ENPM662: Project-02

Manipulative Wheeled Robot for Hospitality Industry

Rohit Reddy Pakhala
UID: 119399125

Yashveer Jain
UID: 119252864

Master's of Engineering Robotics
A. James Clark School of Engineering
University of Maryland- College Park
USA
Date of Submission: 7 December 2022

Contents

List of Figures

List of Tables

1 Introduction

Mobile robots have become more commonplace in commercial and industrial settings. Hospitals have been using autonomous mobile robots to move materials for many years. Warehouses have installed mobile robotic systems to efficiently move materials from stocking shelves to order fulfilment zones. Mobile robots are also a major focus of current research and almost every major university has one or more labs that focus on mobile robot research. Mobile robots are also found in industrial, military and security settings.

In restaurants, robots are very useful and it decreases labour cost and increases labour productivity and service quality. These kinds of robots are very efficient and convenient during the covid times without much human intervention, which will deliver the food from the kitchen to the table. This ensures the hygiene of the food and the safety of the customers. Few others also done the similar task [1], but our model is different and cater bigger customer base, and not just limited to restraunt but to wider set of application such as hospitals etc.

The proposal has 19 sections. The introduction describes the framework for this report and motivation which describes why the problem statement was chosen, the background research, and the interests described with a brief description of the problem statement. Within the section Robot Description, information is provided about the structure of the robot, its size, the sensors it will use, the motors and actuators it will use, as well as the material it will be constructed out of. Under Scope Description, we discuss the possible area of study and the part of the robot's capabilities that will be investigated. Following the next bullet point, Scope Appropriateness, is a description of the significance of the topic chosen, and how this research contributes to the advancement of technology. The following section, Model Assumptions, lists all the assumptions we will consider while designing and simulating the robot. Next, the section on Approach to Performing the Work will describe the implementation process, the tools, and dependencies to be employed, such as SolidWorks, Gazebo, and others. Milestones and Timeline provides a breakdown of the different checkpoints which divide the whole project in half, as well as defining the overall goal of the project and its ambitious aims. In addition to the timeline for the entire project, it will include some buffer time for certain unforeseen factors.

2 Application

There are numerous applications of service robots in the current world. There are many applications instead of serving food, including but not limited to hospitality, home, and patients with covid positive. It delivers the food to the customers at the table. This robot only needs the power to run it like 2.5 dollars per hour unless the workers in the industry. This enhances the good customer experience. This can also use in hospitals where there is a covid ward or to serve food for the patients. These robots are not only limited to restaurants but also it is very useful in health care, hospitality, warehouse automation and many more.

3 Robot Type

There are six main robot types. This project uses a cylindrical robot for the manipulator and Automated Guided Vehicles (AGVs) for the base. The base is a differential drive robot which rotates and moves. It contains one revolute joint at the base of the manipulator and a prismatic joint which moves linearly and it above the revolute Joint.

Differential drive: If both the wheels are driven in the same direction and speed, the robot will go in a straight line. If both wheels are turned with equal speed in opposite directions, as is clear from the diagram shown, the robot will rotate about the central point of the axis. Otherwise, depending on the speed of rotation and its direction, the center of rotation may fall anywhere on the line defined by the two contact points of the tires. While the robot is traveling in a straight line, the center of rotation is an infinite distance from the robot. Since the direction of the robot is dependent on the rate and direction of rotation of the two driven wheels, these quantities should be sensed and controlled precisely.

4 DOFs and Dimension

This is a 5 DOF where the differential drive robot contributes 3 DOF. It can move x,y, and θ (which is wheels rotation). On top of the mobile robot, a 2 DOF manipulator is placed. Fig (1) shows the 2 DOF manipulator.

