PATH PLANNING FOR OPTIMUM NAVIGATION OF AN OMNI –WHEELED MOBILE ROBOT

D. Mahipala[1], G. Maduranga[1], P. de Silva[1], K. Dassanayake[1], S. Wijekoon[1], J. V. Wijayakulasooriya[2]

[1] Department of Mechatronics, Faculty of Engineering, South Asian Institute of Technology and Medicine (SAITM), Sri Lanka.
[2] Department of Electrical and Electronics Engineering, Faculty of Engineering, University of Peradeniya, Sri Lanka Email: jan@ee.pdn.ac.lk

ABSTRACT

Omni – directional wheeled robots have the ability to move in any direction without turning the chassis, as opposed to a differential drive system. This paper presents a mobile robot design using omni – directional wheels. A 4 – wheeled drive system is used in this design as it easier to control its motion compared with a 3WD omni-directional robot. The robot identifies obstacles by the inputs from sensors onboard the robot and plans its motion to correct its path after avoiding obstacles while maintain its current orientation. The test arena used to assess the operation of the robot is also presented.

***Key Words*:** Omni – Directional, Wireless, Self – navigation

1. INTRODUCTION

Omni – directional robots are increasingly being used for optimized navigation. Omni-directional whells are getting popular because of the wheel's ability to move like a normal wheel as well as to support motion perpendicularly to the orientation of the wheel. This is achieved by using rollers/smaller wheels around the wheel which are oriented perpendicular to the orientation of each wheel. When the motor applies a torque on the wheel, because of traction, the wheel will move normally and when the robot is moving in a direction perpendicular to the orientation of the wheel, the rollers help the movement by sliding with minimum friction. This ability gives more versatility to the robot giving it freedom to move in any direction without having to rotate the body. For a short history of Omni directional wheels, refer [1].

Figure 1: The Omni – directional wheel used in this project

Figure 1 shows the wheels used in our design, which is also a very common type of wheel used in RoboCup, [2, 3]. For detailed workings of a 4 wheeled omni – directional drive system, refer [4], and for details on a 3 wheeled drive system, refer [5].

Omni – directional wheels have been used in many path finding robots because of the maneuverability they present. Obstacle navigation can be achieved in a variety of methods, namely by pre - determining the location of obstacles by visual aids or other means beforehand or by detecting obstacles by using local sensors onboard the robot. A comparison of path planning algorithms can be found in [6].

2. BACKGROUND

Omni directional navigation can be accomplished by two different drive mechanisms, namely differential drive system and Omni wheeled drive system.

An Omni wheel based design was implemented in this model due to the ability of an Omni wheel to move in mutually perpendicular directions without turning the robot to the desired direction.

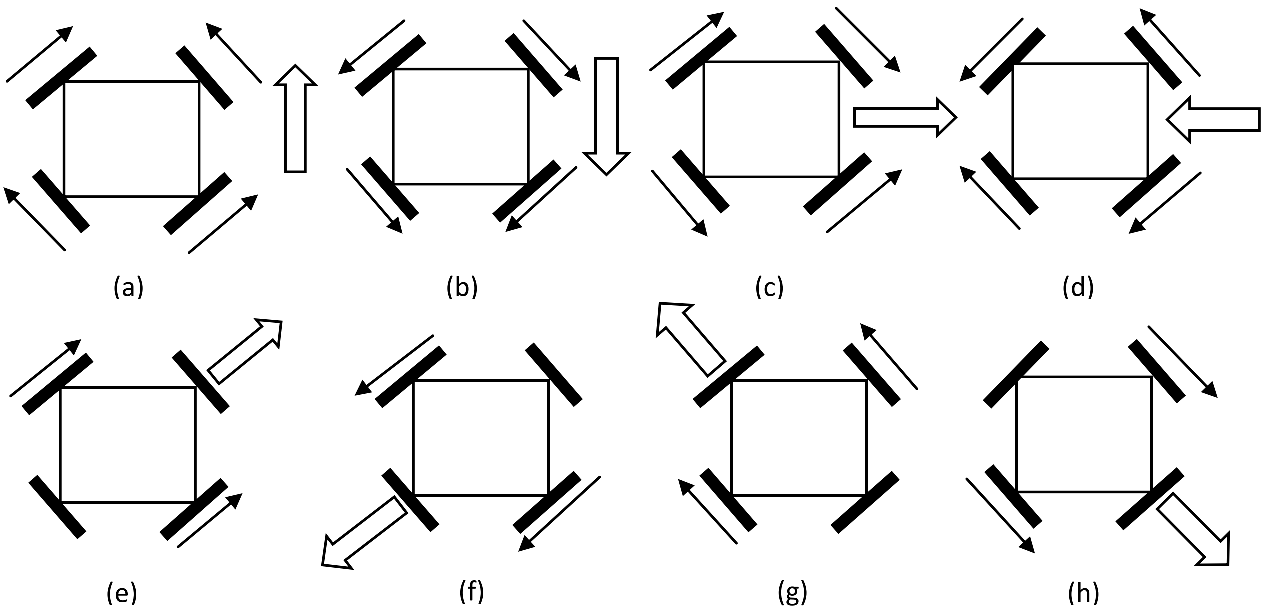


Figure 2: Movement of the wheels to travel in the 8 cardinal directions

A four wheel based design was implemented as opposed to a three wheeled arrangement.

