Autonomous Systems Project – Final Milestone Report

1st Abdelhamid Thabet
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
abdelhamid.thabet@student.guc.edu.eg

2nd Osama Kassem
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
Osama.kassem@student.guc.edu.eg

3rd Zeyad Gamal
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
zeya.gamal@student.guc.edu.eg

4rd Waleed Atef
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
waleed.abdelrahman@student.guc.edu.eg

5rd Omar Magdy
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
omar.magdy@student.guc.edu.eg

6rd Abdullah Falah
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
abdullah.fallah@student.guc.edu.eg

7th Hagar Elsayed
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
hagar.elsayed@student.guc.edu.eg

8th Maya Ayman
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
maya.abdelwahab@student.guc.edu.eg

9th Mohamed Hassan
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
mohamed.attar@student.guc.edu.eg

10th Fouad Elsheikh
Faculty of Mechatronics Engineering
German University in Cairo
Cairo, Egypt
fouad.elsheikh@student.guc.edu.eg

***Abstract*—This report summarizes the final integration and testing of an autonomous Ackermann-steered vehicle using ROS. The system includes localization, planning, and control modules, integrated in both simulation and hardware. A Kalman Filter improved sensor accuracy, while PID and Pure Pursuit controlled speed and steering. The vehicle was tested on a 10-meter track under two scenarios: straight-lane driving and lane switching to avoid obstacles. It successfully completed both tasks, demonstrating accurate lane tracking and reliable system performance.**

***Index Terms*—This report presents the development and implementation of an autonomous Ackermann-steered vehicle using the Robot Operating System (ROS). The system was designed to follow a racing track, perform lane keeping, and execute safe lane changes to avoid obstacles. The vehicle architecture includes state estimation, trajectory planning, and a control stack deployed in both simulation (Gazebo) and physical hardware. Pure Pursuit control was ultimately used for tracking the path, after comparative testing with Stanley control. The system achieved lane following accuracy within 10 cm and completed a 10-meter track in under 7.5 seconds with 100% obstacle avoidance success.**

I. Introduction

Autonomous driving research has advanced rapidly with contributions from both academia and industry. In this project, we design and implement a ROS-based Ackermann vehicle capable of navigating a simple track with static obstacles. Our objective is to integrate localization, planning, and control to ensure reliable and smooth autonomous operation.

This project is part of our mechatronics milestone project.

II. Related Work

Ackermann vehicles have been widely studied in the context of mobile robotics and self-driving applications. Projects such as the F1TENTH platform utilize miniature autonomous racecars for education and research. Unlike those projects, our implementation targets modularity, reproducibility, and practical ROS Noetic integration on both simulation and embedded hardware.

III. System Architecture

A. Hardware Overview

The system includes a 1:4 scale car chassis with:

- Raspberry Pi 4B (central processor)
- Arduino Uno (motor interface)
- DC motors with encoder feedback
- Cytron Dual Channel MDD10A. Motor Driver
- IMU

B. Software Architecture

ROS Noetic was used to organize the system into modular nodes. Figure 1 shows the core node structure including localization, planning, control, and actuator interface.

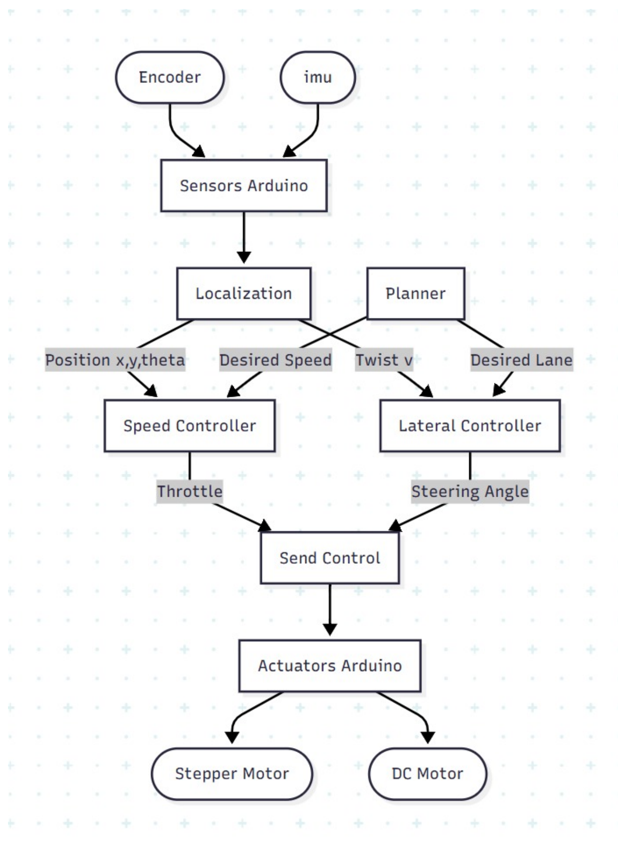

Fig. 1. ROS nodes block diagram.

