2021 IEEE Intelligent Vehicles Symposium (IV)
July 11-17, 2021. Nagoya, Japan

Multi-Constraint Predictive Control System with Auxiliary Emergency Controllers for Autonomous Vehicles

Farhad Partovi Ebrahimpour and Hasan Ferdowsi, *Member, IEEE*

Abstract— **This paper introduces a multi-constraint predictive control algorithm along with a safety layer to guarantee path tracking and object avoidance in emergency situations. A controller switching mechanism is designed which switches the controller between a main and an emergency controller. As the main controller, a nonlinear multi-constraint model predictive controller (MPC) is designed. The MPC algorithm is compared with Stanley and PID methods in terms of their efficiency to validate the MPC as the main controller. However, in unexpected situations, the high computational time of the planner and MPC modules threatens the safety of the vehicle. In order to respond as quickly as possible, emergency braking and maneuver systems are added which could be triggered separately per different situations. Two different emergency scenarios have been implemented in CARLA simulator and Python environment to evaluate the proposed method.**

Keywords – Automated Vehicles, Vehicle Control, Collision Avoidance

I. Introduction

An Autonomous vehicle (AV) is a piece of mechanical equipment capable of discerning its own path by making use of corresponding sensors and processors. In order to safely operate in everyday traffic or in harsh off-road environments, a multitude of problems in the functionality of these vehicles have to be considered [1]. The paramount issue in autonomous driving is safety. Basically, two levels of obstacle avoidance are applied to make sure that the vehicle will not collide with obstacle [2]. Autonomous vehicles are supposed to navigate in complex urban areas with some specific driving rules and other autonomous or human-driven vehicles, pedestrians, and other types of obstacles that are involved. Sometimes, it is difficult to anticipate the other objects' intentions even with good estimation of their positions and headings. Also, it is possible that an object appears suddenly ahead in the road that the perception part was not able to detect it soon enough because of another object occlusion In addition, the prediction of a human driver that violates traffic rules is a possible challenge [3]. It may cause a collision if the driver does not notice the approaching car and could collide with the rear axle line of the car. Also, we need to consider the computational time of the control and other perception modules. When all of these modules fail to ensure the safety, a controller with low computational cost is the last stand. Designing a controller with emergency safety controllers can tackle this challenge. There are longitudinal and lateral control problems which can be solved separately or together. Originally, throttle and brakes are considered as the actuators to implement longitudinal control. On other hand. Lane keeping, lane changing, and yaw stability control to prevent the vehicle from spinning and drifting out are the most common examples of lateral control [4].

The PID feedback control is the most common and widely used model–free algorithm in autonomous vehicle feedback control systems. The simple structure, good control performance and easy implementation are some of the advantages of PID control; however, no parameter optimization, lack of automatic adaptation to environment produced by complexity of vehicle dynamic and uncertainty of the external environment, and the non-holonomic constraints are the drawbacks of this approach [5]. In the autonomous vehicle field, geometric path tracking methods can be considered as one of the most commonly used and popular classes. Normally, obtaining error is done by measuring the error ahead of the vehicle and involving circular arc calculations [6]. Borrelli et al. in 2005 authors of [7] come up with an MPC approach for an autonomous vehicle at low speed, but high speeds are not considered. After two years, they designed an efficient MPC method for relatively high speed, but it needs to have more control inputs such as suspension [8]. In [9], a novel MPC approach is designed based on kinematic model for tracking desired reference trajectory, but at higher sinusoidal speeds, the errors increase significantly. A multi- constraint model predictive controller (MMPC) computes the wheel angle to avoid collision in [9], yet we need another level of obstacle avoidance by using radar data or another backup controller that would be fast enough in unforeseen scenarios. Also, Bae et al. present an obstacle avoidance approach including communication to traffic lights which applies some new constraints; still, the controller could be extended by receding horizon to solve the problem of limited communication distance. Bujarbaruah et al. [10] focus on lane keeping by using steering angle offset. The forward speed is a constant. It could have had more control outputs and optimization goals such as distance from front vehicle. The proposed approach in [11] solves both the problem of lane changing and moves among multiple obstacles, yet it could be extended to deal with uncertainties in predictions. The yaw and lateral stability in order to lane changing and obstacle avoidance is controlled by an offline NMPC method [12]. The approach needs revision to be practical in real-time. In the most recent work [13], an NMPC method is used with a kinematic model for its simplicity to solve the motion planning and control simultaneously with assuming that upstream modules provide a nominal vehicle route and center of the road lane that the vehicle needs to follow. Also, it ensures collision free path by transforming the predicted

Farhad Partovi Ebrahimpour (email: Z1854810@students.niu.edu) and Hasan Ferdowsi (email: hferdowsi@niu.edu) are with the Electrical Engineering Department at Northern Illinois University, DeKalb, IL 60115.

