# Dynamic Path tracking of 6-wheel steering Mobile Robot

Adharsh Mahesh Kumaar[1]

- I thought you are going to compare the paper you read (MPC) with other methods. Is there any reason why you did not do it?
- The first experimental scenario is too simple, and the second one is not a good choice. If you take the data from [11], why does your trajectory look so different with [11]? See [14] for reasonable choices.

***Abstract*—This paper presents different methods used by mobile robots in order to maneuver in off-road environments. Robot stability is a big concern when the robot moves in off-road environments. The stability of the robot should be taken into consideration while performing the path tracking task. In this paper, a waypoint tracking controller is used for path tracking. The longitudinal dynamics are modeled using the tire force distribution algorithm. A Disturbance yaw moment observer is designed for estimating the torque due to the lateral friction force. This control design is then compared with the baseline results in the literature and it is also compared to Stanley controller, another popularly used lateral controller in autonomous vehicles. The main factor which decides the performance of these path tracking controllers is their deviation from the reference trajectory. The efficiency of the path tracking algorithms is tested in MATLAB. Simulation results showed that the trajectory is tracked with an average lateral deviation of 0.148m for the Waypoint tracking controller and 0.06m for the Stanley Controller.**

***Index Terms*—Dynamic Path tracking, Waypoint tracking controller, Stanley Controller.**

## I. Introduction

One of the major components in mobile robotics and autonomous ground vehicles is path tracking. The goal of path tracking is to follow the desired trajectory by controlling the vehicle in longitudinal and lateral motion. Vehicle stability and path tracking accuracy are considered to be a great challenge in this field due to the non-linearity of vehicle dynamics, external disturbances, and road curvature disturbances [1][2]. The ideal controller should take care of rejecting these disturbances and uncertainties by taking the future road information into account [3].

In recent years, extensive research has been conducted in the area of path tracking, and it usually involves optimization or feedforward-feedback control. Previously, tracking controllers were mostly based on feedback control theory and geometric vehicle model due to its stability and simplicity. For example, authors in [4] propose a proportional-derivative-integral (PID) controller architecture to follow the desired trajectory. Similarly, popular tracking controller like Pure-pursuit controller is used as a benchmark in several Autonomous Vehicle DARPA challenges [5]. These methods are simple but not effective since these controllers always calculate errors at one or more preview points, thus unable to capture complete steer dynamics.

[1]First Author is with McGill University, Electrical and Computer Engineering Department, , McGill ID : 260905451 adharsh.maheshkumaar@mail.mcgill.ca

To overcome these limitations, many adaptive methods have been proposed that automatically tune the look-ahead distance depending upon the speed and curvature [6]. In off-road environments, the efficiency of these kinematic controllers can be damaged due to its slippage and vehicle dynamics. Hence, path tracking controllers have started incorporating vehicle dynamic models and predictive strategies. For instance, authors in [7] and [8] design dynamic path tracking controllers by applying nonlinear continuous-time generalized predictive controller (NCGPC) and Linear Quadratic Controller (LQR). However, these controllers do not take into account physical or intrinsic constraints.

To ensure the stability of the mobile robot, we have to consider the lateral and longitudinal dynamics throughout the motion of the robot. So, in this paper, the tire force distribution algorithm is used to compute the amount of force generated at each wheel [10]. The lateral dynamics is represented by the moment due to the friction force ($M_d$). An observer is designed to predict the moment due to friction force [11]. The waypoint tracking algorithm presented in [9] is used to calculate the x, y position of the robot and the velocity v at each waypoint for tracking the trajectory.

Mobile robots that are operated in field environments require high driving performance, steering performance, and stability [12]. Among the available robots, 6-wheel differential drive robots are favorable in off-road environments due to its high mobility on extreme terrains and obstacles. Hence, this paper uses a 6-wheel differential drive robot for the path tracking task. The physical parameters of the 6-wheel robot are taken from [11].

What about [13] and [14]? How do they compare to the others?

This paper is organized as follows. In section II, the dynamic modeling of the 6-wheel robot is highlighted. Section III consists of the different control algorithms used in this paper; (i) Waypoint tracking controller, (ii) Stanley controller. Section IV describes the tire force distribution algorithm used to calculate the amount of longitudinal force is distributed to each wheel. Section V describes the disturbance yaw moment observer designed to observe the moment due to friction, thereby observing the lateral dynamics. Section VI illustrates the experiments performed and the results obtained. It also talks about how the results obtained are compared to the baseline results, and the comparison between the two controllers is also shown. Section VII is dedicated to Conclusion and Future Work. Finally, section VIII is the acknowledgment section.

## II. Dynamic Modeling

This paper incorporates the dynamics of a 6-wheel steering mobile robot. The schematic of the robot is shown in Fig. 1 [12]. The dynamic model of the robot is represented in fixed frame $R_0$ = {O, x, y, z}. The frame attached to the robot is given by $R$ = {G, x, y, z} where G is the Center of gravity of the robot wrt $R_0$. The pose of the robot is given by $[x, y, \phi]^T$ where $[x, y]^T$ is the position of G and $\phi$ is the orientation of $R$ wrt $R_0$.

