Path Tracking Control Based on an Adaptive MPC to Changing Vehicle Dynamics

John M. Guirguis, Sherif Hammad, and Shady A. Maged
Ain Shams University, Cairo, Egypt
Email: jmguirguis@eng.asu.edu.eg, sherif.hammad@eng.asu.edu.eg, Shady.Maged@eng.asu.edu.eg

Abstract—**In this paper, an adaptive Model Predictive Controller (MPC) is proposed as a solution for path tracking control problem for autonomous vehicles. The effect of feeding the MPC with a continuously changing vehicle's mathematical model is studied, so that the controller becomes more adaptable to changing parameter values accompanied with instantaneous states. The proposed MPC is compared with both Stanley controller and a similar MPC that uses a fixed vehicle model. The performance is measured by the ability to minimize both lateral position and heading angle errors. A dynamic bicycle model for the vehicle is deployed in the MPC and the controllers are simulated in CarSim-MATLAB/Simulink co-simulation environment using three common maneuvers: S-Road, double lane change and curved road. Results show that the proposed controller gives better tracking performance than the two others with minimal instantaneous and root mean square RMS errors.**

Index Terms—**autonomous vehicle, model predictive control, path tracking control**

I. INTRODUCTION

Autonomous vehicles have become the core of interest for both researchers and manufacturers in the automotive industry where the latest approaches in sensing technologies, artificial intelligence (AI) and control strategies combine together to produce a vehicle of an autonomy level from *zero* to *five* [1] which is – to an extent – able to eliminate the human role to drive it in order to optimize consumption and pollution emission [2], enhance ride comfort [3] and increase driving safety [4]. To reach a driverless ride, the vehicle must be capable of: 1) Planning an optimal trajectory through which it can move from a starting to a terminal position [5], [6]. 2) Tracking the planned trajectory in terms of position and speed. 3) Perceiving the surrounding environment with all its constraints. 4) Localizing itself with respect to the perceived surroundings.

Trajectory tracking control is based on 1) manipulating the steering wheel angle both accurately and smoothly to assure ride comfort, and 2) achieving the desired speed at every waypoint throughout the tracked path. Various control theories can be used in path tracking problem such as the traditional proportional integral differential (PID) controller, geometric controllers' algorithms as Follow the Carrot, Pure Pursuit [7] and the Stanley controllers [8]. However, Fuzzy logic controllers [9] and model predictive controllers (MPC) [10], [11], [12] are more advanced competitors. In [13] Normey-Rico *et al.* proposed a robust PID controller to control a simplified robot model along a given path, but the tracking control problem is usually a multiple input multiple output (MIMO) problem with many constraints which the PID controller fails to handle. An Adaptive modified Stanley controller with fuzzy supervisory system is constructed in [14] to guide an autonomous armored vehicle while in [15] a fuzzy-logic controller is developed to imitate the human behavior to perform a double lane overtaking maneuver.

The MPC algorithm is thoroughly a multi-constraint multi-variable optimization problem as described in Section-III.A and it helps in dealing with nonlinear systems [16]. Jie Ji et al. used the MPC method in [17] to plan a path and track it in order to avoid vehicle collisions, while, in [18] an adaptive MPC is proposed to deal with moving obstacles by performing short-term path planning. In [19] Hengyang, Wang et al. added some fuzzy improvements to the MPC and tested its behavior through simulations which show enhancements in ride comfort and in the ability to track the desired path even if the vehicle is initially far from it. Mohseni et al. in [20] used distributed-MPC system that cooperate to help in different driving scenarios.

The MPC must acquire the mathematical model of the plant which it controls to predict its future states. We managed here to give the basic MPC a degree of adaptability to the changing parameter values of the vehicle model by feeding it with a new discrete model at each time step according to the instantaneous states returned from the vehicle, as in Section-II we can see how these varying states significantly affect the vehicle's mathematical model. We compared our simulation results with another MPC which uses a fixed vehicle model and a Stanley lateral controller which is used by Hoffmann et al. in [21]. The results show that the proposed controller's behavior is too much better than both competitors.

