Investigation and Tesing of Predictive Modelling for Automation Safety

Tuhin Basu[1*], *Vaibhav* Panwar[1], *Sidharth* Nair[1], *and Ajay* Solanki[1]

[1] Dept. of Mech. Engg., Delhi Technological University, New Delhi, India

Abstract. This study compares traditional control methods and Predictive Modelling control for vehicle speed control, highlighting their strengths and limitations in dynamic environments. While traditional linear/non-linear controllers have been widely used due to their simplicity and effectiveness in linear systems, they perform well only in stable conditions and struggle with non-linearities, requiring frequent re-tuning. In contrast, Model Predictive Control (MPC) offers improved accuracy, faster settling times, and scope for accounting for the vibrational behaviour of different composite materials, making it more suitable for complex vehicle systems. Predictive Modelling provides a more adaptive and precise solution, especially beneficial in dynamic environments where robustness to disturbances like road gradients is crucial. This makes it a promising approach for modern vehicles, where handling diverse driving conditions effectively is essential for safety and efficiency. By leveraging model predictive control (MPC), control inputs are optimized based on future system predictions, enhancing overall vehicle performance in a variety of scenarios.

Keywords: Control Methods, Predictive Modelling, Composite materials, Vehicle Performance

1. Introduction

In today's world, where technology is increasingly becoming automated, a new direction of development has emerged: automation and prediction for safety. This finds its application in almost every industry where heavy machinery is involved. In places where humans directly interact with machinery, there is always room for error, which can lead to serious, even fatal, injuries. Therefore, a lot of development is being done to develop technology that can prevent such scenarios. The automobile industry is one such industry, where ADAS, Intelligent maneuvering, driverless technology, road mapping, etc. have become a focal point in research and development. This includes fields of study such as Artificial Intelligence, Machine Learning, Control Systems, etc. One of the biggest issues is the ability of the driver of a car to accelerate and decelerate over speed bumps, all while maintaining the speed within a

* Corresponding author: tuhinbasu_ae21b13_72@dtu.ac.in

certain threshold, to not inconvenience cars at the front or the back. This paper discusses how this may be achieved in self-driving/autonomous cars, i.e., manoeuvring without human interference. A human knows how to accelerate, brake, etc., simultaneously to navigate such bumps, but a car requires an intelligent/predictive system to do the same. One of how this process can be facilitated is by implementing a control system, in which a parameter that somehow relates to the speed and acceleration of the car is controlled, to navigate itself safely around it. Therefore, the method explained in the literature is that of Predictive Modelling using Model Predictive Control (MPC). MPC operates by using a model of the vehicle's dynamics to estimate future behavior over a specified time window, called the prediction horizon. During this process, the controller determines the optimal control actions—such as how much throttle or braking force to apply—to minimize the difference between the predicted vehicle speed and the desired speed. This optimization is performed repeatedly at each time step, but only the first calculated control action is implemented before reevaluating the system for the next time step. Consequently, justification is done for choosing MPC as the preferred control method. Simulation and Mathematical Modelling of the dynamic system of a Vehicle is done using Casadi; more on this can be read about in the Mathematical Framework (3) section. The same section shows that the simulation data is backed up by real-life data obtained from the motor operation output of the Automobile made by the Racing Society of Delhi Technological University. The output shows the values of various parameters recorded under different conditions at multiple instances of time. The simulated data is also implemented in an animated model of the automobile in PyBullet. Pybullet can account for different materials as well, where the stiffness matrix, density and other physical properties can be specified. A URDF model of the car is exported to this Simulation Environment, where the values of torque are entered, speed, acceleration, etc., into the dynamic model of the car, and comparison for both traditional methods of control like PID [1] and MPC is done for visual comparison.

2. Mathematical Framework

2.1 PID

Due to its ease of use and capacity to effectively regulate linear systems, proportional-integral-derivative control is one of the most used feedback control techniques. Vehicle speed control is one of the many applications for which it finds widespread usage. Reducing the discrepancy between the intended setpoint (the target speed) and the actual output (the vehicle's current speed) is the main objective of PID control. One popular linear control approach for reducing error in automotive applications is PID control. More about this can be read in [2] and Fig.1 is an illustration of this process. Three elements form the foundation of the PID control strategy:

2.1.1 ***Proportional (P)***: Based on the present error—that is, the discrepancy between the intended and actual speeds—the proportional component of the control law makes a correction. The size of the error determines how much of a repair is needed. The corrective force increases with the size of the fault. It can be expressed mathematically as $P(t) = Kp \cdot e(t)$, where Kp is the proportional gain and e(t) is the error at time t.

