IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Factura '20 - National Conference On Emerging Trends In Manufacturing

NSS College of Engineering, Palakkad, Kerala, India

Vol. 7, Special Issue 1, August 2020

Design and Path Planning for A Self-Moving Trolley in Dynamic Environment

Jaseem Sajidh N A[1], Dr. Aneesh Kumar P,[2]

PG Scholar, Mechanical Engineering, NSSCE, Palakkad, Kerala[1]

Assistant Professor, Mechanical Engineering, NSSCE, Palakkad, Kerala[2]

Abstract: Trolley is very familiar utility device used in different areas like industries, airports, shopping malls and household. In airports and shopping malls, we carry our luggage in the trolleys and leave it in different areas of parking or surroundings. The collection of these trolleys requires significant human effort. The project is to design the working model of smart trolley for the airports and shopping malls, which considered a very useful especially for workers. The smart trolley had self-control movements. Self-movement is to improve the process of collecting the trolleys after use them by passengers, so the trolley returns to nearest assembly points around the airport. A simple and effective path planning algorithm is also prepared for the action of the trolley in dynamic environments. The self-moving smart trolley moves to the assembly point through the dynamic environment by the help of special line pattern. So, the path planning includes the algorithms for special line following and obstacle avoidance.

Keywords: Robotics, Smart trolley, Path planning algorithm, dynamic environment, self-moving trolley.

I. INTRODUCTION

Robotics is a field which is much developed and still developing further for helping the humans in satisfying the specific needs of daily life. Robotics is an emerging field in which advancements can be visible in each day. The collaboration of Artificial Intelligence (AI) has changed the face of robotics in productive manner, hence robots are smart and becoming smarter by implementation of these cutting-edge technologies. This technological adaptation involves skilled programming levels, high sensing capabilities and higher computational power. These costly processes make the robots exorbitant and unfamiliar to common people, public places and private households. This high cost issue has inspired to adopt new adaptations and alternative solutions. Robots are of various types, out of which Mobile robots are the ones that mark the field with their wide range of applications. Wheeled Mobile Robots (WMR) are much common as they are less sophisticated in terms of other robot motions. Initially, the application of a mobile robot was limited to manufacturing industries only. But nowadays, it is commonly used in the fields of entertainment, medicine, mining, rescuing, education, military, space, agriculture and many more. Robots are improving in tackling issues which are experienced by people. The use of mobile robots is increasing in modern ubiquity as well as in public places where human involvement and effort is significant. Thus, the difficulties for mobile robots are rising. Some present-day applications for mobile robots are AGVs, programmed motion, cleaning, demining, garden cutting, floor cleaning and programmed reaping of crops. These days the security and energy consumption a significant job for mobile robots. In addition to that, robot's obstacle avoidance is another major criterion that needs to be discussed. The movement of robot in dynamic environment involves more sensors and advanced level of programming. Here the obstacle avoidance is much complicated process. In dynamic environment we cannot predict the presence of obstacle early, and the obstacle itself may be moving in some directions. For specific application like trolley system we have to consider the direction of the motion also. Smart Trolley system is an advanced application of wheeled mobile robots. Automated movement of trolleys in dynamic environment like airport or shopping malls is really challenging since it requires complicated obstacle avoidance system and a directed motion. While performing the task of navigation, the robot is equipped with many intelligent equipments which are required to model the environment and localize its position, control the motion, detect obstacles, and avoid obstacles by using navigational techniques. Safe path planning (by detecting and avoiding the obstacles) from the initial position to the target position is the most important function of any navigational technique. Therefore, the proper selection of the navigational technique is the most important step in the path planning of a robot when working in a simple and complex environment. The unpredictable situations of these areas really affect the development of a common motion algorithm for the situation. Different algorithms like self-organizing and migrating algorithm, multipartite RRTs for rapid replanning in dynamic environments, genetic algorithm, fuzzy logic, neural network, firefly algorithm, particle swarm optimization, ant colony optimization, bacterial foraging optimization, artificial bee colony, cuckoo search are available dynamic conditions. Since the objective of our project is different, we cannot adopt any of existing algorithms completely. So, we have to develop an algorithm based on these available algorithms for the path planning of the smart trolley. In order to achieve the unidirectional motion which is directed to some common assembly point, usually uses the line follower mechanism. Here we modify the line pattern and sensor system in such way that to get the compulsory single directed motion. The determination of position and motion commands to get the correct direction is also a tedious process. This is also developed for the smart trolley system. This work focused on the development and testing of a self-moving trolley in dynamic environment like airports and shopping malls.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Factura '20 - National Conference On Emerging Trends In Manufacturing
NSS College of Engineering, Palakkad, Kerala, India
Vol. 7, Special Issue 1, August 2020

II. DEVELOPMENT OF PATH PLANNING ALGORITHM

Since the existing path planning algorithms are not satisfactory for a dynamic environment like airport or shopping malls, we have to create a new one for our particular application. For the easiness of understanding we can make the total algorithm into two. That is, Special pattern following algorithm and algorithm for dynamic environment. The trolley has basically two modes of operation, manual and automatic. The manual mode conditions are, the manual switch is on or the trolley has a luggage. So before entering the automatic mode we have to check the logic conditions. The conditions are, the trolley weighs nil load and be in stationary position for time limit. The algorithm for entering the automatic mode is shown in Fig 1.

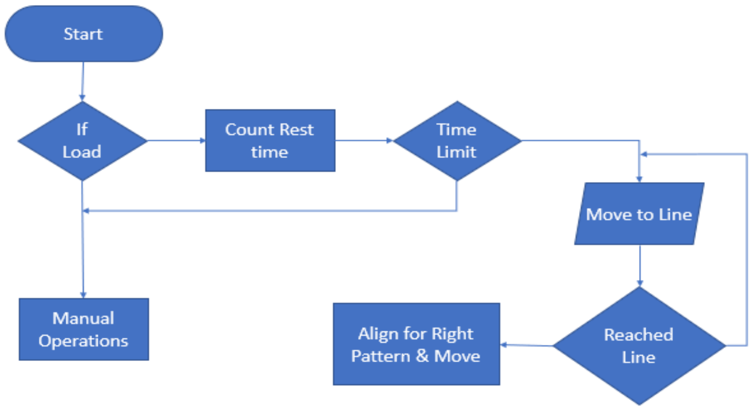


Fig 1 Algorithm for entering the automatic mode

A. Line Follower Algorithm

The movement of the trolley to the assembly point is mainly done by special line pattern follower algorithm. Apart from the normal line follower, this is very different because the unidirectional motion of trolley towards the assembly point is possible. The algorithm is in such a way that, the trolley gets aligned to require direction if it enter to the line from any direction or any orientation.

