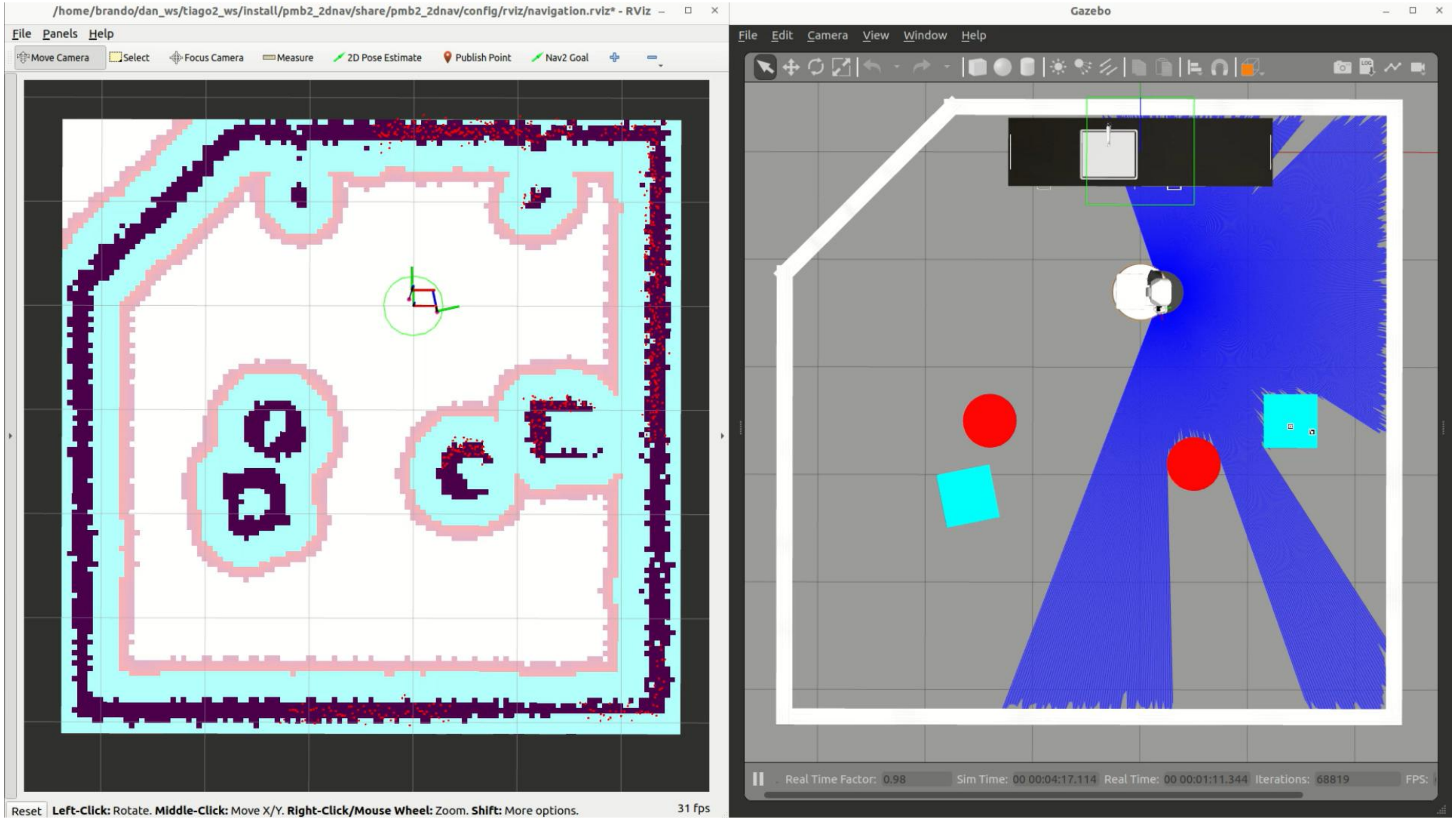
AMC Localization

- Robot localizes itself using the Adaptive Monte Carlo Localization method.
- The robot rotates and collects lidar data, continuously updating its pose and covariance.
- Localization is considered successful when the position and orientation covariances fall below a defined threshold.