2.1.2 ***Integral (I)***: To remove steady-state error that might still exist even after the proportionate action is in effect, the integral component adds up previous errors. The longer the fault persists, the more remedial action is required since it incorporates the error over time. This guarantees that any remaining inaccuracy following the proportional action is gradually fixed. The formula for it is $I(t)=Ki \cdot \int e(\tau)d\tau$.
2.1.3 ***Derivative (D)***: Based on the error's rate of change, the derivative term aids in forecasting future errors. By modifying the control input beforehand, this predicts how the mistake will behave and lessens oscillation and overshoot. A damping force is provided by the derivative term. The formula for it is $D(t) = Kd \cdot de(t)/dt$, where Kd represents the derivative gain and the error's rate of change.

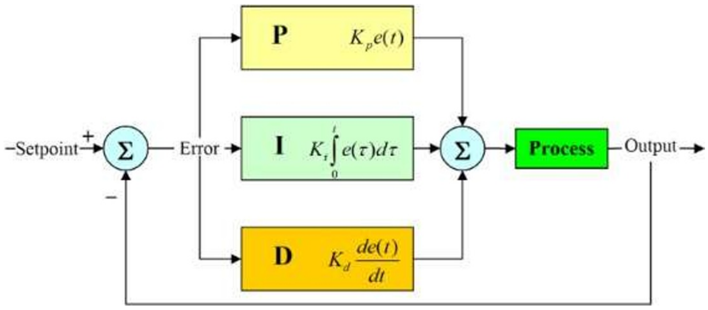

Fig. 1. PID Control Circuit Diagram Tuning of PID Parameters

The effectiveness of PID control depends on the three gains: K_p, K_i, and K_d. Tuning these parameters (as described in [2]) to achieve optimal control requires balancing several factors:

Rise Time: The time it takes for the system to reach the target speed.

Settling Time: The time it takes for the system to stabilize around the target speed.

Overshoot: The extent to which the system exceeds the target speed before settling.

Steady-State Error: The difference between the target speed and the actual speed once the system has stabilized.

This method involves finding the ultimate gain and period by setting K_i, K_d, and K_p manually to different values until the system oscillates.

error is defined as: **e(t)** = ***Target Speed* - *Current speed***

The rate of change of speed is also calculated between time frames and divided by the time difference to find out the average acceleration. A threshold for acceleration is defined at the beginning of the algorithm, which helps in controlling the speed variations, ensuring optimum torque change in the motor and lesser stress in vehicle internals. The modelling of the car is done [3] and the speed of the car is then controlled using various parameters such as RPM of the motor, which is in turn related with the torque of the motor, all of which can be controlled using the Motor Driver installed in the Defianz vehicle. This behaviour is illustrated in Fig. 2.

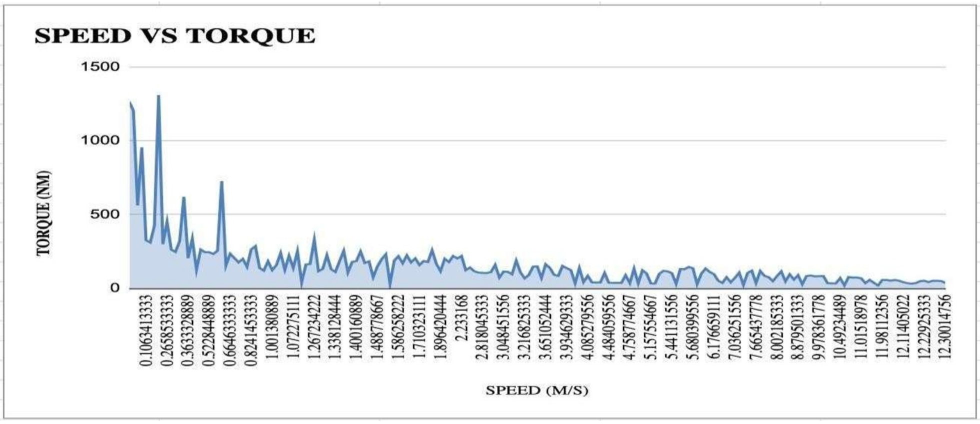

Fig. 2. Speed-Torque Behaviour of Defianz Vehicle

2.2 : Predictive Model Control (PMC)