Fig 2 Special Line Pattern

The Trolley points are interconnected through double line through the boundaries. Trolley moves in a particular direction and touches this line. Once if the trolley touches the line, it ignores the wireless positioning and move through the line. The Line is drawn as double line with different thickness to detect which direction it moves. Figure 2 shows the Line pattern. Keeping thick line in the left side and thin line in the right side, the trolley can easily decide to which direction it has to move. There are five set of sensors to detect the Line. IR sensors are, C0- Centre, R1- Right inside, R2- Right Outside, L1- Left inside, L2- Left outside. The arrangement of sensors and line is shown in Fig 3

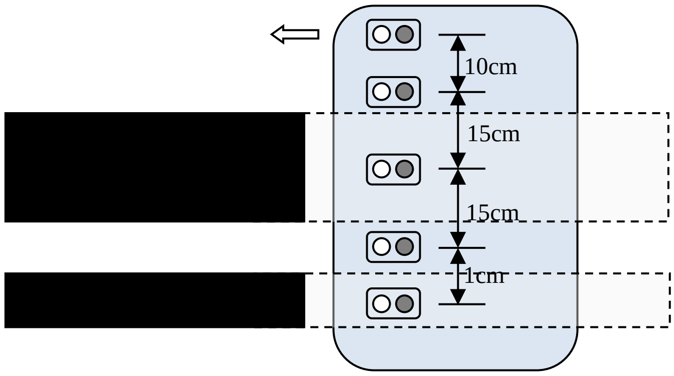


Fig 3 Pattern and IR sensor arrangement in correct orientation

If Centre sensor (C0) and Left most sensor (L2) detects line and others are not online the Trolley moves forward. The table 1 shows the Action for different sensing patterns.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Factura '20 - National Conference On Emerging Trends In Manufacturing
NSS College of Engineering, Palakkad, Kerala, India
Vol. 7, Special Issue 1, August 2020

Table 1 Truth table for special pattern follower

L2	L1	C0	R1	R2	Action	Comment
0	0	0	0	0	None	It's out of line, action depends on wireless position
0	0	0	0	1	Turn Right	To entre completely to track
0	0	0	1	0	Rotate Left	Trolley to be turn and enter to line
0	0	0	1	1	Move Right	To entre completely to track
0	0	1	0	0	Stop	At end of Line
0	0	1	0	1	Move Back	Trolley in reverse direction
0	0	1	1	0	Turn right	Trolley on track but little bit in left
0	0	1	1	1	Move forward	Trolley on track but inclined
0	1	0	0	0	Move Left	Trolley in correct direction but out of line
0	1	0	0	1	--	No possible pattern
0	1	0	1	0	--	No Possible pattern
0	1	1	0	0	Stop	May be end
0	1	1	0	1	**Move Forward**	In correct position
0	1	1	1	0	Move Forward	Trolley on track but inclined
0	1	1	1	1	Move Forward	Trolley just to entre track but inclined
1	0	0	0	0	Move Left	Trolley to enter track
1	0	0	0	1	--	Not Possible pattern
1	0	0	1	0	Continue	Trolley inclined
1	0	0	1	1	Continue	Trolley inclined
1	0	1	0	0	Move Left	Trolley just entre track
1	0	1	0	1	--	Not Possible pattern
1	0	1	1	0	Turn Right	Trolley on track may be turning
1	0	1	1	1	--	Trolley inclined
1	1	0	0	0	Move Forward	Trolley just to entre track
1	1	0	0	1	--	Not Possible pattern
1	1	0	1	0	--	Not Possible pattern
1	1	0	1	1	--	Not Possible pattern
1	1	1	0	0	Move left	Trolley to entre completely to track
1	1	1	0	1	Move Left	On track but turning
1	1	1	1	0	Rotate	Inclined on track
1	1	1	1	1	Rotate	Trolley entered vertically on track

B. Algorithm for Dynamic Environment

In dynamic environment we cannot predict the presence and position of the obstacle. Here we are discussing the movement of the trolley in free space as well as in pattern. We follow same obstacle avoidance algorithm in both cases as they meet an obstacle. We are attached 6 obstacle sensors around the trolley. So, the trolley operates according to the sensor input.

If the trolley moves forward front left sensor senses an obstacle and right side is free. Then the trolley moves right. Moves right is entirely different from turn right. Moves right command includes, unlocking of solenoid switch at 00, turning of rotating plate to 900, locking at 900 and moving forward. This is similarly for Move right command also. So, when the front left senses obstacle and front right is free, then trolley moves right until both the front sensors are free from obstacle. Whenever both front sensors are free, then trolley moves forward (this includes, unlocking at 900, rotating plate rotates to 00, locking at 00, then move forward). While this forward motion, left sensors are active and whenever both the left sensors are open, the trolley returns to the pattern or initial location. Trolley always shows the tendency to return back to the direction opposite to the deviated direction. That is if the trolley is deviated to right the trolley shows the tendency to move left direction, to the initial position.

Logic that follows while obstacle avoidance are;
Set 1

- Front left senses and Front right is free then trolley moves right
- Front right senses and Front left is free then trolley moves left
- Both the front left and front right senses then trolley moves left (by default)

While moving right;

If right sensors senses move back (initial position and move left)
If both front sensors are free moves forward {Set 2}
If front sensors sense then follow set 1
If left sensors are free return to initial (move left)
If right sensors sense then follow set 2

While moving left;

If left sensors senses move back (initial position and move right)
If both front sensors are free moves forward {Set 2}
If front sensors sense then follow set 1
If right sensors are free return to initial (move right)
If right sensors sense then follow set 2

The trolley may get trapped if the obstacle is at three sides, that is at front, left and right. In such cases the trolley tries two times and then stops. Usually such obstacle cases are very rare in airports and shopping malls. And this problem can be solved by providing the reverse drive and backside obstacle sensor. In a dynamic environment, the position, shape and size of the obstacle is not predictable. So, we cannot define all the possibilities graphically. But for an effective path planning algorithm, we have to consider all the possibilities. For this we define some basic possibilities of obstacle presence, which is enough to program for the entire environment.

III. DESIGN OF THE SMART TROLLEY

The design of the smart trolley is initialized by fixing the drive for the robot. We have to make the total arrangement to fulfil the requirements of fully functional smart trolley. For this model we selected differential drive for the robot. Then, the kinematics of the mobile trolley was done and the design with exact dimensions and exact components needs to be finalized. To finalize the robot dimensions, the functions and requirements for a mobile cleaning robot needs to be discussed.

5.1 Basic Functions and Requirements

Since the idea of smart trolley for airports and shopping mall is a new one, we have to make a clear picture on the working of the trolley in such dynamic environments. We have to list out the requirements in details and the functions of the components or the system against the requirements. The requirements are,

- The trolley should have two mode of operation. That is manual and automatic. And the shifting of operation mode should be easy and fulfil the requirements.
- The trolley should start moving automatically to the assembly point only if there is no luggage on the trolley and it is stationary for a particular interval of time.
- Path to the assembly point is defined by special line pattern. Trolley should effectively follow the pattern.
- Trolley should always search for the line pattern if it is in free space.
- The trolley should have the tendency to move back to the pattern if it is deviated due to obstacle.
- Locking and unlocking of trolley body and rotating plate should takes place on required time.
- The movement through line pattern should be unidirectional.

5.2 Fixing the trolley dimensions

Smart trolley is a technological upgradation on the existing utility trolleys. While creating a working model of smart trolley it should not affect the existing utilities as well as it can fulfil the requirements listed above. So, the dimension of each part of the trolley is important. Here we fabricate a working model of smart trolley of standard size, that is 55 cm * 73 cm platform size. Here we increase the thickness of the platform by adding another plate of same size. All the electronic equipment, sensors, battery etc will situate in between the plates. In addition to this, we connect a rotating circular plate with two differential driving wheels. The total load of the trolley and luggage is carried by the driving wheels and four caster wheels. So, all the wheels should align in same level. Length of baseplate is 63 cm and width 45cm. Total thickness of the base plate of the trolley is 10.4 cm. Each baseplates and rotating wheel have a standard thickness of 12mm. Total clearance from the ground to the bottom of the lower baseplate is 10.2cm. Clearance in between rotating plate and lower baseplate is 4.5 cm. Total Hight of the caster wheel is equal to the clearance, that is 10.2 cm.

