Assignment 4

Mobile Manipulator Extra Credit Guide

CS498GC Mobile Robotics

Fall 2025

Early Submission Bonus Points

Submit before **November 9, 2025**
+20 Extra Credit Points

Submit before **November 14, 2025**
+10 Extra Credit Points

Instructor:	Professor Girish Chowdhary
Teaching Assistant:	Kulbir Singh Ahluwalia
Course Website:	kulbir-singh-ahluwalia.com/cs498gc/fa25
Gradescope Code:	KDP5G8

Contents

1 Important Deadlines & Extra Credit Opportunities

✓ Success

Early Submission Bonus Points 🚀

Submit Assignment 4 Part 1 early to receive bonus points:

- Submit before **November 9, 2025**: **+20 Bonus Points**
- Submit before **November 14, 2025**: **+10 Bonus Points**

These bonuses apply to Assignment 4 Part 1 only and reward early completion!

1.1 Regular Deadlines

Assignment	Due Date	Points
Part 1	November 21, 2025 @ 11:00 PM	25 points
Part 2	December 9, 2025 @ 11:00 PM	75 points
Total Possible	With Maximum Early Bonus (+20)	**120 points**

ℹ Information

Gradescope Information

- Course Code: **KDP5G8**
- Submission Format: PDF report + rosbag file + screen recording
- File naming: `assignment4_part1_[netID]/`

2 Demo Resources

2.1 Demo Recording Links

2.1.1 Main Demo Recording

- **Link:** Demo Recording
- **Note:** Only the first video's window was recorded. Audio is present for the Bash script image and PDF slide deck.

2.1.2 Husky Robot Demo (Second Video)

- **Link:** Husky UR3 Gripper Demo
- **Contents:**
  - Husky robot as mobile base
  - UR3 arm with Robotis Gripper
  - Local and global navigation
  - Bash script to launch all files
  - Controllers correctly launched

2.2 Assignment 4 Part 1 Slide Deck PDF

Important

Primary Resource
Latest Version: assignment4_part1_demo_main_VER1.16.pdf
Features: Fully revised, beautifully rendered, no content cutoff
Updated: November 4, 2025 @ 3:28 AM

Alternative Link: Google Drive Slide Deck

3 GitHub Repository

Information

Official Repository

- URL: github.com/kulbir-ahluwalia/husky_ur3_simulator
- Branch: Assignment 4 - Part 1
- Contents: Complete ROS2 package structure with launch files, controllers, and URDF models

3.1 Key Repository Structure

```
husky_ur3_simulator/
|-- launch/
|    |-- gazebo_sim.launch.py
|    |-- controllers.launch.py
|    |-- teleop.launch.py
|-- config/
|    |-- controllers.yaml
|    |-- ros_gz_bridge.yaml
|-- urdf/
|    |-- husky.urdf
|    |-- ur3.urdf
|    |-- gripper.urdf
|-- scripts/
     |-- gripper_controller.py
```

Listing 1: Repository Structure

4 Bash Script for Consistent ROS2 Launches

4.1 Complete Launch Script

Save the following script as `example_bash_script_for_consistent_ros_launch.sh`:

```
#!/bin/bash
# Example Bash Script for Consistent ROS 2 Launch
# CS498GC Mobile Robotics - Assignment 4
# Adapted from ROS 1 Noetic to ROS 2 Jazzy/Humble
# Date: 2025-10-29

# Colors for output
RED=’\033[0;31m’
GREEN=’\033[0;32m’
YELLOW=’\033[1;33m’
NC=’\033[0m’ # No Color
```

```
# ROS 2 workspace paths
ROS_DISTRO="jazzy"  # Change to "humble" if using ROS 2 Humble
ROS_WS="$HOME/ros2_ws"

echo -e "${GREEN}Starting ROS 2 Assignment 4 Launch Sequence...${NC}"

# Check and deactivate conda if active
if [ ! -z "$CONDA_PREFIX" ]; then
    echo -e "${YELLOW}Deactivating conda environment...${NC}"
    conda deactivate 2>/dev/null || true
fi

# STEP 1: Opens up a terminal window 1 - Gazebo Simulation
echo -e "${GREEN}Step 1: Launching Gazebo simulation with Husky + UR3 + Gripper...${
    NC}"
gnome-terminal -- bash -c "source /opt/ros/${ROS_DISTRO}/setup.bash && \
    source ${ROS_WS}/install/setup.bash && \
    ros2 launch husky_ur3_simulation gazebo_sim.launch.py; \
    exec bash"
sleep 5

# STEP 2: Opens a new Terminal window 2 - Robot State Publisher & Controllers
echo -e "${GREEN}Step 2: Launching robot state publisher and controllers...${NC}"
gnome-terminal -- bash -c "source /opt/ros/${ROS_DISTRO}/setup.bash && \
    source ${ROS_WS}/install/setup.bash && \
    ros2 launch husky_ur3_simulation controllers.launch.py; \
    exec bash"
sleep 3

# STEP 3: Opens a new Terminal window 3 - Teleoperation
echo -e "${GREEN}Step 3: Launching teleoperation for mobile base...${NC}"
gnome-terminal -- bash -c "source /opt/ros/${ROS_DISTRO}/setup.bash && \
    source ${ROS_WS}/install/setup.bash && \
    ros2 run teleop_twist_keyboard teleop_twist_keyboard; \
    exec bash"
sleep 3

# STEP 4: Opens a new Terminal window 4 - Gripper Controller
echo -e "${GREEN}Step 4: Launching gripper controller...${NC}"
gnome-terminal -- bash -c "source /opt/ros/${ROS_DISTRO}/setup.bash && \
    source ${ROS_WS}/install/setup.bash && \
    ros2 run husky_ur3_simulation gripper_controller; \
    exec bash"
sleep 3

# STEP 8: Opens a monitoring terminal - Check topics
echo -e "${GREEN}Step 8: Opening monitoring terminal for topic verification...${NC}"
gnome-terminal -- bash -c "source /opt/ros/${ROS_DISTRO}/setup.bash && \
    source ${ROS_WS}/install/setup.bash && \
    echo ’Waiting for topics...’ && sleep 5 && \
    echo ’=== Active Topics ===’ && \
    ros2 topic list && \
    echo ’’ && \
    echo ’=== Topic Frequencies ===’ && \
    timeout 5 ros2 topic hz /odom & \
    timeout 5 ros2 topic hz /imu & \
    timeout 5 ros2 topic hz /scan & \
    wait && \
    exec bash"

echo -e "${GREEN}===========================================${NC}"
echo -e "${GREEN}All terminals launched successfully!${NC}"
echo -e "${GREEN}===========================================${NC}"
```

