Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnLine 53-1 (2020) 524–529

Vehicle Path Generation and Tracking in Mixed Road Traffic ⋆

Parth Deshpande * **Rushikesh Amrutsamanvar** ** **Shankar C. Subramanian** ***

* *Department of Engineering Design, Indian Institute of Technology Madras, Chennai, India (e-mail: parthrdeshpande@gmail.com).*
** *Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, India (e-mail: rushikesh.amrut@gmail.com)*
*** *Department of Engineering Design, Indian Institute of Technology Madras, Chennai, India (e-mail: shankarram@iitm.ac.in)*

Abstract: Given the condition of mixed road traffic, few models exist that can predict the motion of a vehicle in it. Mixed road traffic can be defined as being both, lane indisciplined and heterogeneous. This study aims at developing a model that can analyze a given traffic condition, generate a safe trajectory and provide a control input to the vehicle to follow it. The paper explains the flow of the model, starting with traffic interaction, leader detection, and waypoint derivation. Post this, the trajectory is generated, the tracking errors are discussed and a controller is designed to navigate the trajectory by minimizing the discussed errors. Since the assumption of low speed is made, a kinematic model is used when generating feasibility criteria for the trajectory. Once the trajectory is determined to be feasible, a closed-loop Proportional Integral Derivative (PID) controller provides steering input to the vehicle to follow the trajectory. The controller tuning is performed using a dynamic bicycle model considering the error with respect to the trajectory. The trajectory generation model and the controller for trajectory following are implemented in independent simulator environments. The resulting output is a collision-free trajectory as followed by the subject vehicle (SV) to meet the generated waypoints which are based on the traffic scenario.

Keywords: Intelligent Transportation Systems, Disordered Traffic, Path Planning, Autonomous Vehicles, Vehicle Control

1. INTRODUCTION

With the advent of driver assistance systems, active safety systems, and even autonomous vehicles, there is a need to accurately predict the motion of vehicles at the microscopic level. The problem of tracking vehicle trajectories is challenging in general and becomes extremely difficult under heterogeneous and lane-less traffic conditions. To track a vehicle path under such traffic conditions, a path choice needs to be made, which can provide a desired direction for the vehicle at every time-step based on the traffic environment. This is followed by the determination of waypoints, which may be anywhere along the road as the traffic is disordered, and hence cannot be modeled using standard maneuvers such as lane-changing. Thus, a path tracking system needs to be developed, which can accurately model and trace a trajectory in disordered traffic.

The process of deciding the desired direction of the vehicle has been worked on using path choice models such as multinomial logit and machine learning in Lee et al. (2009) and Amrutsamanvar and Vanajakshi (2019). These methods train a model using available traffic data to give the probable direction of motion as output. Kanagaraj and Treiber (2018) used a two-dimensional force model to predict microscopic vehicle motion, which also incorporates a social force model and a car-following model. However, in these mentioned models, vehicle motion is predicted without a proper trajectory and without checking the feasibility of the maneuver as vehicle dynamics are not considered. Trajectory generation for vehicle maneuvers such as lane changing has been defined in literature using multiple forms of trajectories. Bangarraju et al. (2016) studied the lateral gaps maintained in lane indisciplined traffic using data. Yang et al. (2018) discussed the various ways in which the lateral motion of a vehicle can be maneuvered. The closest estimation can be achieved using a cubic polynomial trajectory, and thus they used it to define a lane-changing maneuver. The trajectory was decided based on safe gaps between vehicles in the adjacent lane and maneuver parameters such as comfortable acceleration and rollover prevention. Mahapatra and Maurya (2018) studied the variation in vehicle parameters such as acceleration and gaps in mixed traffic using an experimental setup and fitting models to study the interparametric relationships. Marino et al. (2011) utilized a nested PID steering control for lane-center following. As compared to tracking only the deviation error, the dual PIDs resulted in better performance because of tracking the yaw rate. These models for decision making, trajec-

⋆ The authors acknowledge the discussions with Prof. Lelitha Vanajakshi and her suggestions.