5.3 Design of the trolley motion

The trolley has to move in the dynamic environments of airport parking or shopping mall surroundings. So, the trolley has to move through the line pattern and also through the free space. So, the trolley must be able to provide sufficient traction for its horizontal motion. While taking a U turn the wheels must be controlled differentially. Also, the special rotation-based movement while interacting with obstacles.

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Factura '20 - National Conference On Emerging Trends In Manufacturing

NSS College of Engineering, Palakkad, Kerala, India

Vol. 7, Special Issue 1, August 2020

5.3.1 Determination of RPM and Speed of the Motion

While moving in a dynamic environment, where humans, vehicles and other obstacles are very high, trolley cannot possess higher velocity. In order to control the trolley more efficiently, the velocity should be lower. So, the trolley moves at a rate of 80cm per second

V_{linear} = 800 mm/s

The diameters of the commercially available wheels are 40mm, 70mm, 100mm etc. Considering the required clearance for the trolley, the wheel diameter is selected to be 100mm.

R_w = 50 mm

Now, finding out the angular velocity required to run the robot at specified speed. Wheel Angular Velocity,

$\omega = V_{linear}/R_w = 800/50 = 16$ rad/s

Now, the rpm of the motor is,

$RPM_{motor} = 60/2\pi \times \omega = 60/2\pi \times 16 = 152.86$ rpm

The required speed of the motor is 153 rpm. The commercially available precision motors have speed limit of 150 rpm. Therefore, the linear speed of the robot is now within the range. The new linear speed from rpm value is,

$V_{linear} = R_\omega \times 2\pi/60 \times RPM = 50 \times 2\pi/60 \times 150 = 785$ mm/s

5.3.2 Wheel Torque Calculation

The drive wheel motor for wheeled mobile robots determines the speed at which the robot is to be moved. To choose motors capable of producing enough torque to propel the robot, it is necessary to determine the total tractive effort (TTE) requirement for the robot.

Total tractive effort (TTE) is

$$TTE\ [\text{kg}] = RR\ [\text{kg}] + GR\ [\text{kg}] + Fa\ [\text{kg}]$$

Where,

$TTE = total\ tractive\ effort$ [kg]

$RR = force\ necessary\ to\ overcome\ rolling\ resistance$ [kg]

$GR = force\ required\ to\ climb\ a\ grade$ [kg]

$Fa = force\ required\ to\ accelerate\ to\ final\ velocity$ [kg]

Each component of the TTE equation can be determined from following steps.

(i) Rolling Resistance (RR)

Rolling Resistance (RR) is the force necessary to propel a vehicle over a particular surface. The possible surface type to be encountered by the robot should be factored into the equation. In our case it is friction of rubber with concrete or floor material

$WR = Robot\ weight$ [kg] = 11 kg

$\mu = Sliding\ friction\ coefficient\ of\ Rubber\ and\ Concrete = 0.6 \rightarrow 0.85$.

The equation for calculating RR is,

$$RR\ [\text{kg}] = WR\ [\text{kg}] \times \mu\ [-]$$

RR [kg] = 11 kg × 0.725 ($average$) = 7.975 kg

(ii) Grade Resistance (GR)

Grade Resistance (GR) is the amount of force necessary to move a vehicle up a slope or grade. This calculation must be made using the maximum angle or grade the vehicle will be expected to climb in normal operation.

$WR = gross\ vehicle\ weight$ [kg] = 11 kg

$\alpha = incline\ angle\ [degrees] = 3°$ (average)

To convert incline angle, α, to grade resistance:

$GR\ [\text{kg}] = WR\ [\text{kg}] \times si(\alpha)$

$GR\ [\text{kg}] = 11 \times sin\ (3°) = 0.575$ kg

(iii) Acceleration Force (Fa)

Acceleration Force (Fa) is the force necessary to accelerate from a stop to maximum speed in a desired time. The parameters required for calculations of Fa are

$WR = Robot\ weight$ [kg] = 11 kg

$g = accleration\ due\ to\ gravity = 9.81\ m/s^2$

$Vmax = maximum\ spee[m/s] = 0.785$ m/s

ta = $time\ required\ to\ achieve\ maximum\ speed\ [s] = 1\ s$

The equation for calculating Fa is,

F_a [kg]= (W_(r) [kg] × V_(max) [m/s])/(g[m/s^2] × t_a)

F_a [kg]= (11 [kg] × .785 [m/s])/(9.81[m/s^2] × 1) = 0.88 kg\

(iv) Total Tractive Effort (TTE)

The Total Tractive Effort (TTE) is the sum of the forces calculated in steps 1, 2, and 3. (On higher speed vehicles friction in drive components may warrant the addition of 10%-15% to the total tractive effort to ensure acceptable vehicle performance.)

TTE [kg]= RR [kg] + GR [kg] + Fa [kg]

TTE [kg]= 7.975 kg + 0.575 kg + 0.88 kg = 20.77 kg ≅9.43 kg

(v) Wheel Motor Torque (τw)

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

IARJSET

International Advanced Research Journal in Science, Engineering and Technology

Factura '20 - National Conference On Emerging Trends In Manufacturing

NSS College of Engineering, Palakkad, Kerala, India

Vol. 7, Special Issue 1, August 2020

To verify the vehicle will perform as designed in regards to tractive effort and acceleration, it is necessary to calculate the required wheel torque (τw) based on the tractive effort. The equation to find the same is given by

$TTE = total\ tractive\ effort$ [kg] = 9.43 kg

$Rw = radius\ of\ the\ wheel/tire$ [cm] = 10 cin

$RF = resistance\ factor$ [−] = 10→15%

The resistance factor accounts for the frictional losses between the caster wheels and their axles and the drag on the motor bearings. Typical values range between 1.1 and 1.15 (or 10 to 15%). Therefore, the torque required for wheels is

τw [kg − cm] = TTE [kg] × Rw [cm] × RF [−]

τw [kg − cm] = 9.43 [kg] × 10 [cm] × 1.15 [−] = 108.445 kg cm

(vi) Reality Check

The final step is to verify the vehicle can transmit the required torque from the drive wheel(s) to the ground. The Maximum Tractive Torque (MTT) a wheel can transmit is equal to the normal load times the friction coefficient between the wheel and the ground times the radius of the drive wheel.

Ww = $weight\ (normal\ load)\ on\ drive\ wheel$ [kg]

μs = $Sliding\ friction\ coefficient\ of\ Rubber\ and\ Concrete$ = 0.6→0.85

$Rw = Radius\ of\ drive\ wheel/tire$ [cm]

Therefore, the Maximum Tractive Torque is equal to

MTT = Ww [kg] × μs [−] × Rw [cm]

MTT = 11 kg × 0.725 × 10 cm = 79.2 kg cm

As there are two wheels which are driven by motors, the MTT is twice of the solution above, which is 158.4 kg cm. The wheel motor torque required for the effective motion and navigation of the robot is about 108.445 kg cm. Hence, the robot will be able to transmit the torque effectively to the ground and impart the required motion to the robot, provided that the motor torque shall never be less than 108.445 kg cm or 1.10 kg.m

5.4 Design of Rotating Plate Mechanism

Rotating plate has a key role in the obstacle avoidance. Whenever the moving trolley interacts with an obstacle, that is the front obstacle sensors senses the presence, the trolley needs to deviate from its current path. In our algorithm usual left and right turn procedure following in line pattern, is not enough. Here we need to move the trolley sideways without affecting its body orientation. Rotating plate plays this role in the system. The requirements of rotating plate are;

- Deviate the driving wheel orientation to 900, -900, 1800 as per the requirements.
- Easy rotation with respect to baseplate.
- Provide the grooves for locking in exact locations.
- Attach the drive motors and wheel system