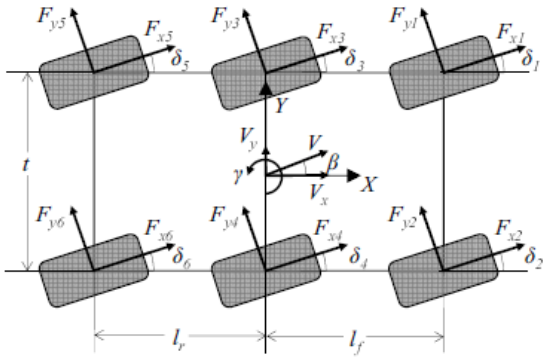


Fig. 1: Schematic of 6-wheel robot

The absolute velocity $[\dot{x}, \dot{y}, \dot{\phi}]$ is transformed to $[u, v, \gamma]$ in the local frame. Their relationship is obtained using transformation matrix given by:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \cos\phi & \sin\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \gamma \end{bmatrix}$$

Here $v_x$ is the component of velocity in x-direction, $v_y$ is the component of velocity in the y-direction in robot frame, and $\gamma$ is the yaw rate. The dynamic model of this system can be represented by the following equations,

$$\dot{x} = v_x \cos(\phi) - v_y \sin(\phi) \tag{1}$$
$$\dot{y} = v_x \cos(\phi) + v_y \sin(\phi) \tag{2}$$
$$m(\dot{v_x} - v_y\gamma) = F_{x1} + F_{x2} + F_{x3} + F_{x4} + F_{x5} + F_{x6} \tag{3}$$
$$m(\dot{v_y} + v_x\gamma) = F_{y1} + F_{y2} + F_{y3} + F_{y4} + F_{y5} + F_{y6} \tag{4}$$
$$\dot{\phi} = \gamma \tag{5}$$
$$I_z\dot{\gamma} = M_z + M_d \tag{6}$$
$$M_z = \frac{t}{2}(F_{x2} + F_{x4} + F_{x6} - F_{x1} - F_{x3} - F_{x5}) \tag{7}$$
$$M_d = l_f(F_{y1} + F_{y2}) - l_r(F_{y5} + F_{y6}) \tag{8}$$

Here m represents the mass of the vehicle, $F_{xi}$ and $F_{yi}$ represents the force in the longitudinal direction, and lateral direction on each wheel i respectively. From Fig. 1 the numbering in the wheels can be observed. $I_z$ represents the moment of inertia of the robot on the z-axis. $t$ represents the tread, which is the distance between the left and right wheels. $M_z$ is called vehicle yaw moment, which is the moment caused due to the longitudinal driving force difference between the tires and $M_d$ represents disturbance moment caused in the tire due to lateral friction force.

## III. Controller Design

The idea used to track the desired trajectory is by using a controller that gives the set of coordinates and velocity to track the trajectory. Since the robot moves in off-road environments, the dynamics of the robot has to be considered at each time step to ensure the stability of the robot. The state of the robot is $\mathbf{z} = [\mathbf{x}, \mathbf{y}, \mathbf{v}_x, \mathbf{v}_y, \Phi, \dot{\Phi}]$, where x, y is the position coordinate of the robot, $v_x, v_y$ is the velocity of the robot in x and y direction and $\phi, \dot{\phi}$ is the yaw and yaw rate respectively. The state-space equations derived from the dynamic model (Equations 1-6) of the robot is given by,

$$\dot{x} = \dot{z(1)} = z(3) * \cos(z(5)) - z(4) * \sin(z(5)) \tag{9}$$
$$\dot{y} = \dot{z(2)} = z(3) * \sin(z(5)) + z(4) * \cos(z(5)) \tag{10}$$
$$\dot{v_x} = \dot{z(3)} = \frac{\sum_{i=1}^{6} F_{xi}}{m} + z(4) * z(6) \tag{11}$$
$$\dot{v_y} = \dot{z(4)} = \frac{\sum_{i=1}^{6} F_{yi}}{m} - z(3) * z(6) \tag{12}$$
$$\dot{\phi} = \gamma = \dot{z(5)} = z(6) \tag{13}$$
$$\dot{\gamma} = \dot{z(6)} = (M_z + M_d)/I_z \tag{14}$$

From equations 1-6, it can be observed that the state-space equations depend on the dynamics of the robot. So, at each time step, the longitudinal dynamics ($F_{xi}$ and $M_z$) and lateral dynamics ($F_{yi}$ and $M_d$) need to be calculated for determining the state of the robot. In this paper, the performance of two controllers is compared (i) Waypoint tracking controller [9] (ii) Stanley controller.

### A. Waypoint Tracking Controller

Generally, the controllers used for tracking try to locate the robot on the $[x_d, y_d, \phi_d]$ coordinate at any given time t where $x_d, y_d, \phi_d$ are the desired location and desired heading angle. These control algorithms can be ineffective in off-road environments because of large disturbances and large tractive forces from the off-road surface. This can make the task of path tracking impossible. So, instead of typical path tracking controllers, this paper uses a forward velocity tracking controller to guide the robot on the desired path. [9]

The waypoints contain the desired [x, y] position coordinate, and the trajectory is generated by connecting these waypoints. The desired velocity and the heading angle is calculated based on the Line of Sight Concept (LOS). At each waypoint, the desired velocity $v_i$ is also assigned. The waypoints can be visualized as in Fig. 2 [9].

