

RESEARCH ARTICLE

Nonlinear Model Predictive Control for Autonomous Vehicle Overtaking with Bezier Curve Path

Hamzah Nur Azzam¹, Ari Santoso*, Yurid Eka Nugraha¹ Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

*Corresponding author (santoso@ee.its.ac.id).

Received: 4 June 2025 ; Revised: 26 July 2025 ; Accepted: 26 July 2025

Abstract

Rapid advancements in autonomous vehicle technology have greatly enhanced driving safety; however, developing reliable overtaking strategies to avoid collisions remains a key challenge. This study presents an algorithmic framework combining Nonlinear Model Predictive Control (NMPC) and real-time Bezier trajectory using LiDAR feedback to enable safe and dynamic overtaking maneuvers. The NMPC processes real-time velocity and steering angle inputs, while the Bezier curves serve as references for the pure pursuit controller to guide the vehicle during overtaking and smoothly return it to its original path. Simulations conducted in the Robot Operating System (ROS) and Gazebo validate the system's performance under dynamic conditions. The results demonstrate significant improvements in control smoothness, with Root Mean Square Error (RMSE) reduced from 0.27 m/s to 0.084 m/s for linear velocity, and from 1.72 rad/s to 0.085 rad/s for steering angle. The overtaking maneuver was successfully and safely executed, maintaining an average clearance distance of 2.05 meters from the vehicle center to the nearest obstacle.

Keywords: autonomous car, bezier curve, NMPC, overtaking, simulation

I. INTRODUCTION

Autonomous vehicles have become a primary focus in automotive and robotics research, driven by their potential to enhance road safety, reduce traffic congestion, and improve transportation efficiency. Among the complex tasks in autonomous driving, overtaking maneuvers present a unique challenge, requiring precise control, real-time decision-making, and accurate prediction of other vehicles' behavior [1],[2],[3],[4]. As the adoption of autonomous vehicles grows, developing reliable control algorithms for overtaking maneuvers has become increasingly critical.

Recent advancements in autonomous vehicle research have concentrated on path planning, obstacle avoidance, and vehicle control. Nonlinear Model Predictive Control (NMPC) has gained attention for its ability to handle nonlinear dynamics and operational constraints, making it well-suited for complex maneuvers like overtaking [5]. While some studies have explored NMPC in autonomous racing scenarios, its application to overtaking maneuvers in simulated environments such as ROS and Gazebo, with static and dynamic obstacles, remains underexplored [6],[7].

Key challenges in autonomous vehicle control for overtaking include predicting the behavior of dynamic obstacles, planning safe trajectories, and ensuring compliance with vehicle dynamics constraints (e.g., maximum speed and steering angle). NMPC offers a promising solution by optimizing control inputs based on future predictions while accounting for constraints such as safe distances and acceleration limits [8]. However, the high computational

demands of NMPC pose challenges for real-time applications [9].

This research aims to develop and evaluate an NMPC controller for overtaking maneuvers in autonomous vehicles within a ROS Noetic and Gazebo simulation environment. Reference trajectories are generated using Bezier curves to ensure smooth and feasible paths [3],[6]. The NMPC controller's performance is assessed based on path-tracking accuracy, adaptability to changing trajectories, and effective obstacle avoidance.

II. METHODS

This section describes the mathematical model of the autonomous vehicle, the formulation of the NMPC controller, path planning using Bezier curves, and the simulation setup in ROS Noetic and Gazebo. In this subsection, all research methods and theoretical foundations are expected to be systematically captured

Autonomous Car Model

The autonomous vehicle is modeled using a kinematic bicycle approach, a method commonly employed in autonomous vehicle research due to its simplicity and computational efficiency (Figure 1) [7]. This approach describes the vehicle's motion based on the geometric relationship between position, orientation, and control inputs, without considering physical forces such as inertia or tire friction. The kinematic model was selected in this study to enable fast state predictions in Nonlinear Model Predictive Control (NMPC) and is well-suited for simulating overtaking

maneuvers in a controlled environment using ROS Noetic and Gazebo [8]. To provide broader context, this kinematic model will also be compared with a dynamic model approach, which accounts for physical forces and vehicle dynamics [9].

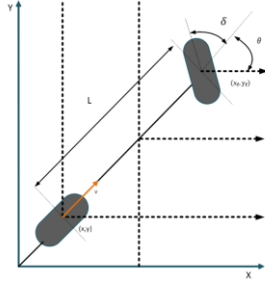


Figure 1 Bicycle Kinematic Model

Based on Figure 1, the kinematics of the autonomous vehicle are represented using the bicycle model. According to this model, the kinematic equations can be formulated as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ \frac{v}{l} \tan \delta \end{bmatrix} \quad (1)$$

This kinematic model represents the vehicle's velocity in relation to the wheel configuration. Here, l denotes the wheelbase length, δ represents the vehicle's orientation angle, and θ indicates the steering angle corresponding to the rotational input of the front wheels [4].