2021 IEEE Intelligent Vehicles Symposium (IV) | 978-1-7281-5394-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/IV48863.2021.9575951

motions and uncertainty of the other objects along with system dynamics and actuator limits into constraints but still, it is not robust to unexpected events like suddenly appearing a pedestrian in the framework because of sensor occlusion.

The main contributions of this paper include: 1) adding multiple cost functions and constraints to the traditional MPC method in order to achieve efficiency standards, which are minimum cross-tracking error, smooth driving in terms of avoiding sudden changes in steering, and less fuel consumption by minimizing applied throttle percentage in relatively high speeds. 2) Making sure that the designed controller meets desired requirements and comparing it with other methods in different scenarios to prove its efficiency and accuracy in regular situations. 3) Designing and incorporating emergency controllers for longitudinal and lateral control of the vehicle separately by using Stanley and PID methods. Due to their low computational requirements, the obstacle can be avoided in situations that would require reaction times that cannot be met with the main controller. Also, developing an algorithm to switch the controller between the main MPC and the emergency controllers in appropriate times.

II. Predictive Control with Safety Layer

In the method (Fig. 1), perception modules receive and perceive the information related to vehicles, pedestrians, lanes, traffic signs and lights and so forth from the corresponding sensors and send the required information to both planning module and controller switching module separately. Planning computes the desired path which could be a sequence of points on the lanes with respect to the specific constraints and objectives. Switching module is a kind of trigger that determines which controller should take the task to send the control signal toward the actuators of plant. If the planned trajectory meets the safety requirements, it sends the planned path to the MPC controller. On the other hand, if there is an emergency safety issue, it will let the emergency control do the obstacle avoidance task until the situation gets back to normal. The safety module and the controllers will be discussed more in the following subsections.

A. Nonlinear Multi-Constraint Model Predictive Control

Kinematic model is used for its simplicity and comparable results with a dynamic one at low speeds. Note that at higher speeds, dynamic characteristics of the vehicle such as tire friction or slip ratio can affect the motion. The equations that represent a kinematic bicycle model (Fig.2) are [14]:

$$\dot{x} = v \times \cos(\psi + \beta) \tag{1}$$

$$\dot{y} = v \times \sin(\psi + \beta) \tag{2}$$

$$\dot{\psi} = \frac{\mathrm{v}}{\mathrm{l_r}} \sin \beta \tag{3}$$

$$\dot{v} = a \tag{4}$$

$$\beta = \tan^{-1}\left(\frac{lr}{l_f + l_r} \times \tan \delta_f\right) \tag{5}$$

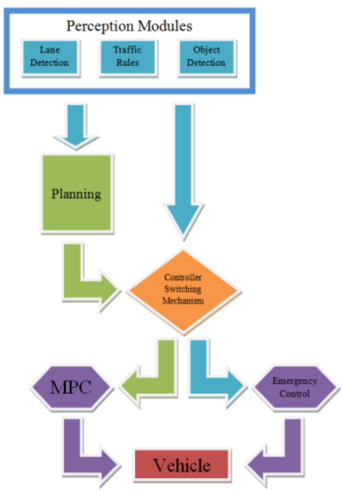

Figure 1. The Overall Safety Architecture

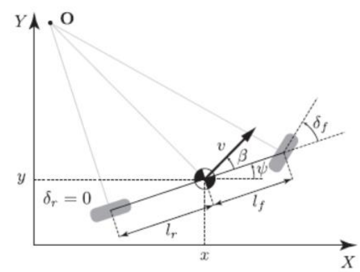

Figure 2. Kinematic Bicycle Model [14]

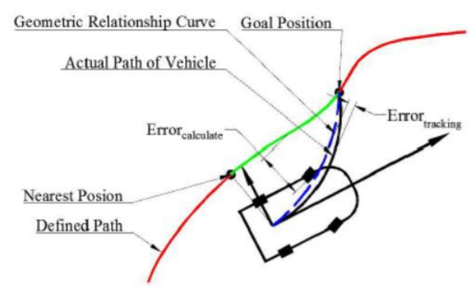

Figure 3. The Geometric Relation Between the Initial Pose and the Goal Pose (blue dashed line) [15]

