

Modeling and Control of an Autonomous Three Wheeled Mobile Robot with Front Steer

Ayush Pandey, Siddharth Jha and Debashish Chakravarty
 Autonomous Ground Vehicle Research Group
 Indian Institute of Technology (IIT)
 Kharagpur, West Bengal, India - 721302
 Email: (ayushpandey, thesidjway, dc)@iitkgp.ac.in

Abstract—Modeling and control strategies for an autonomous three wheeled mobile robot with front wheel steer and front wheel drive have been presented in this paper. Although, three-wheeled vehicles with front wheel steer are commonly used automotives in public transport (especially in Asian countries), the advantages of such a design in navigation and localization of autonomous vehicles are seldom utilized. We present the system model for velocity control and autonomous trajectory control for such a robotic vehicle. Using the obtained model, the velocity control system has been designed in a digital control framework. We propose a novel trajectory control approach for accurate path following to work with the conventional motion planning high level control algorithm. The derived system model and the controllers have been validated using experimental results obtained for the robot vehicle design. Controller performance and robustness issues have also been discussed briefly.

I. INTRODUCTION

With the rise in research and development of autonomous robots in the past decade, there has been an increased focus on control strategies for the robots to achieve robust and optimal performance. A clear application of the research on autonomous robots is the self-driving car, which has already started to change the commute in many cities all around the world. In the coming years, we are bound to discover more such self-driving vehicles on the roads and not just cars. An example is the research on three-wheeled self-driving trikes [1] with an aerodynamic design which could effectively be deployed in the future for shared public transport. Similar is the design of a passively stabilized bicycle [2]. The designs and advantages of these autonomous vehicles are promising, but their autonomous control and stability are still issues that are unresolved. Mercedes-Benz are working on an electric vehicle [3] with a related mechanical design as well. All of these designs are common in the sense that they are front steered and are equivalent to a three wheel vehicle design. Control and stability are major challenges with such three wheeled vehicles as will be discussed in detail in the next section in detail. [4] and references therein give an account of the study of stability of three wheeled vehicles. This paper focuses on the control aspects for such a vehicle design. There are certain distinct advantages that can be had with a three-wheeled robot design. The front wheel steering design is quite close in working to the design of cars. The localization and navigation of such three wheeled vehicles is completely

different when the drive actuation is also in the front wheel rather than the conventional tricycle robot drive mechanism in the two rear wheels. The robot design considered in this paper is also a front wheel steered and driven type and hence the two rear wheels are free. The two unactuated wheels can be effectively used for accurate localization, which would have been otherwise impossible in a rear wheel driven vehicle. The absence of actuators in the wheels gives way to precise localization which in turn helps in better trajectory following and navigation of the vehicle. Although, the modified mechanical design has advantages in navigation, it poses challenges in modeling and control strategies which haven't been discussed in the existing literature concerned with autonomous robots. This paper aims to bridge this gap by identifying the system model and proposing trajectory and velocity control strategies for a three wheeled mobile robot with front steer and front wheel drive.

A. Background

There has been extensive research on low-level control of autonomous mobile robots ([5], [6]). Low-level control strategies for mobile robots (autonomous or otherwise) are largely dependent on the dynamics of the robot. Most common mobile robots today are based on the differential drive model, where the two actuated wheels are used for driving the robot and changing its direction as well. The research on control of such robot vehicles is vast and is not of concern in this paper. However, it is important to note that the control strategies for a differential drive robot are completely different and do not apply to other robot designs such as omnidirectional mobile robots [7] or Ackermann drive robots [8] which are very similar to modern cars. This paper presents a low-level control model for a new kind of steering geometry, consisting of a three wheeled robot which is both steered and actuated using the front wheel. This type of steering geometry has several advantages (as described later in detail) for the purpose of localization and motion planning. Similar kind of designs have been discussed in [9] and [10] but the work on modeling and control of such robot designs is still in a nascent stage. Moreover, most high-level planning algorithms [11] are applicable only to differential drive robots and not for three-wheeled robot designs. Although, there are a few dynamical simulations available for tricycle-like

robots, such as [12], they do not propose any motion planning algorithm to generate trajectories for a given waypoint. Also, the control approach for such robots would be different to the three-wheeled design considered in this paper as the tricycle-like robot consider rear wheel drive while the three-wheeled robot design considered in this paper is front wheel driven. We propose a novel trajectory control method in this paper which enables differential drive motion planning algorithms to work with the three-wheeled robotic designs with front steer and drive.