2405-8963 © 2020, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2020.06.088

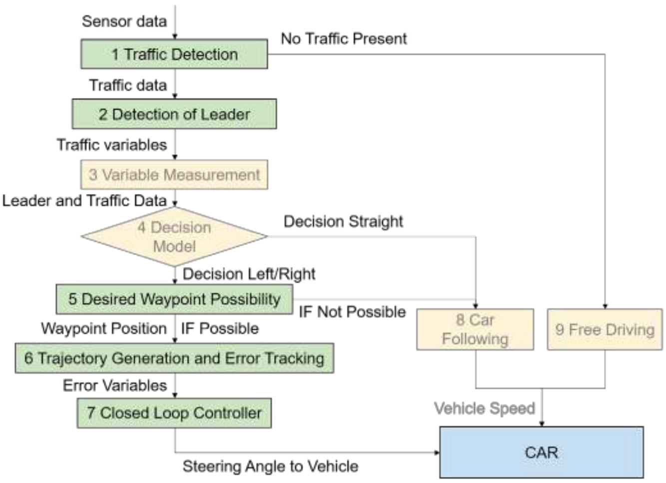

Fig. 1. Overview of the vehicle motion model

tory generation and path following work at different levels have been developed independently. However, they are not specifically designed to work in a traffic scenario consisting of different classes of vehicles without any lane-discipline. There is a need to integrate these models and adapt them to mixed traffic.

This study aims to generate and follow a trajectory for a vehicle in mixed and disordered traffic. The inputs provided to the model are the parameters of vehicles in the environment around the subject vehicle (SV). These include position, speed, and type of vehicle and road parameters such as curb presence and speed limit. The output is to be provided in terms of the steering angle, which is the control input for following the desired trajectory. The current scope of the problem is limited to slow speeds of less than 20 km/h, to be representative of urban roadways. This reduces the dependency of the model on high-speed vehicle dynamics. Further, the heading angle of the vehicle is assumed to be small with respect to the straight road at all times, and the vehicle speed is kept constant during a lateral maneuver. An overview of the proposed model is shown in Fig. 1. The overall process of modeling vehicle motion in traffic involves decision making and both lateral and longitudinal motion modeling. However, as highlighted in Fig. 1, the current study focuses on blocks 1, 2, 5, 6 and 7. Blocks 1 and 2 are universally needed, as any motion planning requires knowledge of the traffic scenario and a leader vehicle, and blocks 5, 6 and 7 are specific to the current study. Blocks 3 and 4 are exclusive to the decision-making process and 8 and 9 pertain to longitudinal motion, and hence they are not considered here.

2. MODEL DEVELOPMENT

2.1 Traffic Interaction

To define lateral maneuvers such as overtaking, a leader vehicle that is to be overtaken has to be defined first. The leader is chosen as the vehicle which may pose a safety risk to the SV and lies in the driver's visible range. The following method is adopted when choosing a leader.

The subject vehicle (SV) is concerned about other vehicles in two different regions in its surroundings, namely the detection region and collision region. The detection region is considered to be less critical than the collision region. The choice of a leader vehicle depends on the position of vehicles in these regions. Both regions are assumed to be rectangular and oriented along the road at every time instant.

The detection region is defined by the visibility region of the rider. The dimensions are calculated using a hazard-perception approach, as shown in Fig. 2.

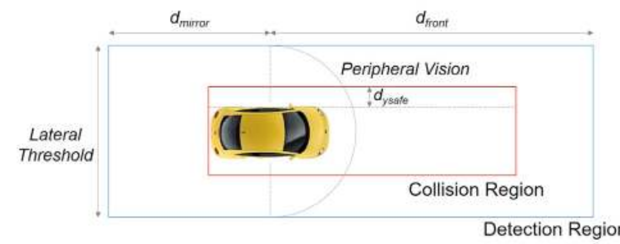

Fig. 2. Interaction region

The front vision distance is calculated to be 11 meters, and the mirror vision distance is considered to be 7.5 meters based on vehicle speed and reaction time (Amrutsamanvar and Vanajakshi (2019)). The lateral threshold depends on the peripheral vision parameters.