Task 2 - Navigation

Robot Localization

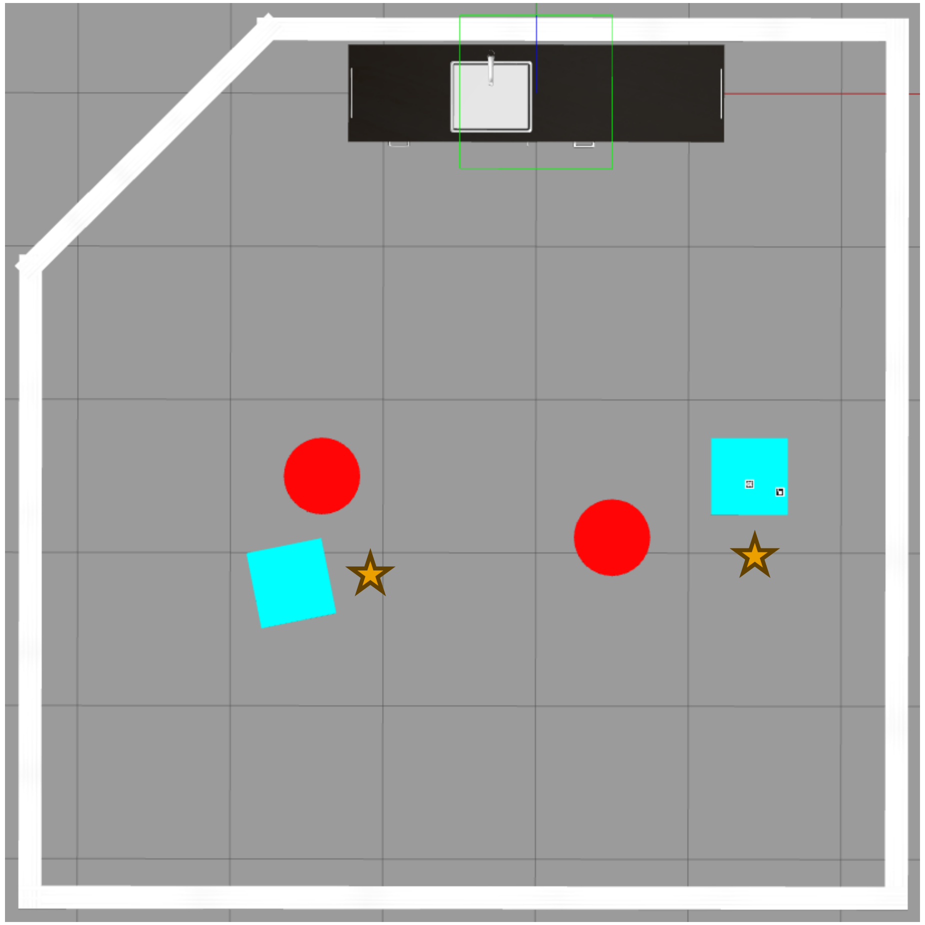

/home/brando/dan_ws/tiago2_ws/install/pmb2_2dnav/share/pmb2_2dnav/config/rviz/navigation.rviz* - RViz
File Panels Help
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
Publish Point
Nav2 Goal
Reset
Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.
31 fps
Gazebo
File Edit Camera View Window Help
Real Time Factor: 0.98
Sim Time: 00 00:04:17.114
Real Time: 00 00:01:11.344
Iterations: 68819
FPS:

Task 2 - Navigation

Navigation Goals

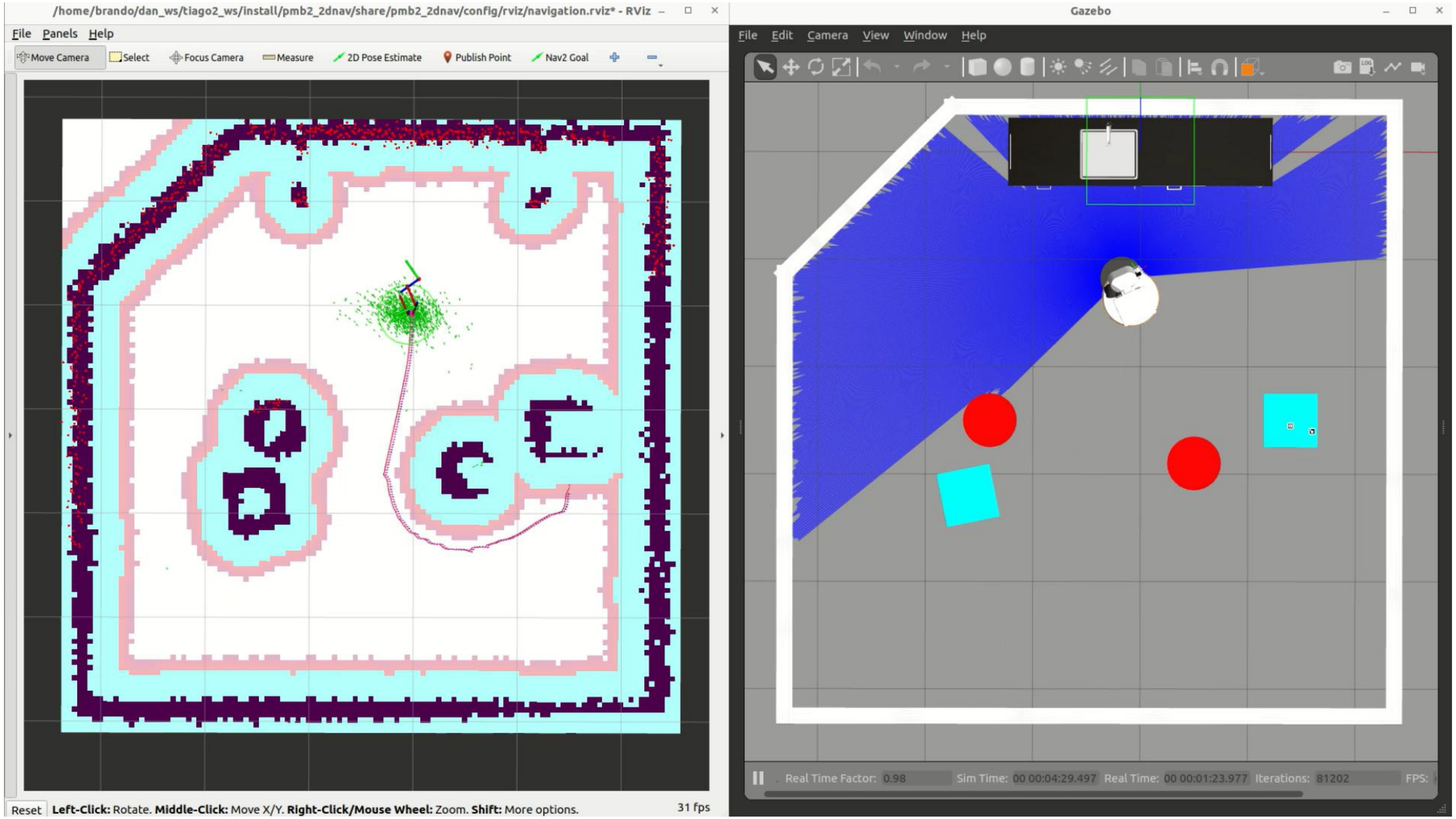
- Exact positions that the robot must reach to properly pick and place the ArUco markers.
- Coordinates and orientation:

	Pick	Place
Position	[1.50, 1.60]	[-0.85, -1.75]
Quaternion orientation	[0, 0, -0.7071, -0.7071]	[0, 0, -0.9963, 0.0853]



Task 2 - Navigation

Obstacle Avoidance and Pick Location



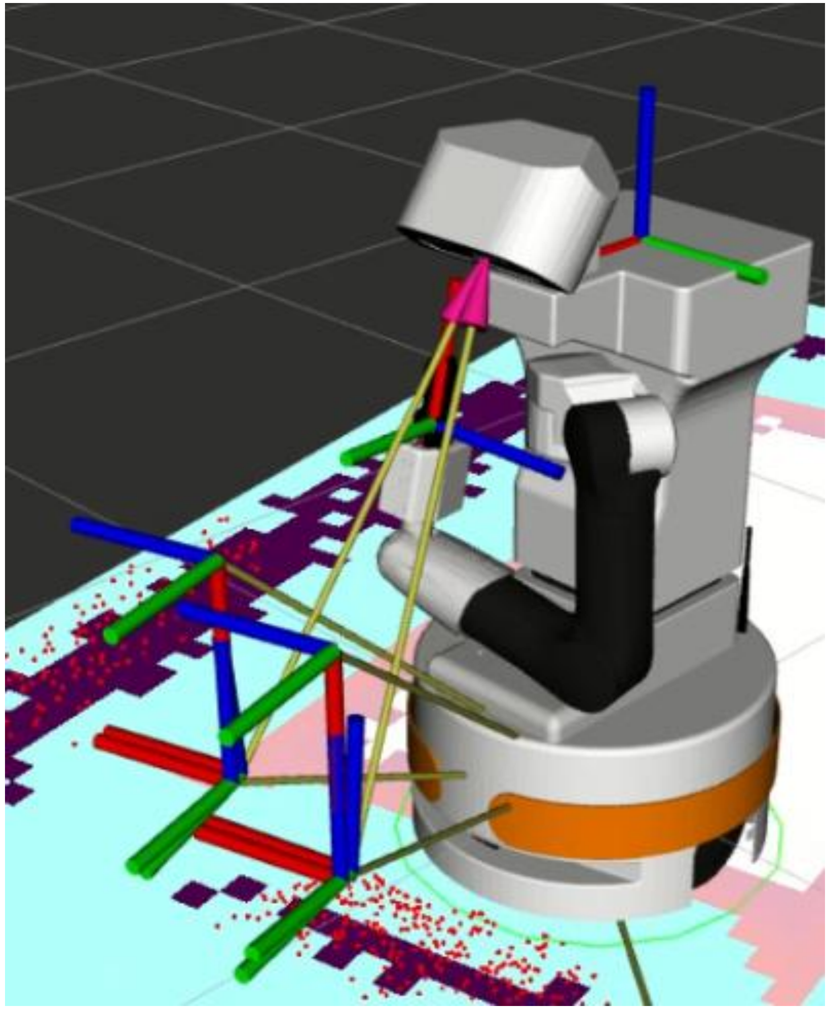
Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.