More realistic and representative approach, which also serves as a development of the kinematic model, a dynamic vehicle model is adopted, as illustrated in Figure 2.

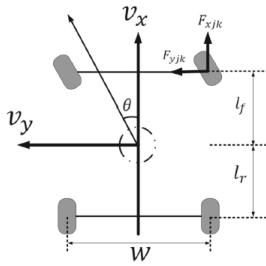


Figure 2 Model Dynamic Autonomous Car

his model is formulated as expressed in the following equation:

$$\dot{x} = v \cos(\theta + \alpha) \quad (2)$$

$$\dot{y} = v \sin(\theta + \alpha) \quad (3)$$

$$\dot{\theta} = r \quad (4)$$

α represents the slip angle, θ denotes the vehicle's heading orientation, and r is the yaw rate angle. To determine the yaw rate angle, several equations must be defined, including the slip angle as follows:

$$\alpha_f = \delta \quad (5)$$

$$\alpha_r = 0 \quad (6)$$

α_f represents the front tire slip angle, and α_r denotes the rear tire slip angle. Since the motion of the rear tire is influenced by the steering of the front wheel (as the rear wheels are fixed), the rear steering angle is assumed to be zero.

Based on the force interactions illustrated in Figure 2, the forces acting on the tires of an autonomous vehicle can be modeled using Pacejka's Magic Formula. This model incorporates trigonometric functions, slip ratios, and other parameters. It aims to provide a more interactive and realistic representation of tire behavior suitable for simulation purposes:

$$F_{yf} = \mu F_z \sin(C \tan^{-1}(B \alpha_f)) \quad (7)$$

$$F_{yr} = \mu F_z \sin(C \tan^{-1}(B \alpha_r)) \quad (8)$$

The lateral force acting along the vehicle's y -axis is influenced by the coefficient of friction (μ), the normal load on each tire (F_z), and the Pacejka parameters B and C . In the longitudinal direction (x -axis), the autonomous vehicle is assumed to experience negligible tire friction, resulting in $F_{xf} = F_{xr} = 0$. When these lateral forces are combined with the vehicle body dynamics and other related components, a comprehensive dynamic equation can be formulated as follows:

$$F_{xf}^{sum} = 2(F_{xf} \cos(\delta) - F_{yf} \sin(\delta)) \quad (9)$$

$$F_{xr}^{sum} = 2 F_{xr} \quad (10)$$

$$F_{yf}^{sum} = 2 (F_{xf} \sin(\delta) - F_{yf} \cos(\delta)) \quad (11)$$

$$F_{yr}^{sum} = 2 F_{yr} \quad (12)$$

Following the derivation of force components in the Cartesian coordinate system along the x and y axes, the comprehensive formulation of the vehicle's yaw rate dynamics can be expressed as follows:

$$\dot{r} = \frac{(\frac{l_f}{2})(F_{xf}^{sum} + F_{xr}^{sum}) + (\frac{l_r}{2})(F_{yf}^{sum} - F_{yr}^{sum}) - l_r \cdot F_{yf}^{sum} + l_f \cdot (F_{yf}^{sum} - F_{xf}^{sum} \cdot \sin(\delta))}{J_z} \quad (13)$$

$\dot{r}$ denotes the yaw acceleration, while l_f and l_r represent the distances from the vehicle's center of gravity (CoG) to the front and rear axles, respectively, as illustrated in Figure 2. Furthermore, J_z denotes the moment of inertia about the vertical (z) axis.

This study adopts parameters from a standard passenger vehicle (Figure 3), which reflects the type of vehicle commonly used in daily driving scenarios. This approach is intended to enhance the model's applicability and realism. The vehicle's schematic representation is shown in Figure 3, while its dimensional parameters are detailed in Table 1.



Figure 3 Autonomous Car Model Simulation

TABLE 1. CAR PARAMETER

Parameters	Symbol	Value
Car dimension (width)	w_c	2.1592 m
Car dimension (Length)	L_c	4.8632 m
Car dimension (height)	H_c	1.2947 m
Car mass	$Mass_c$	~1500 kg
Wheel (radius)	W_{rad}	0.3465 m
Wheel (width)	W_{wth}	0.2363 m
Wheel mass	W_{mass}	~30 kg

Lidar sensor position in each side of the car with $FoV = 180^\circ$, Frequency = 10Hz, vertical laser (channel) = 64. With lidar positioning. In this case, the lidar used is the Velodyne 32 type, which is placed as in Figure 4, where each side will provide feedback as an input signal to the controller. Table 2 represented the coordinate position based on car parameter



Figure 4 LiDAR Velodyne Position

TABLE 2. LIDAR POSITION SIMULATION

Lidar	X	Y	Z
Lidar 1	0.45	0.0	1.2947+0.16
Lidar 2	-0.2	0.4	1.2947+0.16
Lidar 3	-0.2	-0.4	1.2947+0.16
Lidar 4	-0.9	0.0	1.2947+0.16

Lidar data will be evaluated by the controller which controls the steering and wheel of the car, therefore the joint and link

(Figure 7). Diagram interconnection in Figure 7 mentions that sensor link with base link. All joint depending with each actuator that affected one to the others. Revolute mention that they had limitation and to action and continuous containing parameter freedom. the origin car as shown as Table 3 below.