Polynomial fitting method acts such a low-level planning module in the MPC algorithm. Without utilizing this approach, we will not able to consider road curvatures and predict the future. This method transforms global coordinates to vehicle coordinate system and comes up with an n-degree polynomial (usually a cubic order polynomial) that fits selected points of trajectory that should be followed and addressed as road curvature (Fig. 3). f[x(t)] (15) is the evaluation of polynomial f at point x. A polynomial that is fit to the new points will be obtained and addressed as Road Curvature (f[x(t)]) and the derivative of the function is the road desire heading angle (ψ_{des})), note that tracking error is CTE (Cross Tracking Error) and according to (6):

$$\mathrm{Tracking_{error}}(t + 1) = v(t) \times \sin(e\psi(t)) \times dt \tag{6}$$

where dt is the step time and $e\psi(t)$ is the orientation error, which is given by:

$$e\psi(t+1) = \psi(t) - \psi_{des} + \dot{\psi} \times dt \quad (7)$$

Despite the traditional control theory, MPC is based on solving an optimization problem online by minimizing a cost function (J) made of the states (x) and vehicle inputs (U) within imposed constraints. The optimization problem of MPC control can be summarized as the following and the variable states and control inputs in this case are illustrated (8), (9), and (10) respectively (Please see Appendix).

$$J(x(t), U) = \quad (8)$$

$$\sum_{t=0}^{t=N-1} R \times [u(t)^2] + Q \times \left[x - x_{ref}\right]^2 + R_d \times [u(t+1) - u(t)]^2$$

$$x_t = x(t)$$

$$x_{k+1} = f\,(x_k\,, u_k)$$

$$x_{min} < x_k < x_{max}$$

$$u_{min}(t) < U < u_{max}(t)$$

$$[x\,, y\,, v\,, \psi\,, cte\,, e\psi]^T \quad (9)$$

$$[a\,, \delta]^T \quad (10)$$

Equality and inequality constraints will be imposed on the vehicle. The equality constraints are imposed by the vehicle model and road curvature and inequality constraints are the bounds of the actuators. Actuator constraints are adaptive and can be changed throughout the path based on the specific limitations.

Vehicle kinematics will be used for adding constraint. By discretizing (1) to (5) following constraints can be reached which are applied on the vehicle for the prediction horizon length. ($0 \le n \le N-1$)

$$x[n+1] = x[n] + v[n] \times \cos\psi[n] \times dt \quad (11)$$

$$y[n+1] = y[n] + v[n] \times \sin\psi[n] \times dt \quad (12)$$

$$\psi[n+1] = \psi[n] + v[n] \times \delta[n]/l_f \times dt \quad (13)$$

$$v[n+1] = v[n] + v[n] \times a[n] \times dt \quad (14)$$

$$cte[n+1] = f[x(n)] - y[n] + v[n] \times \sin(e\psi) \times dt \quad (15)$$

$$e\psi[n+1] = \psi[n] - \psi_{des}[n] + v[n] \times \delta[n]/l_f \times dt \quad (16)$$

B. Controller Switching Mechanism

The objective is to determine which controller should override the vehicle. As long as d (the distance between objects and the vehicle) is bigger than the planned minimum distance from objects, MPC will be the main controller by tracking the planned path. When d becomes less than the minimum distance, a safety alarm will come and the information such as vehicle states, available lanes, and safe lane positions (there are), will be sent to emergency control module as its inputs and at the end, each controller will send the control signal to the vehicle (Fig. 4).

The emergency module receives various information regarding the road and vehicle, such as: Vehicle states (position, heading angle, speed), speed limit, and lane positions and the availability of left and right lanes based on object positions and traffic rules. The followings are the calculated information for this module, and the raw data are transmitted from perception and planning modules:

- The distance between objects and the vehicle (d).
- The speed (v_{obj}) and direction of objects with respect to the direction of the vehicle (ψ_{obj}-ψ).
- Stopping distance after braking (d_{stop}) that can be simply obtained by (17) in which v_x is forward speed, μ is the coefficient of friction of road, and g stands for gravity acceleration.
- The maximum allowed stopping distance considering the object's movement is calculated by (18).

$$d_{stop} = \frac{v_x^2}{2\mu g} \quad (17)$$

$$d_{stop_m} = d_{stop} - v_{obj}\cos(\psi_{obj} - \psi).\,dt \quad (18)$$

Switching mechanism determines the type of the emergency controller by predicting the stopping distance and checking whether the vehicle can avoid the possible collision by brake or if it needs to use steering to avoid the collision. However, if any of these actions could not confront a collision, it will give the priority to braking. In summary, this module takes the action when the planning and MPC module fail to avoid objects. Fig.5 shows the safety control in detail.