The goal is to make the robot follow the trajectory. This is done by finding the point $[x_{los}, y_{los}]$. This concept is called the Line of Sight (LOS). The point $[x_{los}, y_{los}]$, the desired velocity $v_d$ and the desired heading angle $\phi_d$ needed to track the trajectory can be calculated using the following algorithm.

**Algorithm to find** $[x_{los}, y_{los}, v_d, \phi_d]$:
**Step 1**: Find the parameter u

$$u = \frac{(x - x_n)(x_{n+1} - x_n) + (x - y_n)(y_{n+1} - y_n)}{d_n^2}$$

what is the meaning of this parameter u?

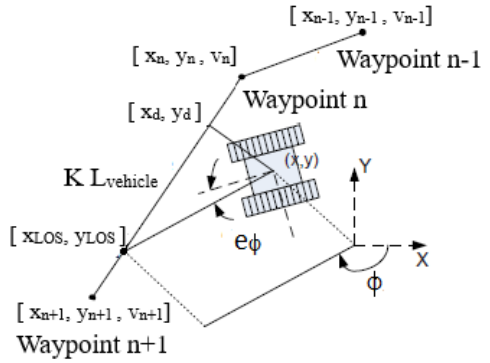


Fig. 2: Visualization of Waypoint tracking controller

figure needs citation

$d_n$ is the distance between waypoint n and waypoint n+1;

**Step 2**: Calculate the point $x_d, y_d$ and $v_d$ using the following equations.

$$x_d = x_n + u(x_{n+1} - x_n)$$

$$y_d = y_n + u(y_{n+1} - y_n)$$

$$v_d = v_n + u(v_{n+1} - v_n)$$

**Step 3**: The lateral error is given by,

$$e_{lateral} = ((x_d - z(1))^2 + (y_d - z(2))^2)^{\frac{1}{2}} sign(y_d - z(2))$$

**Step 4**: The point $x_{los}, y_{los}$ is at a distance of $KL_{vehicle}$ from the point $x_d, y_d$ where K is the LOS constant and $L_{vehicle}$ is the length of the vehicle [11].

$$K = \begin{cases} |v_x|, & \text{for } |v_x| > 1m/s \\ 1, & \text{for } |v_x| \leq 1m/s \end{cases}$$

**Step 5**: After calculating $x_{los}, y_{los}$, the desired heading angle $\phi_d$ can be calculated by,

$$\phi_d = \frac{(y_{los} - y)}{(x_{los} - x)}$$

The velocity error and the heading angle error is given by $e_v$ and $e_\phi$.

$$e_v = v - v_d$$

$$e_\phi = \phi - \phi_d$$

To make the errors zero, the closed loop error dynamics are given by,

$$\dot{e_v} + K_{Pv} e_v + K_{Iv} \int e_v \, dt = 0 \tag{15}$$

$$\ddot{e_\phi} + + K_{D\phi} \dot{e_\phi} + K_{P\phi} e_\phi = 0 \tag{16}$$

$K_{Pv}, K_{Iv}, K_{P\phi}, K_{D\phi}$ are positive control gain values which decide the location of closed-loop poles of each error dynamics equation.
From equations 3 and 6, the total longitudinal force $F_x$ and yaw moment $M_z$ can be calculated using the following equations.

$$\sum F_{xi} = m(K_{Pv} e_v + K_{Iv} \int e_v \, dt) \tag{17}$$

$$M_z = I_z(K_{P\phi} e_\phi + K_{D\phi} \dot{e_\phi}) - M_d \tag{18}$$

Now the total longitudinal force acting on the vehicle is known. So, acceleration in longitudinal direction ($\dot{v_x}$) can be calculated as given in equation 11. This total force has to be distributed to each of the wheels in an optimum way to make sure the force on each wheel is within a limit. This can be done using the Tire force distribution algorithm (Section IV). The method to determine $M_d$ for calculating the vehicle yaw moment $M_z$ (Equation 18) is shown in Section V.

### B. Stanley Controller

Stanley controller is one of the most popularly used lateral controllers in Autonomous Vehicles. This control law was designed originally by Stanford University in the DARPA Grand challenge in 2005. [13] The Stanley control law is given by:

$$\phi_S = \phi + \frac{K e_{lateral}}{v_x} \tag{19}$$

From the above equation, the control law has 2 terms. The first term is to compensate for the angular error in heading angle $\phi$, and the second term is to compensate for the lateral distance error $e_{lateral}$ measured from the center of the front axle to the nearest point in the trajectory. K is a constant. All the above steps of the waypoint control algorithm apply here except step 4 and 5 that is replaced by Equation 19. The formula for vehicle yaw moment $M_z$ is given below.

$$M_z = I_z(K_{P\phi_S} e_{\phi_S} + K_{D\phi_S} \dot{e_{\phi_S}}) - M_d \tag{20}$$

## IV. Tire Force Distribution

The main objective of the Tire Force Distribution algorithm is to calculate how much force is generated at each wheel [10]. The force generated should be within a limit. The force limit $f_l$ is calculated as the ratio of torque limit $\tau_l$ to tire radius $t_r$. Torque limit $\tau_l$ is taken to be 3416Nm and each tire radius ($t_r$) is taken to be 0.547m.

$$f_l = \frac{\tau_l}{t_r} \tag{21}$$

Using the concept of friction circle, the magnitude of force acting on each wheel is proportional to the vertical tire force. Therefore the cost function is given by,

$$J_{cost} = \sum_{i=1}^{6} W_i \frac{X_i^2}{F_{vi}^2} \tag{22}$$

where $X_i$ denotes the amount of force acting on the i-th wheel, $W_i$ denotes the weighting coefficient at the i-th wheel. $W_i$ varies depending on the trajectory but it is taken to be 1 for all the wheels in the experiments conducted. $F_{vi}$ is the vertical force acting on the i-th tire. The vertical force on each wheel is taken to be $(1/6)^{th}$ of the total weight of the vehicle.

