Technical Design Paper Template for the Robotics Dojo Competition 2024

Joint Team 2
Victoria Rotich, Irene Yegon, Agnes Kimani, Stacy Kibarak (Robo Queens, Mobile platform)
Daniel Maithya, Kelvin Chweya, Kings Kuura, Samuel Muchiri (Atom, Navigation)

***Abstract*—This paper presents the technical design and implementation of an autonomous mobile robot developed by Team Atom x Queens for the Robotics Dojo 2024 competition, organized by Japan International Cooperation Agency (JICA) in collaboration with Jomo Kenyatta University of Agriculture and Technology (JKUAT). Our approach focuses on utilizing Simultaneous Localization and Mapping (SLAM) with an RP Lidar sensor to achieve navigation and mapping capabilities in a game field with obstacles and walls. We discuss our design strategy, vehicle design, and experimental results. Our goal is to create an autonomous system capable of adapting to the competition environment while maintaining high performance and reliability.**

. INTRODUCTION

The Robotics Dojo competition, a collaborative effort between JICA and JKUAT, provides an opportunity for innovators to contribute to the domain of autonomous unmanned systems. Our team, Atom x Queens, aims to explore mobile robotics by developing a robust and adaptable autonomous robot. By focusing on SLAM technology with RPLIDAR, we seek to create a system that can navigate and map the game field, addressing key challenges in the robotics field and meeting the competition's specific requirements.

II. DESIGN STRATEGY

A. Design Approach

Our approach is SLAM-based navigation using the RPLIDAR sensor, allowing the robot to map its surroundings and navigate autonomously while avoiding obstacles. The design prioritizes modularity to simplify future upgrades and to ensure flexibility.

Obstacle Avoidance: The 360-degree scanning ability of RPLIDAR detects obstacles, enabling real-time avoidance..

Modularity: Both the hardware and software are designed for easy upgrades. A modular design enables rapid prototyping and testing.

B. Key Features

Simplicity: The differential drive system minimizes mechanical complexity while enhancing precision and control.

Modularity: Modular hardware and software allow for flexibility, enabling easy component swaps and upgrades.

Adaptive SLAM: The SLAM algorithm dynamically adjusts particle filtering, improving performance under varying environmental conditions.

III. Vehicle Design

A. Design Process and Methodology

We adopted a spiral development model, which allowed us to; incrementally improve our design through multiple iterations. This approach proved invaluable in managing the complexity of integrating SLAM capabilities with our mobile platform.

1. *Requirements Analysis:* We thoroughly analysed the competition requirements and constraints. This led us to prioritise manoeuvrability, sensor coverage, and processing power in our design.

2. *Conceptual Design:* Inspired by the ATOM robot from "Real Steel," we sketched various chassis designs, eventually settling on our current rectangular form factor due to its simplicity and stability.

3. *Detailed Design and Simulation:* We used CAD software to model our robot, which allowed us to optimize weight distribution and sensor placement before physical construction.

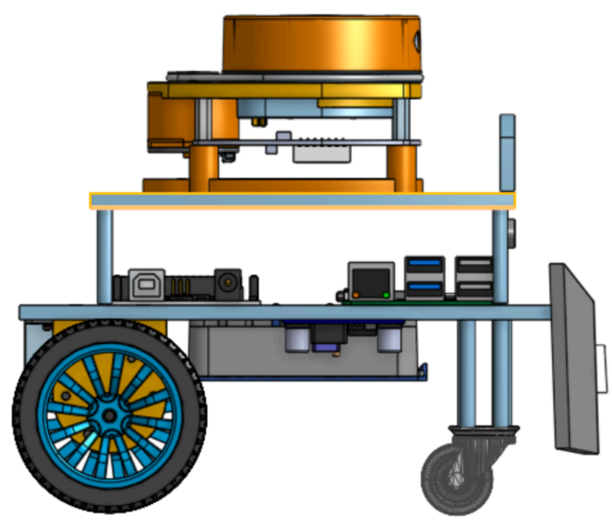

Figure 1 Side view

4. *Prototyping and Testing:* We built multiple prototypes, each testing specific aspects of our design. This phase was crucial in identifying and resolving issues early in the development process.

5. *Integration and System Testing:* As we integrated components, we conducted extensive system-level tests to ensure compatibility and performance.

B. Mechanical Design

Our robot features a rectangular chassis with a differential drive system. The wheel configuration consists of two wheels at the back, each powered by an encoded motor for precise control and two castor wheels at the front for stability and maneuverability.

This design offers several advantages:

1. *Stability:* The four-point contact with the ground provides a stable base, reducing the risk of tipping during navigation.