II. OBJECTIVES

A CAD model of the robot design considered in this paper is shown below. The front wheel is mounted on a steering column which is controlled by a DC servo motor. The two rear wheels are decoupled mechanically and are free to move. For translation, a brushless DC motor has been used in the front wheel itself. The design allows for a more customizable and simple mechanical design because it gets rid of the two motor couplings that would otherwise exist in a rear wheel driven robot, which is usually the case for tricycle-like or differential drive robots. The downside to using a BLDC motor is the high power requirement, but the mechanical robustness and the advantages of this design in autonomous navigation act as compensatory factors. For the three-wheeled autonomous

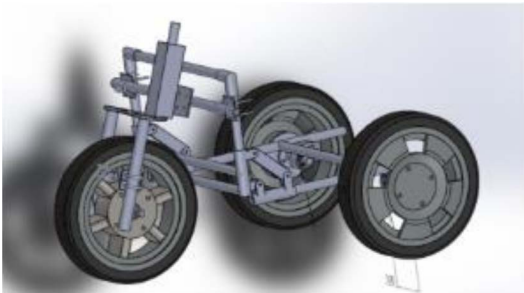


Figure 1: The three wheeled autonomous mobile robot design with front steer

mobile robot shown in Fig.(1), the design for the complete low-level control system has been presented in this paper. There are three main coupled subsystems working in the low-level control of the robot viz. velocity control, steering control and trajectory control. To design appropriate control algorithms, the system model has been identified and validated with experimental results. The trajectory control for this robot design is a challenging task because of the uniqueness of the mechanical design as mentioned earlier. Towards the end of this paper, a trajectory control methodology has been proposed and experimental results for the same as well have been presented.

A. Velocity Control

The front wheel of the robot drives the robot using a brushless DC motor which provides the required thrust. The BLDC

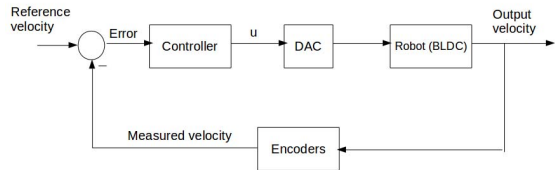


Figure 2: Velocity Control System Block Diagram

is in an outer closed loop control as shown in the velocity control system block diagram in Fig.(2). Approximating the model for the robot in velocity control system by the BLDC motor only, the plant model was identified. A PID controller was designed based on the identified model. The controller implementation and experimental performance analysis have also been considered in the paper.

B. Trajectory and Steering Control

A large class of control problems consist of planning and following a trajectory in the presence of noise and uncertainty [13]. Trajectories become particularly important in autonomous robotics because the target path to be traversed keeps changing dynamically with time. Hence, the trajectory controller for an autonomous robot has to be more robust and dynamic than that for a manually controlled robot [14]. For the robot design considered in this paper, the trajectory control is challenging because there are no high-level planning algorithm implementations that exist for such a design. The trajectory

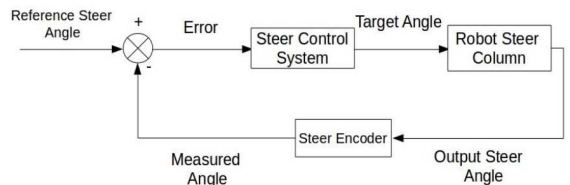


Figure 3: Steering Control System Block Diagram

control interacts with the steering control as shown in Fig.(4). In this work our objective was to design the trajectory control strategy which feeds the steering control loop. The steering control loop has its own controller whose design is also considered in the paper. The steering control block diagram for the robot is shown in Fig.(3).