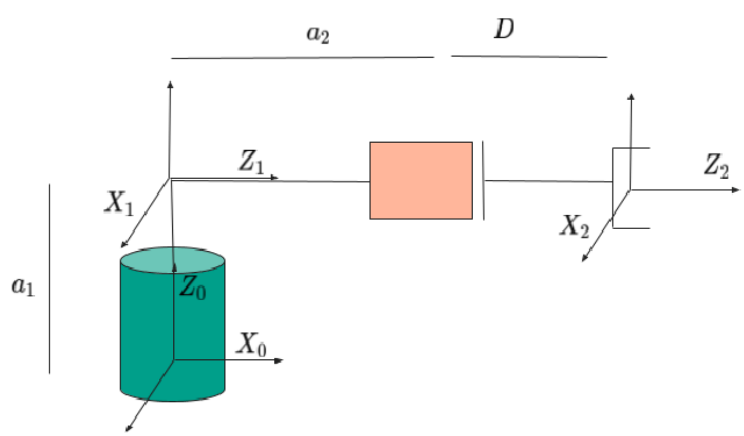


Figure 1: Manipulator

Dimensions:

Table 1: Dimensions

Body	Dimension
table	(length x width x height) = (39.37 in x 39.37 in x 39.37 in)
wheel	Diameter: 8 inches Thickness: 3 inches
Wheel Joint	2in radius, and located at (0,3) inch center robot base wheel joint holes.
Robot body	Wheel base: 30 inches Length: 30 inches Breadth: 30 inches Height: 25 inches
New Design Manipulator link1 (Revolute joint z-axis)	Link length = 15 inch Bottom hole = dia 4 inch , center (-6,0) Top hole = dia 2 inch , center (6,0) Distance b/w link1 and link2 is 12 in
Fixed link	Link length = 8 inch Height = 3 in Width = 6 in Bottom hole = dia 2 inch , center (-2,0) Shape cuboid, Hollow with shape (7,5,1) inch
Material	Aluminum coated and rubber coated wheels
Actuators	DC Motors: 2 Rack and pinion: 1
Link2: End Effector (tray)	Length: 10 in Breadth: 5 in Max reach 10 in. Thickness: 0.5 in
Tray diffusor	Length : 4in

5 CAD Models

The below diagrams show the different parts of the CAD model. It contains:

- Fig (2a) shows wheels of robot.
- Fig (2b) shows link1 of robot.

- Fig (2c) shows robot base of robot.
- Fig (2d) shows the tray diffusor of the robot, which moved the tray from the robot to the table.
- Fig (2e) shows end effector of robot.
- Fig (2f) shows fixed link 2 of the robot, which supports the end effector link.
- Fig (2g) shows base to link 1 joint.
- Fig (2h) shows final assembled robot.

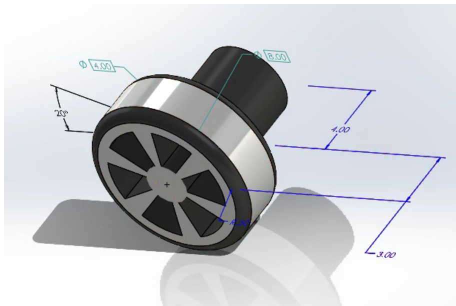
6 D-H parameters

Table (2) represents the DH parameters for the 2DOF manipulator. where,

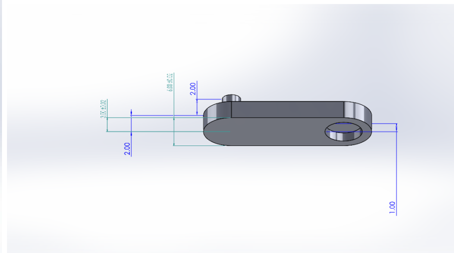
- a_1 is the distance between link 1 from link 2 in z axis from frame0, and for simplicity, we're assuming the $a_1 = 0$.
- a_2 is the length of link1 i.e 12 (in) $+ 7$ (in) of end effector length till tray point. That is $a_2 = 12 + 7 = 19$ (in). In section (9) this is the min length an end effector can reach.
- θ_1 is the rotation of the revolute joint of link 1.
- D is the distance variable covered by the prismatic joint of link 2.