Emergency brake only cancels the throttle value and triggers the braking system of the vehicle. Emergency maneuver consists of two different controllers for two different objectives. A PID controller is used for longitudinal control of the vehicle and Stanley controller is involved to control lateral motions.

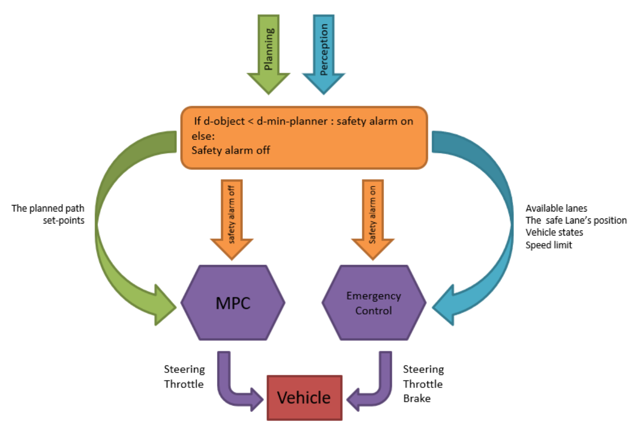

Figure 4. Control Switching Mechanism

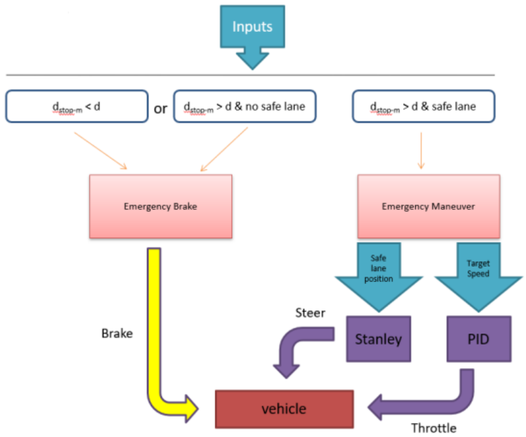

Figure 5. Emergency control

III. SIMULATIONS AND RESULTS

In order to test the proposed method, verify its effectiveness, and compare with existing methods, CARLA simulator has been chosen because of accessibility to simulated urban environment and real car dynamics. The simulations in CARLA are not among the main contributions of this paper, but rather a reliable way to test the performance of the proposed controller. The test scenarios have been created in a way to be as close to real-world conditions as possible without making the simulations too time consuming to run. CARLA simulator is introduced in the following subsection. Also, the programming language used to implement algorithms is Python 3 and several open-source packages have been used, such as SymPy, NumPy and SciPy. All the simulations have been implemented on a laptop CPU (Intel core i5-4200H 2.80 GHz). In the following subsections, PID , Stanley, and the proposed MPC method will all be simulated in CARLA under the same situation.

The prediction and control horizon are set to 10 because with smaller values, the closed-loop predictive control system is not necessarily stable and too many steps will decrease accuracy. Time step is 0.1s, which is the highest possible value for CARLA. MPC parameters values are shown in Table 1. The MPC path tracking is exhibited in Fig. 6.

The methods are implemented in three different velocities and the results are arranged in Table 2. According to Table 2, MPC has the best results among the others. MSE (Mean Squared Error) of "cte" is an estimator that measures the average of the squared errors. It is always non-negative and outcomes closer to zero are better. The second estimator is for smooth steer driving. It has the same structure as MSE_{cte} with only difference in defining the error. The last but not the least is a simple estimator to predict the amount of consumed fuel. The throttle value is constrained between 0 and 1 and the percentage of using the throttle throughout the path could be an acceptable criterion to estimate fuel consumption.

Regarding MSE (Mean Squared Error), the value is increasing as the speed goes up in all of the controller, but the rate of changing of this value is very low in MPC control in comparison with the others. Therefore, the more the speed is increased, the more MPC shows better results than the others. Also, MPC exhibits smooth driving in terms of the rate of consecutive steer value and less fuel consumption, especially at higher speeds. By taking all the estimators into account, we infer that MPC has more preferable tracking result and efficient performance.