TABLE 3. CAR LINK AND JOINT PARAMETER

Name	Type	Parent	Origin (x,y,z)
Front left steering	Revolute	Base_link	(1.5;0.75;0.3)
Front right steering	Revolute	Base_link	(1.5; -0.75;0.3)
Front left wheel joint	Continuous	Front left steering link	(0;0;0)
Front right wheel joint	Continuous	Front Right steering link	(0;0.5;0)
Rear left wheel joint	Continuous	Base_link	(-1.5;0.25;0.3)
Rear right wheel joint	Continuous	Base_link	(-1.5; -0.25;0.3)

Nonlinear Model Predictive Control (NMPC)

Nonlinear Model Predictive Control (NMPC) is an optimal control method that predicts the future states of a vehicle over a specific time horizon and optimizes control inputs to minimize a cost function while adhering to constraints [10]. In this study, NMPC is employed to regulate the linear velocity v and steering angle δ of an autonomous vehicle, enabling it to follow a reference trajectory generated by Bezier curves while avoiding static and dynamic obstacles. [11] NMPC ensures safe and efficient overtaking maneuvers by accounting for vehicle dynamics, the reference trajectory, and environmental data obtained from LiDAR sensors [12],[13]. Therefore, variable state concludes the real number, $x \in \mathbb{R}^4$:

$$X = [x, y, \theta, r]^T \quad (14)$$

The parameters x and y represent the global position with respect to the world frame. The vehicle's orientation is denoted by the yaw angle θ , and its yaw rate is denoted by r . Based on these equations, the vehicle has two control input signals.

$$u = [v, \delta]^T \quad (15)$$

where v denotes the longitudinal velocity, and δ represents the steering angle of the front wheels. Utilizing the mathematical model formulation of the autonomous vehicle, a filter is applied to the reference input. In this study, a low-pass filter is employed for the reference input, as described in the following equation:

$$v_{ref} = \alpha v_{cmd} + (1 - \alpha) v_{prev} \quad (16)$$

$$\delta_{ref} = \alpha \delta_{cmd} + (1 - \alpha) \delta_{prev}, \quad \alpha = 0.1 \quad (17)$$

Initial state X_0 . State current condition state denotes v_{cmd} , and δ_{cmd} . Predictive state denoted v_{prev} and δ_{prev} . Reference control input is evaluated to generate the control input $[v_{ref,k}, \delta_{ref,k}]^T$ $K = 0..N$. prediction that denotes by K is memory horizon that declare as in Table 4. Horizon defected to performance autonomous car that in this research is declared based on manual tuning.

The NMPC can be formulated within a cost function as expressed in the following equation:

$$J = \sum_{k=0}^{N-1} [Q_v(V_k - V_{ref,k})^2 + Q_f(\delta_k - \delta_{ref,k})^2 + R_v(V_R - V_{k-1})^2 + R\delta(\delta_k - \delta_{(k-1)})^2] \quad (18)$$

The constraints in the above equation include limitations on vehicle speed and steering angle, with the following bounds:

$$V_{min} \leq V_k \leq V_{max} \quad (19)$$

$$\delta_{min} \leq \delta_k \leq \delta_{max} \quad (20)$$

Horizon prediction limitation:

$$-\dot{v}_{max} \leq v_k - v_{k-1} \leq \dot{v}_{max} \quad (21)$$

$$-\dot{\delta}_{max} \leq \delta_k - \delta_{k-1} \leq \dot{\delta}_{max} \quad (22)$$

The discrete NMPC model is obtained as presented in the following formulation.

$$X_{k+1} = X_k + \Delta_t f(X_k, U_k) \quad (23)$$

Equation above denotes constraint limitation for input and horizon. The Parameter that supports control NMPC is declared as Table 4.

TABLE 4. NMPC PARAMETER

Parameters	Symbol	Value
Frequency program	f	10Hz
Time sampling	T_s	0.1
Prediction horizon (N)	N	15
Weight velocity tracking (Q_v)	Q_v	0.5
Weight steering angle tracking (R_v)	R_v	50
Weight steering angle tracking (R_δ)	R_δ	50
Minimum velocity (m/s)	v_{min}	0
Maximum velocity (m/s)	v_{max}	5
Steering angle minimum	δ_{min}	-0.5
Steering angle maximum	δ_{max}	0.5