III. MODEL IDENTIFICATION

Standard system identification techniques were used to identify the model of the three wheeled mobile robot with front steer. For the velocity control system, as mentioned above, it was assumed that the robot dynamics are primarily due to the BLDC motor which is responsible for the translation. For a BLDC motor as shown in [15] and similar other works, a second order model was assumed with unknown parameters. By obtaining a set of input and corresponding output measurements, an instrument variable system identification algorithm was used to obtain the unknown parameters [16].

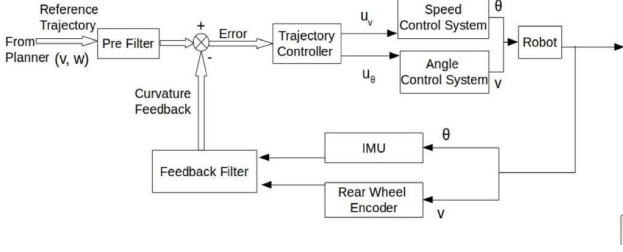


Figure 4: Trajectory Control System Block Diagram

IV. CONTROL DESIGN

For response to a step input in velocity control, a controller was designed based on the model identified. A fast rise time is often the most desirable performance characteristic for any autonomous mobile robot. Other than the high bandwidth requirement, the control design should be such that the closed loop system is insensitive to external disturbances which arise due to undulations in the road terrain and other environmental disturbances. The PID controller designed achieves both the objectives. It has been implemented on a digital microprocessor - the PC. A digital control is not only very easy to implement compared to analog control, but also provides the option to change the reference input and controller parameters easily.

For the second order plant model identified $G(s)$, a discrete-time PID controller $D(z)$ was designed from the given performance specifications. A desired rise time of t_r was for a step input velocity command to the robot and a phase lag to attenuate the disturbances at high frequency were chosen for the design specifications. In the transform frequency domain, the standard controller equations ([17]) were used:

$$D(w) = K_p + \frac{K_i}{w} + K_d w \quad (1)$$

$$D(j\omega_{w_1}) = K_p + \frac{K_i}{j\omega_{w_1}} + K_d j\omega_{w_1} \quad (2)$$

In polar representation

$$D(j\omega_{w_1}) = |D| (\cos(\theta) + j\sin(\theta)) \quad (3)$$

where K_p , K_i and K_d are ideal PID controller constants. At gain crossover frequency, ω_{w_1} , we have

$$|D| = \frac{1}{|G|} \quad (4)$$

where $|D|$ and $|G|$ are magnitude of the controller and the plant at ω_{w_1} . For a given phase lag angle θ , we can use the above equations to write the PID controller parameters as follows

$$K_p = \frac{\cos(\theta)}{|G|} \quad (5)$$

$$K_d \omega_{w_1} - \frac{K_i}{\omega_{w_1}} = \frac{\sin(\theta)}{|G|} \quad (6)$$

Now, using Eq.(5) and Eq.(6), the values of the PID controller parameters were calculated, after choosing the value of one of them depending on the desired performance specifications. This control design methodology was used for the mobile robot shown in Fig.(1) and the results have been given in Section (VII).

V. TRAJECTORY CONTROL

Consider that the three wheeled mobile robot is traversing on a path with a curvature κ . The curvature of the path is defined as the inverse of the instantaneous radius of curvature, centered around a hypothetical center of a circle. The center of curvature is similarly defined as the center of a circle which passes through the path at a given point which has the same tangent and curvature at that point on the path. The curvature for the robot design in consideration in this paper may be calculated as shown below. (Refer Fig.(5)).