The wheel is designed in such a way that it enables the robot to travel like a normal wheel as well as perpendicular to the orientation of the wheel and parallel to the axis of the motor. This movements are achieved by using rollers/small wheels along the circumference along the wheel. When a torque is applied by the motor the wheel will move as if it's a normal wheel due to traction caused by the wheel whilst it will move perpendicularly because of the rollers sliding without friction.

Two or more omni directional wheels are used to drive the robot. Each wheel provides traction parallel to the motor axis. The forces add up and provides the robot a translational motion and rotational motion. When the wheels are rotated in different directions in different velocities, various kinds of movements can be achieved. For the robot to move in the forward direction, the wheel velocities should be as shown in Figure 02 (a). The velocity directions for the backward motion is as shown in Figure 02 (b). Velocity directions for the sideways movements are shown by Figure 02 (c), (d). Movements in the diagonal directions can be achieved by rotating the wheels in the velocity directions as shown in Figure 02 (e), (f), (g), and (h). To determine the position of the robot and to control the speeds of the wheels, four encoders attached to the Omni wheels have been used since it is easier and relatively less expensive compared to other position determining methods.

3. DYNAMICS

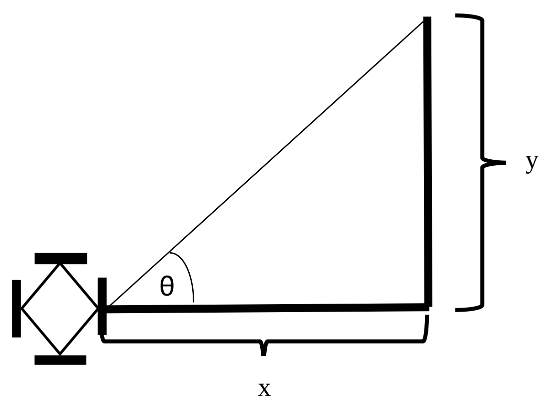


Figure 03: Angle determination for the robot

$$\theta = \tan^{-1}({}^{y}/_{x}) \quad -(1)$$

$$S = \sqrt{x^2 + y^2} \quad -(2)$$

When the destination coordinates are given to the robot via the user interface, the microcontroller will calculate the angle with respect to the X axis. Figure 03 shows the x and y co-ordinates for the point to which the robot would move. From (1) the angle for the robot to move in can be determined. From (2) the distance S the robot should move in the specified direction can be obtained. Encoders are used to determine the distance travelled by the robot and the speeds of the motors. The calculation to find the distance the wheel moves in one second is as follows,

$$s = \left(\frac{2*\pi}{N}\right) * R - (3)$$

Where; N = Number of slots. (Pulses in the output waveform)

R = Radius of the wheel.

s = Distance travelled by the wheel in N number of pulses.

4. DESIGN AND IMPLEMENTATION

4.1. System Block Diagram

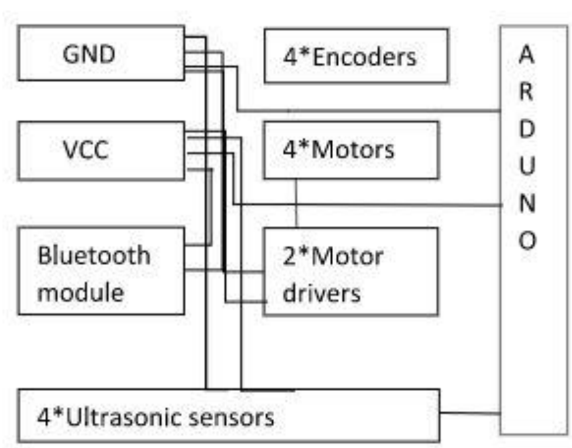


Figure 4: System Block Diagram

The Figure 04 shows the system block diagram of the design used in this paper. An Arduino mega 2560 [8] is used as the main control unit as it gives a wide array of functionality.

4.2 Decision making process

When the robot is moving in an area with obstacles, decisions will be taken as shown in Figure 05 below. If an obstacle is detected the robot will calculate the distance left to the end goal and then will move in a circular path with the distance left as the radius of the particular circular path. When the obstacle is no longer blocking the path, the robot will continue its path towards the end position.

