Defence Science Journal, Vol. 75, No. 6, November 2025, pp. 751-757, DOI : 10.14429/dsj.19596
© 2025, DESIDOC

Simulation of an Autonomous Radiation Detection Robot in Gazebo and Webots using SLAM and A-Star Algorithm

Chanthini Baskar[#], V. Adithya[$], Sasithradevi Anbalagan[$,*] and P. Prakash[!]
[#]School of Electronics Engineering, Vellore Institute of Technology, Chennai - 632 014, India
[$]Centre for Advanced Data Science, Vellore Institute of Technology, Chennai - 632 014, India
[!]Department of Electronics Engineering, Madras Institute of Technology, Chennai - 600 044, India
[*]E-mail: sasithradevi.a@vit.ac.in

ABSTRACT

In recent years, there has been a rise in demand for autonomous robots that can perform hazardous tasks, such as nuclear radiation detection. These robots can reduce the danger of human exposure to radiation and improve the efficiency of radiation detection operations. For the development of such robots, it is essential to have simulation environments that accurately mimic real-world conditions. This research paper provides a comprehensive description of the simulation of an autonomous robot for radiation detection in an unknown virtual environment using simultaneous localization and mapping and the A-star algorithm. This paper seeks to validate and evaluate the effectiveness and robustness of the simulation of an autonomous radiation detection robot using the two most prominent and robust robot simulators, Gazebo and WEBOTS.

Keywords: Simultaneous localization and mapping; Robotic operating system; Unified robot description format; Adaptive monte carlo localization

NOMENCLATURE

AMCL	: Adaptive monte carlo localization
EKF	: Extended kalman filter
GBAD	: Ground-based air defence
PSO	: Particle swarm optimization
RDE	: Rotating disc electrode
ROS	: Robotic operating system
SLAM	: Simultaneous Localization and Mapping
URDF	: Unified robot description format

1. INTRODUCTION

The detection and monitoring of radiation in hazardous environments is a critical issue, especially in settings such as nuclear facilities and disaster zones, where human intervention is limited due to the danger posed by radiation exposure[1]. To address this challenge, specialized robots capable of autonomous navigation and radiation source detection are required. In recent years, simulation tools have become an integral part of the development and testing process for such robots[2]. These tools allow researchers to evaluate and refine the robot's behavior in a virtual environment before physical deployment, ensuring that potential issues can be identified and addressed early[3]. Modern robot simulators, such as Gazebo and WEBOTS, are equipped with powerful physics simulation engines, high-quality graphics, and user-friendly interfaces[4]. These features make it possible to replace physical systems with simulation models, offering a cost-effective and safe alternative to validate and evaluate the performance of conceptual robot designs. This approach helps in testing and refining the robots' functions under controlled, simulated radiation environments[5]. The need for robots capable of safely operating in hazardous environments has become more pronounced with the increasing demand for autonomous systems in critical sectors like nuclear energy and disaster response[6]. These robots must be equipped to navigate complex environments, detect radiation sources, and avoid obstacles autonomously[7]. However, the traditional approach of physical testing is often costly and risky, especially in environments where radiation is present. Simulation tools offer an effective solution by providing a controlled environment in which robot performance can be tested and optimized without the risks associated with real-world trials[8-10]. This work is motivated by the necessity of an accurate simulation technique that can represent the behavior of radiation detection robots in such environments. By leveraging simulation, we can ensure that the robots are not only capable of detecting radiation but also adept at navigating and avoiding obstacles effectively.

The robot will be tested in various scenario-based simulations designed to evaluate its mapping, localization, and radiation detection capabilities. These tests will help assess the system's robustness and efficiency before the robot is deployed in real-world hazardous environments. Our objective is to develop a reliable and efficient system for autonomous radiation detection that integrates

SLAM, A-star, and radiation sensors, with validation carried out through comprehensive simulation[11-14]. The steps

Received : 20 November 2024, Revised : 15 July 2025
Accepted : 18 August 2025, Online published : 06 November 2025

for installing ROS on our Ubuntu system are outlined in (ROS)[15]. References to (SLAM)[16] and (Turtlebot3)[17] describe how SLAM is implemented in Gazebo and ROS using the Turtlebot3. Discussions on testing a SLAM-enabled mobile robot's mapping and traversal capabilities in enclosed spaces have been explored[1]. The evaluation involves creating a robotic model using Robot Operating System (ROS) as the foundation, with Gazebo for robot modeling and Rviz for simulation. The mobile robot, built in the Gazebo environment using ROS, incorporates novel navigation systems for mapping and localization[2-3]. The experimental platform utilizes new hardware and sensors for chassis navigation, integrating inertial and odometer data through the Extended Kalman Filter. The mapping process determines the environmental map, while Dynamic Window Method and A* method are employed for local and global path scheduling, respectively. The developed robot navigation system is tested on an auxiliary operation robot chassis, demonstrating feasibility and effectiveness in a power distribution cabinet switch scenario[4]. The primary objectives of this work are as follows:

- **Implement the SLAM Algorithm**: To enable the robot to autonomously navigate and create an accurate map of an unknown environment.
- **Integrate the A-Star Algorithm**: To enable the robot to plan and follow an optimal path while avoiding obstacles.
- **Simulate Radiation Sources**: To create a virtual environment where radiation sources can be detected, allowing the robot to identify hazardous areas.
- **Conduct Simulation Experiments**: To validate the robot's navigation and radiation detection performance through various scenario-based assessments.

These objectives aim to contribute to the advancement of autonomous robots designed for radiation detection in hazardous environments. Through simulation, we hope to refine the design and functionality of these robots, ensuring that they are capable of performing their tasks efficiently and safely in real-world applications.

2. LITERATURE REVIEW

The development of autonomous robots for radiation detection and navigation [18–23] in hazardous environments has become a significant area of research, especially with the growing need for robots that can operate safely in settings like nuclear facilities and disaster zones. Several studies have explored the use of Simultaneous Localization and Mapping (SLAM) algorithms, pathfinding methods, and robot simulators to address the complexities of autonomous navigation in such environments. The core challenge of autonomous navigation in unknown environments is real-time localization and map generation. SLAM enables robots to create maps of an environment while simultaneously localizing themselves within it. Numerous studies have implemented SLAM using various sensors, including LIDAR and RGB-D cameras, to aid in environmental mapping. For instance, Kannan Megalingam[1], *et al.* demonstrate a ROS-based SLAM implementation for indoor navigation using Turtlebot3, where the Gmapping algorithm is utilized for real-time mapping. Similarly, Ren*, et al.* [2] explored the application of SLAM for an indoor wheeled robot navigation system in a substation, highlighting the efficiency of SLAM in structured indoor environments.

The Gmapping algorithm, based on FastSLAM, has been widely used in mobile robotics due to its ability to handle real-time map generation and localization[4]. It is particularly effective in static environments, making it a popular choice for autonomous indoor navigation tasks. In our work, we adopt Gmapping to build the robot's map and overcome issues related to wheel slippage and drift by using an odometer reference, as proposed by Chikurtev[3].

Once a robot is localized, path planning becomes critical for navigation. The *A (A-star)** algorithm is one of the most well-known methods for determining the shortest path between two points while considering obstacles. The work of Liu[18]„ *et al.* integrates a kinematically constrained A-star algorithm with the Dynamic Window Approach (DWA) to optimize robot navigation, making it more efficient in environments with dynamic obstacles. In the context of radiation detection, pathfinding algorithms like A* enable robots to navigate toward hazardous sources while avoiding obstacles. Our work builds on this foundation by combining A* with AMCL (Adaptive Monte Carlo Localization) for real-time localization correction. This combination is particularly suited for dynamic and cluttered environments, where drift from odometer readings can affect accuracy. The ability of AMCL to adapt to dynamic obstacles and provide robust localization has been validated in multiple studies, including Afanasyev*,, et al.* [7], who explored its application in image-based 3D environments using ROS.

Testing autonomous robots in real-world environments, especially those that are hazardous, is challenging. Therefore, simulation tools such as Gazebo and WEBOTS are extensively used to create controlled environments where robots can be tested. Gazebo, with its high-fidelity physics engine, provides a realistic simulation of robot-environment interaction, which is essential for testing SLAM and path planning algorithms. Several studies, such as Pramod Thale[6]*, et al.*, have used Gazebo for SLAM-based robot navigation in indoor environments. In our study, Gazebo is utilized to simulate a dynamic environment where the robot builds a map and navigates using SLAM. In addition to Gazebo, WEBOTS is used for its simplicity and ease of integrating sensor-based models. The E-Puck robot in WEBOTS has been widely used for simulating mobile robots equipped with various sensors, including proximity sensors and encoders[8]. WEBOTS's ability to simulate robots in a 3D environment, alongside its integration with ROS, makes it ideal for testing tasks like radiation detection. Ali[5]*, et al.* used WEBOTS for simulating a Turtlebot2 in a 2D indoor environment using the Cartographer SLAM algorithm, demonstrating the importance of simulator-based testing for real-world navigation applications.