The length of the collision region is calculated using the vehicle's stopping distance which is speed-dependent, and the width is calculated considering safe gaps from other vehicles.

2.2 Leader Choice

The leader vehicle is considered as the vehicle of the highest priority among all the vehicles. The decisions of speed and overtaking are taken based on the leader's speed and position.

The choice of a leader is based on the following assumptions:

(1) The SV tries to follow the vehicle that is closest to it inside its detection region.
(2) The SV tries to avoid colliding with vehicles present in its collision region.

Based on these assumptions, considering that there are *'m'* vehicles detected in the detection region, out of which *'n'* lie in the collision region at that time instant. The choice of the leader vehicle for that time instant is done by the following steps:

(1) $n = 0$: This means that there are no vehicles in the collision region. Here, the vehicle whose distance to the SV is the least among all the m vehicles is chosen.
(2) $n > 0$: In this case, the vehicle in the collision region whose speed is the lowest among the n vehicles is chosen, as the slowest vehicle in front of the SV would dictate the headway time and the required minimum stopping distance of the SV.

If there are no vehicles present in the detection region, there would not be any leader and hence the free driving model would be evoked.

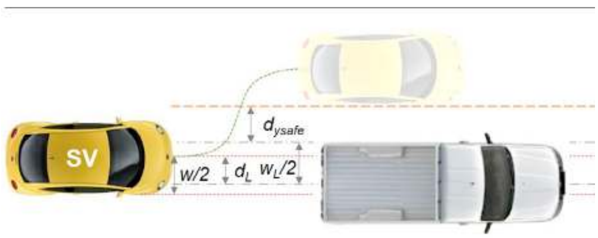

Fig. 3. Desired Lateral Position

2.3 Direction Decision Model

The decision of the direction of motion is based on the environment of the SV. Decision methods typically use multiple environment variables as training data to generate a model that can give the direction of motion (Amrutsamanvar et al. (2019)). One such model is developed by Amrutsamanvar and Vanajakshi (2019), where a logit model is trained on environment data to give the probability of direction of motion. The direction with the highest probability is chosen as the desired direction.

From this known desired direction of the subject vehicle, the present study develops a methodology to decide the exact position in which the vehicle will be at the next time-step. This position is the desired waypoint, which is obtained by applying suitable conditions along the X and Y directions, as discussed below.

2.4 Lateral Waypoint Position

To define the lateral position of the waypoint, it is assumed that the least lateral distance is traveled to maintain a safe lateral distance $d_{y_{safe}}$ when passing the leader vehicle. The SV needs to clear the lateral distance between itself and the leader as well as the safe lateral distance d_{ysafe}. The safe lateral gap can be estimated from data, as demonstrated by Bangarraju et al. (2016).

Here, d_L is the lateral distance between the SV and the leader vehicle, w is the width of the SV and w_L is the width of the leader. The variable dir holds -1 if the desired direction is right and 1 if it is left. The desired position in the Y direction, y_n^f will be given by:

$$y_n^f = dir(d_L + \frac{w_L}{2} + d_{y_{safe}} + \frac{w}{2}). \tag{1}$$

Since the safe gap is defined from the edges of the vehicles, half of the widths of the SV and leader are added in order to clear the leader safely, as seen in Fig. 3.

2.5 Longitudinal Waypoint Position

The desired longitudinal position is used for setting the waypoint as well as checking if the trajectory to that point would be feasible to trace. Current constraints added are based on gaps and the predicted maximum curvature K of the trajectory. The constraints on the curvature typically add a quadratic constraint on the desired X value. Here the minimum and maximum possible values of X are calculated and compared to check the feasibility.