The desired longitudinal force ($F_x$) and the yaw moment ($M_z$) from equations 3 and 7.

$$F_x = X_1 + X_2 + X_3 + X_4 + X_5 + X_6 \tag{23}$$

$$M_z = -\frac{t}{2}(X_1 - X_2 + X_3 - X_4 + X_5 - X_6) \tag{24}$$

From Equation 23 and 24, we can calculate $X_5$ and $X_6$

$$X_5 = \frac{F_x}{2} - \frac{M_z}{t} - X_1 - X_3 \tag{25}$$

$$X_6 = \frac{F_x}{2} + \frac{M_z}{t} - X_2 - X_4 \tag{26}$$

Optimum force $X_i$ can be obtained by optimizing the cost function in Equation 22. By using equation 22, 25 and 26 we get,

$$\frac{\partial J_{cost}}{\partial X_1} = 0 = \frac{2W_1}{F_{v1}^2}.X_1 - \frac{2W_5}{F_{v5}^2}\left[\frac{F_x}{2} - \frac{M_z}{t} - X_1 - X_3\right]$$

$$\frac{\partial J_{cost}}{\partial X_2} = 0 = \frac{2W_2}{F_{v2}^2}.X_2 - \frac{2W_6}{F_{v6}^2}\left[\frac{F_x}{2} + \frac{M_z}{t} - X_2 - X_4\right]$$

$$\frac{\partial J_{cost}}{\partial X_3} = 0 = \frac{2W_3}{F_{v3}^2}.X_3 - \frac{2W_5}{F_{v5}^2}\left[\frac{F_x}{2} - \frac{M_z}{t} - X_1 - X_3\right]$$

$$\frac{\partial J_{cost}}{\partial X_2} = 0 = \frac{2W_4}{F_{v4}^2}.X_4 - \frac{2W_6}{F_{v6}^2}\left[\frac{F_x}{2} + \frac{M_z}{t} - X_2 - X_4\right]$$

From the above equations, the optimum value of $X_i$ can be calculated. If any of the generated force $X_i$ is greater than the force limit ($f_l$) calculated in equation 21, then we assign the force to be equal to the force limit value $f_l$.

## V. Disturbance Yaw Moment ($M_d$) Observer

When the robot moves on a curved path, different driving torques are distributed to the left and right wheels. The moment that interferes with yaw is generated in each tire by the lateral friction force. This moment is called Disturbance Yaw Moment ($M_d$). The Disturbance Yaw Moment is also the moment caused due to the lateral friction force $F_y$. $F_y$ can be calculated using cornering stiffness ($C_s$) and side-slip angle $\alpha$.

$$F_{yi} = \begin{cases} -C_s.\alpha_i, & \text{for } \alpha_i < 0.09rad \\ -C_s.0.09, & \text{for } \alpha_i > 0.09rad \end{cases} \tag{27}$$

$F_{yi}$ is the lateral force acting on each wheel i and $\alpha_i$ is the slip angle of each wheel i. The total lateral force is given by $\sum_{i=1}^{6} F_{yi}$ which can be used in state-space equation 12 and the lateral force $F_{yi}$ can be used to calculate the disturbance yaw moment given in equation 8 thereby calculating the Vehicle yaw moment $M_z$ as given in equation 18. But this type of modeling is not always accurate as the above formula for lateral force is valid only for small slip angles ($\alpha_i < 0.09$). The equation can be approximated for slip angles greater than 0.09, as shown above, but it is not always accurate.

Since the lateral force exhibits a non-linear behavior for large slip angles, it cannot be modeled accurately. To overcome this, an observer is designed to estimate the disturbance yaw moment [11].

The equations for observer are as follows,

$$I_z\dot{\hat{\gamma}} = \hat{M}_d + M_c + l\tilde{\gamma} \tag{28}$$

$$\hat{M}_d = \eta\tilde{\gamma} \tag{29}$$

Here $l$ is a constant, $\dot{\hat{\gamma}}$ is the predicted yaw angular velocity, $\hat{M}_d$ is the predicted disturbance yaw moment, $\tilde{\gamma} = z(6) - \hat{\gamma}$ which is the difference between current yaw rate and predicted yaw rate, and $\eta$ is the observer gain required to calculate the predicted disturbance yaw moment $\hat{M}_d$.

## VI. Results and Experiments

### A. Experiment - Constant Velocity

The controller design with tire force distribution algorithm and disturbance yaw moment observer is tested using the parameters of the robot given in [11]. The mass of the robot (m) is taken to be 6762 kg, front and rear wheelbases ($l_f$ and $l_r$) are taken to be 1.8788m, 1.3216m, and yaw inertia moment ($I_z$) to be 13201 kgm/$s^2$. The algorithms are coded in MATLAB. The data set used to generate this trajectory consists of desired position coordinates (x and y) and the desired velocity at each waypoint. The efficiency of these algorithms is tested on 2 simple trajectories (i) Straight line trajectory (ii) Curved trajectory shown in Fig. 3 and Fig. 4.