In environments where radiation sources need to be detected, robots must be able to localize the sources and navigate accordingly. Several approaches have been proposed for using robots to detect radiation, including using swarm robots for radiation search and localization. Bashyal and Venayagamoorthy[9] investigated the use of a swarm of robots for locating radiation sources in unknown environments, leveraging Particle Swarm Optimization (PSO) for optimal path planning. The hybridization of algorithms like A* and PSO has shown promise in improving the efficiency of radiation source localization. The integration of multiple algorithms has been a key research direction. Zhang[10]*, et al.* propose a Hybrid IACO-A-PSO optimization algorithm* for multi-objective path planning in radioactive environments, demonstrating the effectiveness of combining evolutionary algorithms with pathfinding techniques.

In our work, we adapt this approach by integrating A* and AMCL for navigation and localization, optimizing the robot's ability to navigate toward radiation sources while avoiding obstacles in dynamic environments. Simulating radiation sources in virtual environments is essential for testing radiation detection capabilities. In WEBOTS, Point Light sources are modified to represent radiation, allowing for the simulation of radiation behaviour and its interaction with the environment[12]. Crespo[11] reviewed electrodeposition methods for preparing radiation sources, which can be used to model the radiation levels in simulation environments for robots tasked with detection. The main research gaps exist in the literature are: Lack of Active SLAM integration, Poor integration of radiation mapping with path planning and Insufficient handling of sensor noise and localization errors under radiation simulation.

3. METHODOLOGY

3.1 Simultaneous Localization and Mapping (SLAM) Algorithm

Simultaneous Localization and Mapping (SLAM) is a fundamental task in robotics, enabling a robot to traverse an unknown environment while simultaneously building a real-time map of that space. In our work, we use Gmapping, an open-source implementation based on FastSLAM. Gmapping is widely used for mobile robot localization and mapping, particularly effective in static environments. It operates in real-time and integrates sensor data such as LiDAR or RGB-D cameras, which are simulated in Gazebo for our experiments. The choice of Gmapping is justified by its proven ability to create accurate maps while requiring relatively low computational resources, which is ideal for testing in simulated hazardous environments. The method was implemented in Gazebo, an open-source robot simulator that integrates seamlessly with the Robot Operating System (ROS), providing an effective framework for SLAM. Gazebo is particularly well-suited for testing SLAM algorithms because of its high-fidelity physics engine and realistic 3D modeling capabilities, which allow for the simulation of complex environments where robots must map and localize in real-time. The SLAM method in Gazebo and ROS is used to build the map in Fig. 1.

Simulation Parameters:

- Resolution: The map generated by SLAM is set with a resolution of 0.05 meters to ensure a balance between computational efficiency and the level of detail necessary for accurate navigation.
- Sensor Input: LIDAR and RGB-D cameras are simulated in Gazebo to generate 3D point clouds and depth maps for the robot to process in real-time.

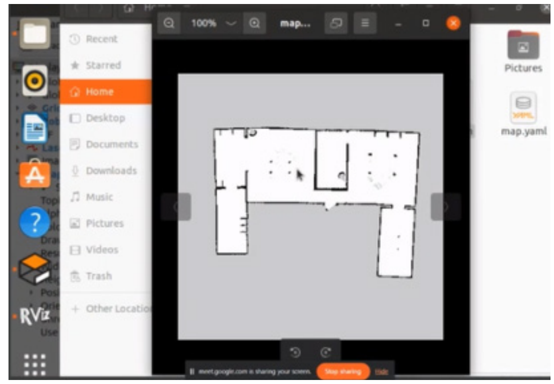

Figure 1. SLAM generated map of the sample environment.

3.1.1 A-STAR Algorithm

The *A (A-star)** algorithm is a well-established pathfinding and graph traversal algorithm that is used to determine the shortest path between two points in a graph. We chose A* for its ability to compute optimal paths while considering obstacles and varying terrain, making it an excellent choice for autonomous navigation in complex environments. *A's** efficiency is enhanced by using Euclidean or Manhattan distance as the heuristic function, depending on the environment's topology, to guide the search process[24]. The choice of A-star is validated by its widespread use in mobile robotics, where it has been proven to be both efficient and effective in a range of pathfinding scenarios[25]low planning efficiency, and insufficient verification of theoretical algorithms. Therefore, a motion control system for an intelligent indoor robot was designed. By optimizing the radar map detecting and positioning, path planning, and chassis motion control, the performance of the system has been improved. First, a map of the warehouse environment is established, and the number of resampling particles interval is set for the Gmapping building process to improve the efficiency of map construction. Second, an improved A* algorithm is proposed, which converts the path solution with obstacles between two points into the path solution without obstacles between multiple points based on the Rapidly expanding Random Trees and Jump Point Search algorithms and further improves the pathfinding speed and efficiency of the A* algorithm by screening the necessary expansion nodes. The Dynamic Window Approach (DWA. From the Fig. 2, we could see the A-Star algorithm finding the most optimistic path to destination.

