Tele-Operated Trajectory Tracking of Differential Drive Wheeled Mobile Robot Using Haptic Robot

Mr. Rahul M[1] Mr. B Abhiram Vishnu[2] Mr. Pranav T[3] Mr. Nandu C Reghu[4] Dr. Mija S J[5]
Department of Electrical Engineering
National Institute of Technology Calicut
Kozhikode, Kerala, India
rahulm.research@gmail.com[1] abhiramvishnu18@gmail.com[2] pranavt09@gmail.com[3]
nandhureghu11@gmail.com[4] mija@nitc.ac.in[5]

***Abstract*—Tele-operation is a kind of semi-autonomous system, with the objective of allowing human to control a device in remote location. This paper proposes such a system implemented with Omni Bundle haptic device and Q-Bot 2 Mobile Robot platforms connected in a master-slave configuration respectively. The paper also provides modelling, trajectory tracking controller for both robots and communication algorithm that takes place between them, in real time with experimental results**

***Keywords*— Tele-operation; PID; Trajectory Tracking; Kinematics; Master; Slave**

I. Introduction

Tele-Operation is a means of controlling and monitoring a remote device (called slave) over long distance by another device (called master).It's a semi-autonomous system, because the master control is provided by human and the slave control is purely autonomous. Tele-Operation is highly advantageous at places or situations where direct human reach is not desirable or possible like high radioactive areas of nuclear plants, lunar or planetary exploration, inspection of underwater structures etc. This is where robots that are tele-operated comes with an advantage. Tele-Operated robots are not only used in situations where human reach is dangerous, but also used for medical applications, such as tele-surgery and rehabilitation [1], to provide health care to patients living in remote location.

Some of the previous works includes [2], a 2D plane based tracking problem where a back stepping technique is used for designing the algorithm for non-holonomic mobile robot and haptic rendering algorithm is designed to generate the haptic feedback based on the position error information. In [3] the robot is tele-manipulated by Omni Bundle haptic device. For this task, a model of the mobile robot is input-output linearized with dynamic extension and it is controlled safely through a tele-operated task in passivity framework. There are Fuzzy based control as in [4] [5], but the slave is a virtual robot.
There are papers where Omni Bundle is utilized as a master robot and tele-operation is achieved on a 6 DOF Slave robot as in [6], where the slave manipulator is equipped with a CCD camera and 3 DOF force sensor for haptic feedback. Further research on this made possible a rehabilitation system where the patient hand is fixed to the robot arm, so the therapist can move the patient hand along the predefined track through tele-operation as in [7].

In this paper, our objective is to create a path using *Omni Bundle* robot manipulator and recreate the trajectory using *Q-Bot 2* mobile robot upon tele operation. The two robots are in master-slave configuration, with Omni Bundle as master and Q-Bot 2 as slave. Q-Bot 2 will be tracking a trajectory in its workspace based on coordinates send from Omni Bundle, of which these coordinates are part of the reference trajectory that is manually taught to master robot. In order to achieve this objective, independent trajectory tracking control for both Omni Bundle and Q-Bot 2 are required which involves dynamic and kinematic modelling of Omni Bundle as a three degree of freedom (DOF) robot manipulator and Q-Bot 2 as a non-holonomic differential drive wheeled mobile robot followed by its controller design.

This work is not merely navigating a mobile robot using a joystick as in [8].We put forward a novel method of using a manipulator for generating reference track, as opposed to use tracker pens, pointer or markers, which requires additional arrangements to develop a coordinate systems, where as in this method, we have a fail proof mathematical model, that gives the coordinates in Cartesian system using forward kinematics of any point within the workspace. This method has huge potential applications of search and rescue, path planning, provided there is physical map within the workspace of manipulator, and this map coordinates can be scaled to real world for the mobile robot.

The initial sections focuses on tele-operation strategy, which is followed by control strategy on Omni Bundle and Q-Bot 2 respectively. The paper ends with experimental results and its discussions, conclusion and future scope.

II. TELEOPERATION ALGORITHM

A. System Overview

Master Robot is Quancer's Omni Bundle Haptic Device (See Fig. 1) which is modelled as a 3 DOF manipulator and Slave Robot is Quancer's Q-Bot 2 (See Fig. 2) which is modelled as Non-Holonomic Differential Drive Wheeled Mobile Robot.

