Smart Route Planning and Object Avoidance of Autonomous Robot

Dr. Sandip Ghosh
Associate Professor
Department of Electrical Engineering,
Indian Institute of Technology (BHU),
Varanasi

Dhruv Aseri
Student
Department of Electrical Engineering,
Indian Institute of Technology (BHU),
Varanasi

Sanskar singh
Student
Department of Electrical Engineering,
Indian Institute of Technology (BHU),
Varanasi

***Abstract*—This report delves into the comprehensive design, implementation, and evaluation of an advanced obstacle detection and avoidance system integrated into the Pololu 3pi+ robot, featuring a smart route navigation system. Through the utilization of ultrasonic sensors, the project aims to bolster the robot's autonomy, enabling it to intelligently navigate its environment while evading potential collisions with obstacles. By harnessing the Pololu 3pi+ 32U4 robot alongside ultrasonic sensors, the system facilitates real-time distance measurements, empowering autonomous decision-making based on nearby object distances within a specific angle range. The inclusion of a smart route navigation system further enhances the robot's capabilities, allowing it to dynamically plan optimal routes and adapt its trajectory based on sensor data and internal mapping. With implications spanning industrial automation, surveillance ect.**

I. Introduction

Pololu stands as a premier provider of robotics and electronics components, serving hobbyists, educators, and professionals alike with their diverse range of offerings. Their catalog spans from individual components like microcontrollers, sensors, and motor drivers to comprehensive platforms such as the 3pi+ 32U4 robot. This compact marvel is equipped with dual motors, line sensors for precise path following, bump sensors to detect obstacles, and a programmable core powered by the ATmega32U4 microcontroller..

What sets Pololu apart is their commitment to user-friendliness, exemplified by their intuitive libraries seamlessly integrated into the Arduino IDE. These libraries simplify the programming process, empowering enthusiasts of all skill levels to breathe life into their robotic creations effortlessly., fostering creativity and innovation in the realm of robotics.

The Pololu 3pi+ 32U4 can be transformed into an Smart Route navigator using ultrasonic sensors and some clever coding.. Pololu libraries and Arduino code work together to interpret this data. If the distance falls below a set threshold, indicating an object is near, the code triggers evasive maneuvers through motor control functions, enabling the robot to avoid collisions. For our smarter Route navigation, we have implemented a pre- programmed map or path-finding algorithm, allowing the robot to make decisions in real time by looking left and right of equal angle and move according to the best route based on sensor data and its internal map .This fusion of object detection, avoidance tactics, and potential route planning empowers the 3pi+ to Plan the smart route in Real-time.

II. Sailent features of the 3pi+ robot

A. *Ports/Features*

The 3pi+ 32U4 stands as a versatile and high-performance robot, boasting user-programmable capabilities within its compact 9.7 cm (3.8) diameter frame. Fueling its operations is the ATmega32U4 AVR microcontroller from Microchip, providing robust processing power. Similar to our A-Star 32U4 programmable controllers, the 3pi+ 32U4 ships with a USB interface and comes preloaded with an Arduino-compatible bootloader, simplifying programming with just a USB A to Micro-B cable. Additionally, a software add-on streamlines programming within the Arduino environment, complemented by our extensive collection of Arduino libraries and example sketches

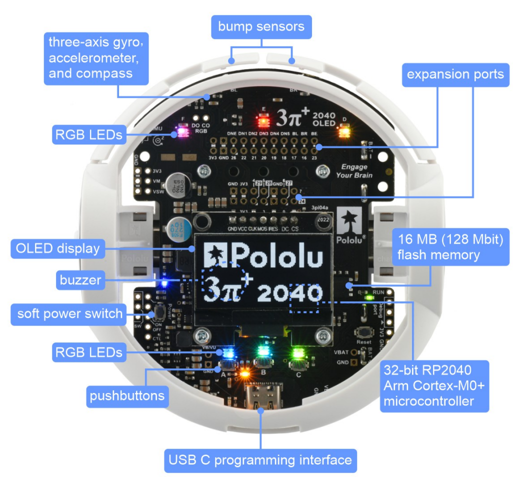

Fig. 1. Top view of 3pi+

B. *Core of 3pi+ Robot*

At its core, the 3pi+ integrates two H-bridge motor drivers and a suite of advanced sensors, including quadrature encoders for precise motor control and a full inertial measurement unit for comprehensive motion tracking. Its array of five downward-facing reflectance sensors enables accurate line following and edge detection, while front-facing bump sensors enhance obstacle avoidance capabilities. User interaction is facilitated through three onboard pushbuttons and a 128x64 graphical OLED display, alongside a buzzer and indicator LEDs for providing feedback.

