**VACUUM CLEANING ROBOT**

A PROJECT REPORT

***Submitted by***

BL.EN.U4AIE19007 APOORVA.M

BL.EN.U4AIE19041 TANUJ.M

BL.EN.U4AIE19068 AISHWARYA.V

***for the course***

***19AIE213 – Introduction to Robotics***

***in partial fulfilment for the award of the degree***

***0f***

**BACHELOR OF TECHNOLOGY**

IN

ARTIFICIAL INTELLIGENCE ENGINEERING



AMRITA SCHOOL OF ENGINEERING, BANGALORE

**BANGALORE 560 035**

**PROBLEM STATEMENT**

Design and implement an vacuum cleaning robot using a custom-built differential drive robot. The robot should traverse to all the locations of the environment, the boundaries of the environment should not be exclusively available it can be indicated by walls indirectly, also assume some static objects placed in the environments. The type of sensors can be team’s choice but the reason to use the sensor should be clear in the report and the presentation.

**PROBLEM STATEMENT EXPLANATION:**

In this problem we can see the working of a vacuum cleaning robot implemented in ROS. The simulation is done in Gazebo and visualisation is done in rviz. We can build the cleaning robot using any differential drive robot. Complete coverage path planning algorithms can be used for the robot to traverse the whole room. There are some static objects placed in room. The cleaning robot needs to make sure that it doesn’t hit the obstacles while cleaning room using sensors and it tries to cover the maximum area in a given environment,

**THEORY**

**PATH PLANNING:**

Path planning and path tracking strategies for mobile robot navigation are highly dependent on the target application. Among those floor cleaning in extended public or industrial areas raise very interesting research challenges when required to be autonomously covered by mobile robots.

**COMPLETE COVERAGE PATH PLANNING:**

Complete coverage path planning is the determination of a path that a robot must take to pass over each point in an environment except reasonably and efficiently. There are two approaches for this problem. Firstly, we have a scenario where the bot is given the map priorly for path planning. The other scenario is where the bot is put in an unknown environment and is expected to traverse it completely using sensors to avoid obstacles. In this project we'll be trying to implement an algorithm for the second scenario.

**SIMULATION & VISUVALIZATION SOFTWARES**

**GAZEBO**

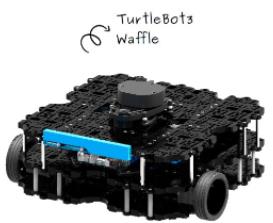
Robot simulation is an essential tool in every roboticist's toolbox. A well-intended simulator will make it possible to rapidly test algorithms, design robots, perform regression testing, and train AI system utilizing realistic scenarios. Gazebo offers the ability to replicate populations of robots accurately and effectively in complex indoor and outdoor environments.

**RVIZ**

[Rviz](http://www.ros.org/wiki/rviz) is a potent robot visualization tool. It provides a handy GUI to envision sensor data, robot models, environment maps, which is handy for developing and debugging your machine controllers.

**DIFFERENTIAL DRIVE ROBOT**

The differential drive robot which we have used in our implementation is Waffle.

****

**Fig1: Waffle Bot**

**SENSORS USED**

**LDS-01:**

360 Laser Distance Sensor LDS-01 is a 2D laser scanner capable of sensing 360 degrees that collects a set of data around the robot to use for SLAM (Simultaneous Localization and Mapping) and Navigation.



**Fig2: LDS-01 Sensor**

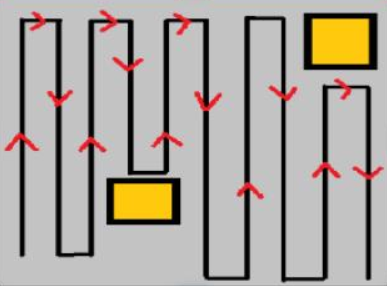
Waffle bot uses LDS-01 sensor for sensing environment. With its unique ability to provide reliable detection and ranging over short to long distances, LiDAR technology is used in multiple industries and applications. Autonomous driving requires LiDAR to complement and complete the existing automotive sensor suite (i.e., Camera vision, Radar, and Ultrasonic technologies), creating a new, fast-growing opportunity for the proliferation of LiDAR with strong demand and large volumes. LiDAR data can be easily converted into 3D Maps to interpret the environment. It generates instantaneous measurements and can be accurate. It is unaffected by ambient light sources.