Bezier Curve Path

Path planning is a critical step in autonomous vehicle simulation to ensure safe and efficient overtaking maneuvers. In this study, the reference trajectory for Nonlinear Model Predictive Control (NMPC) is generated using a quartic Bezier curve, a method selected for its ability to produce smooth, continuous, and easily controllable paths through control points [14]. Bezier curves enable the creation of trajectories that comply with vehicle dynamics constraints, such as the minimum turning radius, and support obstacle avoidance by ensuring smooth transitions during overtaking maneuvers [3]. The generated reference trajectory, consisting of position points (x_{ref}, y_{ref}) and orientation (θ_{ref}) ,

serves as the primary input for NMPC, which optimizes the velocity v and steering angle δ to enable the vehicle to follow the path (Figure 5) while avoiding static and dynamic obstacles. Based model mathematic Bezier curve as follows:

$$B(t) = (1-t)^3 P_0 + 3(1-t)^2 t P_1 + 3(1-t)t^2 P_2 + t^3 P_3, \quad t \in [0,1] \quad (24)$$

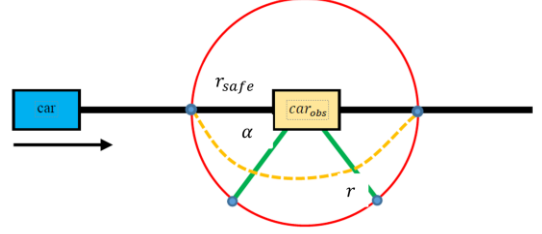


Figure 5 Bezier Path Illustration

The Bezier curve is defined by several control points denoted as P_n . In this study, the Bezier curve serves as an input for the NMPC to determine the route, with the central point of the Bezier curve representing the target obstacle [15]. In this method, P_0 is the current position. Thus, the lateral and longitudinal distances to the obstacle are calculated as follows.

$$\begin{aligned} \text{distance } x &= obs_x - ego_x \\ \text{distance } y &= |obs_y - ego_y| \end{aligned} \quad (25)$$

This distance determines the steering angle of the vehicle to adjust its orientation. Based on this constraint, once the trajectory is generated with coordinate $\begin{bmatrix} x_d \\ y_d \end{bmatrix}$, the result represents the orientation angle along the tangent of the Bezier curve. The tangent of the Bezier curve is obtained by deriving the equation into the following form:

$$B'(t) = 3(1-t)^2(P_1 - P_0) + 6(1-t)t(P_2 - P_1) + 3t^2(P_3 - P_2) \quad (26)$$

Based on its derivative, the representation with respect to the x-axis and y-axis becomes:

$$\frac{dy}{dx} = \frac{3(1-t)^2(y_1 - y_0) + 6(1-t)t(y_2 - y_1) + 3t^2(y_3 - y_2)}{3(1-t)^2(x_1 - x_0) + 6(1-t)t(x_2 - x_1) + 3t^2(x_3 - x_2)} \quad (27)$$

From the equation above, the gradient of the tangent θ_d can be obtained using the arctangent function, resulting in a directional output in form $[x_d \ y_d \ \theta_d]^T$.

In this scenario, the ego car is the controlled vehicle intending to overtake. Therefore, the dynamic safe radius and dynamic angle based in the obstacle vehicle, therefore are calculated as follows [16].

$$r_{safe(dyn)} = r_{safe} Clip \left(\frac{10}{\left(\frac{x_{dist}}{r_{safe}} \right)}, 0.3, 1.0 \right) \quad (28)$$

$$\alpha_{dyn} = \alpha_{max} Clip \left(1.0 - dist \frac{y}{2r_{safe}}, 0.2, 1.0 \right) \quad (29)$$

To avoid collisions, an offset is applied, particularly in the lateral direction, resulting in the following offset [17].

$$y_{off} = \min(r_{safe(dyn)} \sin(\alpha_{dyn}), 0.5r_{safe} \left(r - \frac{dist y}{2r_{safe}} \right) \quad (30)$$

After determining the safe distance, safe angle, and lateral offset, the calculation of points on the Bezier curve is performed as follows:

$$\begin{aligned} P_1 &= \begin{bmatrix} x_0 + r_{safe(dyn)} \cos(\alpha_{dyn}) \\ y_0 + avoid_{dir} y_{off} \end{bmatrix} \\ P_2 &= \begin{bmatrix} x_{obs} + r_{safe(dyn)} \cos(\alpha_{dyn}) \\ x_{obs} + r_{safe(dyn)} \cos(\alpha_{dyn}) \end{bmatrix} \\ P_3 &= \begin{bmatrix} \min(x_{obs} + 3r_{safe(dyn)}, x_{goal}) \\ y_{goal} + (\text{sign}(y_{goal} - y_{obs}) 0.3 y_{offset}) \end{bmatrix} \end{aligned} \quad (31)$$

The clip function is then applied to impose limits.

$$\text{clip}(a, \min, \max) \quad (32)$$

Sign function

$$\begin{aligned} \text{sign}(x) &= 1, \text{ if } x > 0 \\ \text{sign}(x) &= 0, \text{ if } x = 0 \\ \text{sign}(x) &= -1, \text{ if } x < 0 \end{aligned} \quad (33)$$