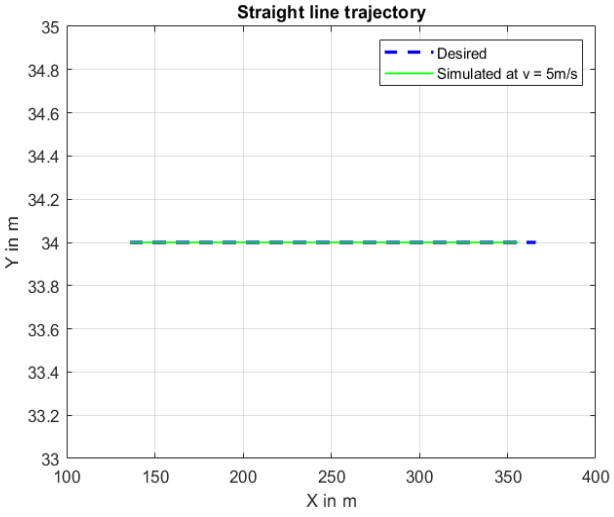


Fig. 3: Straight line trajectory

too simple

Fig. 3 and Fig. 4 represent the tracked trajectory at constant velocity of $V_x = 5m/s$ using the waypoint tracking controller. The most important result in any path tracking problem is the lateral deviation from the desired trajectory. The lateral error and heading angle (yaw) at each time step ($t_s = 0.01s$) till the robot reaches the final waypoint is calculated and plotted in a graph. For the straight-line trajectory, the lateral error and the heading angle was found to be zero as expected. The results

for lateral deviation and heading angle for a curved trajectory is given below in Fig. 5 and Fig. 6.

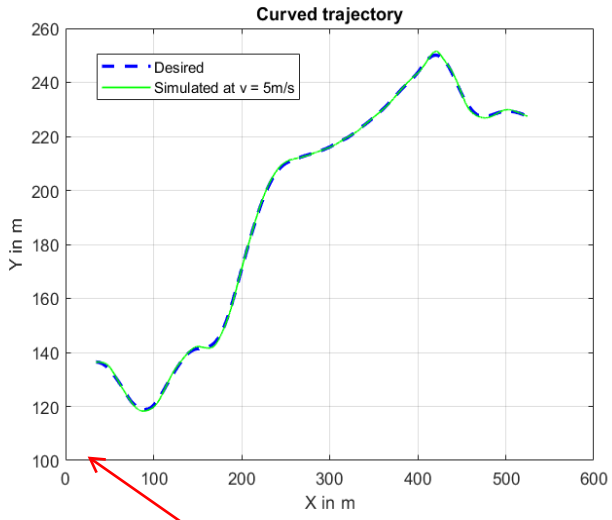


Fig. 4: Curved trajectory

This kind of curved trajectory does not make sense. It looks like a random trajectory rather than some designated ones.

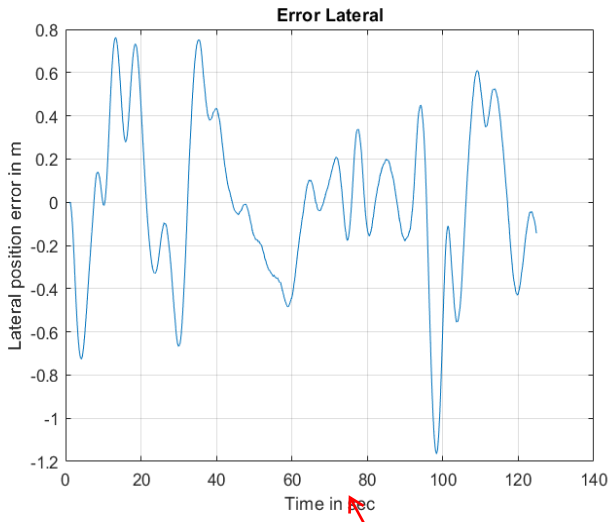


Fig. 5: Lateral Error for curved trajectory at v = 5m/s

The time taken to track the curved trajectory is **125s**. From the graph, it can be observed that the maximum lateral deviation obtained is **1.16m** and the average lateral deviation is **0.29m**. The heading angle exhibits a smooth characteristic without many oscillations, and the obtained values satisfactorily match the desired heading angle. Authors in [14] produced a maximum lateral deviation of **0.3m** when they operated their robot at a constant velocity of 5m/s using their method in simulation. The waypoint controller used in this paper falls short as compared to the Model predictive controller used by the authors in [14].