Yet, the proposed MPC may not be appropriate for the scenarios that need immediate reactions due to the high computational time and the necessity of having a high-level planner for the prediction and optimization task. The computational time of MPC has been measured in each time step and it turns out that the average of measured times trespasses 0.1 seconds, which is the maximum of fixed time step that we can set in CARLA. However, this value for Stanley and PID is about less than 10^{-4} seconds, which is a thousand times smaller than MPC's. Hence, in the emergency cases where a rapid response is required, Stanley

Table 1. MPC Parameters

MPC Parameters	*Value*
Prediction Horizon	10
Control Horizon	10
Step-Time	0.1(s)
Front/Rear-Wheel Base	1.4(m)
Polynomial-Steps	20
Polynomial-Degree	3

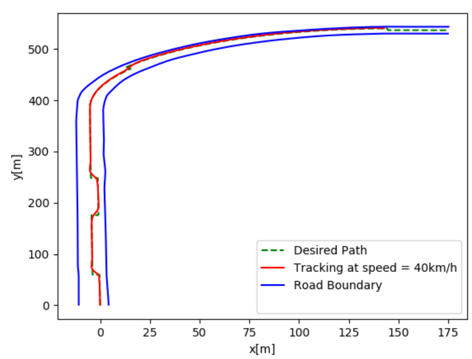

Figure 6. MPC Path Tracking

Table 2. Method Comparison

Controllers	Errors		
	MSE_{cte}(m)	***MSE_{ssd}(× 100)(rad)***	***$ME_{throttle}$(× 100)(%)***
$PID_{v=20\ km/h}$	0.55	0.01	58
$PID_{v=40\ km/h}$	0.9	0.01	72
$PID_{v=50\ km/h}$	3.1	1.5	90
$Stanely_{v=20\ km/h}$	0.29	0.1	56
$Stanley_{v=40\ km/h}$	0.4	0.4	73
$Stanley_{v=50\ km/h}$	2.36	1	88
$MPC_{v=20\ km/h}$	0.08	0.01	66
$MPC_{v=40\ km/h}$	0.09	0.01	67
$MPC_{v=50\ km/h}$	0.1	0.02	68

and PID controller are more appropriate thanks to their faster responses and not requiring a high-level planner.

Two different scenarios are considered for implementing of emergency safety control and related parameters are adjusted based on evaluating the proposed safety control method. These parameters are shown in Table 3 for both scenarios. In the first scenario, the car uses brake to avoid the possible collision. In obedience to the switching mechanism in Fig. 4, if the actual distance between the car and object gets less than minimum allowed distance by the planner, the emergency control module will be triggered, and according to this module in Fig. 5, if the stopping distance is less than the actual distance, it will use the emergency braking module to halt the car. The speed of car is around 35 km/h (Fig. 7_c) when the object distance trespasses minimum distance of the planner (6m) and equation 2.25 gives the stopping distance around 4.45 m. Therefore, the car brakes. The car does not have longitudinal movement (Fig. 7_b) from the moment it reaches yellow line in Fig. 7_a until the safety alarm gets off.

In the second scenario, braking will not help to prevent collision because the speed is higher in this scenario, and stopping distance considering v=50 km\h and μ=0.8 equals to 12 m, and its far greater than 5m, which is the distance of the pedestrian to the car. In this case, we want to make sure that MPC controller cannot avoid the collision. Therefore, MPC controller is used as the emergency maneuver controller.

As shown in Fig. 8, this results in a collision because of the high computational time of MPC and planner due to the optimization and indicates the time that two lines intersect. It occurred when time step was around 200 and the pedestrian exposed herself when time step was around 180; therefore, the car only had 20-time steps to avoid the pedestrian. The reaction time threshold (T_r) can be obtained. It equals to the distance divided by the speed of car:

$$T_r = 4(m) \div 14\ (\mathrm{m/s}) = 0.28(\mathrm{s}) \quad (19)$$

By considering the reaction time, the car should contain a controller with a time step less than 0.01 seconds (0.28 / 20) to react on time, but the proposed MPC controller time step with using the mentioned processor is about 0.1 s, even neglecting the planner computational time. Thus, Stanley and PID controllers can be more appropriate to do this task in view of their computational time, which is a thousand time less than MPC.