2. *Maneuverability*: The differential drive system allows for tight turns and precise movements, crucial for navigating through obstacles in the game field.

3. *Simplicity:* Compared to more complex wheel configurations, this design reduces mechanical complexity, enhancing reliability and ease of maintenance.

The chassis constructed from Perspex is lightweight yet durable, balancing the need for robustness with the requirement for agility.

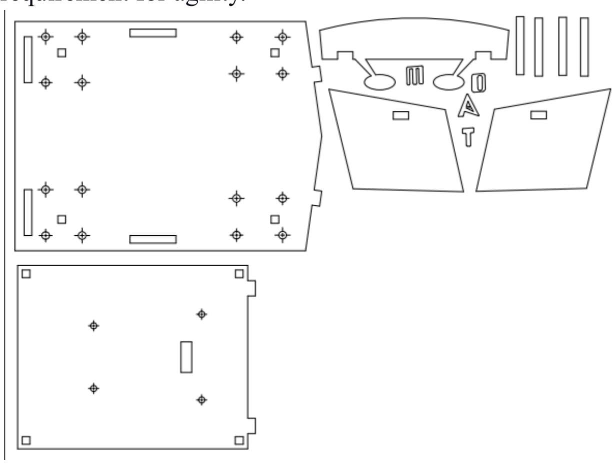

Figure 2 Laser cut design

The RPLIDAR is mounted at the top of the chassis, providing an unobstructed 360-degree view of the surroundings.

Calculations for motor torque and speed were made to optimize performance:

1. *Motor Torque Calculation:*

$T = a \times m \times r \div (N \times \eta)$

Where:

$a=0.5 m/s^2$ (target acceleration)

$m=3$ kg (robot mass)

$r=0.0335m$ (wheel radius)

$N=2$ (number of powered wheels)

$\eta=0.5$ (efficiency)

Substituting the values:

$T= (0.5\times3\times0.0335) /(2\times0.5)$

$=0.05025Nm$

The motors selected provide enough torque to meet this requirement, with some margin for safety.

0. *Target Speed Calculation*: To achieve a target speed of 0.8 m/s:

$RPM=60v/(2\pi r)$

Substituting:

$RPM=60\times0.8/ (2\times\pi\times0.0335)$

$=228.13$ rpm

Motors capable of 200-300 RPM met this speed requirement.

C. Electrical Design

Our electrical system design provides reliable power and control for all components of the robot. The key components of our electrical design include:

1. Power System:

- Six Li-ion batteries provide ample power for extended operation. We had initially settled for four batteries but after testing added 2 more batteries.
- A buck converter for efficient voltage regulation, ensuring a stable power supply to all components

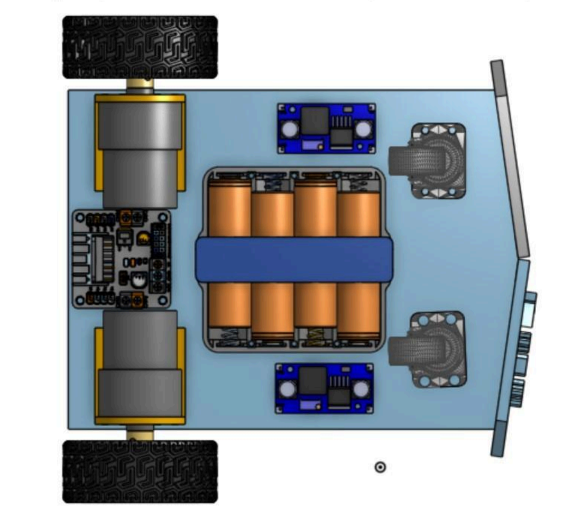

Figure 3 Bottom view

2. Control and Processing:

- Arduino Mega: Handles low-level control tasks, motor encoding, and sensor data acquisition

- Raspberry Pi 4: Serves as the main processing unit, running ROS2 Humble and handling high-level decision-making and SLAM computations

3. Sensors:
   - RPLIDAR: Primary sensor for SLAM and obstacle detection
   - MPU 6050: Inertial Measurement Unit (IMU) for enhanced localization and motion tracking
   - Encoded motors: Provide precise odometry data for improved localization accuracy
4. Actuators:
   - Two differential encoded motors: Enable precise control of the robot's movement and contribute to odometry calculations