Fig. 2. Core/Board of 3pi+

C. *Power supply*

Notably, the 3pi+ boasts a unique power system ensuring consistent motor performance at 8 V, independent of battery charge levels. This feature guarantees stable motor speeds throughout operation, enhancing the robot's reliability and performance across diverse environments.

III. Indroduction of Ultra sonic sensor

Ultrasonic sensors, renowned for their versatility and precision, play a pivotal role in an array of applications spanning industries. Operating on the principle of emitting and detecting sound waves beyond human hearing, typically above 20 kHz, these sensors offer unparalleled measurement and detection capabilities. At their core lies a transducer capable of both emitting and receiving ultrasonic waves, facilitating accurate distance calculations through the time-of-flight method. This feature makes ultrasonic sensors indispensable for tasks requiring precise proximity detection, such as obstacle avoidance in autonomous vehicles or level sensing in industrial tanks.

Furthermore, the robustness of ultrasonic sensors shines through in their remarkable reliability across diverse environmental conditions. Unlike optical sensors susceptible to ambient light interference, ultrasonic sensors remain unfazed by challenging settings, including low-light conditions or environments laden with dust or fog. This resilience not only enhances their suitability for outdoor applications but also positions them as indispensable tools in industrial automation, where consistent performance is paramount for operational efficiency.

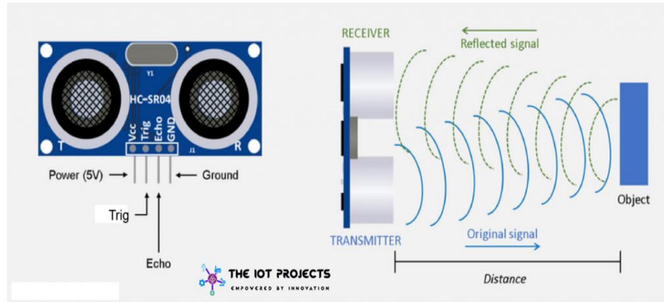

Fig. 3. Ultrasonic Sensor

A. *Distance Measurement Algoritm*

Sensor uses this algortim for calculating distance.

- Clear trigPin to ensure a clean signal and Send a short 10-microsecond pulse on trigPin to trigger the ultrasonic sensor.
- Measure the duration of the pulse on echoPin using the pulseIn() function.
- Calculate the round-trip time of the ultrasonic pulse: duration = pulseIn(echoPin, HIGH);.
- Convert the duration to distance using the speed of sound (approximately 343 meters/second):
- distance = duration * 0.034 / 2.
- Display or use the distance measurement (e.g., print to the serial monitor

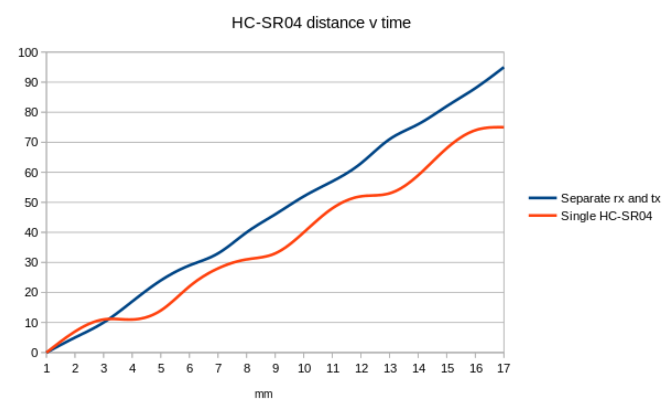

Fig. 4. Distance vs time

B. *Connection pin diagram*

Here is the connection/pin diagram of the 3pi+robot and ultrasonic sensor we have to connect according to the ports mentioned in the code to which we are referring.