Omni Bundle Haptic Device - is a robot with six revolute joints, of which three are actuated and the rest non-actuated. The non-actuated joints forms the end effector stylus, providing a wrist movement. The other three motors can actuate the tip of stylus to span a cartesian workspace. Digital encoders are used for position measurement along X, Y, Z positions and rotations about these axis (roll. pitch, yaw) are measured using potentiometers.

Fig. 1 Degree of Freedom of Omni Bundle [9]

The Phantom Block set for QUARC real-time control software provides the interface to MATLAB and SIMULINK. The three actuated joints are J1, J2 and J3. The non-actuated joints are the three wrist joints, J4, J5 and J6. The end-effector, sometimes referred to as the tool position, is the tip of the stylus shown in Fig.1 [9]

QBot 2 –The Quanser Q-Bot 2 is a multipurpose autonomous ground robot system incorporating a Microsoft Kinect, Gumstix DuoVero embedded computer, Yujin Robot Kobuki platform, and a Quanser DAQ, all of which can be controlled wirelessly. It comes pre-loaded with a wide range of sensors including bumper sensor, wheel drop sensor, cliff sensor, 3-axis gyroscope, Kinect RGBD sensor and wheel encoders with a maximum payload of 4.5Kg.The maximum speed is restricted to 0.6m/s for smooth performance and safety issues [10].

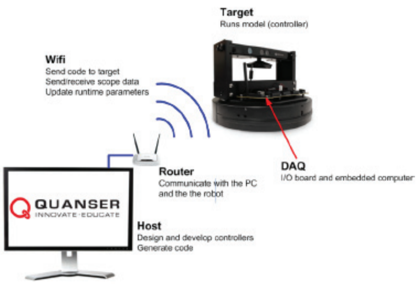

Fig. 2 System Overview of Q-Bot 22 [10]

TABLE 1 System Parameters of Q-Bot 22 [7]

Symbol	Description	Value	Unit
D	Diameter of the QBot 2	0.35	m
d	Distance between the left and right wheels	0.235	m
h	Height of the QBot 2 (with Kinect mounted)	0.27	m
v_{max}	Maximum speed of the QBot 2	0.7	m/s
m	Total mass of the QBot 2	3.79	kg

B. Algorithm and Flow of Control

The Omni bundle have two Ethernet ports, one connected to Master PC and the other connected to WiFi router, which in turn is connected to Slave PC The Q-Bot 2 and Slave PC establishes a wireless connection through the router. In the figure above, brown lines indicate wired and blue indicates wireless connection.

Fig. 3 System Overview

The proposed algorithm for tele-operation is as follows:

1. Set up a Local Area Network as shown in Fig. 3 above.
2. Manually Track the end effecter of Master (Omni Bundle) and guide it through selected points of the desired path.
3. Transmit the coordinates of selected points to Slave PC
4. Upon receiving, convert the coordinates, to fit in the workspace of Slave (Q-Bot 2)
5. Track the coordinates and produce a path using a Trajectory Tracking Control
6. Compare the results of Reference track and Actual track, in simulation and experiment

III. CONTROL STRATEGY – MASTER ROBOT

The overall control system adopted for Master Robot is as shown in the following flow diagram.

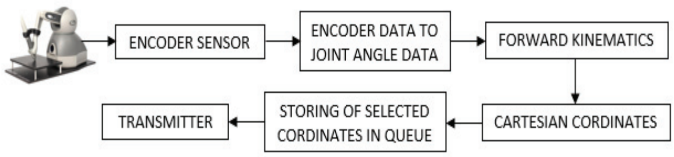

Fig. 4 Data Flow in Master Robot

Using the Quarc Simulink Blocks we will be able to read the encoder values obtained from Omni Bundle and convert them into Joint angle data i.e. θ_1, θ_2, θ_3.

The forward kinematics model shows the relation between the position and orientation of the end effectors and the joint coordinates. The kinematics chain of the Omni haptic device and its representation of variables and constants in the equation is shown in Fig. 5.