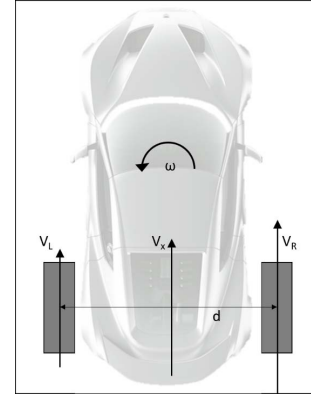


Figure 5: Differential drive geometry

$$V_L = r \times \omega_L \quad (7)$$

$$V_R = r \times \omega_R \quad (8)$$

where r is the radius of a wheel, ω_L is the left wheel angular velocity and ω_R is the right wheel angular velocity.

The rear wheels follow differential drive kinematics, as they are both free to move in both clockwise and anticlockwise directions. The simplistic differential drive model can be used to calculate the kinematic equations of the robot.

$$V_L = V_x - \frac{d}{2} \times \omega \quad (9)$$

$$V_R = V_x + \frac{d}{2} \times \omega \quad (10)$$

where d is the separation between the two wheels, ω is the instantaneous angular velocity of the robot, assumed anticlockwise about a point midway between the wheels. Adding Eq.(9) and Eq.(10), we get

$$V_x = \frac{V_L + V_R}{2} \quad (11)$$

Subtracting Eq.(10) from Eq.(9), we get

$$\omega = \frac{V_L - V_R}{d} \quad (12)$$

A. Curvature estimation

By definition, we can write the curvature as follows

$$\kappa = \frac{1}{R} \quad (13)$$

where κ is the curvature of the path and R is the instantaneous radius of curvature.

Assuming the robot to be a rigid body, we can write

$$\frac{V_L}{R - \frac{d}{2}} = \frac{V_R}{R + \frac{d}{2}} \quad (14)$$

Cross multiplying and solving, we get,

$$\frac{V_R + V_L}{V_R - V_L} = \frac{2R}{d} \quad (15)$$

Rearranging terms, we get

$$\frac{V_L + V_R}{2} \times \frac{1}{\frac{V_R - V_L}{d}} = R \quad (16)$$

Using equations (7) and (8) we can write

$$\kappa = \frac{\omega}{V_x} \quad (17)$$

Using the relation in Eq.(17), the curvature control was designed. It was implemented as a separate PID control loop in the digital controller as shown in the Fig.(4). The steering angle of the robot is controlled using the trajectory controller.

If the robot turns to its left at a constant curvature κ for a

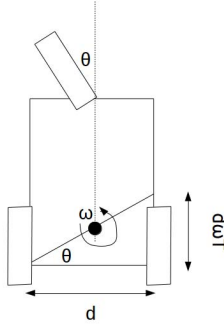


Figure 6: Robot Geometry for Trajectory Control

time T , then analyzing the motion relative to the left wheel (considering it to be stationary), we can write that the right wheel moves a distance of $\omega \times d \times T$, where ω is the instantaneous angular velocity of the robot. From the Fig (6), we can write,

$$\tan(\theta) = \omega \times T \quad (18)$$

Hence, the steer angle is proportional to ω for small values of θ . Using this linearization about small values of θ , we can design the PID controller for the trajectory. Also, since

κ is proportional to ω for a constant velocity, the output of the trajectory controller would be the reference steering input angle to the steering angle control loop.

The controller parameters were experimentally tuned. The same module was also used to provide data for several other tasks not directly linked to low-level control, such as localization and Simultaneous Localization And Mapping (SLAM) for a successful autonomous run.