1 .Pin Diagram of Ultrasonic Sensor: The ultrasonic sensor typically has four pins:

- VCC: Power supply pin (usually connected to +5V)
- GND: Ground pin (connected to GND)
- Trig: Trigger pin (used to send ultrasonic waves)
- Echo: Echo pin (used to receive the reflected ultrasonic waves)

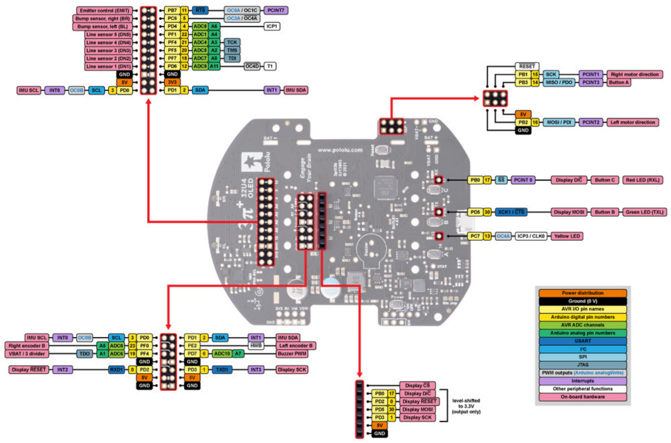

Fig. 5. Pin diagram of the robot

2. Connections: Based on the pin diagram and the Arduino pin assignment, make the following connections

- VCC pin of the ultrasonic sensor to the +5V pin on the Arduino for power supply.
- GND pin of the ultrasonic sensor to any GND pin on the Arduino for ground connection.
- Trig pin of the ultrasonic sensor to Pin 9 on the Arduino.
- Echo pin of the ultrasonic sensor to Pin 10 on the Arduino.

IV. Connection of 3pi+ to Ultrasonic Sensor

Connect the trigger input (Trig) pin to a digital output, the echo (Echo) pin to a digital input on your robot's microcontroller, and the VCC and GND pins to a 5V power source. Wait for a high level on the echo (Echo) pin after at least 10us (microseconds) of high-pulse activity on the trigger (Trig) pin. The distance that the ultrasonic sound travels is represented by the length of time the Echo pin remains high. The robot is getting closer to an obstruction the faster it responds.

- TRIGPIN = 3;
- ECHOPIN = 2;
- VccPIN=5v;

V. Algorithm for Smart Routing

Implementing code in Arudino using C++, using various libraries to access certain features of the robot.

A. Algorithm

The smart navigation algorithm for the Pololu 3pi+ robot typically involves the following steps:

- **Obstacle Detection:** The robot uses sensors as ultrasonic sensors, to detect obstacles in its surroundings by regularly calculating real time distance.
- **Path Planning:**During path planning, the Pololu 3pi+ robot evaluates its environment to identify clear paths

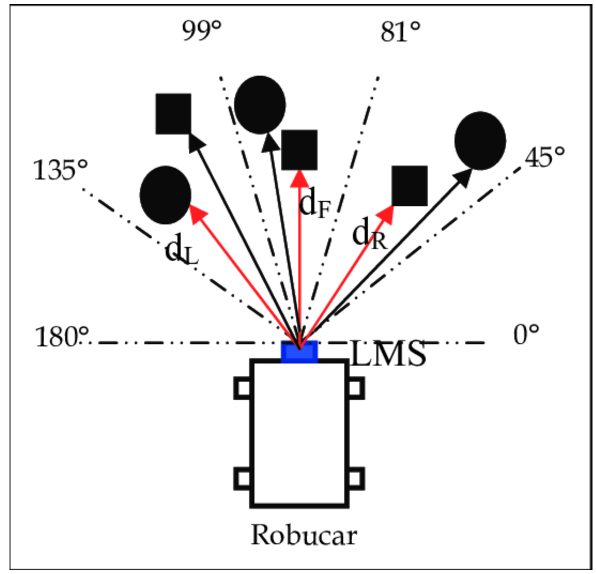

Fig. 6. Sensing data at different angle

while accounting for obstacles. By analyzing obstacle layout, it selects the safest route towards its destination, prioritizing paths with fewer obstructions. This entails assessing obstacle distances and orientations to determine the most viable trajectory.