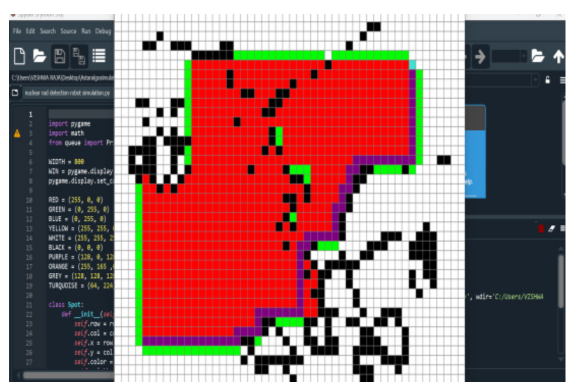

Figure 2. A comprehensive simulation of A-star algorithm using python.

Cost (node, neighbor) is the expenditure to migrate from node to neighbor. f(n) = g(n) + h(n) is the formula for the f-score. Between the node and the objective, the heuristic function h(n) is typically the Euclidean distance or the Manhattan distance. The formula for Euclidean distance is as follows:

$$h(n) = sqrt\,((x_{goal} - x_n)^2 + (y_{goal} - y_n)^2) \quad (1)$$

where, x_{goal} and y_{goal} are the goal's coordinates and x_n and y_n are the node's coordinates. The distance to Manhattan is determined by the formula:

$$h(n) = abs\,(x_{goal} - x_n) + abs\,(y_{goal} - y_n) \quad (2)$$

where, abs is the function for absolute value. The use of two simulators enables us to cross-validate results, ensuring that our simulations are reliable and reproducible. Gazebo is primarily used for large-scale, high-fidelity simulations of complex environments, while Webots is utilized for more straightforward, sensor-based testing in controlled scenarios.

3.2 Adaptive Monte Carlo Localization (AMCL)

It is a popular open-source robot localization algorithm in ROS. AMCL is a form of particle filter algorithm that employs a probabilistic method for estimating the robot's pose[26]. Based on the Monte Carlo Localization (MCL) algorithm, it incorporates adaptive resampling to enhance performance. AMCL necessitates a map of the surrounding environment and sensor data from sources such as LiDAR or RGB-D cameras[27]. AMCL facilitates real-time, global localization for robots in dynamic environments. It ensures precision using adaptive resampling techniques and operates efficiently on mobile platforms. Widely integrated into ROS-based navigation systems, AMCL plays a crucial role in robotics research and enables precise localization even in unfamiliar surroundings.

4. RESULTS AND DISCUSSION

The Fig.3 shows the experimental and environmental setup created to simulate the robot in Gazebo.

4.1 SLAM Implementation and Mapping Accuracy

The simulation successfully implemented SLAM in Gazebo using Gmapping, which generated accurate 2D maps via LIDAR even amid dynamic obstacles. Although minor map distortions occurred due to wheel slippage, these were mitigated by referencing odometry corrections. We visualized SLAM node interactions with RQT Graph, confirming proper data flow. This detailed scrutiny revealed optimization opportunities, particularly for handling non-static environments and uneven terrain. These results demonstrate that a robust SLAM implementation—combining LIDAR mapping, drift correction, and system monitoring—can autonomously map unknown spaces and support operations like identifying safe routes or contamination zones in nuclear or disaster-affected areas.

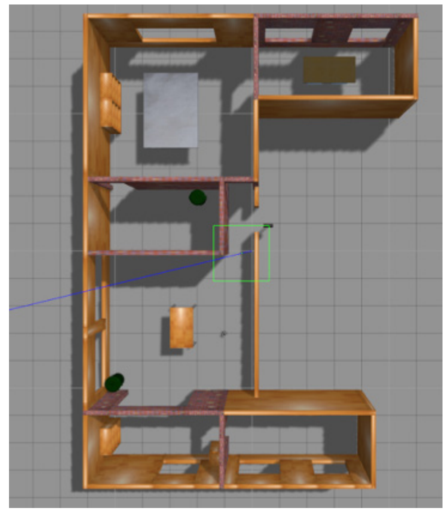

Figure 3. Experimental setup.

4.2 Pathfinding with A Algorithm

We integrated A* pathfinding with AMCL localization to enable real-time navigation in dynamic, obstacle-rich environments. A* efficiently plans shortest paths, while AMCL continuously corrects pose drift from odometry slip, allowing adaptive navigation through tight spaces and moving obstacles. This hybrid system proved robust in simulation and applicable to hazardous environments. Figure 4 shows the step-by-step implementation of A-Star in Gazebo and ROS.