For attaining these requirements, we designed the rotating plate with satisfying dimensions and properties. The rotating plate is designed in SolidWorks. Rotating plate is a circular plate with diameter of 300mm and thickness of 12mm and made up of acrylic. It has the space for attaching the motor and driving wheel on a side and, other side is connected to the baseplates through the sector plate bearing. The grooves for the locking action of the solenoid lock is also present at 0^0, 90^0, -90^0. By the sector plate bearing the rotating plate can have a circular motion without affecting the trolley orientation. The rotation of the plate is possible by the differential motor rotation. The plate rotates 90^0 clockwise by rotating the right motor backward and left motor forward. And the plate rotates 90^0 counter clockwise (-90^0) by rotating the left motor backward and right motor forward. Angle of rotation is cross verified by the angle sensor attached to the shaft on the middle of plate

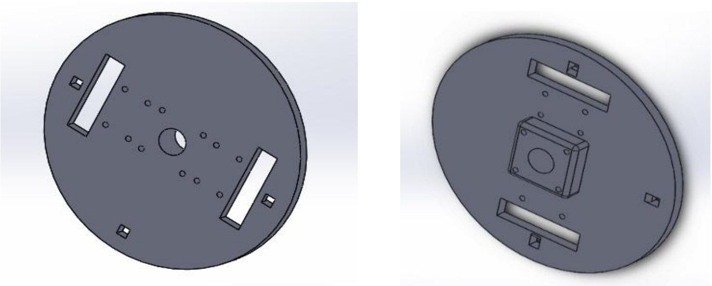


Fig 6 Bottom and upper part of the Rotating Plate

5.5 Design of the Sector Plate Bearing

Sector plate bearings are not common in market because of its limited application. It is usually used to transfer the axial load in between two plates. And it makes possible the independent rotation of plate with respect each other, about the bearing axis. The requirements of the Sector plate bearing are,

- Carry the load from the trolley.
- Easy rotation of plate over counter plate.
- Reduced total thickness in order to fix in the available clearance space.

The maximum load of the trolley is calculated by adding the weight of total trolley system and the weight of the luggage over the trolley. Let's limit the maximum luggage weight as 50kg, and the weight of the trolley 11kg. The total load is calculated as

$W_{total} = W_l + W_t$

Where,

W_l : Load due to luggage (kg.wt) = 50 kg.wt

W_t :Load due to trolley weight (kg.wt) = 11 kg.wt

$W_{total} = W_l + W_t$ = 50+11 = 61 kg.wt = 598 N

So, we need the sector plate bearing with the capacity to carry at least 600 N, and the maximum thickness of 30mm. The available sector plate bearing has the following specifications. And this is suitable for our requirements.

Inside diameter: 2.23 inch or 56.64mm

Outside diameter: 4 inch or 101.6mm

Width: 5/16 inch or 7.93mm

Suggested turntable size: up to 25 inches

Maximum balanced load: 299lbs or 136kg

Thickness: 35mm

5.8 Design of Software Architecture

The working model of trolley requires programming to work autonomously. For this purpose, it has a controller on which specific programs are written. The controller takes input from various sensors and give program specific outputs to the actuators, that is to driving motors and solenoid lock. Here we deal with different sensor inputs and accordingly operated actuator. We have to do the line follower mechanism from the inputs of IR sensor, Obstacle avoidance according to the inputs from ultrasonic sensors, Locking and unlocking according to the rotation angle sensor, Initial movement direction according to GPS module. For doing all these programs simultaneously usually use Raspberry Pi Model 3 B+ Processor. But the processing speed of the Raspberry Pi board was not up to the mark, it took more time than doing it in a PC with processing speed of a 2.3GHz Processor. Hence, the path plan is not done using Raspberry Pi, rather it is done using a PC. The matrix data output is fed as the input to the robot controller Arduino board. Arduino Mega 2560 board based on ATmega2560 controller is used for programming the trolley system. It is having open source IDE (Integrated Development Environment) and programming language software is based on high-level language C++. The various sensory functions and other actuations for the software architecture is shown in the Fig. 7

Smart trolley system has five types of sensors or inputs and two type actuators. Input or sensors are; IR sensor for line pattern detection, Ultrasonic sensors for obstacle detection, Rotation angle sensor, GPS Module, Load sensor, Manual switch. The controller has to integrate the inputs from all the sensors and command the actuators accordingly. The actuators are Drive motors and Solenoid lock

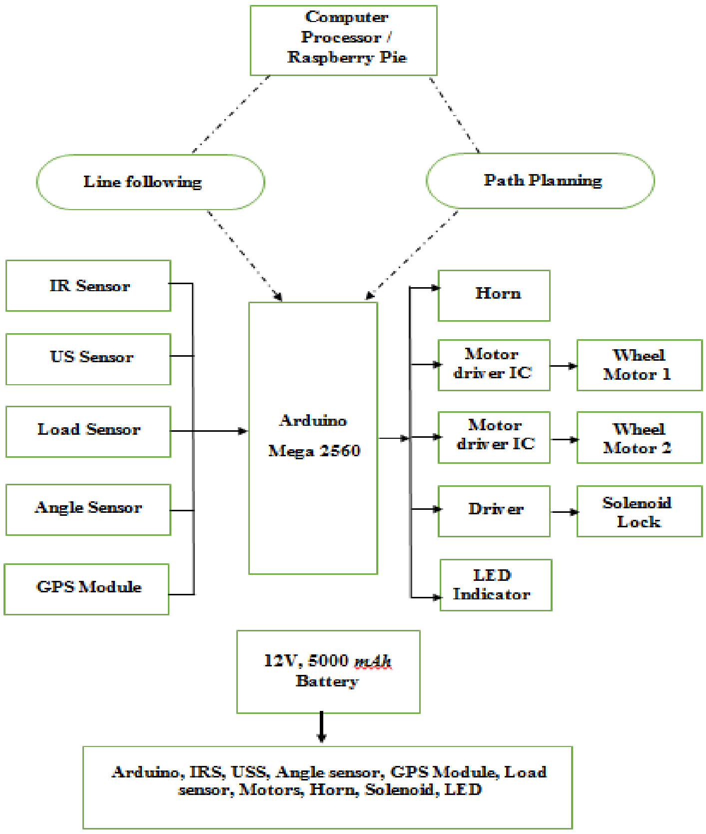


Fig 7 Smart trolley software architecture

IV. FABRICATION AND ASSEMBLY

After the design process, the trolley is assembled with its various components. The smart trolley has various components that need to properly install at its particular positions in the body. The key components are the DC geared motors, the wheels, the controller, line sensors, GPS module, rotation sensor, locking switch, smart load sensor, obstacle sensors, turning mechanism and other electronic components which include the electronic horn, rotation sensor, 5A motor driver modules etc.

6.1 Component Details

6.1.1 Micro Controller Board

For controlling the robot, Arduino MEGA control board is used. Arduino MEGA is a commonly used prototyping board. Fig. 6.1 shows the Arduino MEGA board. Arduino MEGA is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM (Pulse Width Modulation) outputs), 16 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP (In Circuit Serial Programmer) header and a reset button. Arduino can be powered either with USB connection or an external 5V power supply. In this prototype we power the Arduino by connecting it with a standard DC Power jack. The 5V power is obtained by a DC – DC step down circuit.

6.1.2 Drive Motor

The motor used for drive in this prototype is DC geared motor with encoder. DC geared motors are used for obtaining the proper torque at a required rpm. The DC geared motor is a normal permanent magnet DC motor with gearbox attached to the output side. The gearbox contains reducer gears for increasing the torque and reducing the rpm and thereby increasing the torque. Other specifications of the particular model have been mentioned previously in this work.