Real Time Factor: 0.98 Sim Time: 00 00:04:29.497 Real Time: 00 00:01:23.977 Iterations: 81202 FPS:

Task 3 – Pick and Place

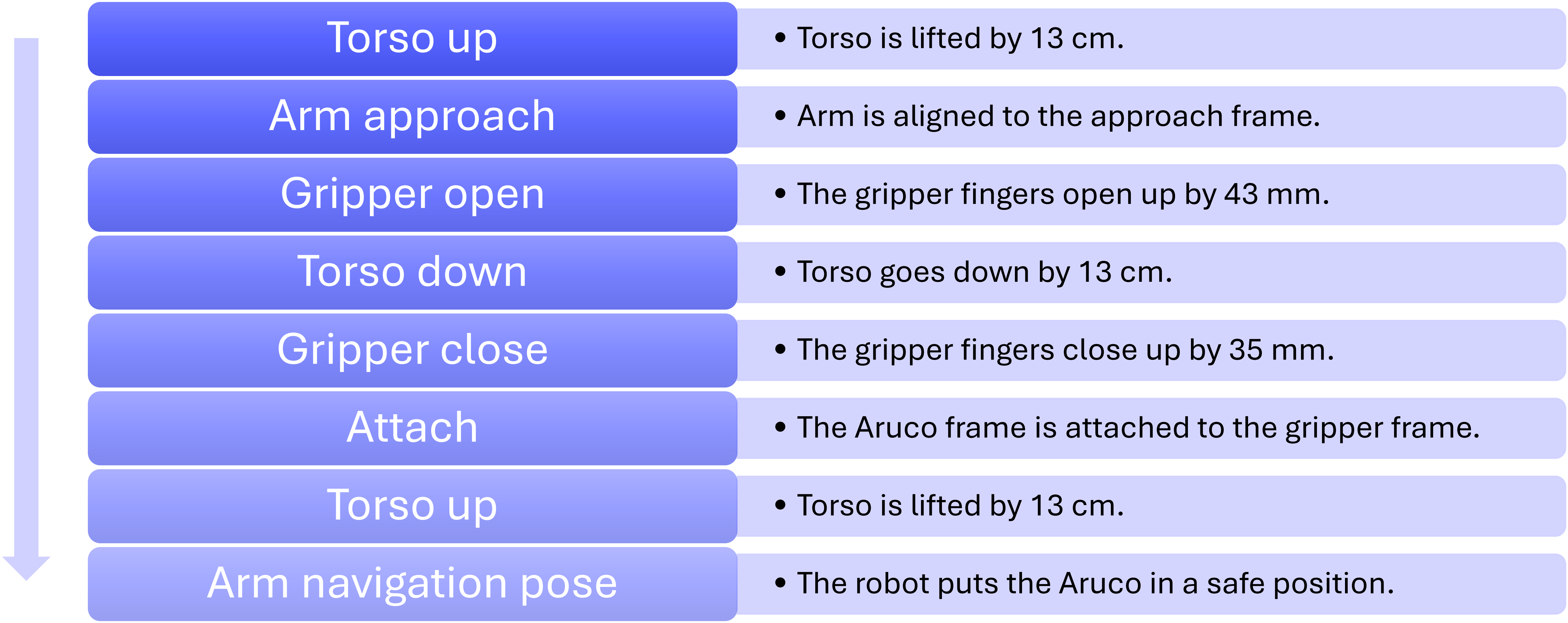
ArUco recognition

- *Aruco_single* node, native in Ros2, recognises the patterns and generates two Frame on the ArUco cubes.
- The custom *Aruco_broadcaster* node reads the frames and process them separately:
  - Aligns them with the Z-axis of the robot, defining the **target frames**.
  - Raises them by 30cm and rotates by 180° around the y-axis, broadcasting the obtained **approach frames**.