VI. MODEL VALIDATION

Using the system identification method described in Section (III), the following continuous transfer function model was obtained for the robot velocity control system. It is important to note here that the for the velocity control, the robot translation has been assumed to be affected primarily only by the BLDC motor which provides the required thrust. Also, as mentioned earlier, the BLDC has been assumed to have a second order dynamics. The system identification data was taken about an equilibrium point about which linearization was carried out. Small perturbations about the nominal velocity value were given and the input-output data was collected. Linearization analysis has been presented at a later stage in this paper as well. Although, the robot system model for velocity control was assumed to be dominated by the BLDC motor only, the effects of other links, joints, other wheels, the electronics and the robot body were taken into account while modeling the system. Other than compensating for the gain in the plant model, an important aspect that the robot displayed was a small process delay. This delay was also modeled and we aimed to appropriately compensate for it in the controller design. The transfer function model obtained after incorporating the delay in the system is shown below.

$$G(s) = \frac{K(s + 2.8)}{(s + 0.44)(s + 5)} \times \exp(-0.3s) \quad (19)$$

The identified model was validated using comparisons with the experimental response of the robot to different inputs. The transfer function given in Eq.(19) is when the BLDC motor is running under the load of the whole robot on a road. A similar model was also identified by running the BLDC motor without load. The BLDC model without load was obtained to be consistent with the robot model for velocity control justifying our previous assumption that for velocity control, the system dynamics are majorly governed by the BLDC motor alone, which provides the translation thrust to the robot. To obtain the identified transfer function as shown in Eq.(19), the robot was excited with a step input around a nominal operating point of $1m/s$ speed. A voltage input step command to the BLDC motor was given and the transient output response of the velocity was recorded. For the operating point ($11V, 1m/s$), the linearized model was obtained. ($11V$ is the averaged voltage value when the BLDC is given a 21% duty cycle input to its source voltage of $48V$). In the next section, we demonstrate the validity of the linearity assumption and the range of speeds for which the obtained model is valid.

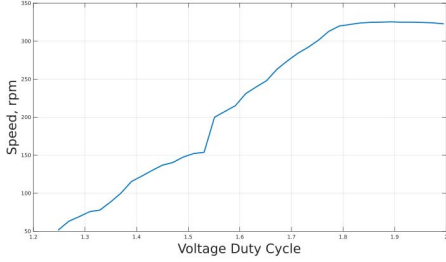


Figure 7: Nonlinear voltage duty cycle and speed characteristics

A. Linearity

For the BLDC motor, the voltage-speed response was obtained experimentally. The resulting motor characteristic is shown in Fig.(7). Clearly, the motor has nonlinear dynamics as the speed saturates after a certain limit. To obtain the linear model given in Eq.(19), linearization was done around the nominal speed of $1m/s$. Since, the model is being used to design the controller which works for the system at different speeds it is important to identify the range for which the robot behaves linearly, which would in effect give the range for which the designed controller would work as expected.

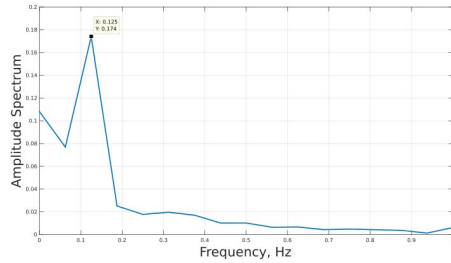


Figure 8: Linearity Validation using Fourier analysis, the maximum frequency component corresponds to 0.125 Hz - the input frequency

To verify the superposition theorem to check the linearity range, the robot was excited with a sinusoidal input signal and the output was recorded. Using Fourier transform, the power density of each frequency component of the output response was obtained. This procedure was repeated for different amplitudes of the sinusoidal input signal around the operating point. From this frequency domain analysis, we observed that the linear model is valid for average voltage amplitudes of up to 28V, i.e. a 10V increment about the nominal operating voltage of 18 V. (These are average voltage values, the source voltage is a constant 48 V, under PWM changing duty cycle). The input signal frequency given was 0.125 Hz. The Fig.(8) shows the frequency component amplitudes in the output response. Clearly, the 0.125 Hz frequency has the maximum power, proving the fact that for this increment the linear model holds. This increment in voltage corresponds to $4m/s$ velocity. Hence, we operate our designed controller in this velocity

range.