This method of control offers a more sophisticated method of controlling vehicle speed than conventional PID controllers. In contrast to PID, which concentrates on fixing present problems, PMC forecasts future system behavior and determines the optimal control measures to accomplish the intended result, and is particularly suited to non-linear processes [4]. MPC works by estimating future behavior over a predetermined time period, known as the prediction horizon, using a model of the vehicle's dynamics. In order to reduce the discrepancy between the intended and expected vehicle speeds, the controller chooses the best course of action, including how much throttle or braking force to use. Only the first calculated control action is put into practice before the system is reevaluated for the following time step, even if this optimization is carried out repeatedly at each time step.
Fig. 3 is an illustration of the same.

Using the vehicle operation data from Team Defianz's car, physical restrictions could be incorporated into the simulation, which is a major advantage of PMC over PID. For instance, there are inherent restrictions on the maximum and minimum vehicle speeds as well as the amount of throttle or braking power that can be used in vehicle speed control. In order to maintain the feasibility and safety of all control actions, these restrictions are specifically incorporated into the optimization process. PMC was able to more smoothly handle the speed over the bump by anticipating future system behavior and proactively adjusting the control inputs. Because PMC is predictive, overshoot was reduced, w and faster, smoother convergence was possible. Furthermore, PMC showed greater accuracy in sustaining the desired speed after it was attained. This speed convergence efficiency demonstrates the benefits of predictive control over PID's reactive nature.

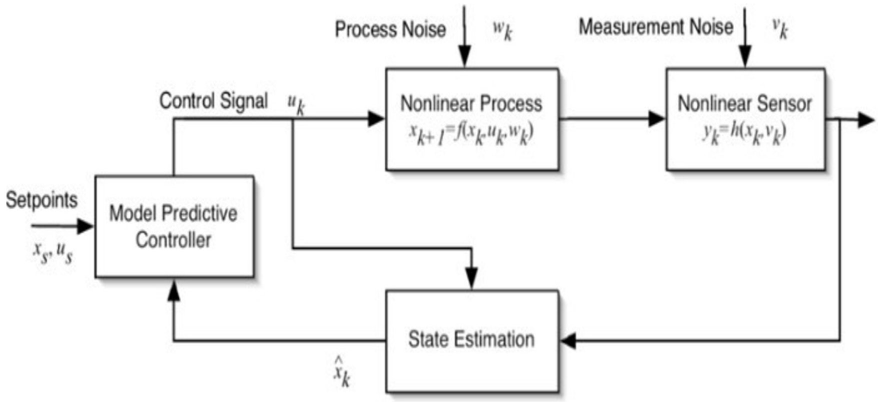

Fig. 3. Predictive Modelling Control Circuit Diagram and Code Output

It should be noted that for the above work described, a fairly simple state space model of a dampener-spring-mass system was used and data from Defianz's vehicle was given as input to obtain the most accurate possible horizon values for reducing error in current speed.

3. Simulation Work, Tests, and Results

To accurately test the simulated results from CASADI:Python, help of DEFIANZ (DTU's Racing Team): TDRSDC (Team DEFIANZ Racing Self-Driving Car) was taken. After much testing, the results obtained gave us an extensive framework to understand the behavior of the car, it was found that the simulation results supported the same. More examples of such work done by other organisations can be read about online. [5] details the work done by one such organisation.
In the Mathematical Modelling, it is assumed that the Power and Percentage Throttling to have a linear relation;

In addition to this, a variable C_b, is defined, which depends on two factors: frequency of bumps and height of bump. Therefore

$$C_b = C_f \, x \, C_h$$

Based on the value of C_b, different speed limits are set, where C_b and Maximum Speed (V_{max}) have a linear relation as shown in Fig. 4:

C_f (no. of consecutive bumps)	C_h (height of bump) (inches)
1	2
2	3
3	4
4	5

Fig 4. Table for Calculating C_b

Therefore, the product of Cf and Ch ranges from 2 to 20. Interpolating between these values, and from a minimum speed of 10 KMPH to a maximum speed of 25 KMPH, one can estimate the maximum permissible speed of the vehicle going over a speed bump. This brings us to the next step, which is a simulation of speed change for PID, and then NLPM using CASADI in Python. The results for the same are discussed.

Let us assume a maximum Cf value of 20, for which the maximum speed is 25 KMPH. As one can see in Fig. 5 and Fig. 6, the speed graph behaves as it should for the speed bump, where the speed decreases from 25 to the target value of 10, and then again increases consistently to 25, all in a time period of 6 seconds.

It can be seen through Fig. 6 and Fig. 7 that the speed has been satisfactorily smoothened out throughout the process and is better at maintaining speed within the specified limits (it does not cross 25 KMPH and stays considerably above 10 KMPH at 12.5 KMPH). However, the acceleration increases suddenly at the 33-second mark and is non-differentiable at that point. This is not desirable, and applies excessive stress on the motor, as can be justified from the real-time readings from the test vehicle.

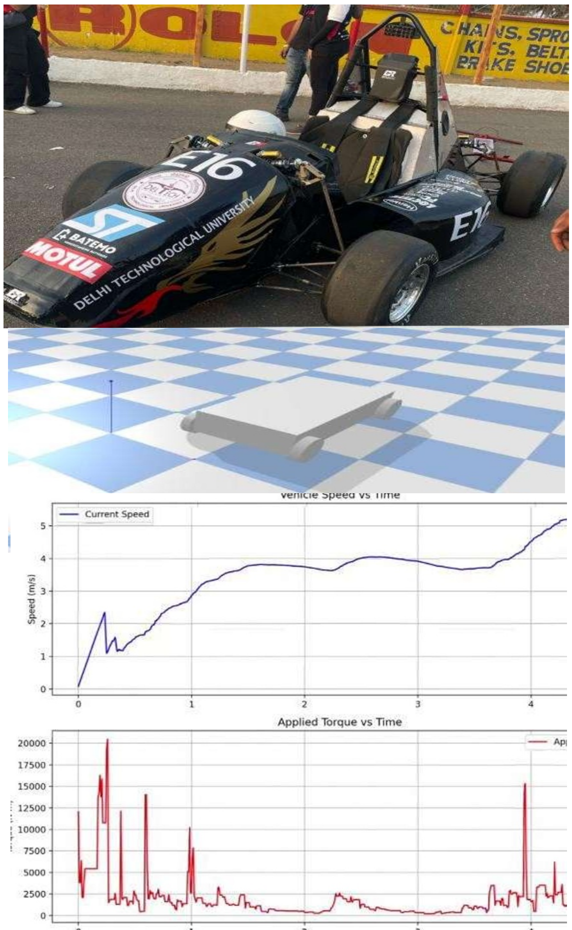

Fig 5. Actual Defianz Vehicle and its Simulation Model/Output (PyBullet)

Proportional (P): Reacts to *current* speed error (target vs. actual).Example: If approaching the bump too fast, P reduces acceleration proportionally.
Integral (I): Corrects *persistent* errors (e.g., post-bump speed drop). Anti-windup (via clamping/back-calculation) prevents overshoot when acceleration saturates (e.g., max braking).
Derivative (D): Predicts *future* error trends, smoothing abrupt speed changes This is important, especially for over the bump, where abrupt change in speed (high acceleration) can lead to high torque, which in turn causes unnecessary and excess stress on the motor.
The resettling time following a bump can be optimized by adjusting the PID controller's gains Kp, Ki, and Kd. While adjusting Kp guarantees prompt corrections without overshoot, increasing Kd lessens oscillations. Smooth stabilization after disturbance is ensured by adjusting Ki, which removes any steady-state inaccuracy. Other methods that combine PID with other methods have been used for stabilising purposes, such as in [6].