**IMPLEMENTATION**

**U-TURN ALGORITHM**

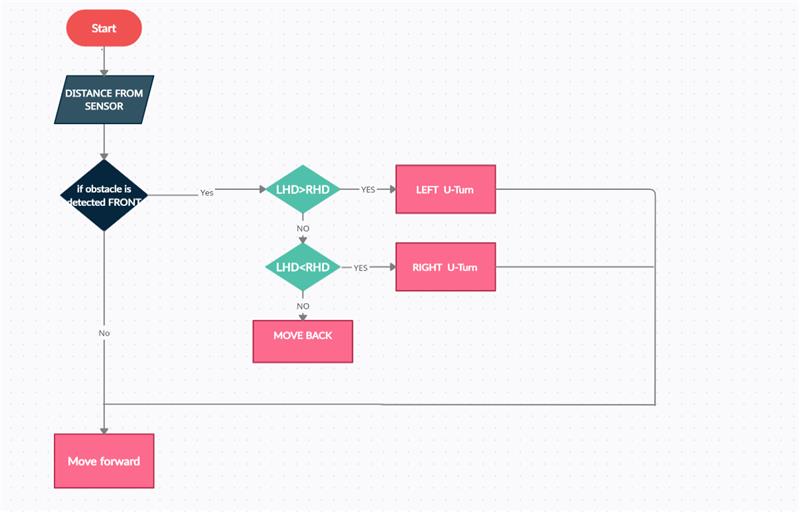
We have tried to come up with a simple algorithm using Lidar Sensor keeping in mind cost effectivity and efficiency.

* The bot is instructed to move forward until it encounters an obstacle in its path.
* Whenever it encounters an obstacle in front, the bot takes a Left U-turn (rotates 180 counter clockwise).
* Whereas when it encounters an obstacle in front as well as left or right, it is instructed to take a Right U-turn (rotates 180 clockwise) or Left U-turn (rotates 180 counter clockwise respectively.



**Fig3: Proposed U-turn Algorithm**

**FLOWCHART**



**Fig4: Working of U-Turn Algorithm**

**CREATING WORLD IN GAZEBO**

This is a simple house environment that we have created in Gazebo for the project Implementation. The House consists of a living room and a small room along with some obstacles such as a Cafe table, blocks, and a dustbin.

Chart, box and whisker chart

Description automatically generated

**Fig5: Gazebo environment**

**CODE**

Graphical user interface, text, application, email

Description automatically generated

**Fig6: Code**

**RESULTS**

This is the output that was attained in Rviz during implementation.

Chart

Description automatically generated

**Fig7: Rviz environment**

**CONCLUSION**

The implementation of the vacuum bot involved the study and analyzation of various navigation algorithms. Even though this implementation was based on a basic U-turn algorithm, it did require a lot of fine tuning. For an unknown environment, the bot worked well using just the LiDAR sensor. It works for both static and dynamic obstacles. The aim was to keep the logic as simple as possible and to increase the robustness of the Vacuum cleaner, which has been partially achieved. As a future prospect of the project, we have planned to work on decreasing the time taken to cover the complete environment. This can be done by storing all the previously traversed positions so that the bot does not waste time in cleaning the same place repeatedly.

**REFERENCES**

**1.Manreet Kaur, Preeti Abrol “Design and Development of Floor Cleaner Robot (Automatic and Manual) “International Journal of Computer Applications (0975 – 8887) Volume 97– No.19, July 2014.**

**2.Xueshan Gao, Kejie Li, Yan Wang, Guangliang Men, Dawei zhou and Koki Kikuchi. A floor cleaner robot using Swedish wheels.In IEEE international conference on robotics and biomimetics December 15-18, 2007, Sanya, China.**

**3.J Frolizzi C.Disalvo. Service robots in the domestic environment: A study of Roomba**