frame	$\boldsymbol{\theta}$	$\boldsymbol{\alpha}$	$\boldsymbol{r}$	$\boldsymbol{d}$
$0 \rightarrow 1$	$\theta_1 - 90\,\text{deg}$	$-90\,\text{deg}$	0	a_1
$1 \rightarrow 2$	0	0	0	$a_2 + D$

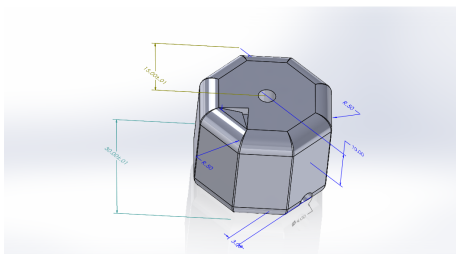
Table 2: DH Parameters



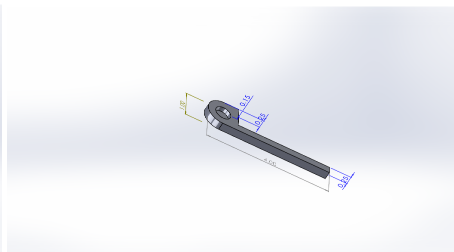
(a) Wheel



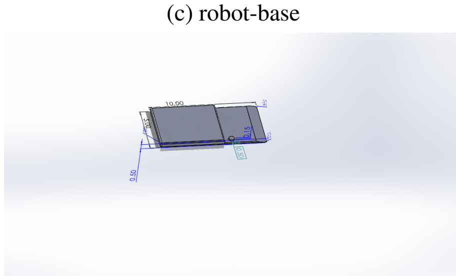
(b) Link1



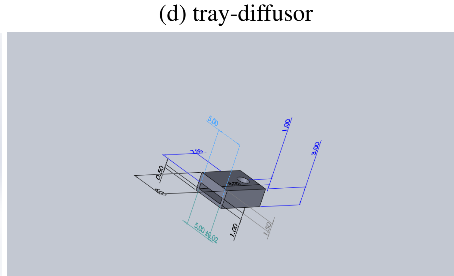
(c) robot-base



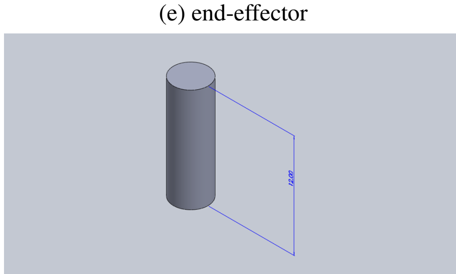
(d) tray-diffusor



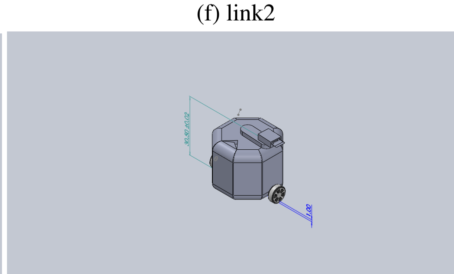
(e) end-effector



(f) link2



(g) base-to-link1-joint



(h) final-assembled-robot

Figure 2: CAD Models

7 Forward Kinematics

7.1 Differential Drive Robot

The forward kinematics of the Differential Drive robot is given by:

When moving in a straight line:

$$\Delta x = V cos(\theta) \Delta t \tag{1}$$

$$\Delta y = V sin(\theta) \Delta t \tag{2}$$

$$\theta = 0 \tag{3}$$

When rotating:

$$\Delta x = 0 \tag{4}$$

$$\Delta y = 0 \tag{5}$$

$$\Delta = (2v\Delta t)/l \tag{6}$$

7.2 Manipulator

Using the DH convention [2] the generalized Transformation Matrix is:

$$H_i^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & r_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & r_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Using this convention, we can write homogeneous matrices for link1 and link2.

$$H_1^0 = \begin{bmatrix} \sin(\theta_1) & 0 & \cos(\theta_1) & 0 \\ -\cos(\theta_1) & 0 & \sin(\theta_1) & 0 \\ 0 & -1 & 0 & a_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

$$H_2^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & a_2 + D \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