(1) Constraints on trajectory curvature: The curvature of the trajectory is constrained by various parameters pertaining to the dynamics of a vehicle, and these need to be considered to generate a trajectory that would be safe to manoeuver. For this, the maximum values of curvature are generated from the following constraints:

- Lateral traction limit: The centrifugal force on the vehicle should not exceed the lateral force provided by friction, given by μ times the normal reaction. K_{speed} is the maximum allowable curvature to support this, given by:

$$K_{speed} = \frac{\mu_{lat} g}{v^2}. \tag{2}$$

- Vehicle steering angle constraints: The steering angle of the vehicle required to trace the trajectory should not exceed the maximum possible steering angle of the vehicle in the lock position. At low speeds, the steering angle δ required to trace a curvature K tends to be lK. K_{steer} is the maximum allowable curvature considering the steering configuration, given by:

$$K_{steer} = \frac{\delta_{max}}{l}. \tag{3}$$

- Lateral acceleration constraints for comfort: the lateral acceleration required to trace a curve of curvature K is given by v^2K. The lateral acceleration adopted by various drivers would be different and would depend on the driver aggressiveness (DA) parameter. It is assumed to be ranging from 1 m/s^2 for a safe driver (DA=0) and 4 m/s^2 for an aggressive driver (DA=100). If K_{comf} is the curvature with a limit of a_{comf} on the lateral acceleration,

$$K_{comf} = \frac{a_{comf}}{v^2}, \tag{4}$$

where

$$a_{comf} = 1 + 3\frac{DA}{100}. \tag{5}$$

The maximum and minimum lateral acceleration can be estimated using experimental data, as done by Mahapatra and Maurya (2013).

To ensure that all the conditions are met, the maximum possible curvature that can be traced by the vehicle is taken as the minimum of all the calculated K values:

$$K_{max} = min(K_{speed}, K_{steer}, K_{comf}). \tag{6}$$

Once the value of K_{max} is obtained, the condition that the maximum curvature of a cubic polynomial trajectory should be less than K_{max} is applied, where the maximum curvature is given by (Yang et al. (2018)):

$$K_{max} = \left| \frac{2x_n^f tan(\theta_i) - 6y_n^f}{(x_n^f)^2} \right|. \tag{7}$$

Here, θ_i is the heading angle of the vehicle at the initial point. The heading angle at the final point is assumed to be zero. The minimum value of x (x_{fmin}) is calculated from equation (7).

(2) Maximum constraint based on gaps: To avoid a collision, the overtaking should be complete before the longitudinal position of the SV crosses that of the leader vehicle. The time taken for overtaking is $t = \frac{L}{v}$, where L is the length of the trajectory, which is

calculated by integrating d_s over the distance x_{fmax}. d_x is the longitudinal distance between the SV and the leader. The length of the segment is given by:

$$d_s = \sqrt{1 + \left(tan(\theta_i) + x\frac{6y_n^f - 4x_{fmax}tan(\theta_i)}{x_{fmax}^2} + x^2\frac{3x_{fmax}tan(\theta_i) - 6y_n^f}{x_{fmax}^3}\right)^2}. \quad (8)$$

The maximum value of x (x_{fmax}) is calculated using a geometric formulation considering the total length of the trajectory and assuming that the vehicle speed remains constant, as (Yang et al. (2018)):

$$x_{fmax} = d_x + \frac{v_L L_{total}}{v}, \quad (9)$$

where

$$L_{total} = \int_0^{x_{fmax}} d_s dx. \quad (10)$$

The maximum and minimum values of x obtained are compared to check the feasibility of the trajectory and decide the desired point accordingly. If $x_{fmax} > x_{fmin}$, the trajectory is possible and will be executed, else the maneuver will be cancelled and the vehicle will continue straight motion.