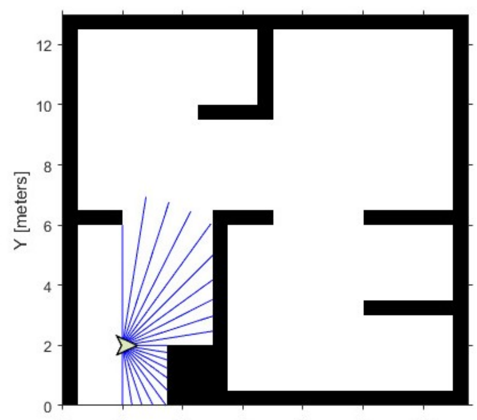

Fig. 7. Planning the path in real time

- **Decision Making:** In order to navigate efficiently during decision-making phases, the robot makes use of the data obtained from path planning. The robot makes decisions about its movements in real time, such as turning left or right to avoid obstacles and stay on course, based on the positions of obstacles and the selected path. As the robot moves forward, these choices are dynamically modified to guarantee ongoing adaptation to the shifting surroundings and safe passage to the intended destination.
- **Continuous Monitoring:** The robot continuously monitors its surroundings to adapt its route if new obstacles are detected or if the environment changes.

B. Initialization(SETUP()FUNCTION)

ECHO-PIN and TRIG-PIN are defined for the ultrasonic sensor (connected to specific digital pins). motors object is created to control the robot's motors. Serial Communication is established with the baud rate set to 9600 for debugging and data visualization.
pinMode(ECHO-PIN, INPUT) sets the echo pin as an input, waiting for signals from the ultrasonic sensor. pinMode(TRIG-PIN, OUTPUT) sets the trigger pin as an output, sending signals to the ultrasonic sensor. bumpSensors.calibrate() calibrates the bump sensors for accurate readings.

C. Distance Calculation

The project incorporates a precise method for calculating distances to detected obstacles. This calculation is essential for informing the robot's navigation decisions and ensuring safe manoeuvrability in its environment. It uses this formula to calculate the distance to the obstacle.

distance = duration * 0.0343 / 2

The duration represents the time measured in microseconds using the loop function of the Arduino platform. The duration is obtained by initiating a pulse of ultrasonic waves and measuring the time taken for the waves to return after bouncing off an obstacle. where duration is the time measured in microseconds using loop Function

This crucial information enables the robot to make informed decisions regarding obstacle avoidance and path planning, contributing to its overall autonomy and efficiency in navigation.

- digitalWrite(TRIG-PIN, HIGH);//sets high for 10microseconds
- delayMicroseconds(10);//delay high for 10microseconds
- digitalWrite(TRIG-PIN, LOW);//sets low for 10microseconds

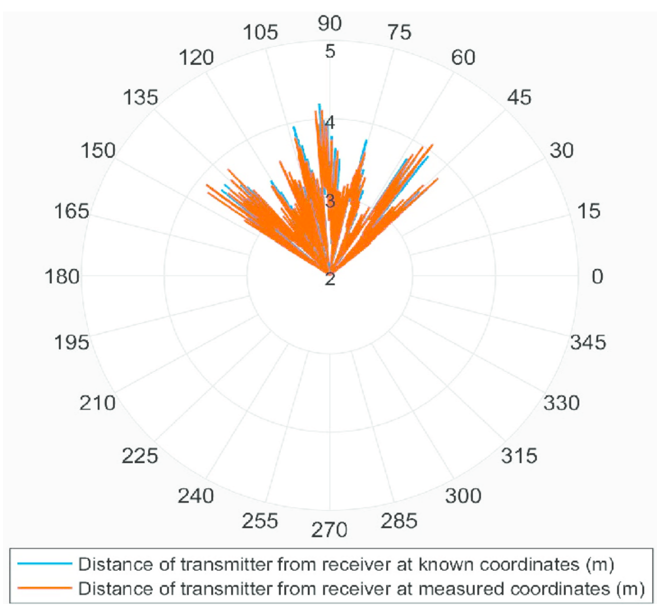

Fig. 8. Storing data at different angles

D. Motor Speed Adjustment

The current implementation includes a lower limit check to ensure that motor speeds do not fall below a predefined threshold, MOTOR-MIN-SPEED. Additionally, optional motor compensation factors, L-MOTOR-FACTOR and R-MOTOR-FACTOR, can be applied to adjust for potential differences in left and right motor speeds.
Algorithm: Motor Speed Adjustment and Obstacle Avoidance