Task 3 – Pick and Place

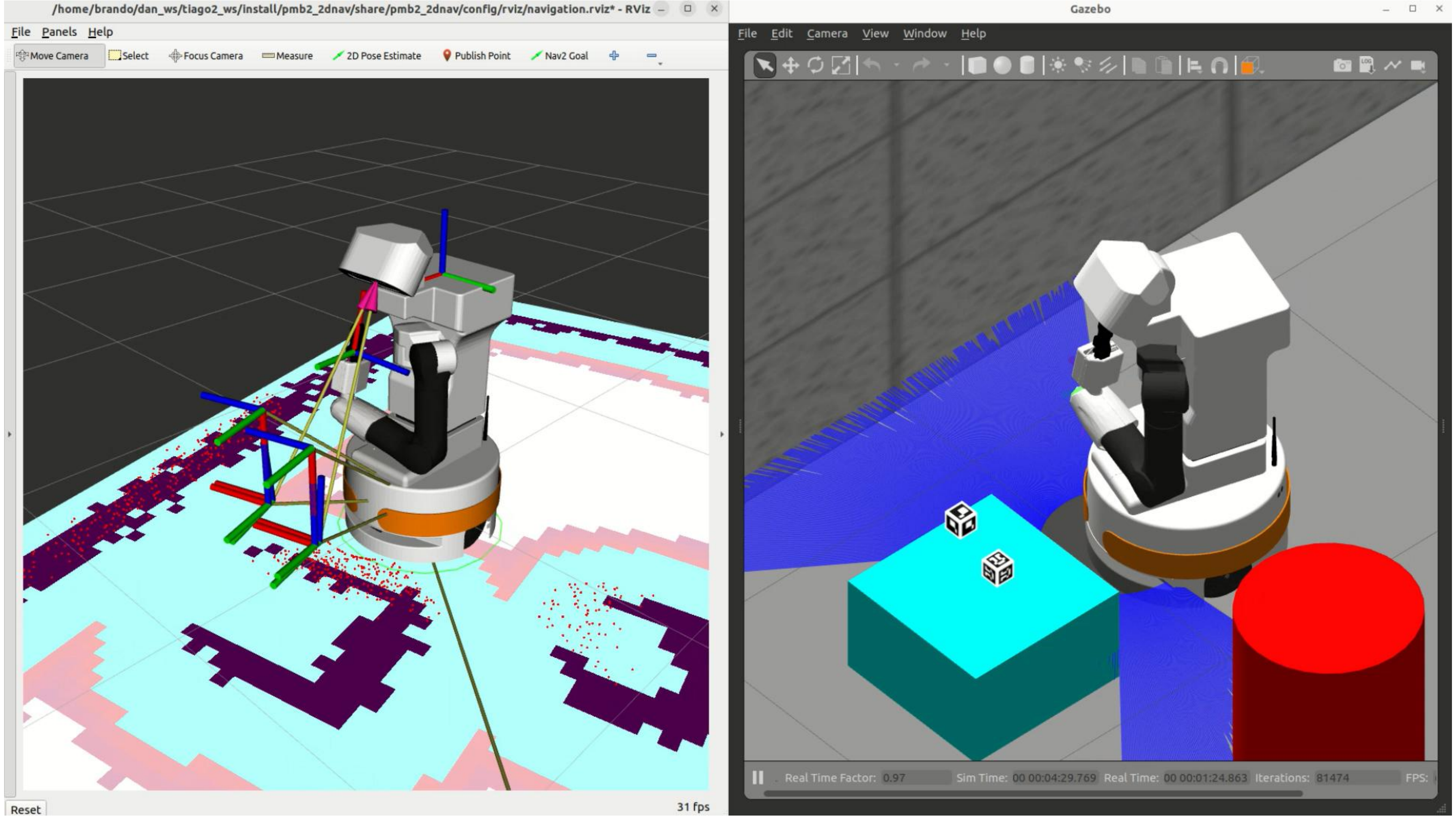
ArUco picking process



Torso up	• Torso is lifted by 13 cm.
Arm approach	• Arm is aligned to the approach frame.
Gripper open	• The gripper fingers open up by 43 mm.
Torso down	• Torso goes down by 13 cm.
Gripper close	• The gripper fingers close up by 35 mm.
Attach	• The Aruco frame is attached to the gripper frame.
Torso up	• Torso is lifted by 13 cm.
Arm navigation pose	• The robot puts the Aruco in a safe position.