After employing the Denavit-Hartenberg convention on the manipulator, the transformation matrix is found and the kinematics can then be expressed as

$$X_m = -\sin\theta_1(L_2\sin\theta_3 + L_1\cos\theta_2) \quad (1)$$

$$Y_m = -L_2\cos\theta_3 + L_1\sin\theta_2 + L_3 \quad (2)$$

$$Z_m = L_2\cos\theta_1\sin\theta_3 + L_1\cos\theta_1\cos\theta_2 - L_4 \quad (3)$$

Where, L_1 = 133.35 mm, L_2 = 133.35 mm, L_3 = 23.35 mm, L_4 = 168.35 mm. L_1 and L_2 are the lengths of link 1 and 2, and L_3 and L_4 are the workspace transformation offsets between the origin of the end effectors and the first joint [11] [12].

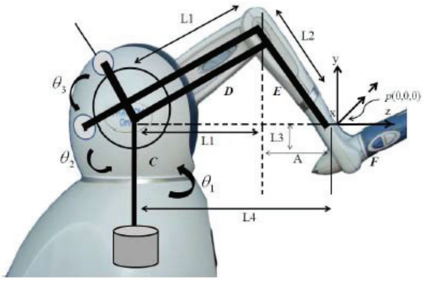

Fig. 5 Forward Kinematics Parameters [8]

The X and Y coordinates of Omni Bundle are need to be transmitted to the PC operating Q-Bot 2 over a Local Area Network. This LAN is established by connecting an Ethernet cable from Omni Bundle's 2nd LAN port to a Wi-Fi router that is connected to PC operating Q-Bot 2. PC is connected to Q-Bot 2 over Wi-Fi.

The required coordinates are stored in two separate queue that can be pushed and popped as needed. Apart from the coordinates input, there is trigger input which is connected to a push button on the stylus of the Omni Bundle. Whenever a desired point is reached, the user need to press the push button on the stylus, which will generate a pulse in the trigger input, this will push in the new coordinate in to the queue. Once the data is available in queue, it is popped in to a server-client program based on Quarc Software package for Simulink. For this there are some IP configurations as per the TCP/IP protocol, need to be configured in order to establish Omni Bundle as a server and Q-Bot 2 as a client.

IV. CONTROL STRATEGY – SLAVE ROBOT

The overall control system adopted for Slave Robot will be as follows:

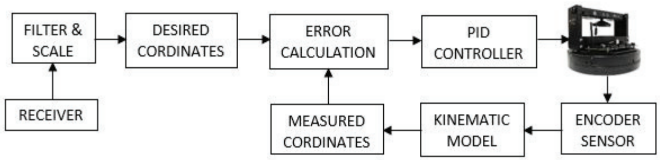

Fig. 6 Data Flow in Slave Robot

The coordinates is received by the client program that runs in Slave PC, the coordinates are then stored in an array. This is followed by a filtering and scaling program that removes repetitive coordinates and noise, finally resulting in a new array of desired coordinates scaled to the workspace of Q-Bot 2.

The second part is the path tracking algorithm, for which a set of measured coordinates and desired coordinates are required. Measured coordinates is a result of the feedback loop. The wheel encoders of Q-Bot 2 gives the right and left wheel velocities, which is transformed into X_m, Y_m and orientation coordinate (Φ_m) using the Kinematic model of non-holonomic DDWMR.

$$distance_{error} = \sqrt{(X_d - X_m)^2 + (Y_d - Y_m)^2} \quad (4)$$

$$\Phi_d = K \pm \tan^{-1}\frac{Y_d - Y_m}{X_d - X_m} \quad (5)$$

$$angle_{error} = \Phi_m - \Phi_d \quad (6)$$

$$u(t) = K_P\, e(t) + K_I \int e(t)\, dt + K_D \frac{d}{dt} e(t) \quad (7)$$

Where K takes different values for each quadrant.
X_d,Y_d are desired coordinates. X_m,Y_m are measured coordinates. Φ_m is the measured orientation. u(t) is the control value; K_P, K_D, K_I are the PID gains and e(t) stands for respective distance or angle error.