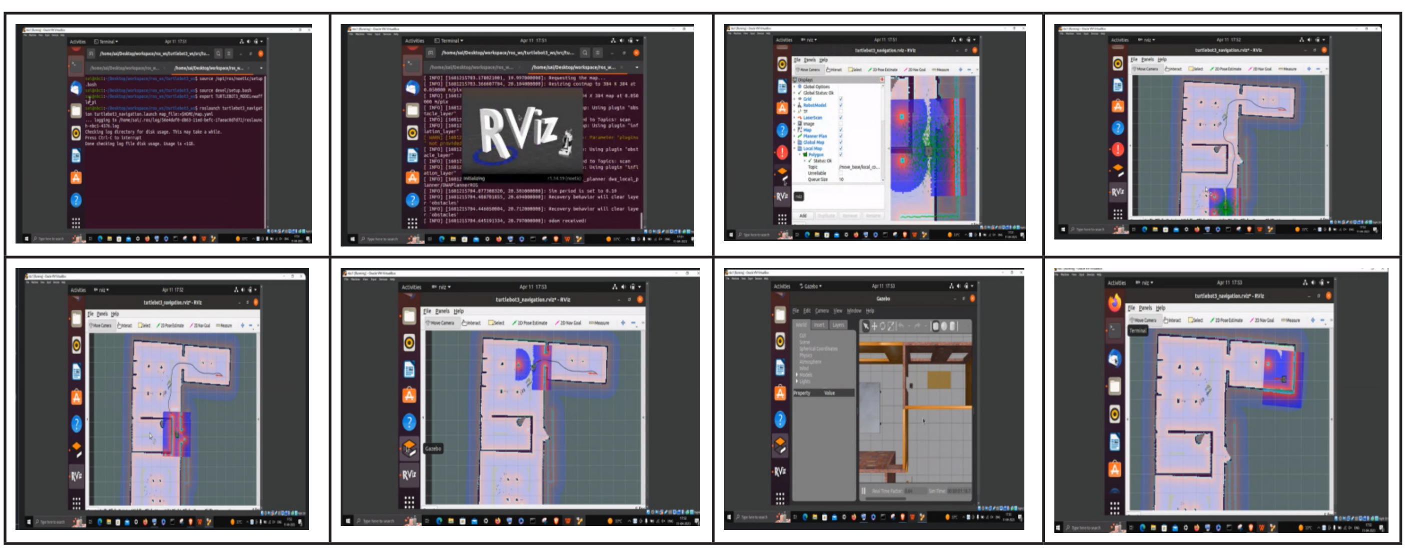

Figure 4. A step-by-step implementation of A-star path-planning in Gazebo.

4.3 Radiation Detection and Robot Behavior

We integrated a custom radiation sensor onto a WEBOTS-simulated EPuck, using Recursive Wavefront path planning to autonomously seek radiation sources. As radiation readings increased, the robot adapted its trajectory in real time, validating its ability to detect and respond to hazards. This demonstrates strong potential for deployment in unpredictable, radiation-prone environments.

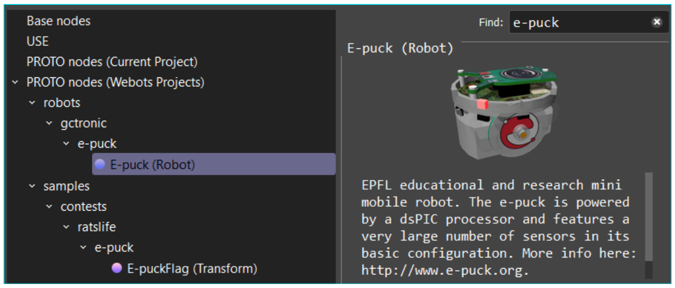

Figure 5. E-puck robot.

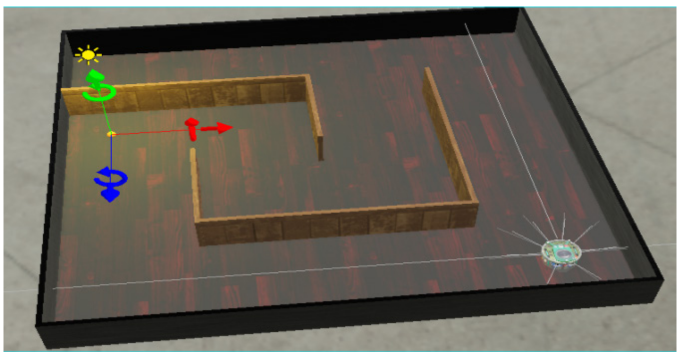

Figure 6. Radiation source.