3. TRAJECTORY GENERATION AND TRACKING

3.1 Trajectory Equation

Once the desired waypoint is obtained, the model generates a trajectory to reach it. Using a cubic polynomial trajectory (Yang et al. (2018)):

$$y = a_1 x + a_2 x^2 + a_3 x^3, \quad (11)$$

where

$$a_1 = tan(\theta_i),\ a_2 = \frac{3y_n^f - 2x_n^f tan(\theta_i)}{{x_n^f}^2}, \quad a_3 = \frac{x_n^f tan(\theta_i) - 2y_n^f}{{x_n^f}^3}. \quad (12)$$

3.2 Error tracking

To implement a path tracking controller, the lateral offset error must be provided to the control block. The path following problem is modeled as a lane-keeping problem, where the lane center is the desired trajectory. In standard lane-keeping models (Rajamani (2011)), the error variable is defined as the look-ahead offset y_L, which is the predicted lateral deviation at a look-ahead distance (d_s). e_1 is the lateral deviation and e_2 is the heading angle deviation, both at the current vehicle position.

Here, a component of the curvature of the trajectory, K_e, at the point on the trajectory, is considered when finding e_1 and e_2. This results in the following relationship (Deshpande (2017)):

$$y_L = e_1 + d_s tan(e_2) + d_s tan\left(\frac{sin^{-1}(d_s K_e)}{2}\right). \quad (13)$$

The time-step for the decision loop (outer loop) is taken as $t_d = 1$ s because of its computationally intense nature, and for the control loop (inner loop) as $t_s = 0.001$ s for accurate error tracking, as shown in Fig. 4.

To utilize this error variable, the parameters e_1, e_2 and K_e are calculated with respect to the desired trajectory.

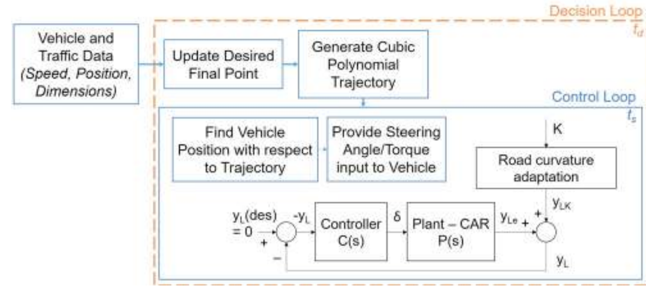

Fig. 4. Time loops and time-steps

(x_c, y_c) is the current position of the SV with respect to the global road coordinate frame with heading angle θ_c. (x_{prev}, y_{prev}) is the position of the SV at the previous outer loop time-step with heading angle θ_i. Then, the position of the SV with respect to the local coordinate system will be:

$$x_{rel} = x_c - x_{prev},\ y_{rel} = y_c - y_{prev}. \quad (14)$$

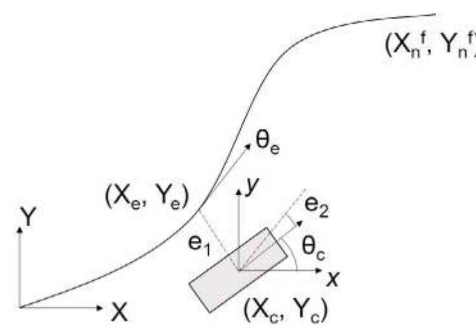

Fig. 5. Trajectory coordinates for error tracking

The first step is to find x_e. Here, x_e lies on the line passing through (x_{rel}, y_{rel}) and perpendicular to the heading of the SV, as well as on the trajectory. The point x_e is calculated by equating the equations of the line and the curve, thus arriving at equation (15):

$$a_3 x^3 + a_2 x^2 + (a_1 + tan(\theta_c))x = y_{rel} + x_{rel} tan(\theta_c). \quad (15)$$

Once x_e is obtained as the desired point, y_e is calculated from equation (11).