Task 3 – Pick and Place

ArUco picking process



Real Time Factor: 0.97 Sim Time: 00 00:04:29.769 Real Time: 00 00:01:24.863 Iterations: 81474 FPS:

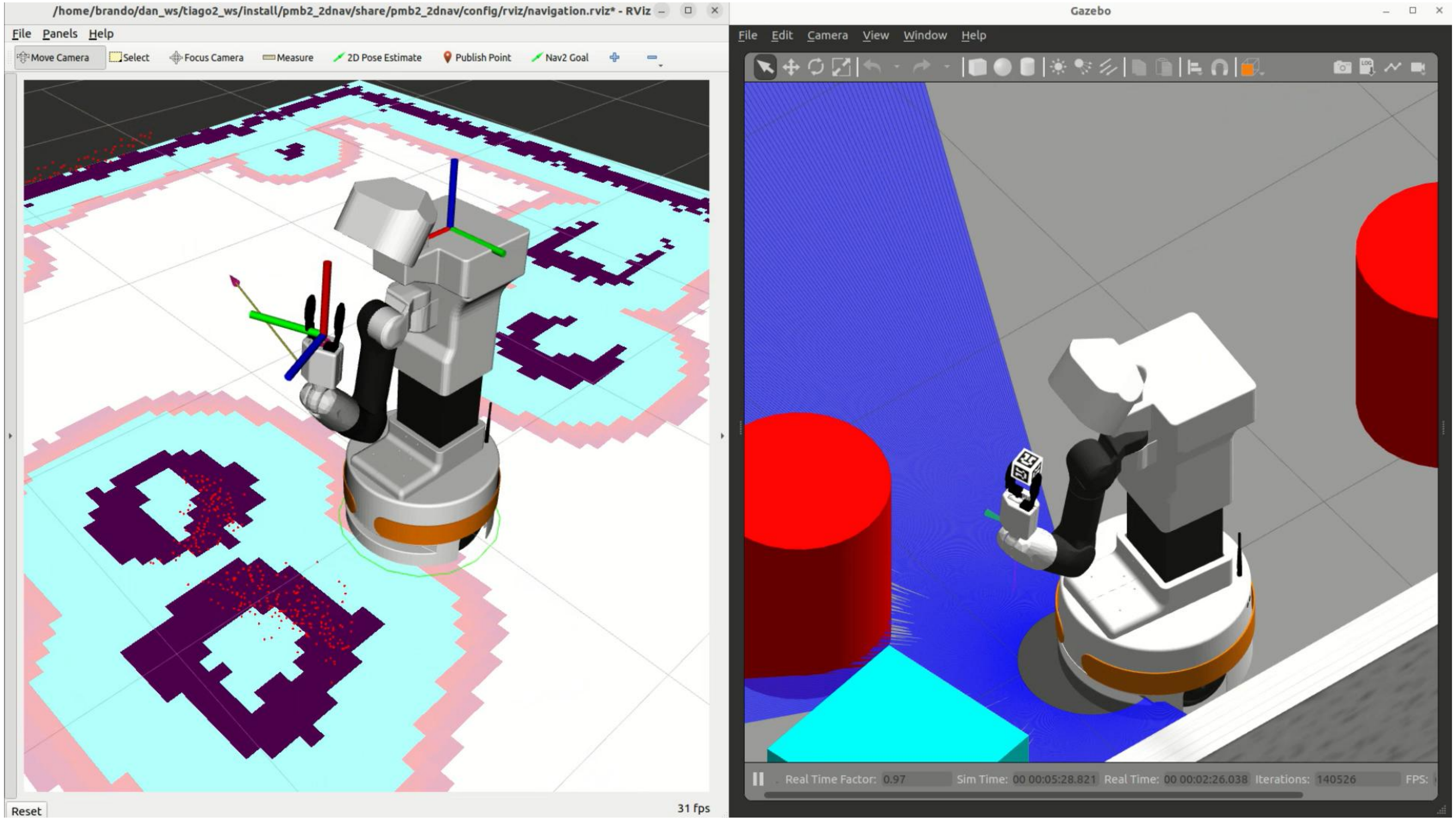
Task 3 – Pick and Place

ArUco placing process

Arm place	• The arm is set to release the Aruco marker on the table.
Torso down	• Torso goes down.
Gripper open	• The gripper fingers open up by 43 mm.
Detach	• The Aruco marker is detatched from the gripper.
Torso up	• Torso is lifted up by 13 cm.
Arm default pose	• The arm is set in the initial default position.