The remainder of this paper is organized as follows: In Section-II the vehicle's mathematical dynamics model is obtained. Section-III introduces the proposed controllers, the MPC optimization problem and how the controller adapts to the changing vehicle dynamics. In Section-IV the simulation environment is explained together with comparison results between the proposed controllers. And

Manuscript received February 1, 2022; revised June 5, 2022.

doi: 10.18178/ijmerr.11.7.535-541

finally, the conclusion and the future work are presented in Section-V.

II. Mathematical Model

In this section we will derive the mathematical model for the vehicle (car) and the tire model used in our control strategy since it is a must to define the plant's mathematical model for any model-based controllers such as basic MPC and AMPC. The bicycle model of an Ackerman steered vehicle [22] is an effective model despite its simplicity, and so it is used in many vehicle control applications and gives results that are close enough to reality, Table I shows all model parameters and symbols.

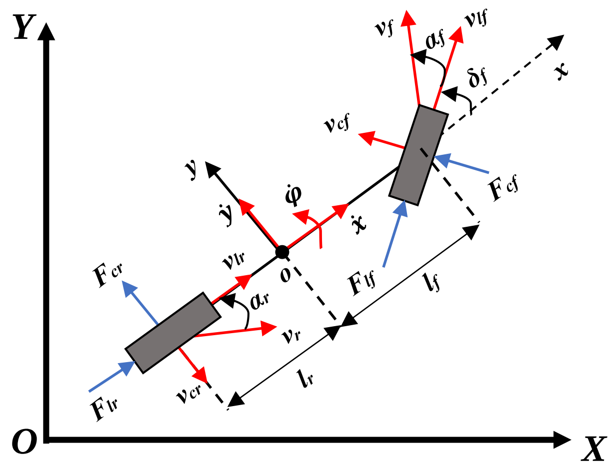

Figure 1. Bicycle model of vehicle dynamics. XOY represents global coordinates and xoy represents local vehicle-fixed coordinates.

TABLE I. Vehicle Parameters and Model Symbols.

Symbol	Definition
F_{lr} / F_{cr}	*Longitudinal / Lateral rear wheel force*
F_{lf} / F_{cf}	*Longitudinal / Lateral front wheel force*
v_{lr} / v_{cr}	*Longitudinal / Lateral rear wheel velocity*
v_{lf} / v_{cf}	*Longitudinal / Lateral front wheel velocity*
α_r / α_f	*Slip angle for rear / front wheel*
φ / $\dot{\varphi}$	*Yaw angle / yaw rate*
l_f / l_r	*Distance from center of mass to front / rear wheel*
δ_f	*Front wheel steering angle*
$\dot{x}$	*Vehicle longitudinal speed in xoy*
$\dot{y}$	*Vehicle lateral speed in xoy*
m	*Vehicle's mass*
I_Z	*Vehicle's body yaw inertia*
S_f / S_r	*Front / Rear tire slip ratios*
C_{lf} / C_{lr}	*Longitudinal stiffness for front/rear tires*
C_{cf} / C_{cr}	*Lateral stiffness for front/rear tires*

As shown in Fig. 1, this model -for simplicity- assumes: 1) That the two front wheels are reduced to one wheel placed at the center of the front axle, and the same for the two rear wheels. 2) Neglection of the aerodynamic forces and suspension effects. 3) Neglection of rolling resistance.