Once the errors are calculated, a PID controller is designed which outputs Velocity of center of mass (V_{COM}) to reduce the distance error using equation (4), and another PID controller which outputs Angular Velocity of center of mass (W_{COM}) to reduce the angle error using equation (5) and (6). V_{COM} and W_{COM} are the system control inputs. Being a nonlinear system, the stability of system is very sensitive to the gain parameters [13].This V_{COM} and W_{COM} can be converted into Left and Right wheel velocities using simple mathematical equations (8) and (9).

$$V_R = (\,2V_{COM} + W_{COM}\,L\,)/2R \quad (8)$$

$$V_L = (\,2V_{COM} - W_{COM}\,L\,)/2R \quad (9)$$

The parameter L stands for distance between wheels and R stands for radius of wheel.

The Simulink Block model for Simulation (Robot is only a Model) of Trajectory Track is as in Fig. 7

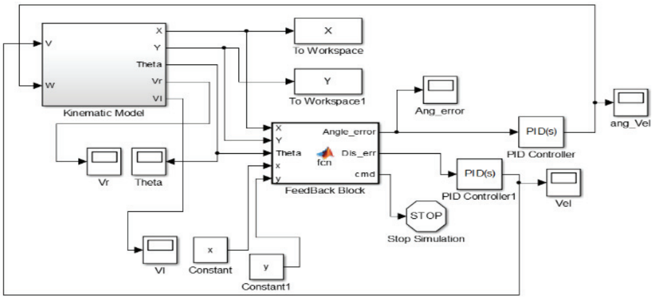

Fig. 7 Simulink Model for Simulation of Trajectory Tracking

Since the mobile robot has a maximum speed of 0.6m/s, a kinematic model is enough to model the robot for simulations and provides satisfactory results on experiments.

After integrating the hardware, the Simulink Block model will be as shown in Fig. 8

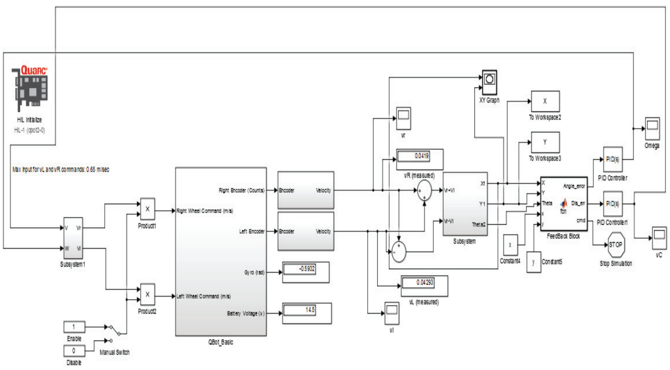

Fig. 8 Simulink Model for Hardware Implementation of Trajectory Tracking

V. EXPERIMENT AND RESULT

As part of several experiments and simulations on mobile robot, these are some the results obtained for trajectory tracking.

The coordinate (1, 1) is the required goal position, the robot (shown in triangle) is placed at several initial points and it is observed that the robot reaches the goal point (See Fig. 9).

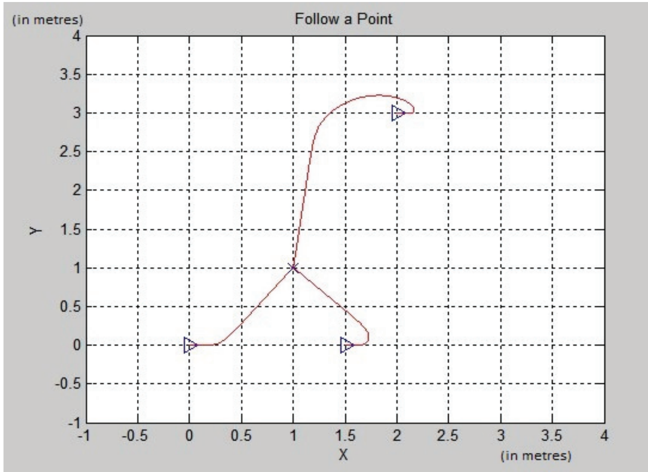

Fig. 9 Move to a Point

The robot starts from (1, 1) (shown with a x mark) follows the reference line 1 then moves into reference line 2 that has a different slope. The reference lines are in blue, actual tracking of robot in red in Fig. 10.