Task 3 – Pick and Place

ArUco placing process


/home/brando/dan_ws/tiago2_ws/install/pmb2_2dnav/share/pmb2_2dnav/config/rviz/navigation.rviz* - RViz
File Panels Help
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
Publish Point
Nav2 Goal
Reset
31 fps
Gazebo
File Edit Camera View Window Help

Real Time Factor: 0.97 Sim Time: 00 00:05:28.821 Real Time: 00 00:02:26.038 Iterations: 140526 FPS:

State machine

Ros2 nodes used

Localization: Performs the starting autonomous localization.

Navigation: Manages the navigation to the pick and place locations

Aruco_single (x2): Recognizes the ArUco and broadcast a frame for them.

Aruco_broadcaster: Generates target and approach frames.

Arm: Handles movements for the arm.

Gripper: Handles movements for the gripper.

Torso: Handles movements for the torso.

Attacher: Operates attachment and detachment of the ArUco

State machine

State machine operating principle

01

After setting up Gazebo and RViz, a custom launch file starts all required nodes.

02

The state machine sends commands on */state_machine_cmd* topic, which all nodes subcribes to.

03

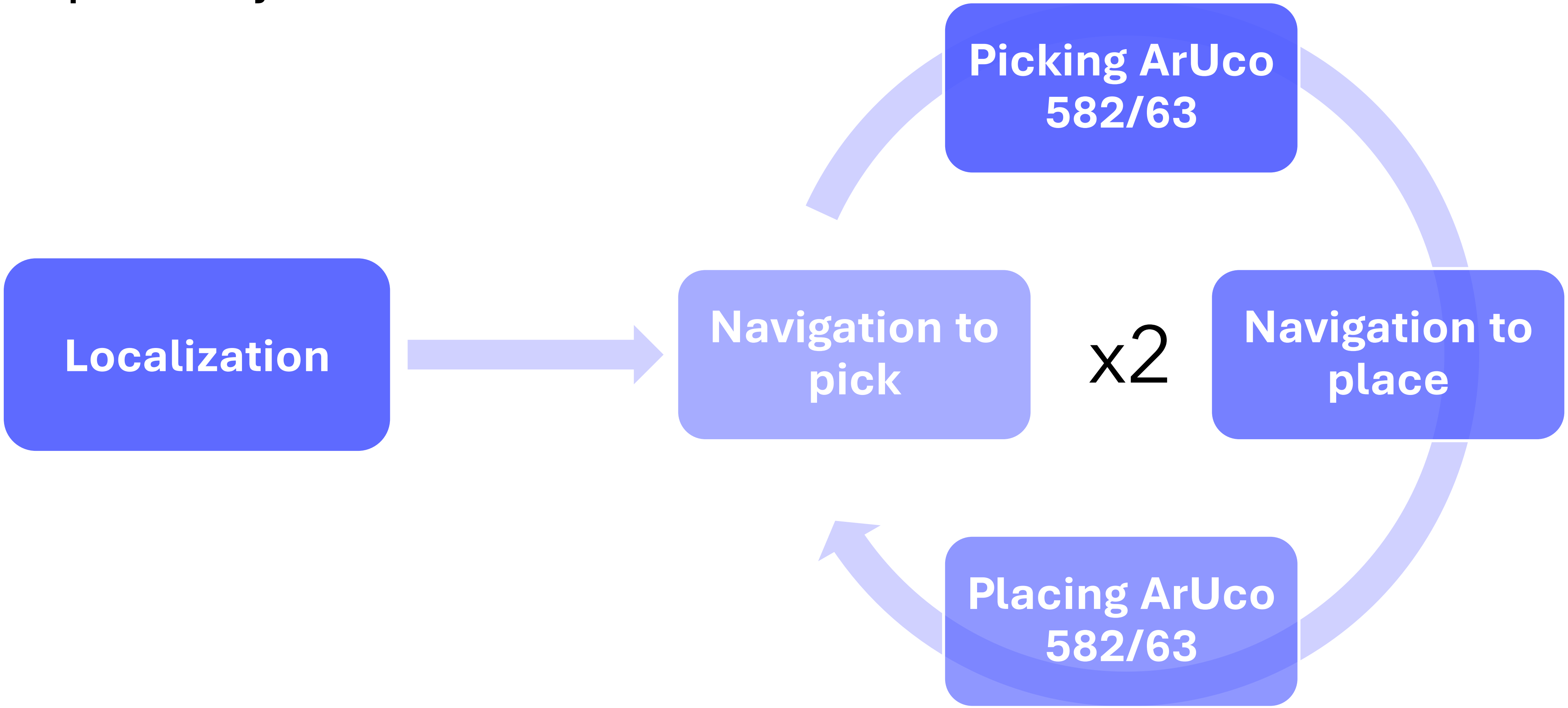
Upon receiving the command or completing a task, nodes send feedback on */state_machine_fb* topic.