The value of the desired heading angle, θ_e, at (x_e, y_e) is calculated using equation (16) which gives the slope of the cubic polynomial curve as (Yang et al. (2018)):

$$\theta_e = tan^{-1}\left(tan(\theta_i) + \frac{(3y_n^f - 2x_n^f tan(\theta_i))x_e}{{x_n^f}^2} + \frac{(x_n^f tan(\theta_i) - 2y_n^f)x_e^2}{{x_n^f}^3}\right). \quad (16)$$

Since e_1 is the lateral deviation between the SV and the trajectory, it can be positive or negative depending on which side of the SV does the trajectory lie. To determine the sign of e_1, an angle θ_{e1} is defined from the SV to the desired point (x_e, y_e) with respect to the X-axis. This angle is given by:

$$\theta_{e1} = atan\left(\frac{y_e}{(x_e - x_{rel} + \frac{y_{rel}}{tan(\theta_c)})}\right). \quad (17)$$

The sign of e_1 is determined by comparing θ_{e1} with the vehicle heading θ_c as shown below:

(1) If $(\theta_{e1} > \theta_c)$:
$e_1 = -\sqrt{(y_{rel} - y_e)^2 + (x_{rel} - x_e)^2}$.
(2) Else:
$e_1 = \sqrt{(y_{rel} - y_e)^2 + (x_{rel} - x_e)^2}$.

The heading angle deviation of the SV with respect to the trajectory is given by:

$$e_2 = \theta_c - \theta_e. \tag{18}$$

4. CONTROLLER SIMULATION AND TESTING

4.1 Control Variables

A closed-loop control system is used for steering control, which inputs the steering angle as the control input and tracks the look-ahead offset as the error. The additional curvature term is added after the error is obtained from the plant, which can then be said as the disturbance to the closed-loop system. P and PI controllers are used for converting the error to the control input.

4.2 System Modelling

For tuning the P and PI controllers before implementation, a lower order model is used. The dynamic bicycle model in terms of error with respect to the road as given in Rajamani (2011) is used here.

The state vector is given as:

$$\mathbf{x} = \begin{bmatrix} e_1 \\ \dot{e_1} \\ e_2 \\ \dot{e_2} \end{bmatrix}, \tag{19}$$

and the state equations are:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}\delta, \\ y_L &= \mathbf{c} \cdot \mathbf{x}. \end{aligned} \tag{20}$$

The matrices $\mathbf{A}$, $\mathbf{b}$ and $\mathbf{c}$ can be seen in Rajamani (2011). The state-space model from equation (20) is converted into a transfer function between the steering angle and the error output. The vehicle parameters used are for for a test car IPG CarMaker, as the controller is being tuned for implementation in CarMaker. The data is shown in Appendix A. These parameters are substituted into the transfer function obtained to get:

$$P(s) = \frac{2496.1s^2 + 21570s + 18230}{s^4 + 83.21s^3 + 1721s^2 - 1.042s10^{-13}}. \tag{21}$$

4.3 Controller Tuning using Root Locus

In order to tune the controller, the performance criteria are set as a 10% peak overshoot and 1 s settling time. These criteria, when plotted with the root locus of the system, are shown in Fig. 6.

To tune a PI controller, the assumption made is that $\frac{K_i}{K_p} = \beta$, where β is a very small value (=0.01), and K_p and K_i are the proportional and integral gains respectively.

Then, the system transfer function is modified as:

$$P_{PI}(s) = \left(1 + \frac{\beta}{s}\right) P(s). \tag{22}$$

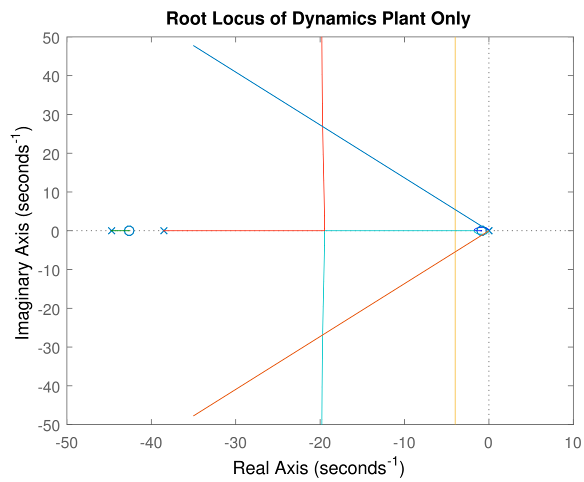

Fig. 6. Root locus of transfer function from δ to y_L

The root locus of this transfer function when plotted with the performance criteria gives $K_p = 2.23$.
Thus, $K_i = \beta K_p = 0.0223$.