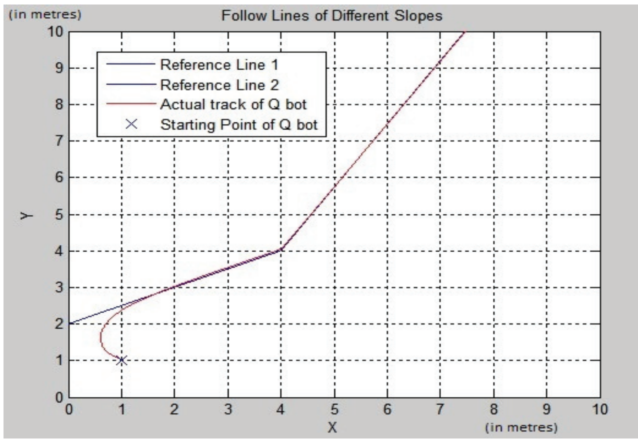

Fig. 10 Follow many lines

Robot's initial position is at (-1,-1) and it moves along many desired coordinates (See Fig. 11), red line is the robot's path.

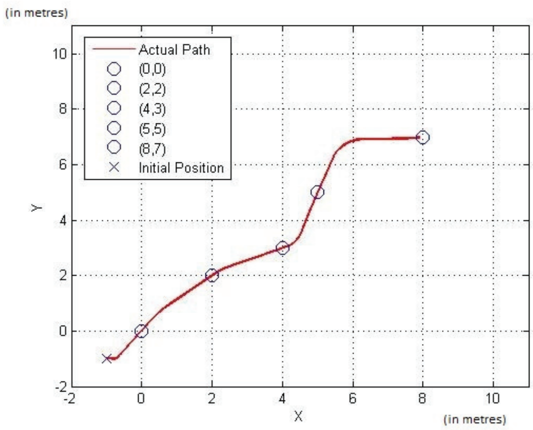

Fig. 11 Path tracking through several points

The robot starts from (1.5, -0.5) and moves through different points of a circle. The different points are highlighted, path of the robot is shown in red. The above set of points were now tested in hardware (See Fig 13), upon positive results from simulation (See Fig 12). The results obtained where satisfactory and within all limits, now the robot is ready for tele operation.

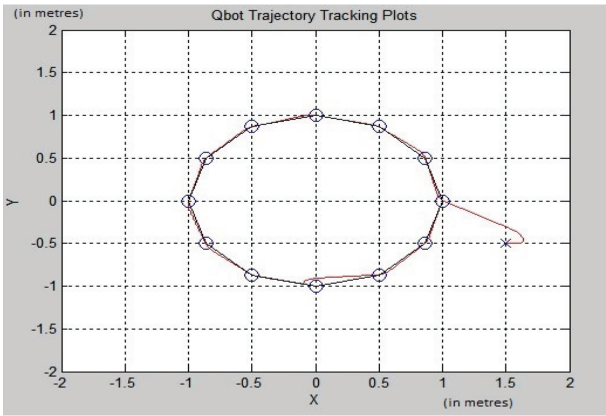

Fig. 12 Tracking several points of a circle

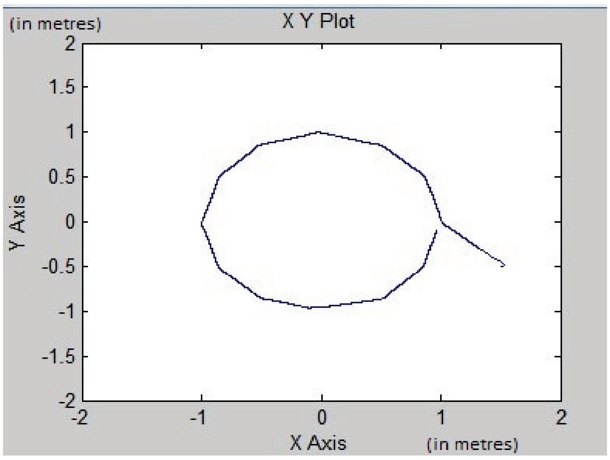

Fig. 13 Hardware implementation of Tracking

A set of points were transmitted from Omni Bundle and the Fig. 14 shows the simulation results of tracking algorithm of Q-Bot 2, As the simulation result (See Fig. 14) was satisfactory, experimentation was carried out (See Fig. 15).