- if (leftSpeed ¡= L-MOTOR-FACTOR-THRESHOLD) leftSpeed *= L-MOTOR-FACTOR;

- if (rightSpeed ¡= R-MOTOR-FACTOR-THRESHOLD) rightSpeed *= R-MOTOR-FACTOR;

E. Stop distance Check

If the detected distance (distance) is less than or equal to the STOP-DISTANCE, the robot smartly navigates and finds the best possible angle for which the object distance is maximum. Implementing the logic for left or right turns based on distance readings.

- if (distance is less than STOP-DISTANCE) millisElapsed = 0;
  leftSpeed = 0;
  rightSpeed = 0;
  predict(leftSpeed, rightSpeed);
  motors.setSpeeds(leftSpeed +1),rightSpeed);

F. Smart Navigation

After assessing its surroundings and detecting an obstacle within the stop distance, the robot initiates a process of fine-tuning its orientation for optimal obstacle avoidance. This involves rotating both clockwise and anticlockwise by a predefined angle, typically 60 degrees, to survey the environment and identify the direction with the greatest distance from the obstacle.

To accomplish this, the robot utilizes data collected from the angle , prioritizing angles corresponding to the largest distances from the obstacle. By systematically rotating and measuring distances, the robot identifies the angle associated with the maximum distance, indicating the safest direction to manoeuvre. Once the optimal angle is determined, the robot corrects its path and adjusts its rotor speeds accordingly to facilitate smooth and collision-free motion in that specific direction. This adjustment ensures that the robot transitions seamlessly from a stationary state to a moving state while avoiding obstacles and maintaining rotor speeds below a predefined threshold for stability.

Fine-tuning the robot's rotation involves precisely adjusting both motor speeds and the duration for which each motor rotates in a particular direction. This meticulous calibration ensures accurate angle rotation, allowing the robot to navigate with precision and efficiency in complex environments.

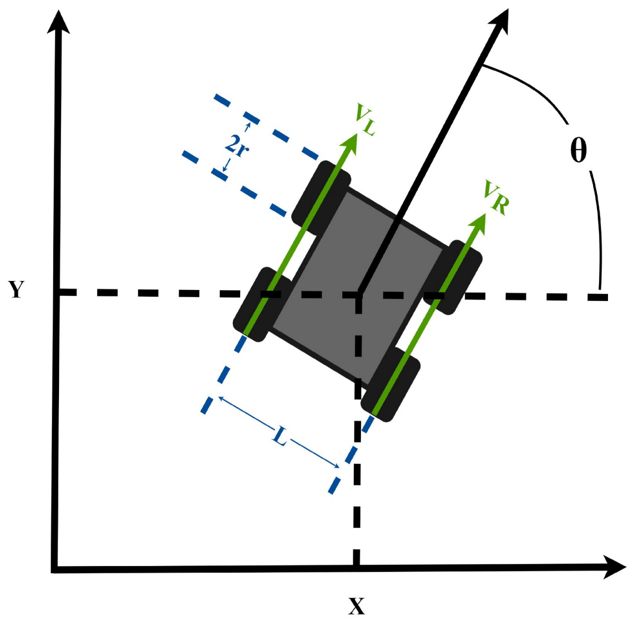

Fig. 9. Enter Caption

VI. Implementation challenges

Numerous challenges arose that demanded careful attention and innovative solutions. One prominent obstacle was the issue of sensor inaccuracies and the diverse reflective properties of surfaces encountered in real-world settings. These inaccuracies posed a significant challenge to the reliability of distance measurements obtained from the ultrasonic sensor.

Addressing this challenge necessitates the development of sophisticated calibration techniques aimed at mitigating sensor inaccuracies. Through a meticulous and iterative process, the sensor's parameters were fine-tuned, and software-based compensation mechanisms were devised and implemented. These measures aimed to enhance the sensor's accuracy and consistency, thereby improving the reliability of the distance measurements crucial for effective obstacle detection and navigation.