TABLE 5. BEZIER PARAMETER

Parameters	Symbol	Value
Safe distance	r_{safe}	5 m
Angle maximum	α	45°
Avoid direction	$Avoid_{dir}$	4.8632 m
Alignment threshold	$Align_{trh}$	0.5 m

Simulation

Simulation is a critical step in the development and testing of autonomous vehicles, enabling the evaluation of control system performance without the risks associated with physical testing [18]. In this study, simulations are conducted using Robot Operating System (ROS) Noetic and Gazebo to test the overtaking maneuver strategy of an autonomous vehicle employing Nonlinear Model Predictive Control (NMPC). ROS Noetic, the final version of ROS 1, provides a modular framework for robotics system development, while Gazebo, a 3D physics simulator, supports realistic visualization and simulation of environmental dynamics [19]. This section details the simulation setup, environment configuration, simulation scenarios, and integration with the NMPC method and Bezier curve path planning [20]. The block diagram in Figure 6 illustrates the control process of the autonomous vehicle, which receives feedback from the LiDAR sensor and utilizes NMPC as the main controller, with Pure Pursuit employed for path following along the Bezier trajectory. Figure 7 describes the communication between sensors and actuators as implemented in the simulation.

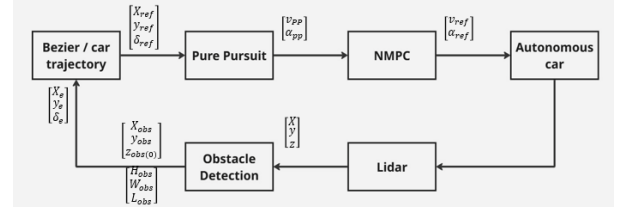


Figure 6 Block Diagram Overview

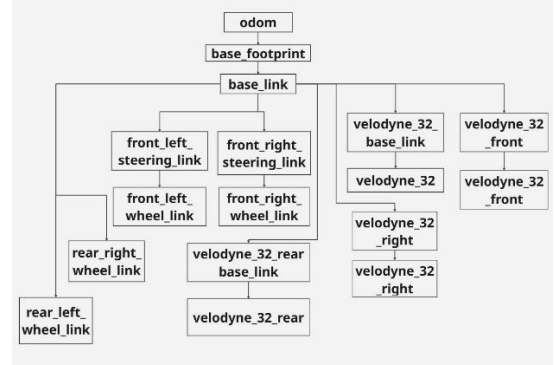


Figure 7. Topology Simulation ROS and Gazebo Link

III. RESULTS AND DISCUSSION

This section will discuss the algorithm, implementation, and simulation results related to autonomous cars for overtaking maneuvers. To test robustness, the evaluation was conducted using a dynamically moving obstacle at a constant speed of 10 m/s.

Maneuver Overtaking Algorithm

Based on the implementation of the model, parameters, and methods discussed in the previous subsection, the decision algorithm for performing overtaking maneuvers is obtained as shown in Figure 3 below.

The challenge here is to develop a dynamic Bezier curve that adapts to the conditions of obstacles as input for the Nonlinear Model Predictive Control (NMPC). The algorithm's flowchart explains that two decision computations are required when detecting an obstacle. The first decision occurs when the obstacle is in close proximity and the second when the obstacle is being overtaken. These two conditions directly influence the Bezier curve.

- When the obstacle is near, the control points $P_0 \dots P_n$ of the Bezier curve are adjusted. If the obstacle is not yet close, the path is extended linearly, maintaining a straight trajectory.
- During the overtaking process, the Bezier curve waypoints are stored in the prediction horizon for computational purposes. When the positions of the ego vehicle and the obstacle align appropriately, the system reuses the plan from the NMPC history. Once the obstacle has been overtaken, the system reverts to the last path without modifying the curve's trajectory.

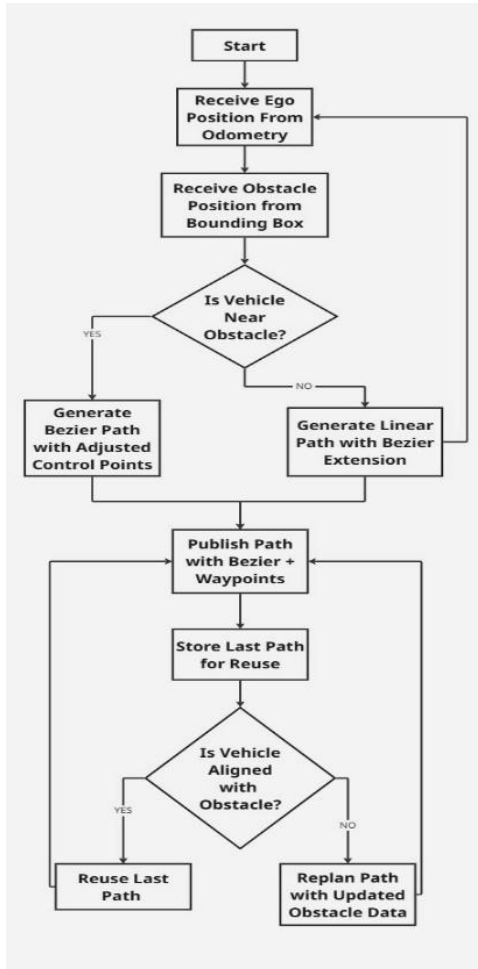


Figure 8 Overtaking Algorithm with Bezier Curve

Overtaking Result

The overtaking results, based on the plotted route trajectory, are illustrated in Figure 4 below.