5. RESULTS

5.1 MATLAB® simulation results (without controller)

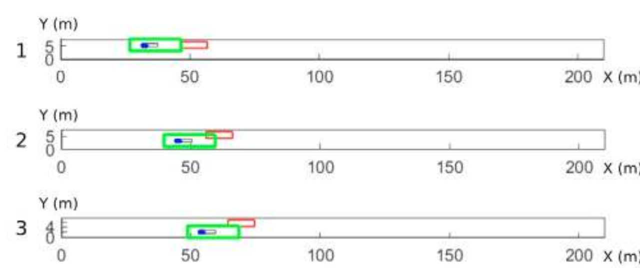

Fig. 7. Step-wise overtaking manoeuvre

The trajectory generation module is tested out by developing a simulator environment in MATLAB®. The test model includes the blocks starting from obtaining the desired direction to the generation of the trajectory. The execution is modeled without a controller or plant, by directly calculating the position of the SV at the next time-step considering its speed and assuming that the trajectory is exactly followed. The scenario tested includes one vehicle, which is detected as the leader and is then overtaken.

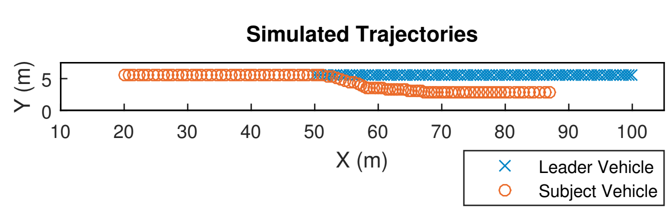

Fig. 8. Overall trajectories (MATLAB® simulation)

The step-wise process is shown in Fig. 7. The SV is depicted in blue, the detection region is shown by the green box around it, and the collision region is shown by the smaller black area in front of the SV. In step 1, the SV is seen closing up to the vehicle in front, and then once the vehicle lies in the detection region, it is chosen as the leader because of being the diagonally closest vehicle. The trajectory feasibility is then checked and the trajectory is generated. An intermediate time-stamp during the maneuver is shown in step 2. In step 3, since the vehicle is outside the detection region, there is no leader and hence the lateral maneuver is complete, post which the SV continues straight motion. The overall trajectories are shown in Fig. 8.

5.2 Controller implementation on IPG CarMaker

The controller is tested independently in IPG CarMaker with the vehicle as a plant. This tested model involves blocks from generating a trajectory to the execution. For testing, the waypoints are defined along the lane centers, and the result is shown in Fig. 9. The red line shows the desired waypoints and the blue line is the followed trajectory. The waypoints are along $Y = -1.5$ m till $X = 80$ m and thereafter along $Y = 1.5$ m.

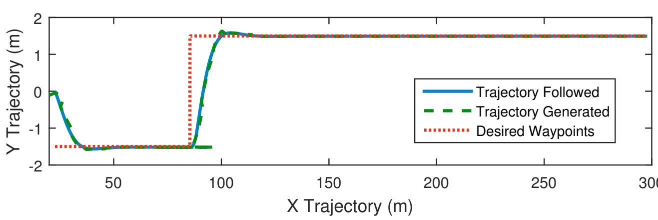

Fig. 9. Trajectories obtained in CarMaker

6. CONCLUSION AND FUTURE TASKS

A model for generating a safe trajectory considering the nearby traffic and vehicle dynamics, and following the trajectory using a steering controller has been developed. The algorithm for leader detection is developed specifically for disordered traffic, as the leader may be present anywhere and may not always be the vehicle right in front of the SV. Unlike existing methods, the waypoints are decided based on the feasibility of the trajectory first. Finally, a standard lane change model is modified to trace a cubic polynomial-based trajectory, and the PID controller is tuned using the root locus method. The model has been tested in two phases – trajectory generation and trajectory tracking, with an overlap in the type of the trajectory. The model covers the gaps in the existing literature by integrating vehicle and traffic models that have been otherwise independently developed.