So, the final transformation from frame 2 to frame 0 is,

$$H_2^0 = H_1^0 * H_2^1 \tag{9}$$

$$H_2^0 = \begin{bmatrix} \sin(\theta_1) & 0 & \cos(\theta_1) & (a_2 + D)\cos(\theta_1) \\ -\cos(\theta_1) & 0 & \sin(\theta_1) & (a_2 + D)\sin(\theta_1) \\ 0 & -1 & 0 & a_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

from eq (10), it can be said that if a point is on end-effector i.e. $P_2 = [0, 0, 0, 1]^T$, then the point location with reference to the base frame will be:

$$P_2^0 = H_2^0 * P_2$$

$$P_2^0 = \begin{bmatrix} \sin(\theta_1) & 0 & \cos(\theta_1) & (a_2 + D)\cos(\theta_1) \\ -\cos(\theta_1) & 0 & \sin(\theta_1) & (a_2 + D)\sin(\theta_1) \\ 0 & -1 & 0 & a_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$P_2^0 = \begin{bmatrix} (a_2 + D)\cos(\theta_1) \\ (a_2 + D)\sin(\theta_1) \\ a_1 \\ 1 \end{bmatrix} \tag{11}$$

Given $a_2 = 19$ and $a_1 = 0$, the P_2^0 will be:

$$P_2^0 = \begin{bmatrix} (19 + D)\cos(\theta_1) \\ (19 + D)\sin(\theta_1) \\ 0 \\ 1 \end{bmatrix} \tag{12}$$

7.3 Forward Kinematics Validation

From eq (11) we can validate the Forward kinematics that the end effector will generate a circular path as shown in section (9).

8 Inverse Kinematics

Inverse Kinematics of the manipulator:

$$\theta = tan^{-1}(y_f/x_f) - 90 \tag{13}$$

As we have a $-90\deg$ offset which is shown in table (2) for θ_1 we need to take into account while computing the inverse kinematics for θ.

$$D + a_2 = y_f/cos(\theta) = -x_f/sin(\theta) \tag{14}$$

where, x_f and y_f are the final position where end-effector need to reach w.r.t to robot frame. θ is the θ_1 rotation along link1, D is the variable distance end-effector can move, a_2 is the link1 length, as stated in section (6)

9 Workspace Study

Figure (3) demonstrates the workspace of the robot manipulator. It shows the distance/area covered by the manipulator. The maximum range of the endpoint of the end-effector is 29 (in) and the minimum range is 19 (in) and has circular coverage.

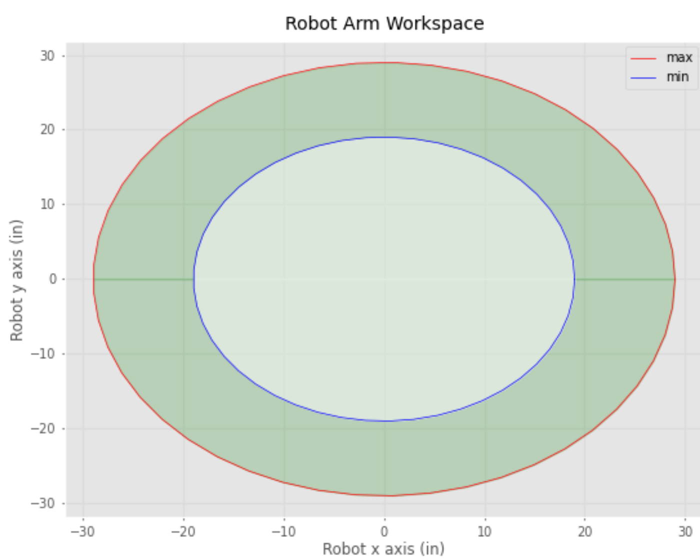


Figure 3: Arm Workspace

10 Assumptions

There are a few assumptions taken into consideration before developing the project

1. Tables are located in such a way that a robot in a single location can cater to 2 tables, one is on its right and another one will be on its left and are within reach

of the robot arm (that is within 10 inches on each side). This pattern can be followed by any even number of tables. In our case, we have only taken 6 tables arranged in fixed locations, defined by the above constraint.