04

This ensures synchronized and robust task execution across all subsystems.

State Machine

Total operation cycle



Localization

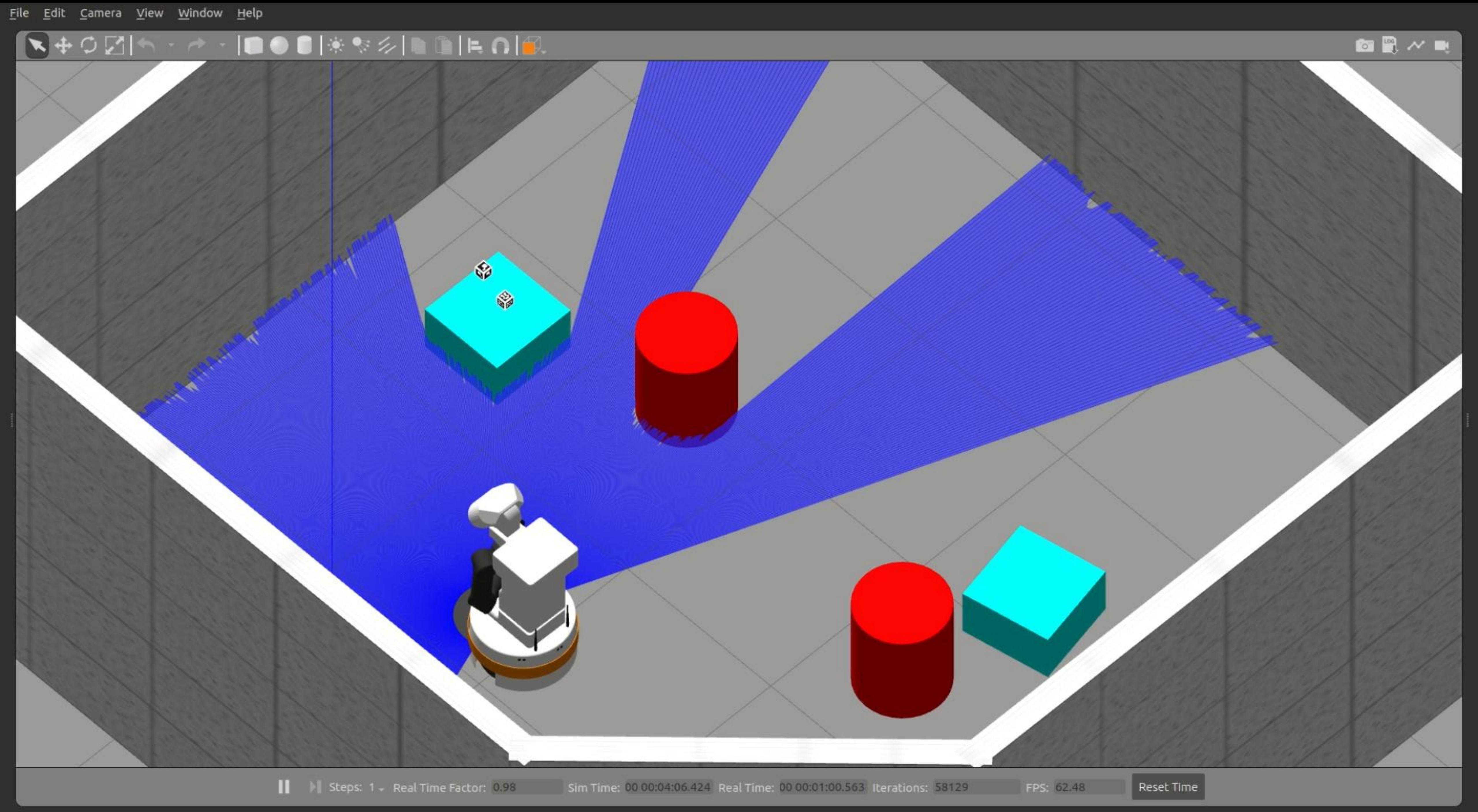
Navigation to pick

Picking ArUco 582/63

Navigation to place

Placing ArUco 582/63

x2



Steps: 1 Real Time Factor: 0.98 Sim Time: 00 00:04:06.424 Real Time: 00 00:01:00.563 Iterations: 58129 FPS: 62.48

Conclusions

☑ **Main goal achieved:** Successful implementation of all the tasks and of a state machine that synchronize all o them

☑ **Project strengths**

- Reliability thanks to modular architecture and simple individual components
- Efficient state machine for general execution

⚠ **Project limitations**

- Not scalable
- Computationally demanding

THANKS FOR YOUR ATTENTION