B. Open Loop Performance

To validate the robot model obtained for velocity control system, the open loop performance of the robot using the experimental results was compared with the response as calculated from the model. A comparison is shown in Fig.(9),Fig.(10) and Fig.(11) for two step inputs of different amplitudes and a ramp input.

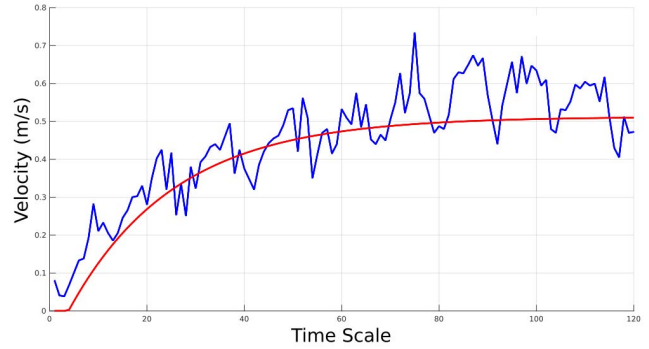


Figure 9: Model validation - Step input of amplitude 5 V (10 % duty cycle) above 11 V operating point. Red - Identified model result, Blue - Experimental result

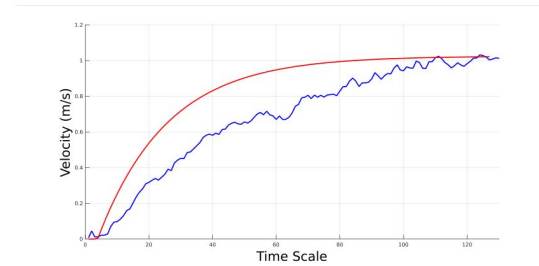


Figure 10: Model validation - Step input of amplitude 10 V (20 % duty cycle) above 11 V operating point. Red - Identified model result, Blue - Experimental result

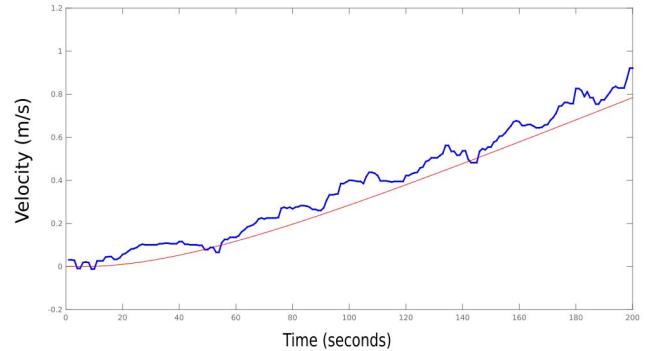


Figure 11: Model validation - Ramp input. Red - Identified model result, Blue - Experimental result

Clearly, the model matches the experimental robot data. On statistical analysis of the error between the model and the experimental results, a mean error of less than 10% was obtained.

VII. CONTROL DESIGN AND IMPLEMENTATION

A PID controller was designed for velocity control based on the identified system model. To achieve a rise time of 0.5 seconds and a phase lag of 5° to attenuate the high frequency noise, we used equations (5 and 6) to calculate the values for the controller parameters. On choosing the value of K_i to achieve the desired lag response, we calculated the values for K_p and K_d using the equations given above. The discrete-time controller can then be written as shown below.

$$D(z) = K_p + \frac{K_i T z + 1}{2 z - 1} + \frac{K_d(z - 1)}{Tz} \quad (20)$$

The discrete time controller equation was obtained by using bilinear transformation from the transform domain to z-domain for the integrator and the backward difference method for the differentiator to incorporate the finite bandwidth differentiator in the controller. The sampling time for the implemented controller was 0.05 seconds. Using the PID parameter values and the sampling time, the controller was implemented on a computer (Intel 64 bit microprocessor) in the discrete-time domain. The control input was given to a DAC which provides the input to the BLDC motor in terms of the duty cycle according to the given control input. For system analysis, the effect of this DAC was incorporated by obtaining a zero-order hold equivalent of the continuous-time plant. The closed loop performance was then analyzed in discrete-time domain. The response of the closed loop system to a step input was similar to the experimental closed loop response. The robot performance is shown in Fig.(12).