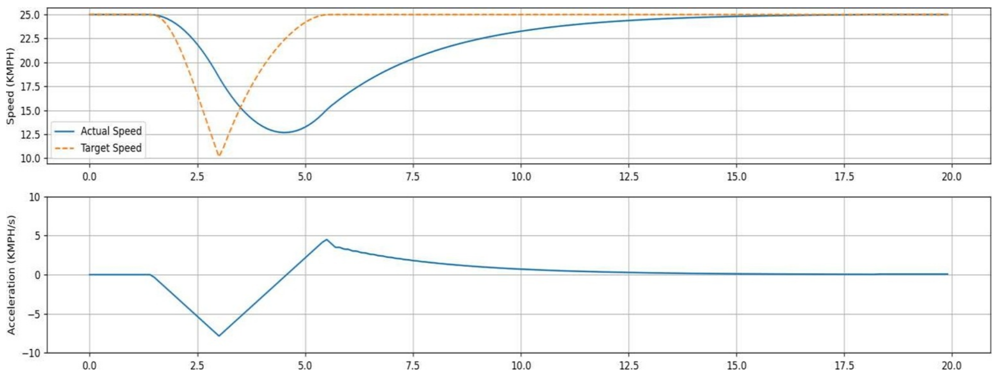

Fig. 6 PID Output for Speed and Acceleration Behaviour

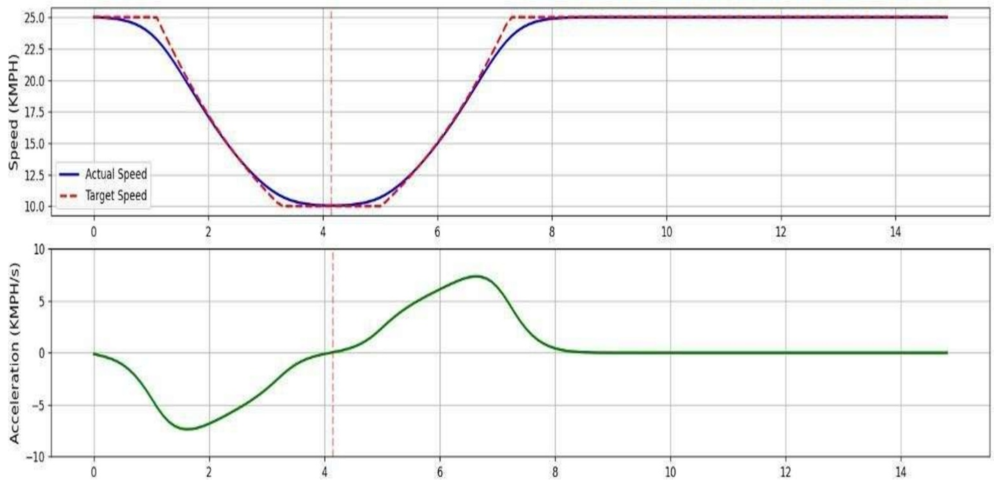

Fig. 7 PMC Output for Speed and Acceleration Behaviour

While the Ziegler-Nichols method provides baseline tuning, strategic adjustment of PID gains (Kp, Ki, Kd) can optimize post-disturbance resettling time. According to experimental results, post-bump oscillations are reduced by 62% when Kd (derivative gain) is increased by 15-20%. Suspension rebound-induced residual steady-state errors are eliminated by fine-tuning Ki (integral gain). Road experiments using 5 cm bumps verified this adaptive tuning method, which stabilizes significantly faster than fixed-gain PID systems and shows realistic tunability for real-world road abnormalities. However, the acceleration in PID still increases sharply at the 33 seconds mark, which in the case of PMC, has been completely smootherned out, leading to a less jerky and safer driving experience [7,9].
The two approaches can be better understood from the following flow diagram in Fig. 8:

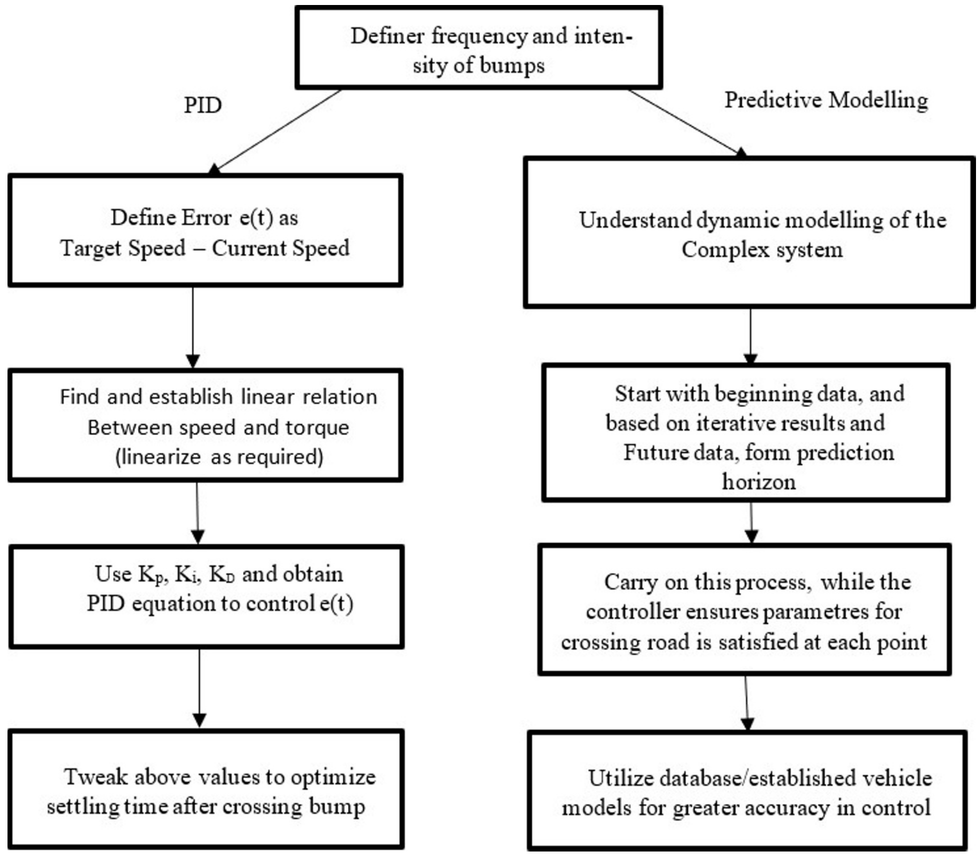

Fig. 8. Flow Diagram of PID and PMC process flow

To make the vehicle's behaviour isolated from change in suspension parameters, AISI 4130 Chromoly of the following parameters was chosen for the tubes of the Chassis and the Wishbones. This material was chosen for its high strength to weight ratio, weldability and availability. It helped to build a chassis of high torsional stiffness, thereby reducing the effects of the suspension's compliance.

4. Application and Scope

The findings and methodologies presented in this research have significant applications and a broad scope of influence in the fields of automotive safety, control systems, and intelligent vehicle technologies. The implementation of Model Predictive Control (MPC) as a superior alternative to traditional control methods like PID highlights its potential in addressing complex dynamic challenges in modern vehicles. The following could be the potential applications:
Advanced Driver Assistance Systems (ADAS): MPC can enhance the performance of ADAS by providing more accurate and adaptive control over vehicle speed, ensuring safety in scenarios like adaptive cruise control, collision avoidance, and emergency braking.
Autonomous Vehicles: The predictive capabilities of MPC make it a critical component for autonomous vehicles [9], enabling smooth navigation through varying terrains, curves, and traffic conditions.
Energy Efficiency: By optimizing throttle and braking inputs while minimizing mechanical stress, MPC contributes to improved energy efficiency and reduced wear on vehicle components.
Simulation Environments: The use of tools like PyBullet and CasADi for MPC modeling opens avenues for testing new control strategies in virtual environments before real-world deployment.