6.1.3 Wheels

The diameter of the wheel is 100 mm and has a width of 20 mm. This is a commonly commercially available wheel in the market. The wheel diameter is obtained from calculation which involves rpm of the motor, velocity required for the robot etc.

6.1.4 Ultrasonic Range Sensor

The robot uses ultrasonic range sensor HC-SR04 for detecting obstacles and bend. The HC-SR04 ultrasonic sensor uses ultrasound to determine distance to an object. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package. It can detect or measure the position of an object from 2cm to 400 cm. Its operation is not affected by sunlight or black material. It comes complete with ultrasonic transmitter and receiver module. It has four pins. VCC and GND are power supply pins and are connected to the 5V output of the stepdown converter. The trigger and echo are connected to the digital pins of arduino. The echo pin is to be set as input and trigger pin as output. It works by sending an ultrasonic 'chirp' through the transmitter and calculating the time required for the 'chirp' to reflect back from the object. The distance is calculated then by using multiplying speed and time where speed is the speed of sound.

6.1.5 Infrared Sensor

An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measure only infrared radiation, rather than emitting it that is called as a passive IR sensor. Usually in the infrared spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes, that can be detected by an infrared sensor. The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, the resistances and these output voltages, change in proportion to the magnitude of the IR light received.

6.1.6 Battery

A 12 Volt 3S 30C 2200 mAh battery is used in this prototype, this can run the prototype for about 30 minutes. The 3S denotes 3 cells are connected in series for this particular battery. Similarly, 30C denotes the discharge capacity. It is the constant current that can be drawn from the battery without harming it. For a 2200 mA 30C battery we can draw a maximum current of 66 $Amps$. This time is only required, as this is only a prototype model. For commercial applications a larger battery capacity with higher mAh value could be used.

6.1.7 Load Sensor

A load cell (or loadcell) is a transducer which converts force into a measurable electrical output. Although there are many varieties of force sensors, strain gauge load cells are the most commonly used type. A load cell works by converting mechanical force into digital values that the user can read and record. The inner working of a load cell differs based on the load cell that we choose. Here we use a 30 kg load cell attached with the upper baseplate of trolley.

6.1.8 Solenoid Lock

The most basic forms of electronic locks are solenoids. When a user "unlocks" the lock, a current traveling through a coil of wire creates a magnetic field that attracts a metal plunger or locking pin. The locking pin moves into an unlocked position state against a spring. Here we use the solenoid lock for arresting the independent movement of rotating drive plate with respect to base plate of the trolley.

6.1.9 Shaft rotation sensor

This rotation sensor or incremental encoder senses the angle of rotation of rotating plate with respect to the trolley base plate. It can also detect clockwise and counter clockwise rotation and that's why it has 2 pins (SIA and SIB) for detecting rotation. Basically, it translates the rotary motion into a sequence of pulse signals. By comparing the square wave generated from each of

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Factura '20 - National Conference On Emerging Trends In Manufacturing
NSS College of Engineering, Palakkad, Kerala, India
Vol. 7, Special Issue 1, August 2020

the two pins you can determine the direction of rotation and by counting the number of pulses you can determine both the rotation angle and the RPM of the rotating shaft.

6.1.10 GPS Module

A Complete GPS module based on the Ublox NEO-6M is used to get GPS location. This unit uses the latest technology from Ublox to give the best possible positioning information and includes a larger built-in 25 x 25mm active GPS antenna with a UART TTL socket. A battery is also included so that you can obtain a GPS lock faster. This is an updated GPS module that can be used with ardupilot mega v2. This GPS module gives the best possible position information, allowing for better performance with your Ardupilot or other Multirotor control platform.

6.1.11 Sector Plate Bearing

Here we use a sector plate bearing, type of thrust bearing, to connect the rotating plate and trolley's base plate. It allows the independent rotation of the plates with respect to each other and effectively carry the load. The dimensions of sector plate bearing are; Inside diameter: 1.1/4 inch or 31.75mm, Outside diameter: 3 inch or 76.2mm, Width: 5/16 inch or 7.93mm, Suggested turntable size: up to 18 inches, Maximum balanced load: 200lbs or 90kg

6.2 Electronic Circuit of Trolley

The circuit of the trolley include 13 sensors, GPS module, driver ICs, motors, solenoid lock and controller. The electronic circuit is prepared and tested using the software Proteus. The circuit diagram is shown at Fig 8. For the testing of circuit using proteus, we cannot attach the sensors as it is. So, for the easiness here the sensors are indicated as switches. Sensing of each sensor is assumed as the on position of the switch.

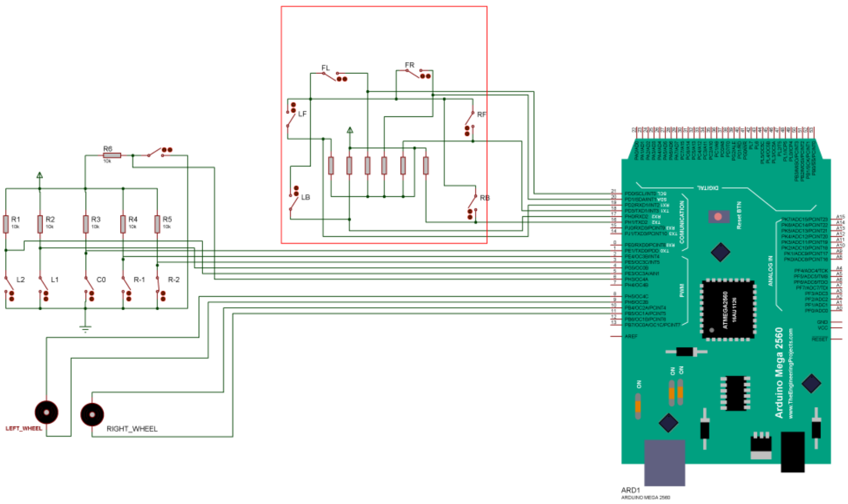


Fig 8 Circuit diagram of smart trolley plotted in proteus

V. WORKING AND TESTING

After the assembly of the whole system we have to check whether it is properly working or not. In this chapter we detailly explain the working of the smart trolley system. And also explains different testing strategies used here. The total operations are subdivided into two, that is manual and automatic. The automatic mode of operation is explained as four stages, they are initiation of automatic movement, movement in free space, movement on pattern and obstacle avoidance.