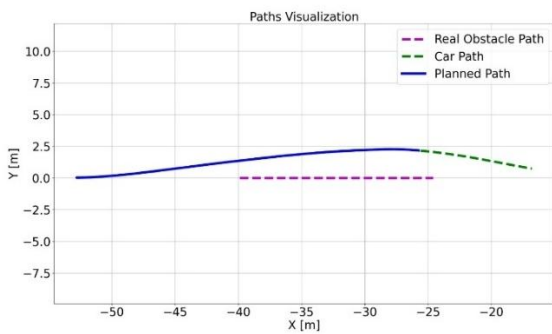


Figure 9 Overtaking Dynamic Obstacle

In this overtaking simulation, the green curve represents a path that continuously adapts due to the dynamic movement of the obstacle. Once the ego vehicle is aligned with or surpasses the obstacle by approximately 0.5 meters (offset), the Bezier curve is closed to return to the original route.

The inputs for the Nonlinear Model Predictive Control (NMPC) consist of velocity and steering angle for the overtaking process. Consequently, feedback from the LiDAR sensor provides information on route changes, triggering

regeneration of the Bezier curve. By incorporating reference data from the Bezier curve as inputs for velocity and steering angle, the resulting reference velocity and steering angle are obtained to generate the NMPC algorithm. The reference velocity and steering angle from the pure pursuit method are depicted in Figure 5.

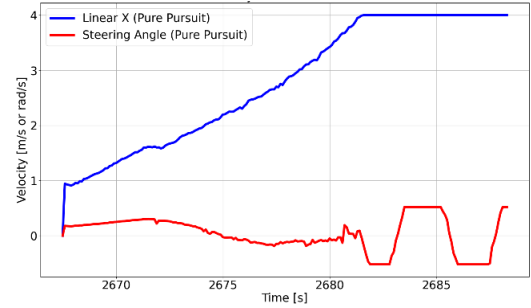


Figure 10 Velocity and Angular Steering Reference

This refinement of the reference inputs yields the output reference velocity and steering angle for the NMPC (used for the vehicle's actuators), as shown in Figure 6 below.

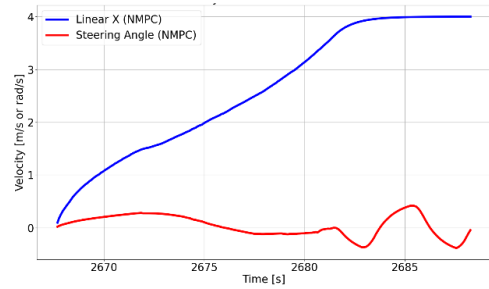


Figure 11 NMPC velocity and angular steering result for car

Evaluate the performance of the proposed NMPC based controller, a comparative analysis is performed between the system with NMPC implementation and a baseline without NMPC. The evaluation considers both linear and angular velocity profiles. As depicted in Figure 12, the NMPC provides improved linear velocity tracking, while Figure 13 demonstrates enhanced angular response, indicating smoother and more stable maneuver execution. Based on the comparison, a significant reduction in the Root Mean Square Error (RMSE) is observed. The linear velocity RMSE decreases from 0.27 m/s to 0.084 m/s , while the RMSE of the steering angle reduces from 0.72 rad/s to 0.085 rad/s , indicating improved control accuracy and trajectory tracking performance.

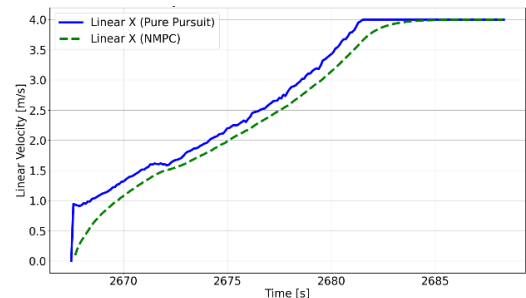


Figure 12 Comparison Velocity Performance

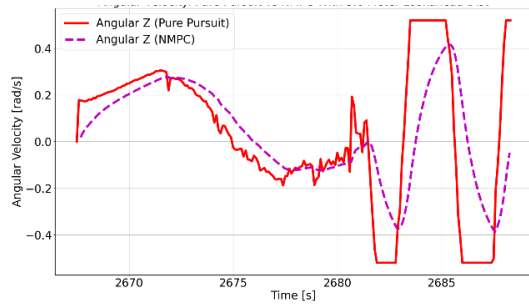


Figure 13 Comparison Angular, Steering Angle

The overtaking path illustrated in Figure 9 demonstrates the successful performance of the autonomous vehicle in executing an overtaking maneuver around a dynamic obstacle. This process is critical, particularly regarding the overtaking distance, to ensure collision avoidance. A detailed analysis was conducted to identify key central data points representing the nearest and farthest positions during the overtaking maneuver. Figure 14 presents the average distance from the vehicle's center to the nearest point of the obstacle is 2.05 M, while Table 7 provides the transformed LiDAR readings showing the lateral distance between the vehicle's sides and the closest obstacle edge.