Furthermore, ensuring precise control over the robot's motors presented another formidable challenge in the implementation process. Achieving seamless coordination between the robot's movements and the feedback received from the ultrasonic sensor demanded a high degree of motor control precision. However, variations in motor performance, mechanical limitations, and unforeseen external influences introduced complexities that needed to be carefully managed.

To address these challenges, rigorous motor calibration procedures were developed, incorporating real-time data analysis techniques to identify and rectify any deviations from desired motor behavior. This involved fine-tuning motor parameters and implementing adaptive control strategies to maintain optimal performance under varying conditions.

In summary, the successful implementation of the ultrasonic sensor for obstacle detection in the autonomous robot project required overcoming significant challenges related to sensor inaccuracies and motor control precision. By employing meticulous calibration techniques and innovative control strategies, these challenges were effectively addressed, paving the way for improved autonomy and navigation capabilities in the robot's operation.

VII. Conclusion

In conclusion, the integration of ultrasonic sensors for obstacle detection in autonomous robotics, as demonstrated by the Pololu 3pi+ 32U4 robot, marks a significant milestone in the field. Through meticulous calibration and innovative algorithms, this project has propelled the robot's capabilities to new heights, enabling it to navigate complex environments with heightened autonomy and efficiency. Challenges such as sensor inaccuracies and motor control precision were addressed with careful consideration and robust solutions, ensuring the reliability and effectiveness of the obstacle detection system.

Moreover, the Pololu 3pi+ 32U4 robot, enhanced with ultrasonic sensors and intelligent coding, showcases remarkable adaptability and versatility in real-world scenarios. Leveraging Pololu's user-friendly libraries and Arduino compatibility, enthusiasts of all skill levels can easily harness the power of robotics to innovate and explore new possibilities. The implementation of a smart routing algorithm further elevates the robot's capabilities, facilitating real-time decision-making based on sensor data and internal mapping. This fusion of object detection, avoidance tactics, and route planning empowers the Pololu 3pi+ 32U4 to navigate dynamically changing environments with precision and confidence.

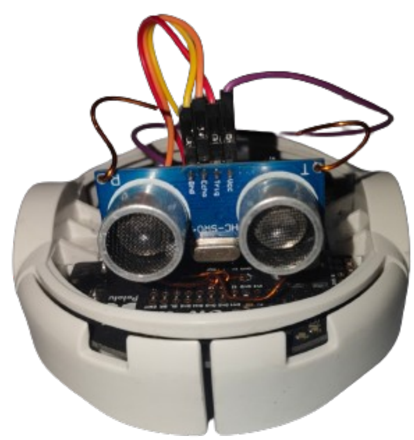

Fig. 10. Final robot look

Overall, these projects represent transformative ventures within the realm of robotics. They highlight the practicality and effectiveness of ultrasonic sensors in enhancing autonomy and navigation capabilities. The success of these endeavors underscores the significance of noncontact sensors like ultrasonic sensors in robotics, showcasing their resilience in various environmental conditions and their versatility in addressing complex challenges. As such, they pave the way for future advancements in the field, promising exciting opportunities for industrial automation, surveillance, and educational robotics alike.

References

[1]. T. Yamada, Y.L. Sa, and A. Ohya." Moving obstacle avoidance for mobile robot moving on designated path." in Ubiquitous Robots and Ambient Intelligence (URAI),2013. 10th international conference on pp. 509- 514. IEEE 2013..

[2].G. Mester." Obstacle voidance of mobile robot in unknown environment." in Intelligent System and Informatics (SISY) 2007.5th International Symposium on pp.123-127. IEEE.

[3]J. Duan, J. Yao, D. Liu, and G. Liu. "A path tracking control algorithm with speed adjustment for intelligent vehicle." In Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on, pp. 2397-2402. IEEE, 2013.

[4].S. K. Malu, and J. Majumdar. "Kinematics, localization and control of differential drive mobile robot." Global Journal of Research in Engineering (2014)

[5].G. Gallego and A. Gil "A vision -based multi -camera system for 2d localization of vehicles "

[6].T. Y. Ali, and M.M. Ali." Robotino Obstacles Avoidance Capability Using Infrared Sensors." in Applied Electrical Engineering and Computing Technologies (AEECT), IEEE Jordan Conference, pp. 1-6. 2015