In order to obtain the dynamic equations that describe this physical model, Newton's equations of motion can be used after analyzing lateral and longitudinal forces that act on the vehicle to construct the three equations for the three degrees of freedom (DOF) of the vehicle: lateral, longitudinal and yaw. The main forces that act upon the vehicle to accelerate, brake, or rotate are generated on the tires (F_{lf}, F_{lr}, F_{cf} *and* F_{cr}). This can yield to the following dynamic model [19]:

$$\begin{cases} m\ddot{x} = m\dot{y}\dot{\varphi} + 2\left(F_{lf}\cos\delta_f - F_{cf}\sin\delta_f\right) + 2F_{lr} \\ m\ddot{y} = -m\dot{x}\dot{\varphi} + 2\left(F_{lf}\sin\delta_f + F_{cf}\cos\delta_f\right) + 2F_{cr} \\ I_z\ddot{\varphi} = 2l_f\left(F_{lf}\sin\delta_f + F_{cf}\cos\delta_f\right) - 2l_rF_{cr} \end{cases} \tag{1}$$

Due to the effect of tire forces, it is very important to follow a tire model to obtain these forces in a way that is close enough to reality. A Pacejka tire model is used here, which is a semi-empirical nonlinear model [23], [24]. When the cornering angle and slip ratio of the tires are small, the linearized and simplified formulae of longitudinal and lateral forces in the tire model are given by:

$$\begin{cases} F_{lf} = C_{lf}s_f \\ F_{cf} = C_{cf}\alpha_f \\ F_{lr} = C_{lr}s_r \\ F_{cr} = C_{cr}\alpha_r \end{cases} \tag{2}$$

where the longitudinal force (F_{lf} & F_{lr}) is proportional to the tire slip ratio (s_f & s_r) with the longitudinal stiffness (C_{lf} & C_{lr}) and the lateral force (F_{cf} & F_{cr}) is proportional to the tire slip angle (α_f & α_r) with the lateral stiffness (C_{cf} & C_{cr}), and this applies for the front and rear wheels. From the geometry of the bicycle model in Fig. 1, we can deduce that:

$$\begin{cases} \alpha_f = \tan^{-1}\dfrac{v_{cf}}{v_{lf}} \\ \alpha_r = \tan^{-1}\dfrac{v_{cr}}{v_{lr}} \end{cases} \tag{3}$$

The slip ratio of the for both front and rear tires during acceleration is defined as:

$$S = \frac{\omega R - v_l}{\omega R} \tag{4}$$

where (ω and R) are the rotational speed and the rolling radius of the wheel respectively, and (v_l) is the actual linear speed of the wheel in the longitudinal direction.

The longitudinal and lateral speeds of the front and rear wheels (v_{lf}, v_{cf}, v_{lr} and v_{cr}) can be expressed in terms of local-body-coordinates' speeds ($\dot{x}, \dot{y}$ *and* $\dot{\varphi}$) as follows:

$$\begin{cases} v_{cf} = (\dot{y} + l_f\dot{\varphi})\cos\delta_f - \dot{x}\sin\delta_f \\ v_{lf} = (\dot{y} + l_f\dot{\varphi})\sin\delta_f - \dot{x}\cos\delta_f \\ v_{cr} = l_r\dot{\varphi} - \dot{y} \\ v_{lr} = \dot{x} \end{cases} \tag{5}$$

Substituting by Eq. (2), (3) and (5) in the equations of motion in Eq. (1) yields to the continuous dynamic model