7.1 Manual and Automatic mode of operation

Usually the passengers or customers use the trolley manually to carry their luggage. For carrying the luggage to desired destination is completely done in manual mode. But the advantage is, since we have the differentially driving motors the person needs not to push the trolley. It moves by the motors and the person need to control it as per the required destination. For the controlling of the motion two switches are provided on the hand wheel. These two switches separately control the two motors. Simultaneous on position of the switches rotated both the wheels forward, and the trolley moves forward. If the person needs to take a right turn, he has to off the right switch and left switch keeps in on position. Similarly, for taking the left turn, keeps the left switch off and right switch on. For pulling back the trolley, that is to move in reverse direction on both the switches in reverse direction. Then both the motors rotate backward and we get a reverse motion. By controlling the switch, we can have the left and right turn in reverse direction also. The conditions of switches and corresponding movement are shown in table 2. S1 controls M1 (Right motor) and S2 controls M2 (Left motor). During the whole manual operation period the solenoid lock will be in locked position. That is the independent action of rotating plate is completely arrested. So, while taking the turning or rotations, the whole trolley rotates.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Factura '20 - National Conference On Emerging Trends In Manufacturing
NSS College of Engineering, Palakkad, Kerala, India
Vol. 7, Special Issue 1, August 2020

Table 2 Switch positions and movement

S1	S2	M1	M2	Motion Status
OFF	OFF	OFF	OFF	Stationary
ON (+)	ON (+)	Forward	Forward	Moves forward
ON (+)	OFF	Forward	OFF	Forward right turning
OFF	ON (+)	OFF	Forward	Forward left turning
ON (-)	ON (-)	Reverse	Reverse	Reverse movement
ON (-)	OFF	Reverse	OFF	Reverse left turning
OFF	ON (-)	OFF	Reverse	Reverse right turning
ON (+)	ON (-)	Forward	Reverse	Counter clockwise rotation
ON (-)	ON (+)	Reverse	Forward	Clockwise rotation

When the luggage of the trolley is cleared and manual controlling switches are not in use, then the resting time period of trolley is tested by the controller. Whenever it exceeds the time delay, here we set it as 120 seconds, the trolley operation will be shifted to automatic mode. The time delay can be varied according to the need. So, the necessary conditions for shifting to automatic mode are;

- Trolley is not in manual control, that is both S1 and S2 are in OFF position.
- Trolley is luggage free, that is the load sensor senses the load over the top baseplate is nil.
- Trolley is stationary for a certain period of time, about 2 minutes. The mode shifted occurs only at all the conditions gets true.

7.2 Initiation of automatic movement

When the conditions for automatic mode are satisfied, then the trolley starts automatic self-movement. Before starting the movement, the GPS module inside the trolley panel identifies the current position of the trolley and the position of the final point, that is assembly point. So, the trolley makes the first movement in such a way that, the distance between assembly point and the current trolley must be reduced. If the trolley is on the line pattern, then the GPS module need not to act. Because the line pattern is specialised for single directed motion, that only towards the assembly point. For the easy understanding of automatic motion initiation, we can explain is as some cases. Before that, we understand that the aim of this initial movement is to find the line pattern.

Case 1: Automatic mode gets on when trolley at free space.

- GPS Module checks the current position coordinates and destination coordinates.
- Trolley takes a direction, which gives reduced distance between current and final positions on movement.
- Trolley starts moving forward.
- Trolley moves forward till the IR sensor senses the line pattern.
- If it faces the obstacle before that, enters the algorithm for obstacle avoidance.

Case 1: Automatic mode gets on when trolley at line pattern.

- IR sensor senses the current position on line pattern.
- Enters the line follower algorithm directly and the trolley get aligned correctly for the exact line movement.
- Here the GPS module need not be active

7. 3 Trolley movement on free space

The free space trolley movement is the stage starts with mode shifting to automatic and ends usually with finding the line pattern. During this, the direction of motion is determined only by the GPS module attached in the system until it meets an obstacle. Whenever obstacle comes into picture, then obstacle avoidance algorithm is followed. Priority order in the free space movement are; 1st Checking for the line pattern, 2nd Obstacle avoidance, 3rd GPS Module During the entire motion of the trolley in free space it always check for the line pattern. Even though the trolley enters the obstacle avoidance algorithm, then also it enters the line follower algorithm and then deals the obstacle as a condition in line follower movement. GPS module has the role only at the case of initiation of motion at free space. For understanding the priority, we can consider a condition. That is, the trolley shifted to automatic mode while being stationary on the line pattern and an obstacle is situated in front. Then the trolley identifies the pattern first and then deals the obstacle as it is in the line follower movement. GPS module is not come into account here.

7.4 Movement over the line pattern.

The movement of the trolley to the assembly point is mainly done by special line pattern follower algorithm. Apart from the normal line follower, this is very different because the unidirectional motion of trolley towards the assembly point is possible. The algorithm is in such a way that, the trolley gets aligned to require direction if it enter to the line from any direction or any orientation. The Trolley points are interconnected through double line through the boundaries. Trolley moves in a particular direction and touches this line. Once if the trolley touches the line, it ignores the wireless positioning and move through the line.

The Line is drawn as double line with different thickness to detect which direction it moves. Since we have 5 IR sensors for determining the special line pattern, we get a good result for the pattern following. IR sensor input checks almost all the possibilities and controller directs the motor to attain the unidirectional forward motion. If the trolley position on line is just opposite to the forward motion condition, then the trolley makes an 180^0 rotation and moves forward. That is the, reverse movement id completely arrested in the case of line follower. Fig 7.1 shows the position of trolley in forward motion.

Fig 9. Trolley movement in line

7.5 Obstacle avoidance

During the motion in free space and line pattern there is the possibility to meet with obstacle. Generally, the obstacle avoidance algorithm works on the principle of obstacle coverage, not on redirecting principle. Whenever the trolley meets the obstacle, it deviates from its path for only covering the obstacle and get back to the initial flow. Before gets deviated from the flow, the trolley makes a horn sound for the obstacle, and wait for a while to get the path cleared. If the obstacle exists there even after the delay time, then the trolley moves according to the simple logics used in the algorithm. The obstacle sensors give the inputs for the coverage of obstacle. After entering the line follower algorithm, the main priority will be given to obstacle avoidance. The execution of the logic is expressed in the Fig 10. Logic that follows while obstacle avoidance are;

Set 1

- Front left senses and Front right is free then trolley moves right
- Front right senses and Front left is free then trolley moves left
- Both the front left and front right senses then trolley moves left (by default)

While moving right;

If right sensors senses moves back (initial position and move left)
If oth front sensors are free moves forward *{Set 2}*
If front sensors senses then follow set 1
If left sensors are free return to initial (move left)
If right sensors senses then follow set 2

While moving left;

If left sensors senses moves back (initial position and move right)
If both front sensors are free moves forward *{Set 3}*
If front sensors senses then follow set 1
If right sensors are free return to initial (move right)
If right sensors senses then follow set 2

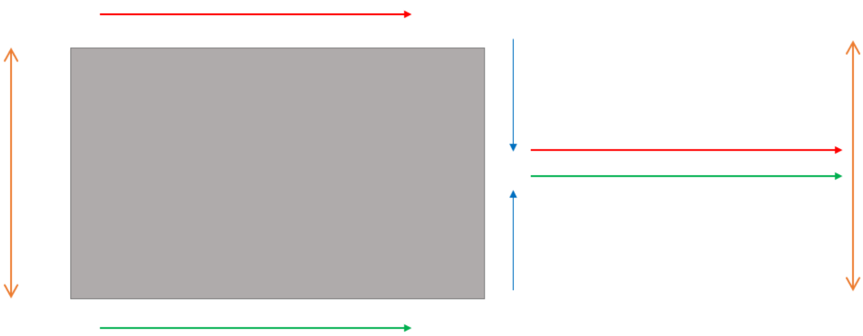


Fig 10. The execution of the obstacle avoidance logic

7. 6 Working of rotating plate

Rotating plate has a key role in the obstacle avoidance. Whenever the moving trolley interacts with an obstacle, that is the front obstacle sensors senses the presence, the trolley needs to deviate from its current path. In our algorithm usual left and right turn procedure following in line pattern, is not enough. Here we need to move the trolley sideways without affecting its body orientation. Rotating plate plays this role in the system Rotating plate have two drive wheels on the bottom and attached to the

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Factura '20 - National Conference On Emerging Trends In Manufacturing
NSS College of Engineering, Palakkad, Kerala, India
Vol. 7, Special Issue 1, August 2020

baseplate through bearings. By the help of wheel movement, we can rotate the plate as we need. But the problem is we need the rotation of the plate alone in some times and the rotation of whole trolley some times. This separation is achieved by the use of solenoid lock in the system. By the sector plate bearing the rotating plate can have a circular motion without affecting the trolley orientation. The rotation of the plate is possible by the differential motor rotation. The plate rotates 90^0 clockwise by rotating the right motor backward and left motor forward. And the plate rotates 90^0 counter clockwise (-90^0) by rotating the left motor backward and right motor forward. Angle of rotation is cross verified by the angle sensor attached to the shaft on the middle of plate. We need the independent rotating plate action only in the obstacle avoidance algorithm, particularly just in the instant of changing drive direction. If the plate attains the position then the lock is come into action.