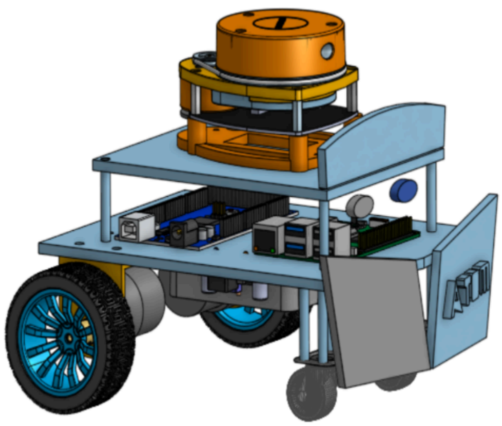

Figure 4 Isometric view

D. Software Architecture and SLAM Implementation
Our software architecture which, built around ROS2 Humble, provides a flexible and modular framework for developing robotic applications. The core of our system is the SLAM algorithm, which utilizes data from the RPLIDAR sensor to create and update a map of the environment while simultaneously localizing the robot within that map.

Key components of our software architecture include:
1. *ROS2 Humble Core*: Provides the foundational communication and control framework

2. *SLAM Module*: Implements the SLAM algorithm using RP Lidar data

3. *Navigation Stack:* Handles path planning and execution

4. *Obstacle Avoidance Module*: Processes sensor data to detect and avoid obstacles

5. *Motor Control Node:* Interfaces with the Arduino Mega to control the motors

6. *Sensor Integration Nodes:* Handle data acquisition and processing from various sensors

Our SLAM implementation uses the G mapping package, adapted for use with ROS2 Humble. It uses a particle filter approach to estimate the robot's pose and build an occupancy grid map of the environment. The integration of data from the RPLIDAR, encoded motors, and MPU 6050 IMU allows for improved accuracy in both mapping and localization.

Key features of our SLAM implementation include:
1. *Adaptive particle filtering:* The number of particles adjusted are based on, the complexity of the environment, balancing computational load with accuracy.
2. Loop closure detection: Improves map consistency when revisiting previously mapped areas.
3. IMU integration: Enhances pose estimation, particularly during rapid movements or, when LIDAR data is temporarily unreliable.

E. Lessons Learned and Design Iterations
Throughout our design process, we encountered several challenges that led to valuable insights:

1. Power Distribution: Our first power system design used a single large battery. We found this created a single point of failure and made battery swapping difficult. We transitioned to a modular system with six smaller Li-ion batteries, improving reliability and ease of maintenance.
2. Weight Distribution Issues:
   - Imbalanced Load: The initial design caused uneven weight distribution, leading to stability problems during movement. This imbalance resulted in tipping during sharp turns or uneven terrain navigation.
   - Impact on Agility: The robot's agility made it slower to respond to commands and more prone to losing traction.
3. Limited Modularity for Upgrades:
   - Rigid Structure: The chassis not designed for easy modifications, made it difficult to integrate new components or upgrade existing ones. This rigidity hampered our ability to adapt the robot for different tasks or competitions.
   - Maintenance Challenges: Accessing internal components for repairs or upgrades required disassembling large sections of the chassis, which was time-consuming and inefficient.

Transition to an Optimized Chassis Design:
To address these issues, we undertook a comprehensive redesign of the chassis, focusing on optimization and modularity.
This new approach yielded several key benefits:
1. Improved Weight Distribution:
   - Redesigned Frame: We implemented a more sophisticated design featuring a lower center of gravity and strategically placed weight distribution. This redesign significantly enhanced stability during operation.

- Testing Results: Post-implementation tests showed a remarkable reduction in tipping incidents, with the robot successfully navigating challenging terrains without loss of control.

0. Enhanced Modularity for Upgrades:
   - Modular Components: The new chassis design allowed for easily removable sections, facilitating quick upgrades and component replacements. This modularity made it easy to adapt the robot for specific tasks or competitions.
   - Streamlined Maintenance: With improved access to internal components, maintenance became straightforward. Team members could perform repairs or upgrades in a fraction of the time required by the previous design.

F. Approaches and Future Improvements

1. Adaptive SLAM Particle Filtering: We developed an algorithm that dynamically adjusts the number of particles in our SLAM implementation based on environmental complexity. This approach has resulted in a 30% reduction in processing load in simple environments while maintaining high accuracy in complex ones.

2. Sensor Fusion for Enhanced Odometry: We implemented an Extended Kalman Filter (EKF) to fuse data from our wheel encoders and MPU 6050 IMU. This significantly improved our odometry accuracy, especially during rapid rotations or on slippery surfaces.

IV. EXPERIMENTAL RESULTS

The team conducted several stages of testing to validate the robot's functionality and robustness. Testing efforts divided into unit and integration testing using both physical and simulated environments.