In Fig 14 the black line represents the reference track and red line is the simulation tracking. The Fig. 15 shows the hardware implementation of the same.

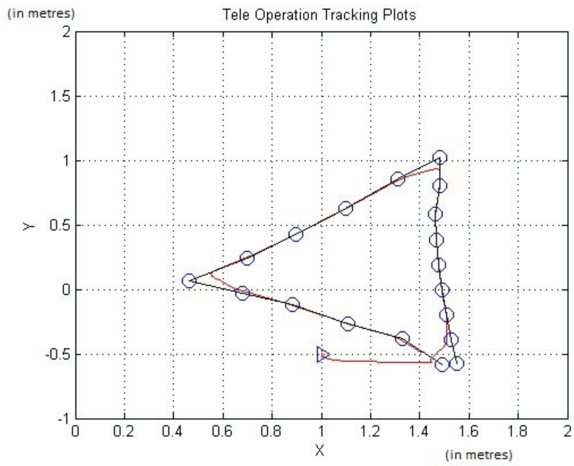

Fig. 14 Simulation result for tele-operation

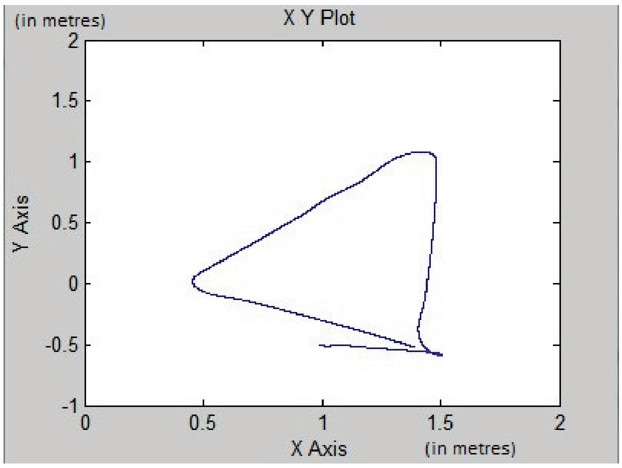

Fig. 15 Hardware Tracking of Q-Bot 2 for tele-operation

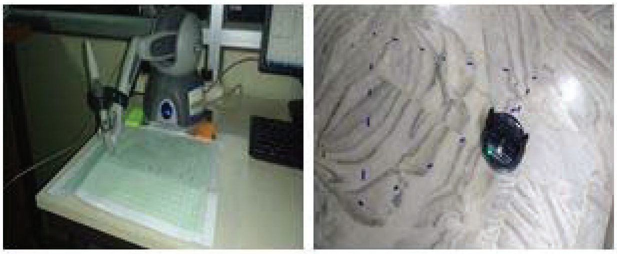

Fig. 16 Hardware setting up for tele-operation

It was observed that both the wheel velocities (See Fig. 17 and Fig. 18) where within limits of + or - 0.6m/s and the control variables (See Fig. 19 and Fig. 20) responded satisfactorily. Though the changes in velocities are not smooth, a better tuning can make the difference. The PID controller outputs V_{COM} and W_{COM}, which are then transformed into Right and Left wheel velocities