Table 3. Scenario Parameters

Scenarios	**Parameters**			
	V_{car}(km/h)	***$V_{pedestrain}$ (m/s)***	***Minimum distance (m)***	***Friction coefficient(μ)***
1	35	0.8	6	1
2	50	1.0	5	0.8

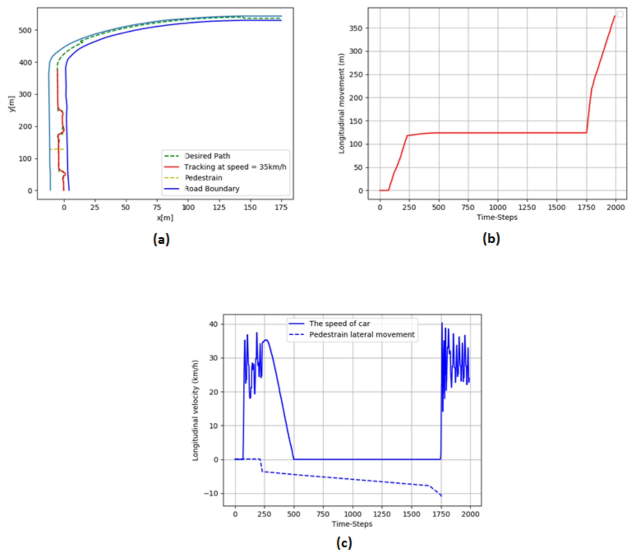

Figure 7. The Emergency Braking Scenario

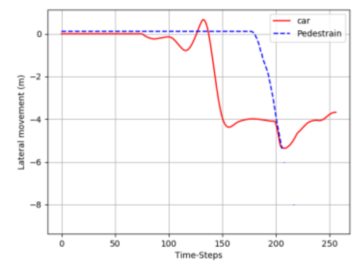

Figure 8. Lateral Movements of Actors (MPC)

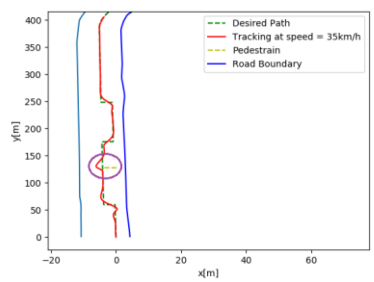

Figure 9. Tracking the Path with the Emergency Maneuver

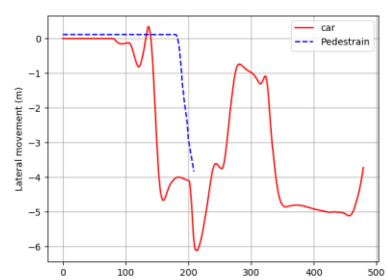

Figure 10. Lateral Movements of Actors (Safety Control)

Scenario 2 is simulated again with Stanley and PID controllers as emergency controllers for controlling steering and throttle, respectively. The result is promising and the proposed safety layer mechanism succeeds to steer clear of the collision. Both actors' paths are also shown in Fig. 9 and the safety maneuver is highlighted with a circle.

By comparing Fig. 8 and Fig. 10, it is clear that the emergency controller avoided the pedestrian in the same reaction time threshold and there is no intersection when the step time is around 200, at which the collision happened in the previous simulation.

IV. CONCLUSIONS

In this paper, the proposed MPC was compared with Stanley and PID methods in terms of efficiency. It was drawn from simulations and comparisons that the MPC controller is far better than the others in different aspects especially in higher speeds. However, under unforeseen circumstances, the high computational time of the planner and MPC modules jeopardizes the safety of the vehicle. In order to react as fast as possible, an emergency brake system incorporates with the main controller. In scenarios that the controller predicts the emergency braking system fails via calculating stopping distance, another emergency control system takes the responsibility of obstacle avoidance by maneuvering properly. Future work can be devoted to investigating safety at higher speeds where the prediction requires vehicle dynamics to be considered because of the impact of tire forces.