The result obtained can be improved further by not fixing the velocity to a constant value. In real-time, the robot can't maneuver at a constant velocity in off-road environments. As an example, a robot moves slower on a curved surface than

They did not run on straight line trajectories neither. I understand that you may not be able to re-implement [14], but at least you can choose something similar to compare with

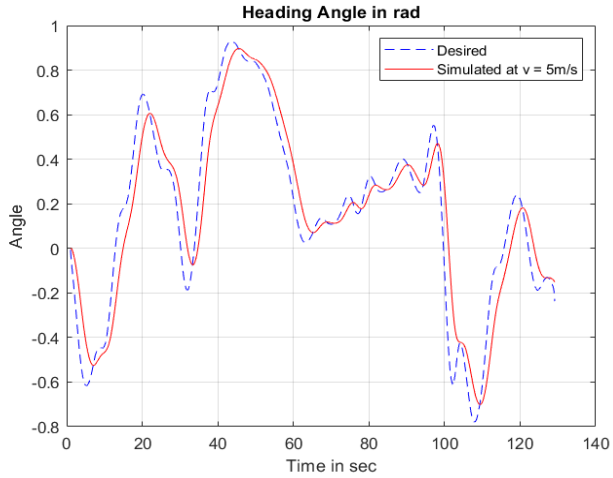


Fig. 6: Heading angle for curved trajectory

on a straight road as the centripetal acceleration is larger on a curved surface. Hence a curved trajectory is difficult to track with the same velocity as in a straight-line trajectory. So, the performance of the path tracking controllers can be improved by varying the velocity based on the curvature of the path. The following section compares the performance of the two controllers discussed in Section III when the robot moves at a variable velocity.

### *B. Experiment - Variable velocity*

The data set for this experiment is modified from the data set provided by the author of [11]. The data set consists of (x, y) position coordinate and velocity at each waypoint. The velocity ranges from 0.3m/s to 6.8m/s. The goal here is to track the desired trajectory with the desired velocity.

*1) Waypoint tracking controller:* The time taken to track the trajectory is **160s**. The velocity, heading angle, and lateral position error at each time step is calculated and is plotted on a graph shown in Fig. 7 and Fig. 8.

From Fig. 7 it can be observed that both velocity and heading angles track the desired values. The heading angle has a smooth characteristic whereas the velocity does not. This is one of the drawbacks of this model. From Fig. 8, it can be observed that the maximum lateral error **0.49m** and the average lateral error was found to be **0.148m**. The maximum and average lateral error is much lesser than in our constant velocity experiment. Fig. 9 represents the input torque given to the forward left and right wheels that is obtained from the tire force distribution algorithm (Section IV) and the distributed yaw moment from the observer model (Section V). It can also be observed that the torque does not exceed the torque limit ($\tau_l = 3416Nm$) at any point in its motion. The torque from the tire force distribution algorithm does not provide a smooth characteristic. This can be a problem when applied in real life. This is another drawback of the model.

Can you explain why you did not make the same comparison with the Stanley controller here?

what are those black lines

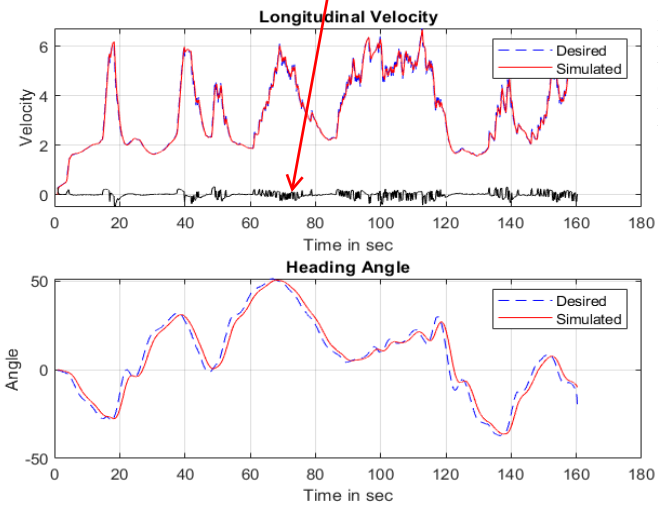


Fig. 7: (i) Velocity (ii) Heading angle

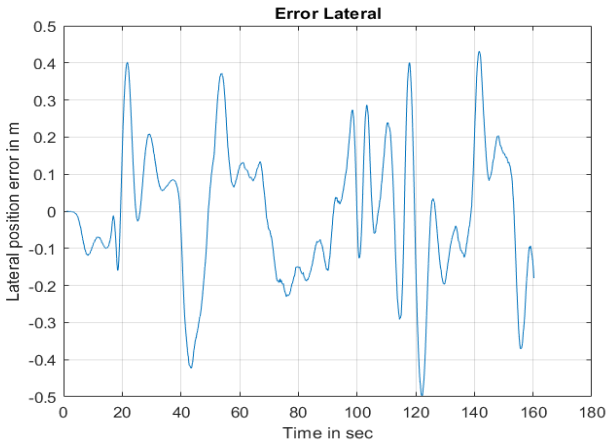


Fig. 8: Lateral Error for curved trajectory at variable speed

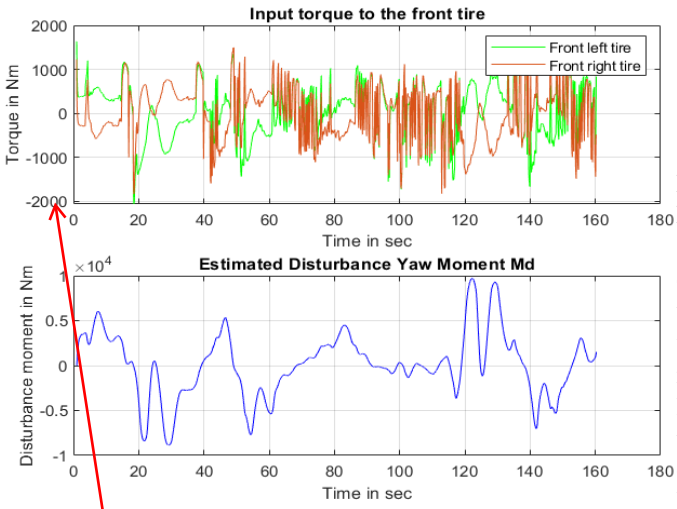


Fig. 9: (i) Torque applied to the front tire (ii) Disturbance Yaw Moment ($M_d$)

Are those values realistic?
What can you conclude from this?