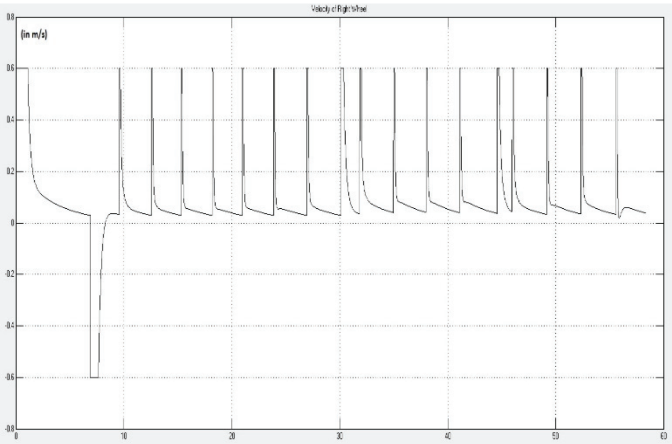

Fig. 17 Velocity of Right Wheel in m/s

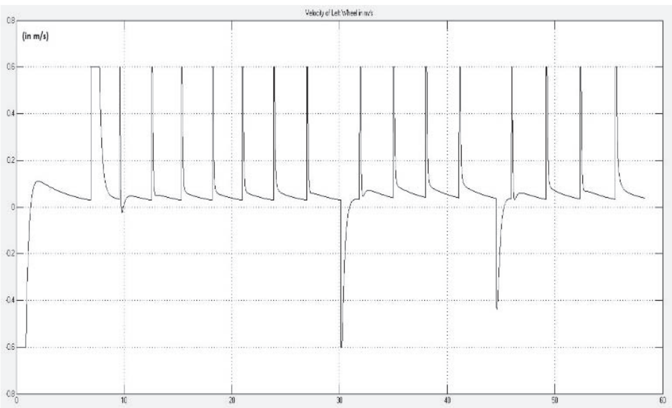

Fig. 18 Velocity of Left Wheel in m/s

CONCLUSION

The tracking results were having close correlation with the simulation results, though in the actual hardware there is error in tracking due to slippage of wheels and friction. Independent trajectory tracking control of Q-Bot 2 was satisfactory with PID, but observed that, the stability of the system was very sensitive to the gains. As a future scope, the changes in velocities can be smoothen to have a better tracking. Control Gains can be improved. Better control strategies can be implemented and a comparative study can be performed.

REFERENCES

[1] Andreas Nygaard "High-Level Control System for Remote Controlled Surgical Robots", Master of Science Thesis, Norwegian University of Science and Technology, 2008

[2] Huanran Wang and Xiaoping P. Liu, "Human-Robot Interaction Via Haptic Device"

[3] Pablo Falcon, Antonio Barreiro, Miguel D. Cacho, Emma Delgado, "Passive tele-operation of mobile robot with input-output linearization and dynamic extension", The 4th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems, Hong Kong, China, 2014

[4] Wei-dong Chen, Qi-guang Zhu and Ying Chen, "Research on the Fuzzy PID-based Control of Haptic System", International Conference on Computer Science and Network Technology, 2011.

[5] Wei-dong Chen, Qi-guang Zhu, Ying Chen "Research on the Fuzzy PID-based Control of Haptic System" International Conference on Computer Science and Network Technology,2010

[6] Gang Song, Shuxiang Guo and Qiang Wang "A Tele-operation system based on haptic feedback" IEEE International Conference on Information Acquisition, Weihai, Shandong, China ,2006.Q-Bot 2 2 – Manual

[7] Gang Song and Shuxiang Guo, "Development of a Novel Tele-rehabilitation System" IEEE International Conference on Robotics and Biomimetics,Kunming, China,2006.

[8] Rajibul Huq, Herv´e Lacheray, Cameron Fulford, Derek Wight, and Jacob Apkarian "Q-Bot 2: An Educational Mobile Robot Controlled In MATLAB Simulink Environment"

[9] Omni Bundle Haptic Device User Manual

[10] Q Bot 2 User Manual

[11] Alejandro Jarillo-Silva, Omar A. Dom´ınguez-Ram´ırez, Vicente Parra-Vega, J. Patricio Ordaz-Oliverz "PHANToM OMNI Haptic Device: Kinematic and Manipulability" Electronics, Robotics and Automotive Mechanics Conference, 2008

[12] Thitipong Sansanayuth, Itthisek Nilkhamhang, Kanokvate Tungpimolrat "Teleoperation with Inverse Dynamics Control for PHANToM Omni Haptic Device" SICE Annual Conference, 2012

[13] Stephen Armah, Sun Yi, Taher Abu-Lebdeh "Implementation of Autonomous Navigation Algorithms On Two-Wheeled Ground Mobile Robot" American Journal of Engineering and Applied Sciences pp.149-164, 2014.