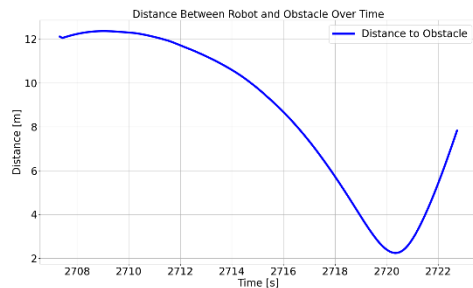


Figure 14 Distance from Center of Car to Nearest Obstacle

TABLE 6. DISTANCE DATA AUTONOMOUS TO OBSTACLE

Distance Position	Nearest Distance (m)
Front to rear obstacle	2,12
Left to right obstacle	1,81
Rear to front obstacle	2,01
Left to rear edge obstacle	1,92
Left to front edge obstacle	1,83

Simulation Visualization

The results obtained from the previous simulations were further implemented and visualized within the Robot Operating System (ROS) and Gazebo environments, providing a more realistic and representative depiction of autonomous vehicle behavior under dynamic conditions. This implementation not only validated the proposed algorithm in near real-world scenarios but also facilitated the identification of algorithmic discrepancies and performance limitations. The simulation environment allowed for the observation of sensor responses, control actions, and vehicle dynamics in detail, which in turn supported iterative improvements and refinements of the control strategy. Figures 15-18 illustrate the

outcomes of the ROS based simulations, showcasing the system's ability to adapt to complex overtaking scenarios while maintaining safe and stable performance.

- Initial Position

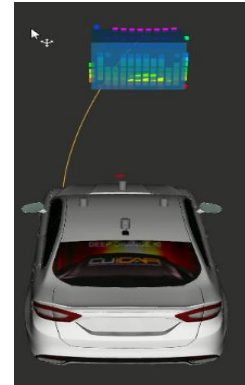


Figure 15 ROS Initial Identification Obstacle

- Overtaking process

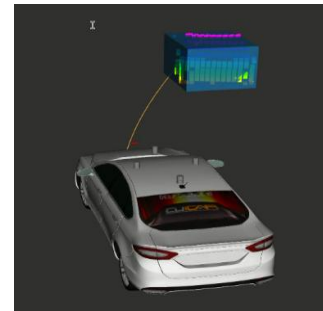


Figure 16 ROS Overtaking Process

- Alignment with obstacle

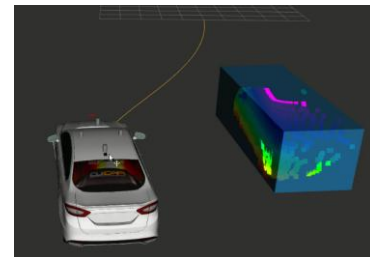


Figure 17 ROS Alignment Process

- Return to route

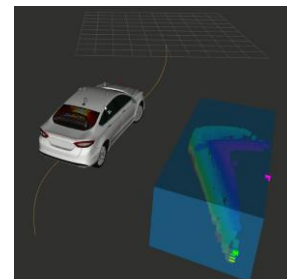


Figure 18 ROS Return to the Route Process

IV. CONCLUSION

The simulation results confirm that the autonomous vehicle successfully performed a dynamic overtaking maneuver on an obstacle moving at a constant speed. This was achieved using a Nonlinear Model Predictive Control (NMPC) framework that incorporates time-varying Bezier curve trajectories and historical behavior data to adaptively manage velocity and steering angle inputs. The integration of LiDAR-based perception and Bezier-based trajectory planning enabled the system to anticipate obstacle motion and maintain smooth maneuverability.

Comparative analysis revealed that the NMPC controller produced significantly smoother linear and angular velocity profiles. Specifically, the Root Mean Square Error (RMSE) for linear velocity decreased from 0.27 m/s to 0.084 m/s, and for steering angle from 1.72 rad/s to 0.085 rad/s. The overtaking process maintained a safe distance, with an average clearance of 2.05 meters from the vehicle center to the nearest obstacle.

The proposed method was validated in the Robot Operating System (ROS) and Gazebo environments, ensuring realistic conditions and robust system behavior. These results demonstrate the effectiveness of the NMPC-trajectory integration for enabling safe, real-time overtaking in autonomous vehicle navigation. Furthermore, the framework supports adaptability in dynamic environments and serves as a scalable solution for more complex multi-agent traffic scenarios. Future work may explore its implementation in real-world autonomous platforms to further validate its applicability and robustness.