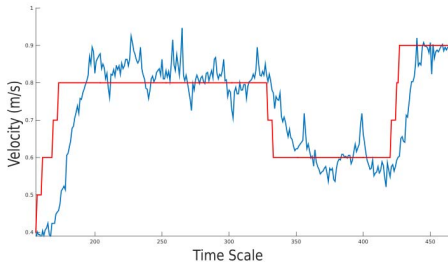


Figure 12: Closed loop performance of the robot using the designed controller

VIII. TRAJECTORY CONTROL AND SOFTWARE IMPLEMENTATION

Apart from velocity control, the trajectory control of the three wheeled autonomous mobile robot with front steer poses newer challenges which are previously uncovered in the existing literature. This was described in detail in Section (V). The trajectory controller was designed for the robot using the method given in curvature estimation section. The

controller was implemented in the structure as shown in Fig.(4) and the performance of the robot was analyzed.

The low level control has been implemented in C++ on a ROS based environment. The discrete-time PID loops run at a constant loop rate of $20Hz$. The command signal is sent using serial communication to a microcontroller (Arduino Due) and a programmable motor driver (Roboteq MDC2230) in order to control the motors (the steering and the BLDC for translation). The microcontroller is also responsible for collection of encoder data for the low level control and also other purposes like odometry which is used in localization of the autonomous robot. The constraint for this particular type of design model is that it cannot turn at an arbitrary angle or rotate in its own position (i.e. a zero radius turn).

Since, most available high level planners are made for differential drive robots that assume that both the aforementioned feats are possible. Planners such as TP-RRT [18] have been implemented for non-holonomic designs like that of a car and may be used. However, the robot design presented in this paper lies between the holonomic differential drive and the non-holonomic car-like design. Hence, there is no existing high-level motion planning implementation that would work with the design in consideration. In order to counter this situation, the high level planner was modified to assume that our robot is differential driven but with added certain constraints for this robot design which is a non-holonomic tricycle with free rear wheels. For example, maximum radius of curvature, maximum angular velocity and a maximum linear velocity were a few constraints that were limited to modify the existing planner implementation. After addition of these and other required constraints, the high level planner decides the best possible path and sends the target linear velocity and angular velocity to the low level control node. The TP-RRT planner is used for this purpose as it works for non-holonomic robot designs such as ours. It is implemented using Mobile Robot Programming Toolkit (MRPT). To make this planner work for the robot design in consideration, a trajectory controller has been implemented as described previously which leads to accurate trajectory following for this design as well (even though the motion planner has not been implemented for this design).

IX. FUTURE WORK

The work on autonomous three wheeled robots has a long way to go before such vehicles are realized onto the roads because of many control and stability related challenges. We described a particular kind of three wheeled robot mechanical design which has various advantages in autonomous navigation and localization, but is difficult to properly control on a trajectory. A novel trajectory control method was proposed for the robot design and the velocity controller was also designed in the low-level control system by identifying the robot translation system model. This work could be carried forward in various directions such as design of controllers using the robust and optimal control theoretic techniques

so that the robot performs under various uncertainties while consuming as less battery power as possible. It would also be interesting to design adaptive PID controllers for which the parameters change according to the environment conditions. Although, there has been a significant amount of research on adaptive and fuzzy PID control designs, but extending such results to this robot design would be an interesting problem to consider given the different challenges that this front wheel driven & steered design poses.

X. CONCLUSION

We identified a linearized model for a three wheeled autonomous mobile robot. The robot design considered in the paper is a front wheel steer design. The model was validated by comparing the derived model response with the experimental results of the autonomous robot. The linearity of the model was also investigated thoroughly and the range of linearity was calculated by analyzing the experimental data. A PID controller was designed based on the identified model and was implemented in a discrete-time controller hardware. The trajectory control problem was touched upon briefly in this paper and some promising initial results were demonstrated. A high level planner designed for holonomic differential drive robots only, when used with the designed trajectory controller for the three wheeled robot design followed the desired trajectory.