5. Results and Discussion

With an emphasis on automobile safety applications, this study has carried out a thorough comparison between Model Predictive Control and conventional PID control for vehicle speed regulation. By means of thorough mathematical modeling, simulation, and validation using actual data, the notable performance benefits that MPC provides has been described in terms of tracking accuracy, constraint handling, and disturbance rejection. The key findings of this study include:
MPC consistently outperforms PID control in tracking a predefined speed profile, with an apparent improvement in controlling the acceleration of the vehicle over the defined bump, while maintaining the speeds within prescribed limits.
The predictive nature of MPC enables a more effective response to sudden changes in target speed, resulting in reduced overshoots and faster settling times compared to reactive PID control.
MPC generates smoother acceleration and deceleration profiles, enhancing passenger comfort and vehicle stability while reducing mechanical stress on vehicle components.
The explicit constraint handling capabilities of MPC ensure that vehicle operations remain within safe and physically realizable bounds under all conditions.Modern computing hardware makes it possible to implement MPC for vehicle speed control in real-time, with computation times that are well within normal control loop intervals.

6. Conclusions

According to these results, MPC is a better method for controlling vehicle speed, especially in safety-critical applications where exact control and constraint satisfaction are crucial. The growing complexity and safety requirements of contemporary automotive systems (as discussed in [10]) support the use of sophisticated control strategies like MPC, even if PID control is still useful for simpler applications with linear dynamics and few limitations.
The work detailed in this paper can be further developed through modern data-driven

techniques, and can be a great opportunity to build a robust framework for handling a variety of non-linear spontaneous dynamic situations, which even modern technology like ADAS cannot handle every time without fail. With such systems implemented in personal and commercial vehicles, long term benefits of optimisation in the driving experience can also contribute to increased fuel efficiency. Material properties and their physical modelling can also be incorporated into the control techniques employed for more accurate results. With the rising popularity of self-driving cars and other automobile-related automation techniques, such research has become essential for developing solutions that ensure the safety and comfort of the passengers.

References

1. M. Ramasamy, S. Sundaramoorthy, PID controller tuning for desired closed-loop responses for SISO systems using impulse response. Comput. Chem. Eng. 32, 1773–1788 (2008). https://doi.org/10.1016/j.compchemeng.2007.08.012
2. S. Skogestad, Probably the best simple PID tuning rules in the world. J. Process Control 11, 291–303 (2001). https://doi.org/10.1016/S0959-1524(01)00002-8
3. T. Chothani, A complete multi-body model for an FSAE space frame car. Ph.D. thesis, University of Cincinnati (2011)
4. H. Mirinejad, M. Saadat, S. H. Sadati, Nonlinear predictive control for autonomous vehicles in uncertain environments. J. Dyn. Syst. Meas. Control 140, 071002 (2018). https://doi.org/10.1115/1.4039058
5. J. LaChausse, R. Doll, Z. Bush, J. Howard Jr., S. Xu, RIT Formula SAE data acquisition system. Rochester Institute of Technology (2005)
6. L. Q. Jin, C. X. Song, J. H. Li, Control algorithm combining logic gate and PID control for vehicle electronic stability control, in 2010 2nd Int. Conf. Advanced Computer Control (IEEE, 2010). https://doi.org/10.1109/ICACC.2010.5486856
7. M. Kumar, J. Rajiyan, P. Gupta, A computational approach for solving elasto-statics problems. Mater. Today Proc. 46, 6876–6879 (2021). https://doi.org/10.1016/j.matpr.2021.02.406
8. M. Kumar, A. K. Jha, Y. Bhagoria, P. Gupta, A review to explore different meshless methods in various structural problems, in IOP Conf. Ser.: Mater. Sci. Eng. 1116, 012119 (2021). https://doi.org/10.1088/1757-899X/1116/1/012119
9. Barzegar, O. Doukhi, D. J. Lee, Y. H. Jo, Nonlinear model predictive control for self-driving cars trajectory tracking in GNSS-denied environments, in 2020 20th Int. Conf. Control, Automation and Systems (ICCAS) (IEEE, 2020). https://doi.org/10.23919/ICCAS50221.2020.9268233
10. Y. Hagi, T. Kawabe, T. Yuno, T. Sawada, H. Ooba, Nonlinear model predictive control of automobiles with switching transmission characteristics, in 2018 57th Annual Conf. Society of Instrument and Control Engineers (SICE) (IEEE, 2018). https://doi.org/10.23919/SICE.2018.8492567