2. so the final end-effector position can be $(-0.55, 0)$ (m) for right w.r.t manipulator base frame, or $(0.55, 0)$ (m) for left w.r.t manipulator base frame.

3. Robot first link is in front position at $270 \deg$ rotation. That means the tray will be pointing in front of the robot, when first link θ_1 is $270 \deg$.

4. Robot will always start from its origin, and so the kitchen is located at the origin.

5. Robot will be operated using teleop or key commands.

6. There are no dynamic obstacles, that is no person is moving at a time of robot motion.

7. Tray will be picked up by human customers, according to their orders.

11 Robot Control

The robot is designed taking into consideration the task, which needs to be done which is the delivery of the Dish from the kitchen to the table. For this, there are mobile manipulator was best suited for the task.

For mobile motion, we choose the differential drive system, for easy maneuverability and command control, the gazebo plugin 'differential_drive_controller' (refer to [3] and [4] for more detail) was used to control the motion of the robot, with this plugin we also got the odometry data and transform of the robot with the world frame. This was helpful for better control and moving the robot more robustly.

For Manipulator, we used ROS 'joint_position_controller' for 3 links, and tuned the P, I, and D gain for each link's respective PID controller, for more detail on PID controller refer to [5],

$1.$ Link1 which rotates $360 \deg$.

$2.$ Link2 which is a prismatic joint and acts as a tray, moving in and out, within range i.e $[0, 10]$ (in).

$3.$ Tray diffusor, which is used to serve the dish to the table, by pushing the tray from the end-effector (link2) to the table, this is still hard to control because the PID parameter tuning is difficult, for this link, due to its low mass, so for now, we stabilized it but not able to control it fully

12 Gazebo Visualization

In Gazebo Visualization, Fig (4) shows the Gazebo visualization, in this we ran the robot in the custom environment, using teleop to move the robot from the kitchen (i.e. origin in world frame) to a fixed table location, and then pressing table number to move the arm to serve the tray on that table (for our case it was 5).

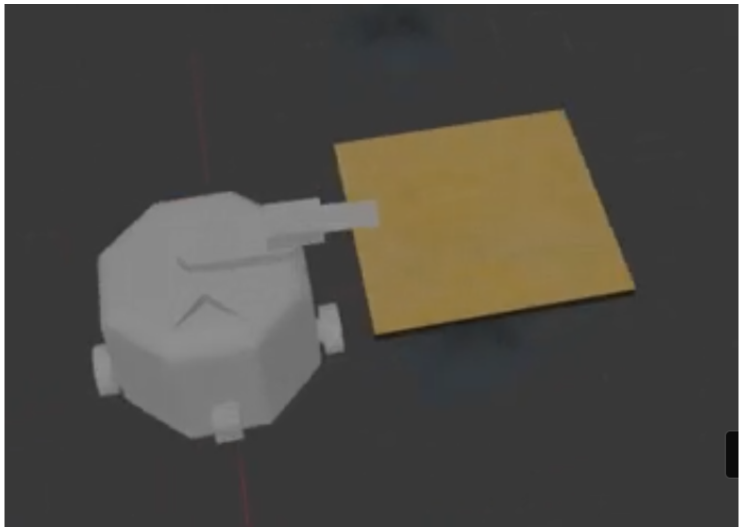


Figure 4: Gazebo visulisation

13 RViz Visalization

And Fig (5) shows the Rviz visualization, which demonstrates the motion of each joint, such as wheels (left and right), link1 joint, end-effector (link2) joint, and the tray diffusor.