Listing 2: ROS2 Launch Script

4.2 Script Features

Success

Your bash script automatically launches:

1. **Gazebo simulation** with Husky + UR3 + Gripper
2. **Controllers** for robot control
3. **Teleoperation** for mobile base control
4. **Gripper controller**
5. **Monitoring terminal** to verify all topics

4.3 Running the Script

```
cd /your_directory/demo_assignment4_part1/
chmod +x ./example_bash_script_for_consistent_ros_launch.sh
./example_bash_script_for_consistent_ros_launch.sh
```

Listing 3: Execute Script

The script will:

- Open multiple terminal windows
- Provide colored status messages to track the launch sequence
- Include corrected gripper commands (0.0 for open, 1.05 for close)

5 Deliverables Checklist

5.1 Part 1 Requirements (25 points + up to 20 bonus)

5.1.1 ☑ Rosbag Recording (30 seconds)

```
ros2 bag record -a -o assignment4_[netID] --max-bag-duration 30
```

Information

Required Topics:

- `/cmd_vel` - Velocity commands
- `/odom` - Odometry (50 Hz)
- `/joint_states` - All joint positions
- `/imu` - IMU data (100 Hz)
- `/scan` - Laser scan (10 Hz)
- `/gripper_position_controller/command` - Gripper commands

5.1.2 ☑ Screen Recording (30 seconds)

- **10 seconds:** Mobile base movement (forward, backward, turn)
- **20 seconds:** Gripper demonstration (open and close)
- **Format:** MP4 or MOV
- **Tool:** OBS Studio or QuickTime

5.1.3 ☑ Submission Format

1. Create folder: `assignment4_part1_[netID]/`
2. Include files:
   - `rosbag/` - Contains recorded bag file
   - `screen_recording.mp4` - Screen capture
   - `README.txt` - Brief description

5.2 Part 2 Requirements (75 points)

- MoveIt2 integration
- Navigation stack implementation
- Pick and place demonstration
- Detailed documentation

6 Gripper Control Commands

6.1 Updated Commands (Corrected Values)

⚠ Important

Important: These are the corrected gripper values as discussed in class

6.1.1 Open Gripper

```
ros2 topic pub -1 /rh_p12_rn_position/command std_msgs/msg/Float64 '{data:
    0.0}'
```

6.1.2 Close Gripper

```
ros2 topic pub -1 /rh_p12_rn_position/command std_msgs/msg/Float64 '{data:
    1.05}'
```

7 Troubleshooting

7.1 Common Issues and Solutions

8 Additional Resources

8.1 Course Links

- **Course Website:** kulbir-singh-ahluwalia.com/cs498gc/fa25
- **Assignment 4 Page:** Assignments Page
- **Campuswire:** For questions and discussions
- **Canvas:** For grade tracking

Issue	Solution

Qt5/GPU Errors	`export QT_QPA_PLATFORM=xcb`
	`export LIBGL_ALWAYS_SOFTWARE=1`

Missing Topics	Check ros_gz_bridge configuration: `ros2 param get /ros_gz_bridge config_file`

Transform Errors	`ros2 run tf2_tools view_frames`
	`evince frames.pdf`

Controller Not Loading	`ros2 control list_controllers`
	`ros2 control load_controller gripper_position_controller`
	`ros2 control set_controller_state`
	`gripper_position_controller active`

8.2 Documentation

- **ROS2 Documentation:** docs.ros.org
- **Gazebo Sim:** gazebosim.org
- **MoveIt2:** moveit.picknik.ai

8.3 Office Hours

- **TA Office Hours:** Wednesdays 1:30-2:30 PM @ SC 4407
- **Special Sessions:** Sept 17 & Oct 1 via Zoom

9 Important Notes

⚠ Important

From the Announcement (October 24, 2025)

"Prof Girish and I are both concerned that many students are not starting early enough for Assignment 4 which is the hands-on ROS semester long project for your learning and benefit."

9.1 Key Points

- Start early to maximize learning
- Attend office hours for help
- Ask questions on Campuswire
- Watch demo videos for guidance

10 Contact Information

Last Updated: November 7, 2025

Role	**Contact Details**
Teaching Assistant	Kulbir Singh Ahluwalia Email: ksa5@illinois.edu Website: kulbir-singh-ahluwalia.com
Instructor	Professor Girish Chowdhary Email: girishc@illinois.edu Lab Website: daslab.illinois.edu