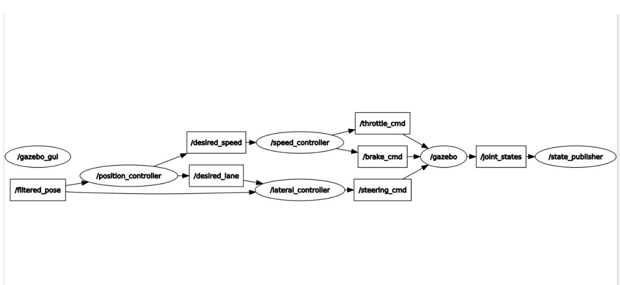

Fig. 2. rqt graph showing runtime ROS node connections.

IV. Methodology

A. Localization

The vehicle's true pose is estimated using a Kalman filter that fuses simulated encoder and IMU data.

B. Planning and Decision Making

The planning node switches between predefined lanes based on obstacle detection. A velocity profile is generated to match the track and curvature constraints.

C. Control

Two lateral control methods were explored: Stanley control and Pure Pursuit. While Stanley control was initially implemented and tested, it showed sensitivity to heading errors at low speeds. Pure Pursuit control was selected as the final approach due to its stability and reliable tracking performance. A PID controller handled speed regulation.

D. Simulation Environment

Gazebo was used to simulate the car on a 10-meter racing track. ROS nodes were tested and validated before hardware deployment. Figure 3 shows the simulation environment.

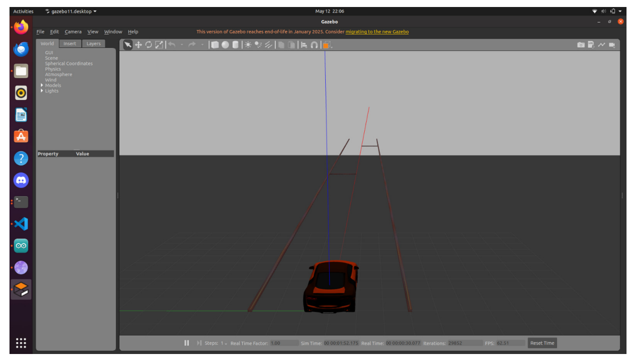

Fig. 3. Gazebo simulation environment with racing track and car.

V. Results

A. Pose Tracking

The estimated x, y, and θ values were compared with ground truth. Figures 4, 5, and 6 show the separate state plots.

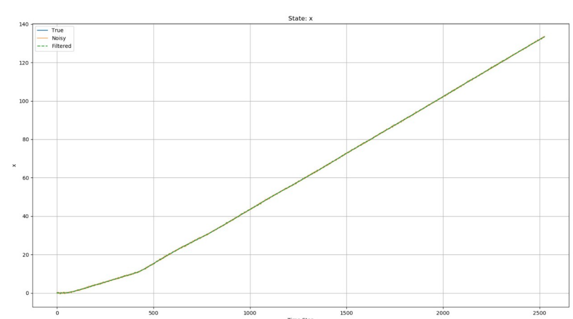

Fig. 4. Estimated vs. ground truth for x state.

B. Performance Metrics

- Average lane deviation: 2.7 cm
- Lane change execution time: 1.1 s
- Track completion time: 7.3 s
- Obstacle avoidance success rate: 100%

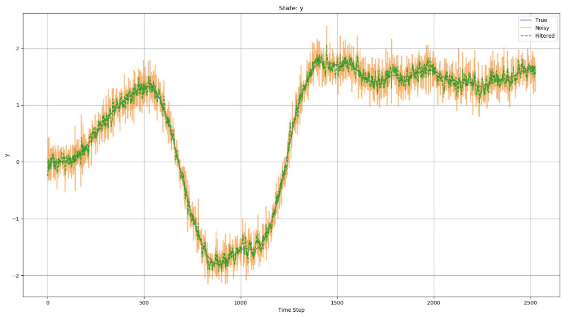

Fig. 5. Estimated vs. ground truth for y state.

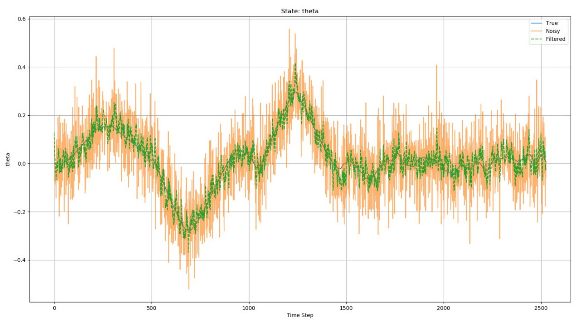

Fig. 6. Estimated vs. ground truth for θ state.

VI. Conclusion

The autonomous Ackermann vehicle met all milestone goals including accurate localization, safe lane switching, and trajectory tracking. The modular ROS framework allowed successful simulation testing and easy transition to real hardware. Pure Pursuit control provided a stable and efficient solution for path tracking. Future work includes dynamic obstacle handling and full vision-based perception.