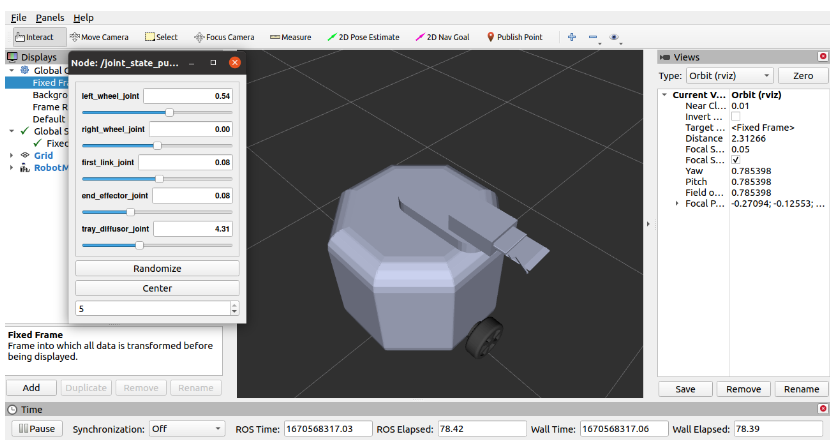


Figure 5: Rviz visulisation

14 Problems

We Faced many challenges when working on the project, such as:

1. Difficult to tune the PID controller parameters for the robot to function.
2. Deploying the Solidwork model in the gazebo environment, the robot was breaking apart and not able to move.
3. Adding inverse kinematics to the code gave us a lot of errors.
4. Designing the world and placing the table coordinates as it is assumed to be located at a particular distance.
5. Assumption tables are placed in a way that the robot manipulator reaches them.
6. Integrating the gazebo plugin to the custom robot, was difficult, as it contains certain packages installed, and parameters need to be edited to work.

15 Lessons Learned

Through this project, We have encountered problems and learned a lot by solving those problems. Such as:

- How to add a gazebo plugin and integrate it with a custom robot.
- Make plans and discuss the details of the project before implementing.
- How to make Cad Models and export it to urdf.
- Understand in-depth knowledge of Inverse kinematics and implementation.
- Working with ROS and gazebo.
- Debugging methods.
- Always keeps fallback plans when implementing the project.

16 Implementation details

Our Code and Cad model of Robot is present on github. The GitHub repository contains a readme file which has the instructions to run the package. For our project we used robo_motion_v3_only_arm_ik.py code. And our code is built on top of ROS Noetic and python3.8.

17 Conclusion

These models will help the hospitality industry to grow its profits. It not only decreases manpower but also increases service quality and labor productivity. Uses less manpower and decreases human intervention. Although, the use of this solution may decrease human errors even the robots may make some mistakes and have some limitations such as the end-effector length.

18 Future Work

Like everything, it has limitations over its capability. The robot can serve the customers according to the table and the position they are arranged. This data is already fed to the robots. Thus, if you change the table position the robot will serve the wrong table. Though robots are in the serving and decreasing manual intervention, manual work still exists as robots can only serve and takes back the plate. Future work includes the robot running autonomously even though the position of the table changes.

References

[1] Takeshi Shimmura, Ryosuke Ichikari, Takashi Okuma, Hiroyuki Ito, Kei Okada, and Tomomi Nonaka. Service robot introduction to a restaurant enhances both labor productivity and service quality. *Procedia CIRP*, 88:589–594, 2020. 13th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 17-19 July 2019, Gulf of Naples, Italy.

[2] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. Wiley select coursepack. Wiley, 2005.

[3] Julio V. Méndez Bethencourt, Qiang Ling, and Ángel Valera Fernández. Controller design and implementation for a differential drive wheeled mobile robot. In *2011 Chinese Control and Decision Conference (CCDC)*, pages 4038–4043, 2011.

[4] Fahmizal Fahmizal, Angga Priyatmoko, Esa Apriaskar, and Afrizal Mayub. Heading control on differential drive wheeled mobile robot with odometry for tracking problem. In *2019 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, pages 47–52, 2019.

[5] Kiam Heong Ang, G. Chong, and Yun Li. Pid control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*, 13(4):559–576, 2005.