ACKNOWLEDGMENT

The authors would like to thank all the members of the Autonomous Ground Vehicle (AGV) research group, IIT Kharagpur for their continuous support in our research work. We would like to extend special gratitude towards Jignesh Sindha (PhD research scholar in the group) for his help with the robot mechanical design. We would also like to thank Gopabandhu Hota, Yash Gaurkar and Aakash Yadav for letting us use their velocity measurement sensor to obtain the experimental results. We are grateful to Sponsored Research and Industrial Consultancy (SRIC), IIT Kharagpur for funding the research in our group.

REFERENCES

- [1] Tyler C. Folsom. "Self-driving Tricycles". In: *University of Washington* (2013).
- [2] Ayush Pandey et al. "Low cost autonomous navigation and control of a mechanically balanced bicycle with dual locomotion mode". In: *Transportation Electrification Conference (ITEC), 2015 IEEE International*. IEEE. 2015, pp. 1–10.
- [3] "Mercedes F 300 - Concept Cars". In: *Mercedes-Benz* (2016).
- [4] Jignesh Sindha, Basab Chakraborty, and Debashish Chakravarty. "Rigid body modeling of three wheel vehicle to determine the dynamic stability - A practical approach". In: *Transportation Electrification Conference (ITEC), 2015 IEEE International*. IEEE. 2015, pp. 1–8.
- [5] Fumio Miyazaki Yutaka Kanayama Yoshihiko Kimura and Tetsuo Noguchi. "A Stable Tracking Control Method for an Autonomous Mobile Robot". In: *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on* (1990).
- [6] Yilin Zhao and S.L. BeMent. "Kinematics, dynamics and control of wheeled mobile robots". In: *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on* (1992).
- [7] K. Watanabe. "Control of an omnidirectional mobile robot". In: *Knowledge-Based Intelligent Electronic Systems, 1998. Proceedings KES '98. 1998 Second International Conference on* (1998).
- [8] Bjorn Astrand and Albert-Jan Baerveldt. "An Agricultural Mobile Robot with Vision-Based Perception for Mechanical Weed Control". In: *Autonomous Robots - Springer* (2002).
- [9] Behnam Dadashzadeh and M.J. Mahjoob. "Fuzzy Logic Control vs. Nonlinear P Control of a Three Wheeled Mobile Robot (TWMR)". In: *Mechatronics and Automation, 2007. ICMA 2007. International Conference on* (2007).
- [10] Brueske Vern Alvin. "Self-propelled, electric, three wheel maintenance cart". In: *US Patent 3572455 A* (1968).
- [11] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, 2006.
- [12] Markus Bader. "Gazebo ROS Tricycle Drive Simulation". In: (2014).
- [13] R M Murray. "Chapter 12, Optimization Based Control". In: (2010).
- [14] Felipe N. Martinsa et al. "An adaptive dynamic controller for autonomous mobile robot trajectory tracking". In: *Control Engineering Practice* (2008).
- [15] Oludayo Oguntoyinbo. "PID Control of Brushless DC Motor and Robot Trajectory Planning and Simulation with MATLAB/Simuling". In: *KTH Royal : PhD Thesis* (2009).
- [16] Torsten Soderstrom and Petre Stoica. "Instrumental variable methods for system identification". In: *Circuits Systems and Signal Processing* (2002).
- [17] Katsuhiko Ogata. *Discrete-time control systems*. Prentice Hall, 2nd Edition, 1987.
- [18] Jose Luis Blanco, Mauro Bellone, and Antonio Gimenez-Fernandez. "TP-Space RRT — Kinematic Path Planning of Non-Holonomic Any-Shape Vehicles". In: *International Journal of Advanced Robotic Systems* (2015).