REFERENCES

- [1] [A. Buyval, A. Gabdullin, and A. Klimchik, "Deriving overtaking strategy from nonlinear model predictive control for a race car," in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), Sep. 2017, pp. 1–6.
- [2] I. Lamouik, A. Yahyaoui, and M. A. Sabri, "Model predictive control for full autonomous vehicle overtaking," *Transp. Res. Rec.*, vol. 2677, no. 1, pp. 1–12, Jan. 2023.
- [3] [S. Dixit, U. Montanaro, S. Fallah, M. Dianati, D. Oxtoby, and T. Mizutani, "Trajectory planning for autonomous high-speed overtaking using MPC with terminal set constraints," in Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC), Maui, HI, USA, Nov. 2018, pp. 1061–1068.
- [4] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1:43 scale RC cars," in Proc. IEEE Conf. Decis. Control (CDC), Melbourne, VIC, Australia, Dec. 2017, pp. 2545–2550.
- [5] P. Petrov and F. Nashashibi, "A nonlinear adaptive controller for a two-vehicle automated overtaking maneuver," in Proc. IEEE Int. Conf. Intell. Transp. Syst., Oct. 2020, pp. 1–7.
- [6] T. Brüdigam, A. Capone, S. Hirche, D. Wollherr, and M. Leibold, "Gaussian process-based stochastic model predictive control for overtaking in autonomous racing," in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), 2021, pp. 1–8.
- [7] L. Zheng, "Bézier curve-based trajectory planning for autonomous vehicles with collision avoidance," *IET Intell. Transp. Syst.*, vol. 14, no. 8, pp. 1–10, Aug. 2020.
- [8] Z. Zhang, Z. Wang, and J. Li, "A path and velocity planning method for lane changing collision avoidance of intelligent vehicle based on cubic 3-D Bezier curve," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 1–12, Sep. 2021.
- [9] A. Franco and V. Santos, "Short-term path planning with multiple moving obstacle avoidance based on adaptive MPC," in Proc. IEEE Int. Conf. Auton. Robot Syst. Competitions (ICARSC), Porto, Portugal, Apr. 2019, pp. 1–7.
- [10] A. Arab and J. Yi, "Safety-guaranteed learning-predictive control for aggressive autonomous vehicle maneuvers," in Proc. IEEE Int. Conf. Intell. Transp. Syst., Sep. 2020, pp. 1–6.
- [11] E. Alcalá, O. Sename, V. Puig, and J. Quevedo, "Learning-based nonlinear model predictive control for autonomous vehicles," in Proc. IFAC World Congr., Berlin, Germany, Jul. 2020, pp. 1–6.
- [12] F. Rossi, A. Norouzi, and M. Diehl, "Real-time learning-based nonlinear model predictive control of a virtual motorcycle employing grey-box modeling through Gaussian processes," *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 2, pp. 1–15, Mar. 2023.
- [13] M. O'Kelly, H. Zheng, D. Karthik, and R. Mangharam, "F1TENTH: An open-source evaluation environment for continuous control and reinforcement learning," in Proc. NeurIPS Compet. Demonstr. Track, Dec. 2019, pp. 1–10.
- [14] A. Raji, A. Liniger, and M. Bertogna, "Multi-layer motion planning and control architecture for autonomous racing," in Proc. IEEE Int. Conf. Intell. Transp. Syst., Oct. 2021, pp. 1–8.
- [15] J. V. Frasch, A. Gray, and M. Diehl, "Real-time obstacle avoidance using spatial nonlinear model predictive control," in Proc. IEEE Conf. Decis. Control, Dec. 2018, pp. 1–6.
- [16] K. Kritayakirana and J. C. Gerdes, "Nonlinear moving horizon estimation for combined state and friction coefficient estimation in autonomous driving," in Proc. IEEE Int. Conf. Intell. Transp. Syst., Sep. 2019, pp. 1–7.
- [17] X. Huang, W. Zhang, and P. Li, "A path planning method for vehicle overtaking maneuver using sigmoid functions," in Proc. IFAC-PapersOnLine, vol. 52, no. 8, 2019, pp. 422–427.
- [18] B. Kim, D. Kim, S. Park, Y. Jung, and K. Yi, "Automated complex urban driving based on enhanced environment representation with GPS/map, radar, LiDAR, and vision," in Proc. IFAC-PapersOnLine, vol. 49, no. 11, 2016, pp. 190–195.
- [19] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive cruise control with autonomous overtaking," in Proc. IEEE Conf. Decis. Control, Dec. 2018, pp. 1–6.
- [20] A. Feher, A. Arpad, and C. Pozna, "Deep reinforcement learning for autonomous car path planning and control," in Proc. IEEE Int. Conf. Auton. Robot Syst., Mar. 2024, pp. 1–8.