APPENDIX

The following matrices pertaining to associated cost functions that are minimized in presented MPC module.

$$\text{State error cost}_{\text{penalties}} = \begin{bmatrix} Q_1 & Q_2 & Q_3 \\ \vdots & \vdots & \vdots \\ Q_1 & Q_2 & Q_3 \end{bmatrix}_{N\times 3} \times \begin{bmatrix} cte(t)^2 & cte(t+1)^2 & \dots & cte(t+N-1)^2 \\ e\psi(t)^2 & e\psi(t+1)^2 & \dots & e\psi(t+N-1)^2 \\ V_e(t)^2 & V_e(t+1)^2 & \dots & V_e(t+N-1)^2 \end{bmatrix}_{3\times N}$$

$$Actuator_{penalties} = \begin{bmatrix} R_1 & R_2 & R_3 \\ \vdots & \vdots & \vdots \\ R_1 & R_2 & R_3 \end{bmatrix}_{N\times 3} \times \begin{bmatrix} a(t)^2 & a(t+1)^2 & \dots & a(t+N-1)^2 \\ \delta(t)^2 & \delta(t+1)^2 & \dots & \delta(t+N-1)^2 \\ a(t)V(t)^2 & a(t+1)V(t+1)^2 & \dots & a(t+N-1)V(t+N-1)^2 \end{bmatrix}_{3\times N}$$

$$onsecutiveActuator_{penalties} = \begin{bmatrix} R_{d1} & R_{d2} \\ \vdots & \vdots \\ R_{d1} & R_{d2} \end{bmatrix}_{(N-1)\times 2} \times \begin{bmatrix} (a(t+1)-a(t))^2 & \dots & (a(t+N-1)-a(t+N-2)^2 \\ (\delta(t+1)-\delta(t))^2 & \dots & (\delta(t+N-1)-\delta(t+N-2)^2 \end{bmatrix}_{3\times(N-1)}$$

Cost Function	***Penalty***
CTE (Q_1)	800
Eψ (Q_2)	100
Speed (Q_3)	10

Cost Function	***Penalty***
Acceleration (R_1)	50
Steering (R_2)	20
High speed High delta (R_3)	5
Consecutive Acceleration (R_{d1})	1000
Consecutive Steering (R_{d2})	100000

REFERENCES

[1] T. Luettel, M. Himmelsbach, and H.-J. Wuensche, "Autonomous Ground Vehicles-Concepts and a Path to the Future.," *Proc. IEEE*, vol. 100, no. Centennial-Issue, pp. 1831–1839, 2012.

[2] S. Liu, L. Li, J. Tang, S. Wu, and J.-L. Gaudiot, "Creating Autonomous Vehicle Systems," *Synth. Lect. Comput. Sci.*, vol. 6, no. 1, p. i-186, 2017.

[3] G. Colin and A. Matthew, "EMERGENCY LANE KEEPING (ELK) SYSTEM TEST DEVELOPMENT," 2013.

[4] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[5] P. Zhao, J. Chen, Y. Song, X. Tao, T. Xu, and T. Mei, "Design of a control system for an autonomous vehicle based on adaptive-pid," *Int. J. Adv. Robot. Syst.*, vol. 9, no. 2, p. 44, 2012.

[6] J. S. Wit, "Vector pursuit path tracking for autonomous ground vehicles," FLORIDA UNIV GAINESVILLE CENTER FOR INTELLIGENT MACHINES AND ROBOTICS, 2000.

[7] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat, "MPC-based approach to active steering for autonomous vehicle systems," *Int. J. Veh. Auton. Syst.*, vol. 3, no. 2–4, pp. 265–291, 2005.

[8] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 3, pp. 566–580, 2007.

[9] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multi-constraints," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 952–964, 2017.

[10] M. Bujarbaruah, X. Zhang, H. E. Tseng, and F. Borrelli, "Adaptive MPC for Autonomous Lane Keeping," *arXiv Prepr. arXiv1806.04335*, 2018.

[11] M. Babu, Y. Oza, A. K. Singh, K. M. Krishna, and S. Medasani, "Model Predictive Control for Autonomous Driving Based on Time Scaled Collision Cone," in *2018 European Control Conference (ECC)*, 2018, pp. 641–648.

[12] M. Metzler, D. Tavernini, A. Sorniotti, and P. Gruber, "An Explicit Nonlinear MPC Approach to Vehicle Stability Control," in *Proceedings of The 14th International Symposium on Advanced Vehicle Control*, 2018.

[13] I. Batkovic, M. Zanon, M. Ali, and P. Falcone, "Real-time constrained trajectory planning and vehicle control for proactive autonomous driving with road users," *arXiv Prepr. arXiv1903.07743*, 2019.

[14] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1094–1099.

[15] Y. Shan, C. Chen, W. Yang, and B. Li, "Improved path tracking approach for unmanned vehicles based on clothoid curve," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 493–498.