Figure 5 gives an overview of the E-Puck robot. The “Scene Tree Window” in WEBOTS is used to build the radiation source. The modified version of Point Light in the WEBOTS library is the source of the radiation. By studying various radiation sources and their ability to penetrate barriers, adjustments are created. The E-puck robot is equipped with a custom-made sensor plugin that serves as a radiation detector during the simulation. Figure 6 shows the modified pointlight source that acts as a radiation source.

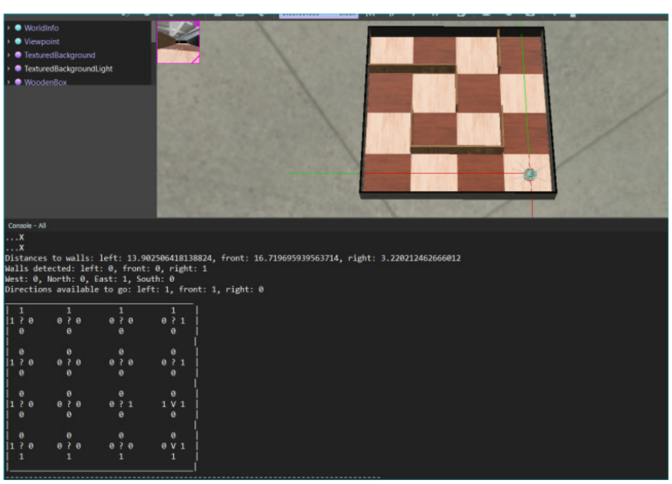

Figure 7. SLAM implementation in WEBOTS.

WEBOTS’ “Scene Tree window” is used to create the simulation’s Environment. It is a straightforward rectangular arena. To make the simulation’s geography more complicated, walls are included. The Environment is mapped using the SLAM technique in Fig. 7. This algorithm was run using cameras and lidar. Following that, the map is kept in the robot’s memory. In the environment, a radiation source is installed. The radiation detector will activate the robot, and the direction will be determined based on the data. Using the recorded map and the Recursive Wavefront technique, a path to the source is found. The robot will then go down the path until it arrives at its objective.

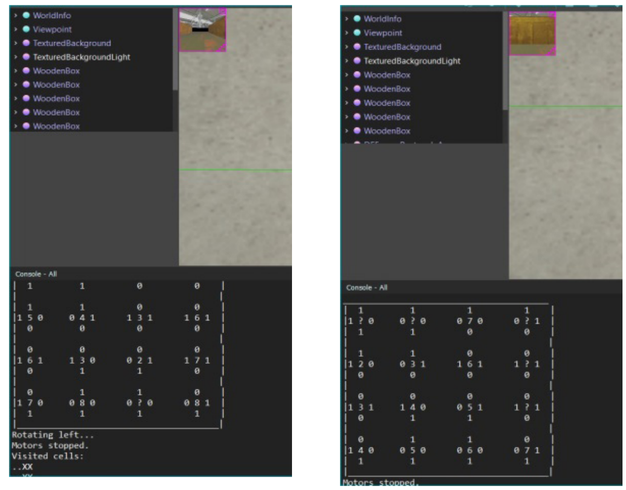

Figure 8. Testing and validation in two scenarios.

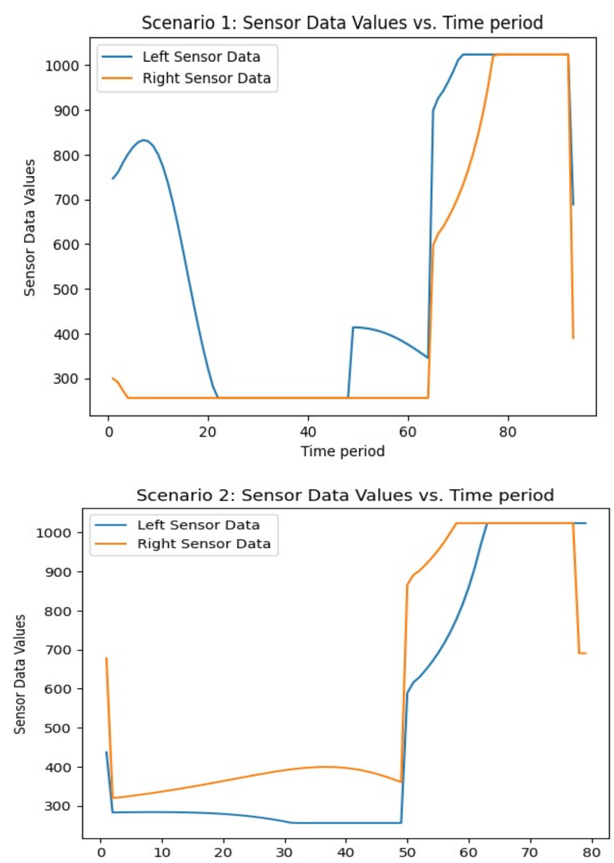

Figure 9. Radiation sensor data from two scenarios.