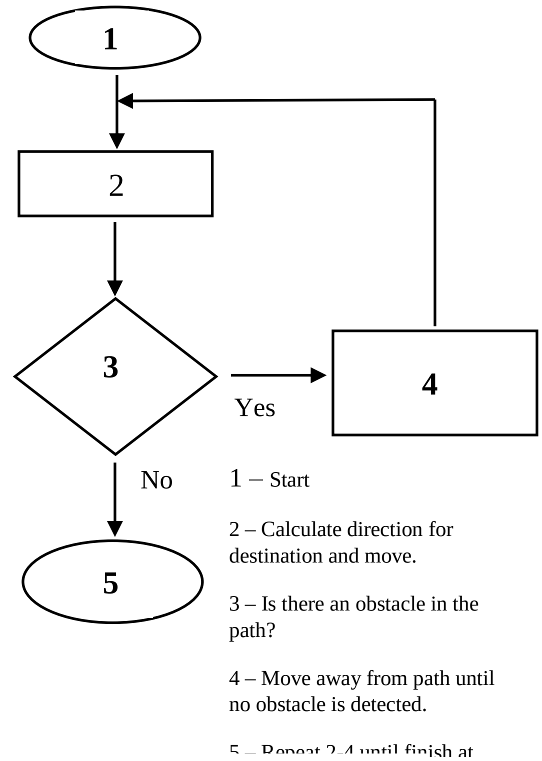


Figure 05: Decision process

4.3. PI Algorithm

When a position is given, the destination angle is calculated and the robot will automatically turn in the direction of the position so that two mutually opposing wheels will be facing the direction it is supposed to move. Then the robot will only use two wheels to move forward. When moving forward to maintain the correct path, a PI algorithm is implemented as follows. To calculate the error, the microcontroller is calculating the number of slots passed within a second from both the wheels generating a square waveform with each slot generating a pulse in the waveform.

$$Error = N_L - N_R$$

Where;

N_L = No. of pulses generated from the left wheel

N_R = No. of pulses generated from the left wheel.

Proportional = Error * K_P

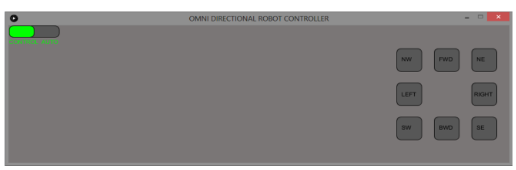
Integral = Error – lastError * K_I

Total_error = Proportional + Integral

4.4. User interface

As the first part we have used Processing 3.0.2 JAVA based platform to develop the user interface of the robot. The purpose of having a wireless communication with the robot is to have the ability to update the position it needs to go in real time. The interface used to submit the positional co-ordinate is as shown in Figure 06 (a). If a situation arises where the robot is unable to continue without manual assistance, the manual control mode is then activated enabling the robot to successfully pass the obstacles in question. The wireless communication in this model is the HC – 06 Bluetooth module [7].

(a)



(b)

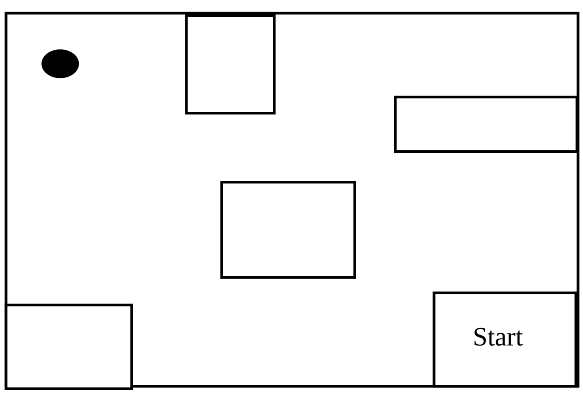


Figure 6: Graphical User Interface with (a) auto and (b) manual control modes

Figure 7: Test Arena

4.5. Test

To test the capabilities of the omni – directional wheeled drive system and the navigational capabilities a test arena is required. The arena should possess the ability to fully test the abilities of the wheels to move in any direction and also its implementation in navigation through an area with multiple number of obstacles. The proposed design for the track is as shown in Figure 07.

5. CONCLUSION

Wireless communication was successfully established using the Bluetooth module, thereby enabling manual control and also the ability to update the position in real time. Obstacle navigation and optimal path planning will be further improved in the future, by considering arbitrary shaped objects and dynamic objects.

6. REFERENCES

[1] R. Rojas: "*A short history of omnidirectional wheels*".

[2] A. Egorova: "*FU Fighters Small Size Team 2004",* RoboCup-2004: Robot Soccer World Cup VIII, Springer-Verlag, 2005.

[3] F. V. Hundelshausen: "*FU-Fighters Team Description 2003",* RoboCup-2003 - Proc. of the Int. Symposium, 2003.

[4] R. Rojas: *"Holonomic Control of a robot with an omnidirectional drive.",* KI - K¨unstliche Intelligenz, B¨ottcherIT Verlag, 2006.

[5] F. Ribeiro: *"Three Omni – directional wheels control on a mobile robot*".

[6] F. Haro: "*A Comparison of Path Planning Algorithms for Omni-Directional Robots in Dynamic Environments"*, LARS, 2006.

[7] Detailed document available at https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf

[8] Detailed document available at http://www.mantech.co.za/datasheets/products/A000047.pdf