Case 1: Move Right

This move right command includes

1. Unlock the solenoid lock
2. Turn the rotation plate 90^0 clockwise
3. Verify the angle of rotation
4. Lock the rotating plate at 90^0
5. Move forward

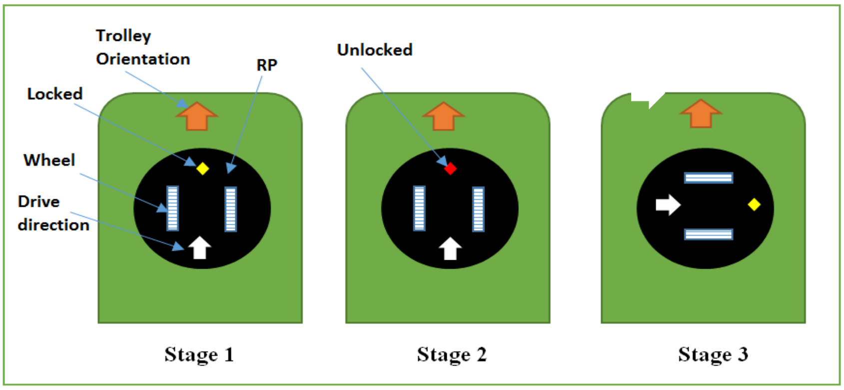


Fig 11. Action of RP in move right command

Case 2: Move Left

This move right command includes

1. Unlock the solenoid lock
2. Turn the rotation plate -90^0 anticlockwise
3. Verify the angle of rotation
4. Lock the rotating plate at -90^0
5. Move forward

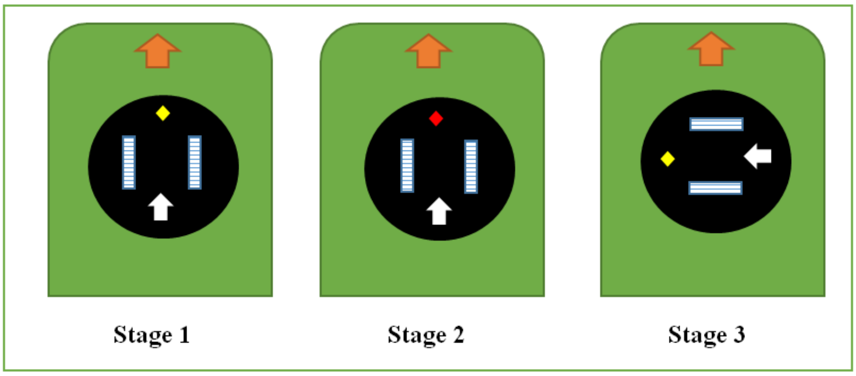


Fig 12. Action of RP in move left command

Case 3: Move Backward (Obstacle sensed at side in move right/move left case)

This move right command includes

1. Unlock the solenoid lock
2. Turn the rotation plate 180^0 clockwise/anticlockwise
3. Verify the angle of rotation
4. Lock rotating plate at -90^0 (if from 90^0) and at 90^0
5. Move forward

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Factura '20 - National Conference On Emerging Trends In Manufacturing
NSS College of Engineering, Palakkad, Kerala, India
Vol. 7, Special Issue 1, August 2020

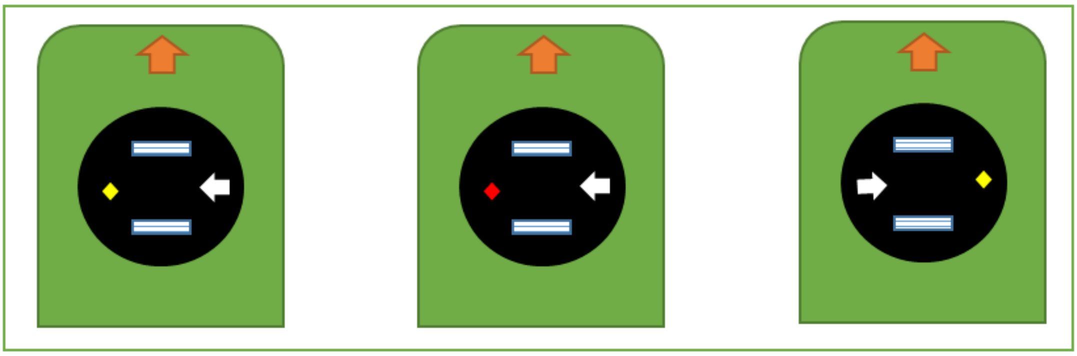


Fig 12. Action of RP in move backward command

7.2 Testing of Trolley

After the assembly all the parts and circuiting, the working of the trolley has tested in realistic conditions. The switching of automatic and manual phases, entering to the line pattern and Different set of obstacle conditions are tested.

7.2.1 Switching of mode test

As per our program logic, the trolley has to switch to automatic mode if it has no load and being unmoved for 2 minutes. This switching process mainly driven according to the input from load sensor. Here we tested the trolley with different load levels on the trolley. We have an upper load limit of 50 kg. We carried out the test by putting the weights of 5kg, 10kg, 20kg, 40kg, and 50kg. The results are shown in table 3.

Table 3 Results of mode switching test

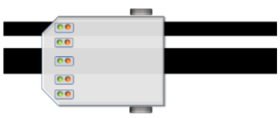
Load	Stationary time	Status
50 kg	120 sec	Manual mode
40 kg	120 sec	Manual mode
20 kg	120 sec	Manual mode
10 kg	120 sec	Manual mode
5 kg	120 sec	Automatic mode
0 kg	120 sec	Automatic mode

From the test we got that the trolley detects up to 5 kg as. This is due to the resolution of the load cell used. The mode switching happens at exact time delay after sensing the weight. For better accuracy we have to modify the load sensor to detect the small amount of loading also.

7.2.2 Testing the entry of trolley to the pattern.

The trolley has number of possible orientations to enter the line pattern. While creating the algorithm we have considered almost all the possibilities. Here we are testing some of them. The line follower works on the input data from set of IR sensors. Major 5 orientations are tested here. The details and results are tabulated in the Table 4. From this test we observe that, the trolley follows the algorithm absolutely and moves in required scenario.

Table 4. Testing the entry of trolley to the pattern

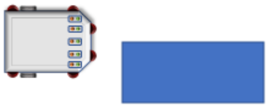
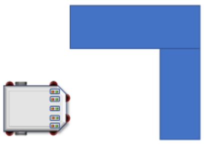
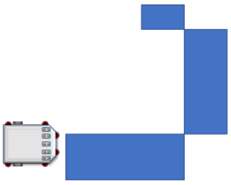
No	Orientation	IR Status	Action
1		L2 = 0 L1 = 0 C0 = 0 R1 = 1 R2 = 1	Turns right and enter the line pattern
2		L2 = 1 L1 = 1 C0 = 0 R1 = 0 R2 = 0	Turn left and enter the line pattern
3		L2 = 0 L1 = 0 C0 = 1 R1 = 0 R2 = 1	Moves back (180^0 rotation and move forward.