4.4 Testing and Validation

We evaluated the robot in two simulated scenarios (Fig. 8), tracking radiation sensor readings (Fig. 9) as it navigated toward and away from the source. Readings increased when approaching and decreased when retreating, and the robot adjusted its path to avoid obstacles, consistently maintaining adaptability and reliable hazard response in dynamic environments.

5.5 Implications and Future Directions

We successfully simulated an autonomous radiation-detection robot in Gazebo and Webots, enabling navigation through dynamic, cluttered environments using a hybrid SLAM + A* + AMCL approach and a custom radiation sensor for real-time hazard response. The robot demonstrated effective obstacle avoidance, adaptive path adjustments toward radiation sources, and accurate detection capabilities, showcasing the potential of these simulation platforms for complex robotic tasks These results underscore the promise of deploying autonomous robots in safety-critical scenarios—such as nuclear facilities and disaster zones—with future work aimed at sensor accuracy improvements and transitioning to real-world testing.

5. CONCLUSION AND FUTURE WORK

This study leveraged Gazebo and WEBOTS simulations to create and test an autonomous radiationdetection robot integrating SLAM, A*, and AMCL. The system used LIDAR for mapping, while a custom radiation sensor triggered realtime path adjustments. Gmapping SLAM enabled accurate map generation in complex environments, despite minor odometry drift corrected via odometer referencing and AMCL localization. Communication nodal flow was tracked using RQT Graph, supporting data integrity and pinpointing optimization needs. This hybrid framework demonstrated robust navigation and precise localization in cluttered settings, successfully guiding the robot toward radiation sources. Our results validate that combining SLAM, A*, and AMCL with a responsive radiation sensor is effective for hazardous environment navigation. These findings align with recent advancements in multisensor SLAM and hybrid navigation methodologies The approach holds promise for real-world applications in nuclear facilities and disaster zones. Future work will refine algorithmic performance in dynamic settings and move toward real-world validation.

REFERENCES

1. Megalingam RK, Chinta R, Sreekanth S, Raj A. ROS based Autonomous Indoor Navigation Simulation Using SLAM Algorithm. 2019.
2. Ren J, Wu T, Zhou X, Yang C, Sun J, Li M,, *et al.* SLAM: Path planning algorithm and application research of an indoor substation wheeled robot navigation system. Electronics [Internet]. 2022, Jun 9;11(12),1838.
3. Chikurtev D. Mobile robot simulation and navigation in ROS and Gazebo. In: 2020 International Conference Automatics and Informatics (ICAI) [Internet]. IEEE; 2020. pp. 1–6.
4. Marshal Revanth C, Saravanakumar D, Jegadeeshwaran R & Sakthivel G. Simultaneous localization and mapping of mobile robot using GMapping algorithm. In: 2020 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS) [Internet]. IEEE; 2020. pp. 56–60.
5. Ali W, Ahmed W. Robot Operating System-Based SLAM for a Gazebo-Simulated Turtlebot2 in 2d Indoor Environment with Cartographer Algorithm. 2021.
6. Pramod Thale S, Mangesh Prabhu M, Vinod Thakur P & Kadam P. ROS based SLAM implementation for Autonomous navigation using Turtlebot. Patil MD, Vyawahare VA, editors. ITM Web Conf [Internet]. 2020 Jul 29;32:01011.
7. Afanasyev I, Sagitov A, Magid E. ROS-Based SLAM for a Gazebo-Simulated Mobile Robot in Image-Based 3D Model of Indoor Environment. In 2015. pp. 273–83.
8. Adam YM, Binti Sariff N & Algeelani NA. E-puck Mobile Robot Obstacles Avoidance Controller Using the Fuzzy Logic Approach. In: 2021 2nd International Conference on Smart Computing and Electronic Enterprise (ICSCEE) [Internet]. IEEE; 2021. pp. 107–12.
9. Bashyal S & Venayagamoorthy GK. Human swarm interaction for radiation source search and localization. In: 2008 IEEE Swarm Intelligence Symposium [Internet]. IEEE; 2008. pp. 1–8.
10. Zhang D, Yin Y bo, Luo R & Zou S liang. Hybrid IACO-A*-PSO optimization algorithm for solving multiobjective path planning problem of mobile robot in radioactive environment. Prog Nucl Energy [Internet]. 2023 May,159,104651.
11. Crespo MT. A review of electrodeposition methods for the preparation of alpha-radiation sources. Appl Radiat Isot [Internet]. 2012, 70(1), 210–5.
12. Stary V & Gacho L. Webots open source robot simulator capabilities for modelling and simulation of ground-based air defence. In: 2022 20th International Conference on Mechatronics - Mechatronika (ME) [Internet]. IEEE; 2022. pp. 1–5.
13. Stibinger P, Baca T & Saska M. Localization of ionizing radiation sources by cooperating micro aerial vehicles with pixel detectors in real-time. IEEE Robot Autom Lett [Internet]. 2020, 5(2), 3634–41.
14. Wright T, West A, Licata M, Hawes N & Lennox B. Simulating ionising radiation in Gazebo for robotic nuclear inspection challenges. Robotics [Internet]. 2021 Jul ,10(3), 86.
15. SLAM [Internet]. Available from: https://medium.com/@olesyakrindach/testing-different-slam-algorithms-with-turtlebot3-simulation-34c741db96ea
16. Olalekan AF, Sagor JA, Hasan MH & Oluwatobi AS. Comparison of Two SLAM Algorithms Provided by ROS (Robot Operating System). In: 2021 2nd International Conference for Emerging Technology (INCET). 2021. pp. 1–5.
17. Juneja A, Bhandari L , Mohammadbagherpoor H, Singh A & Grant E. A Comparative Study of SLAM Algorithms for Indoor Navigation of Autonomous Wheelchairs. In:

2019 IEEE International Conference on Cyborg and Bionic Systems (CBS). 2019. pp. 261–6.

18. Fan C, Wei S & Yang Y. file:///C:/Users/Admin/Downloads/citation-339311999.ris. In: 2024 International Conference on Electrical Drives, Power Electronics & Engineering (EDPEE). 2024. pp. 624–9.
19. Abdul Rahman N, Sahari K, Hamid N & Cheong Hou Y. A coverage path planning approach for autonomous radiation mapping with a mobile robot. Int J Adv Robot Syst., 2022, 19, 172988062211164.
20. Takaya K, Asai T, Kroumov V & Smarandache F. Simulation environment for mobile robots testing using ROS and Gazebo. In: 2016 20th International Conference on System Theory, Control and Computing (ICSTCC) [Internet]. IEEE, 2016. pp. 96–101.
21. Rivera ZB, De Simone MC & Guida D. Unmanned ground vehicle modelling in Gazebo/ROS-based environments. *Machines* [Internet]. 2019, 7(2), 42.
22. Liu Y, Wang C, Wu H & Wei Y. Mobile robot path planning based on kinematically constrained A-Star algorithm and DWA fusion algorithm. *Mathematics,* 2023 5, 11(21), 4552.
23. Mi Z, Xiao H & Huang C. Path planning of indoor mobile robot based on improved A* algorithm incorporating RRT and JPS. *AIP Adv* 2023, 1, 13(4).
24. He F & Zhang L. Design of indoor security robot based on robot operating system. J Comput Commun, 2023,11(05), 93–107.
25. Trybała P, Szrek J, Dębogórski B, Ziętek B, Blachowski J, Wodecki J„ *et al.* Analysis of lidar actuator system influence on the quality of dense 3D point cloud obtained with SLAM. Sensors, 2023, 23(2), 1–19.

CONTRIBUTORS

Dr Chanthini Baskar obtained his PhD in the area of Internet of Things from SASTRA Deemed University, Thanjavur and working as Associate Professor in Vellore Institute of Technology, Chennai.

In the current study, she developed and validated the Gazebo and Webots simulation environments, including radiation source modeling and SLAM sensor setup.

Mr V. Adithya is an undergraduate student of School of Electronics and Communication Engineering, Vellore Institute of Technology, Chennai.

In the current study, he implemented and finetuned the SLAM pipeline (e.g., AMCL/Gmapping) in ROS, processed sensor data, and optimized mapping accuracy.

Dr Sasithradevi Anbalagan obtained her PhD in the area of Video Retrieval from Anna University and working as Associate Professor in Vellore Institute of Technology, Chennai.

She conceived the project, designed the SLAMbased radiation detection framework, and led integration of the A* algorithm for optimal robot path planning.

Dr P. Prakash obtained PhD in the area of Signal Processing at Anna University and working as Professor in the Department of Electronics Engineering, MIT Campus, Anna University, Chennai.

In the current study he conducted performance analysis of navigation via AStar, compared planned vs. executed paths, and drafted results/discussions on algorithm efficiency and realworld implications.