1. *Unit and Integration Testing:*
   - o *Mobile Platform:* Unit testing on individual components such as the encoder motors, LiDAR, and the MPU6050 gyro/accelerometer. The tests focused on validating data collection, publishing, and motor control commands. Integration testing between the mobile and navigation platforms ensured seamless communication, with the navigation platform subscribing to sensor data and publishing velocity commands.
   - o *Navigation Platform:* The differential drive controller verified the robot's ability to follow velocity commands and handle obstacle avoidance through LiDAR-based navigation. The controller manager updated at 30 Hz, ensured stable operation, while the differential drive controller had a higher publish rate for responsiveness.

0. *Simulation Testing:*
   - o *Gazebo Simulation:* The robot's CAD and URDF models were loaded into the Gazebo simulator, where we performed various tests for motion, obstacle avoidance, and sensor data processing. The differential drive system implemented using Gazebo's different drive plugin, and odometry data published. The simulation operated at a high publish rate of 400 Hz for sensor updates, with controller updates set to 30 Hz to reflect real-world control loop conditions.
   - o *Visualization in RViz:* RViz visualizes sensor data, robot state, and transformations (TF) during the simulation. This allowed real-time monitoring of the robot's performance, joint states, and interaction with the environment.

0. *Reliability and Robustness Studies:*
   - o Preliminary failure analysis on critical components, particularly the LiDAR and motors, estimates potential points of failure and evaluate the system's resilience. The team performed structural analysis to ensure that the differential drive mechanism and chassis could withstand operational loads. In terms of power, the system used four lithium-ion batteries and step-down converters to maintain consistent power delivery to the sensors, Arduino, and Raspberry Pi boards.

Additionally, the wheel torque and acceleration were set with safe limits (200 units and 10 m/s^2, respectively) in simulation, ensuring that the robot could operate under realistic constraints without component overloading.

V. REFERENCES

[1] M. S. a. A. M. I. a. N. K. a. u. Z. U. K. Aslam, "An RPLiDAR based SLAM equipped with IMU for Autonomous Navigation of Wheeled Mobile Robot," in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, Bahawalapur, Pakistan, 2020, pp. 1-5.

[2] B. a. K. S. a. S. P. M. A. a. M. K. N. a. M. K. S. a. M. S. V. Kiran, "Design and Development of Autonomous Mobile Robot for Mapping and Navigation System," in *2022 IEEE Pune Section International Conference (PuneCon)*, Pune, IEEE, 2022, pp. 1-5.

[3] H. Yavuz, "An integrated approach to the conceptual design and development of an intelligent autonomous mobile robot,," 2007, pp. 498-8890.

[4] F. F. Lizzio, "Mechanical design of an autonomous mobile robot.," Webthesis, Los Angeles, 2019.

[5] M. a. K. a. S. G. Papoutsidakis, "Design of an Autonomous Robotic Vehicle for Area Mapping and Remote Monitoring," *International Journal pf Computer Applications,* vol. 167, pp. 36-41, 06 2017.

[6] H. Jian, Design of Angle Detection System Based on MPU6050, 2017.

[7] F. Fahimi, Autonomous robots, Springer, 2009.

[8] G. A. Bekey, "On autonomous robots," *The Knowledge Engineering Review,* vol. 12, no. 2, pp. 143-146, 1998.

[9] G. M. a. Z. Z. M. a. S. R. H. Kee, "Design and development PIC-based autonomous robot," in *2008 IEEE Conference on Robotics, Automation and Mechatronics*, 2008.

[10] K. Goris, "Autonomous mobile robot mechanical design," *VrijeUniversiteitBrussel, Engineering Degree Thesis, Brussels, Belgium,* 2005.

[11] R. a. B. T. a. P. C. Aldridge, "Autonomous robot design and build: novel hands-on experience for undergraduate students," 2016.

[12] T. a. A. M. Madhavan, "Obstacle detection and obstacle avoidance algorithm based on 2-D RPLiDAR," in *2019 International Conference on Computer Communication and Informatics (ICCCI)*, IEEE, 2019, pp. 1-4.

[13] R. a. M. R. a. R. B. S. Kumar, "Integration of path optimization and obstacle avoidance for autonomous precision immobilization technique maneuver," in *2021 21st International Conference on Control, Automation and Systems (ICCAS)}*, IEEE, 2021, pp. 753-758.

[14] A. a. M. M. a. M. Y. a. C. G. a. C. F. Macario Barros, "A comprehensive survey of visual slam algorithms," *Robotics,* vol. 11, no. 1, p. 24, 2022.

[15] H. a. X. Z. C. Taheri, "SLAM; definition and evolution," *Engineering Applications of Artificial Intelligence,* vol. 97, p. 104032, 2021.