*2) Stanley Controller:* The time taken to track the trajectory is **153s** which is faster than waypoint controller. The velocity, heading angle and lateral position error at each time step is calculated and is plotted on a graph as shown in Fig. 10 and Fig. 11.

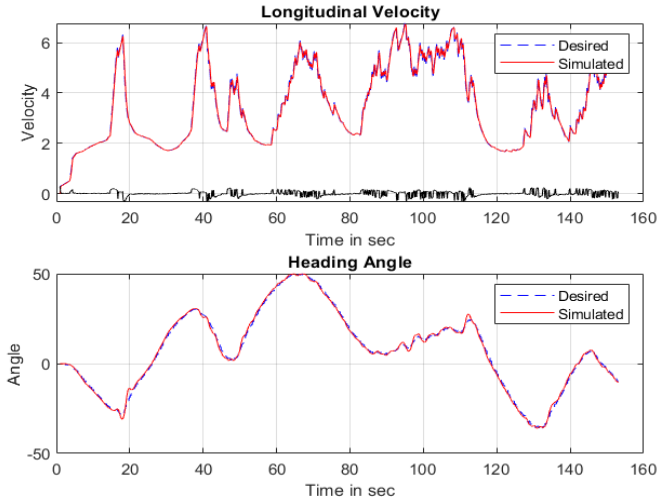


Fig. 10: (i) Velocity (ii) Heading angle

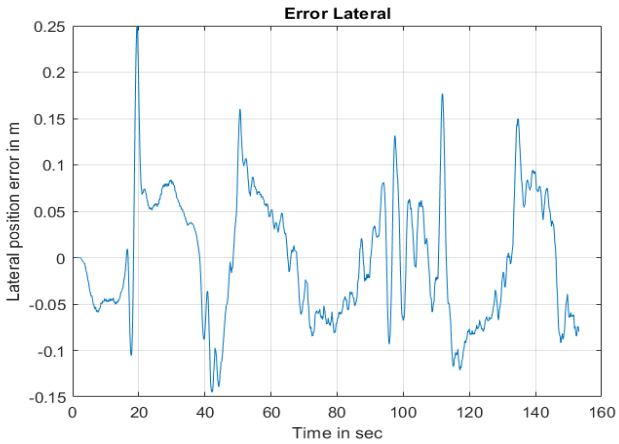


Fig. 11: Lateral Error for curved trajectory

From Fig. 10 it can be observed that both velocity and heading angles track the desired values. The heading angle has a smooth characteristic whereas the velocity does not, as in the case of the waypoint controller. From Fig. 11, it can be observed that the maximum lateral error **0.25m** and the average lateral error was found to be **0.06m** which is very close to 0. The maximum and average lateral error is much lesser than in our constant velocity experiment and less than the waypoint controller. Fig. 12 represents the input torque given to the forward left and right wheels that is obtained from the tire force distribution algorithm (Section IV) and the distributed yaw moment from the observer model (Section V). It can also be observed that the torque does not exceed the torque limit ($\tau_l = 3416Nm$) at any point in its motion. The torque from the tire force distribution algorithm

what about MPC? Should we use MPC instead?

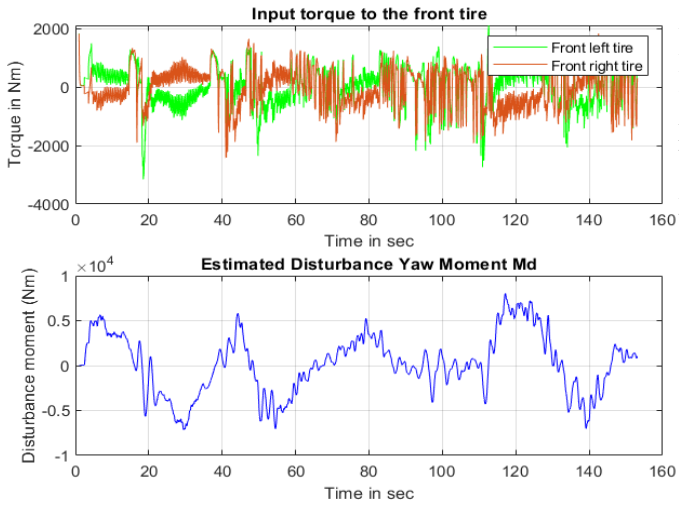


Fig. 12: (i) Torque applied to the front tire (ii) Disturbance Yaw Moment ($M_d$)

| Parameters | Waypoint Controller | Stanley Controller |
|---|---|---|
| Time | 160 s | 153 s |
| Max. lateral error | 0.497 m | 0.25 m |
| Avg. lateral error | 0.148 m | 0.06 m |

TABLE I: Comparison of controllers in experiment 2

does not provide a smooth characteristic. Stanley controller produces more oscillations in velocity, input torque to the tires and disturbance yaw moment but is more accurate than the waypoint tracking controller.