4		L2 = 1 L1 = 1 C0 = 1 R1 = 1 R2 = 1	Rotate the trolley to right
5		L2 = 0 L1 = 0 C0 = 0 R1 = 1 R2 =0	Moves Right and entered the pattern

7.2.3 Obstacle condition testing

In the path planning algorithm for dynamic environment we have to deal with many obstacle conditions. Even though we programmed the trolley by considering the common and basic conditions, we have to confirm the proper working by testing in some common obstacle conditions. The test details are described in Table 5

Table 5. Obstacle condition testing

No	Obstacle	Actions followed by the trolley	Inference
1		Move left Move forward Move right	Covers the obstacle through left
2		Move right Move forward Move left	Covers the obstacle through right
3		Move left Move forward Move right	Covers the obstacle through left
4		Move left Move backward Move forward Move left	Follows the logic and covered the obstacle through the right
5		Move left Move forward Move left Move forward Move right Move forward Move right Move forward Move right	Covers the obstacle through left. Shows the tendency to return to initial path
6		Move left Move forward Move left Move backward Move backward Stop	Trolley trapped inside the obstacle area after trying twice.

By checking the action of trolley in different obstacle condition gives almost 100 percentage results. The trolley follows the logics and covers the obstacle successfully. While doing all these practical tests, the motor action, rotating plate action and locking and unlocking efficiency are also verified. From the positive results of test, we can ensure the action of motor, rotating plate, solenoid lock and angle sensor are proper.

VI. CONCLUSION

The design and fabrication of the smart trolley is completed. The unique and innovative idea of a fully functional smart trolley system, which have capability to assemble in single point automatically after use, is manufactured and tested. An effective and simple path planning algorithm is created for the action of the wheeled mobile robot, especially for the trolley system. The developed working model produced good results on testing of the various requirements. All the components worked well and the objectives were satisfied to a very good degree. Literature survey for this work points on effective path planning algorithm for dynamic environments. Various path planning algorithms were studied, adopt the ideas to create a new path planning algorithm. Survey also discusses the different works which are done in fabricating a wheeled mobile robot for different applications. And we found that the idea of smart trolley is a unique one. The preparation of path planning algorithm detailly explained. The possible conditions of obstacle, movement over a special line pattern and free space movement are well described here. It is confident to say that; the algorithm considers all the possibilities of an ordinary dynamic environment. The dynamics of the mechanical components of the trolley is explained. Discussed the Forward and Inverse Kinematics of the differential drive robot and formulated the required equations for robot motion here. The design of the smart trolley is discussed, it includes the 3D modelling, design of motion, design of rotating mechanism, design of bearing, robot software architecture and the battery selection. The stepwise explanation of the working of trolley is explained. The action of trolley in different conditions, logical working of algorithm, action of rotating plate mechanism are detailed here. Also, the testing to ensure the trolley action and its results are discussed in the later portion of the chapter.

8.1 Future Work

The smart trolley system opens wide possibilities of future work. The working model of smart trolley can be upgraded in many ways. The future possibilities can be discussed related to some areas, like upgradations in structure, upgradation in sensing resolutions, upgradation in the mechanisms and upgradation in the algorithm Being a utility device, the trolley should have a particular structure. Here the structure of the working model is not suitable for airports and shopping malls. So, upgradation in physical structure in without affecting other system needs is essential. We can make the trolley smarter by incorporating the ideas of IoT. The follow me trolleys are available already. The incorporation of follow me trolley features with this automatic assembly feature produce a really smart trolley. We can hope in future. Enabling additional sensors or modules like Bluetooth will gives more connectivity with a central point. The addition of image processing idea will open the applicability in very strange environment movements.

REFERENCES

[1] Bremermann HJ. "The evolution of intelligence. The Nervous system as a model of its environment," Technical Report no.1, contract no. 477(17). Washington, Seattle: Dept. Mathematics, Univ.; July 1958.
[2] Xia J, Michalewicz Z, Zhang L, Trojanowski K. Adaptive evolutionary planner/navigator for mobile robot. IEEE Transcation on Evolutionary Computation Apr 1997;1(1).
[3] Zadeh LA. Fuzzy sets. Inf Control 1965;8(3):338e53.
[4] Zavlangas PG, Tzafestas SG. Motion control for mobile robot obstacle avoidance and navigation: a fuzzy logic-based approach. Syst Anal Model Simulat 2003;43(12):1625e37.
[5] Janglova D. Neural networks in mobile robot motion. Int J Adv Robot Syst 2004;1:15e22.
[6] Yang XS. Nature-inspired metaheuristic algorithm. Luniver press; 2008.
[7] Hidalgo-Paniagua A, Miguel A, VegaeRodriguez, Ferruz J, Pavon N. Solving the multi-objective path planning problem in mobile robotics with a fireflybased approach. Soft Computing 2015;1e16.{123}
[8] Eberhart RC, Kennedy JA. A new optimizer using particle swarm theory. In: Proceedings of the sixth international symposium on micro machine and human science. Piscataway, NJ, NAGOYA, Japan: IEEE service center; 1995. p. 39e43.
[9] Tang X, Li L, Jiang B. Mobile robot SLAM method based on multi-agent particle swarm optimized particle filter. J China Univ Posts Telecommun 2014;21(6):78e86.
[10] Macro D. Ant colony system: a Cooperative learning approach to the travelling salesman problem. IEEE transaction on evolutionary computation 1997;1(1):53e66.
[11] Guan-Zheng T, HE Huan, Aaron S. Ant colony system algorithm for real time globally optimal path planning of mobile robots. Acta Autom Sin 2007;33(3): 279e85.
[12] Karaboga D. An idea based on honey bee swarm for numerical optimization. Erciyes university, engineering faculty, computer engineering department; 2005. Technical report-tr06.
[13] Contreras-Cruz MA, Ayala-Ramirez V, Hernandez-Belmonte UH. Mobile robot path planning using artificial bee colony and evolutionary programming. Appl Soft Comput 2015;30:319e28.
[14] Yang XS, Deb S. Cuckoo search via Levy flights. In: Nature and biologically inspired computing, NaBIC 2009, IEEE world congress; 2009. p. 210e4.
[15] Zelinka, I.. Somaself-organizing migrating algorithm. In: New optimization techniques in engineering. Springer; 2004, p. 167–217.
[16] Bao, D. Q., & Zelinka, I. (2019). Obstacle Avoidance for Swarm Robot Based on Self-Organizing Migrating Algorithm. Procedia Computer Science, 150, 425–432. doi:10.1016/j.procs.2019.02.073
[17] Jean-Claude Latombe, Robot Motion Planning, Kluwer Academic Publishers, Boston, MA, 1991.
[18] Chan, R. P. M., Stol, K. A., & Halkyard, C. R. (2013). Review of modelling and control of two-wheeled robots. Annual Reviews in Control, 37(1), 89–103.
[19] Nițulescu, M. (2003). Using Mobile Robots in Intelligent Assembly or Disassembly Process. IFAC Proceedings Volumes, 36(23), 49–54. doi:10.1016/s1474-6670(17)37660-7
[20] Armesto, L., Torres, J. C., & Tornero, J. (2003). Hardware architectures for mobile robots. IFAC Proceedings Volumes, 36(17), 375–380. doi:10.1016/s1474-6670(17)33423-7
[21] Budanov, V. M., & Devyanin, Y. A. (2003). The motion of wheeled robots. Journal of Applied Mathematics and Mechanics, 67(2), 215–225. doi:10.1016/s0021-8928(03)90008-6