The future scope of the study includes adopting the model for high-speed maneuvers, where tire dynamics play an important role. For vehicle control, a nested PID system as seen in Marino et al. (2011) can be used as well depending on the application. Once the end-to-end model is developed in a simulation environment, the developed model can be installed onto multiple vehicles using agent-based modeling, so that a disordered traffic scenario is developed. This can be used for testing advanced driver assistance systems (ADAS) or autonomous vehicle functionality.

REFERENCES

Amrutsamanvar, R.B., Muthurajan, B.R., and Vanajakshi, L.D. (2019). Extraction and analysis of microscopic traffic data in disordered heterogeneous traffic conditions. *Transportation Letters*. doi: 10.1080/19427867.2019.1695563.

Amrutsamanvar, R. and Vanajakshi, L. (2019). Modeling path choice behavior of powered-two-wheelers in disordered heterogeneous traffic. In *Transportation Research Board, National Research Council, Washington, D. C.*

Bangarraju, V., Ravishankar, K., and Mathew, T.V. (2016). Analysis of lateral distance keeping behaviour in mixed traffic conditions with little lane discipline. *International Journal for Traffic and Transport Engineering (IJTTE)*, 6, 431–443.

Deshpande, P. (2017). Lane keeping and lane changing system for passenger car. Technical report, The Automotive Research Association of India (ARAI).

Kanagaraj, V. and Treiber, M. (2018). Self-driven particle model for mixed traffic and other disordered flows. *Physica A: Statistical Mechanics and its Applications*, 509, 1 – 11.

Lee, T.C., Polak, J., and Bell, M. (2009). New approach to modeling mixed traffic containing motorcycles in urban areas. *Transportation Res. Record*, 2140, 195–205.

Mahapatra, G. and Maurya, A. (2013). Study of vehicles lateral movement in non-lane discipline traffic stream on a straight road. volume 104. doi: 10.1016/j.sbspro.2013.11.128.

Mahapatra, G. and Maurya, A.K. (2018). Dynamic parameters of vehicles under heterogeneous traffic stream with non-lane discipline: An experimental study. *Journal of Traffic and Transportation Engineering (English Edition)*, 5(5), 386 – 405.

Marino, R., Scalzi, S., and Netto, M. (2011). Nested pid steering control for lane keeping in autonomous vehicles. *Control Engineering Practice*, 19(12), 1459 – 1467.

Rajamani, R. (2011). *Vehicle Dynamics and Control*. Mechanical Engineering Series. Springer US.

Yang, D., Zheng, S., Wen, C., Jin, P.J., and Ran, B. (2018). A dynamic lane-changing trajectory planning model for automated vehicles. *Transportation Research Part C: Emerging Technologies*, 95, 228 – 247.

Appendix A. VEHICLE PARAMETERS FOR CONTROL IMPLEMENTATION

The parameters used for tuning the controller are shown in Table A.1. Since the model here is tested in IPG CarMaker, the data is taken from a model car present in the same.

Symbol	Description	Value
d_s	Look-ahead distance	5 m
v	Speed of SV	20 km/h
l	Wheelbase of the SV	2.888 m
a	Distance of CoG from front axle	1.313 m
b	Distance of CoG from rear axle	1.575 m
m	Mass of SV	1564 kg
I_z	Yaw moment of inertia of SV	2800.246 kgm^2
C_f	Front tyre cornering stiffness	83130.4 N/rad
C_r	Rear tyre cornering stiffness	83130.4 N/rad

Table A.1. Parameters used