Table I provides a summary of the comparison provided in this section. Time represents the time needed to track the trajectory in s, and Lateral error is the lateral deviation from the trajectory.

## VII. Conclusion and Future Work

This paper presents the dynamic path tracking of a 6-wheel steering mobile robot using 2 different controllers; (i) Waypoint tracking controller (ii) Stanley controller (Section III). Section IV describes the tire force distribution algorithm which is used to distribute the total force to each of the 6-wheels in an optimal way. A cost function is optimized to find the optimum force required by each of the wheels to track the desired trajectory. The force is distributed in such a way that it does not exceed the maximum force limit of the wheel. Section V describes the method used to calculate the lateral force generated at each wheel. Since the method used to calculate lateral force is inaccurate, an observer is designed to estimate the disturbance yaw moment to accurately calculate the moment caused due to the lateral force. The experiments conducted and the results obtained are described in Section VI. Broadly, 2 experiments were conducted. The first experiment was conducted by maintaining a constant velocity of 5m/s for the robot, and the second experiment was conducted by varying the velocity of the robot. The maximum lateral error obtained in the first experiment was 1.16m and that did not produce a satisfactory result compared to our baseline result. The second experiment compared the efficiency of the two controllers described in section III. The results for velocity, lateral error, and input torque were compared for both the controllers. Stanley controller produced a smaller lateral deviation than Waypoint controller but produced more oscillations in velocity and input torque. A smooth characteristic for heading angle is obtained for both the controllers in both the experiments.

Future work focuses on:

- Performing the experiments in real-time on off-road environments to properly evaluate the efficiency of the proposed model of path tracking.
- Decreasing the oscillations in velocity and input torque given to the tires.

## Acknowledgment

I would like to thank Dr.Hyosung Hong, the author of [11] for open sourcing the data set. This data set was modified for performing experiment 2.

## References

[1] N. R. Kapania and J. C. Gerdes, “Design of a feedback-feedforward steering controller for accurate path tracking and stability at the limits of handling,” Vehicle Syst. Dyn., vol. 53, no. 12, pp. 16871704, Dec. 2015.

[2] G. Tagne, R. Talj, and A. Charara, “Design and comparison of robust nonlinear controllers for the lateral dynamics of intelligent vehicles,” IEEE Trans. Intell. Transp. Syst., vol. 17, no. 3, pp. 796809, Mar. 2016.

[3] L. Tang, F. Yan, B. Zou, K. Wang and C. Lv, ”An Improved Kinematic Model Predictive Control for High-Speed Path Tracking of Autonomous Vehicles,” in IEEE Access, vol. 8, pp. 51400-51413, 2020.

[4] R. Marino, S. Scalzi, G. Orlando, and M. Netto, “A nested PID steering control for lane keeping in vision based autonomous vehicles,” in Proc. Amer. Control Conf., 2009, pp. 28852890.

[5] M. Buehler, K. Iagnemma, and S. Singh, The 2005 DARPA Grand Challenge: The Great Robot Race, vol. 36. Berlin, Germany: Springer, 2007.

[6] R. Lenain, B. Thuilot, C. Cariou and P. Martinet. Model Predictive Control for Vehicle Guidance in Presence of Sliding: Application to Farm Vehicles Path Tracking. In IEEE International Conference on Robotics and Automation (ICRA), pp. 885-890, 2005.

[7] M. Krid, F. Ben Amar and R. Lenain. A new explicit dynamic path tracking controller using generalized predictive control. International Journal of Control, Automation and Systems, pp. 303-314, 2017.

[8] M. Fnadi, B. Menkouz, F. Plumet and F. Benamar, Path tracking control for a double steering off-road mobile robot. In : ROMANSY 22-Robot Design, Dynamics and Control. Springer, pp. 441-449, 2019.

[9] Hong, S., Choi, JS., Kim, HW. et al. A path tracking control algorithm for underwater mining vehicles. J Mech Sci Technol 23, 2030–2037 (2009).

[10] Kang, J., Kim, W., Lee, J. et al. Design, implementation, and test of skid steering-based autonomous driving controller for a robotic vehicle with articulated suspension. J Mech Sci Technol 24, 793–800 (2010).

[11] H. Hong, J. B. Han, H. Song, S. Jung, S. S. Kim, W. S. Yoo, et al., ”Path Tracking Control of 6X6 Skid Steering Unmanned Ground Vehicle for Real Time Traversability”, Transactions of the Korean Society of Mechanical Engineers - A, vol. 41, no. 7, pp. 599-605, 2017.

[12] Kim, C., Ashfaq, A.M., Kim, S. et al. Motion Control of a 6WD/6WS wheeled platform with in-wheel motors to improve its maneuverability. Int. J. Control Autom. Syst. 13, 434–442 (2015).

[13] S. Dominguez, A. Ali, G. Garcia and P. Martinet, ”Comparison of lateral controllers for autonomous vehicle: Experimental results,” 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, 2016, pp. 1418-1423.

[14] M. Fnadi, F. Plumet and F. Benamar, ”Model Predictive Control based Dynamic Path Tracking of a Four-Wheel Steering Mobile Robot,” 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019, pp. 4518-4523.