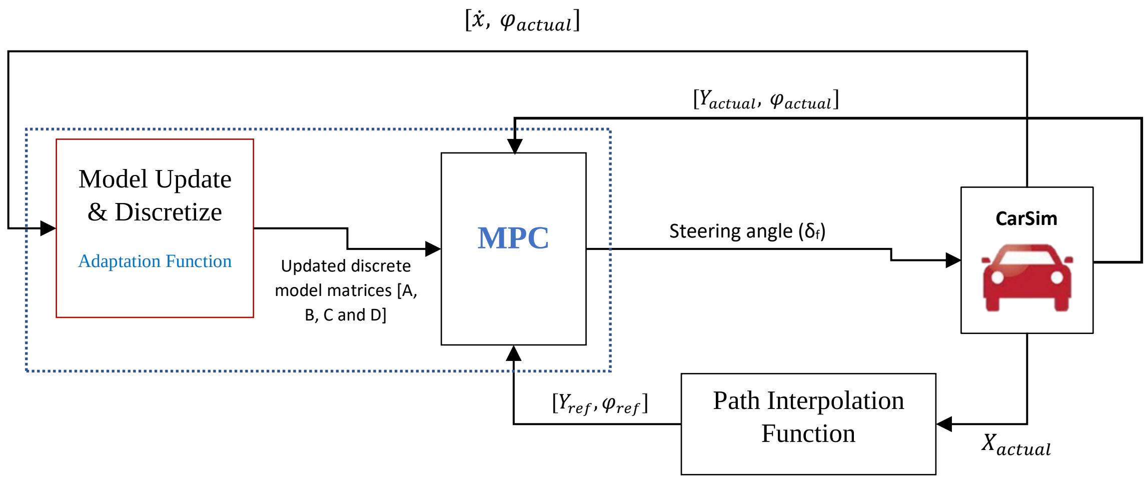

Figure 2. Block diagram of simulation process

of the vehicle. Local-body coordinates (xoy) can be converted to the global coordinates (XOY) as:

$$\begin{cases} \dot{X} = \dot{x}\cos\varphi - \dot{y}\sin\varphi \\ \dot{Y} = \dot{x}\sin\varphi + \dot{y}\cos\varphi \end{cases} \tag{6}$$

As the MPC utilizes the state-space representation of the plant model, the obtained model is converted to a four-state-variable model as follows:

$$\dot{X}_t = A_t X_t + B_t u_t \tag{7}$$

$$Y_t = C_t X_t + D_t u_t \tag{8}$$

where the state vector is $X_t = [\dot{y}, \varphi, \dot{\varphi}, Y]^T$, the steering angle ($\delta_f$) is the only control action passed to the plant, so, the input vector is $u_t = \delta_f$ and the two controlled variables (*i.e.*, plant's outputs) are the lateral position (Y) and the yaw angle (φ), so, the output vector is $Y_t = [Y \quad \varphi]^T$. The state-space matrices (A_t, B_t, C_t and D_t) are obtained from the above equations as:

$$A_t = \begin{bmatrix} \frac{-2(C_{cf}+C_{cr})}{m\dot{x}} & 0 & -\dot{x} - \frac{2(C_{cf}l_f - C_{cr}l_r)}{m\dot{x}} & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-2(C_{cf}l_f - C_{cr}l_r)}{I_z\dot{x}} & 0 & \frac{-2(C_{cf}l_f^2 + C_{cr}l_r^2)}{I_z\dot{x}} & 0 \\ \cos\varphi & \dot{x}\cos\varphi - \dot{y}\sin\varphi & 0 & 0 \end{bmatrix} \tag{9}$$

$$B_t = \begin{bmatrix} \frac{2C_{cf}}{m} \\ 0 \\ \frac{2C_{cf}l_f}{I_z} \\ 0 \end{bmatrix} \tag{10}$$

$$C_t = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \; and \; D_t = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{11}$$

Table II gives the values of the vehicle specific parameters according to the model chosen from CarSim.

TABLE II. VEHICLE PARAMETERS' VALUES ACCORDING TO CARSIM FOR THE USED VEHICLE.

m (kg)	***1110***
I_Z (kg.m^2)	*1343*
l_f (m)	*1.04*
l_r (m)	*1.56*
C_{lf} / C_{lr} (N)	*3000 / 2200*
C_{cf} / C_{cr} (N/deg)	*3200 / 2400*

III. CONTROL STRATEGIES

A. *Stanley Controller*

The Stanley controller is a member of the family of the geometric controllers that uses only the kinematic model of the vehicle ignoring both the dynamic forces and the slipping. Eq. (12) [14] shows how the Stanley controller computes the steering wheel angle (δ), where ($e(t)$) is the continuous error in lateral position and (k) is the controller's gain.

$$\delta(t) = \begin{cases} \varphi(t) + \tan^{-1}\left(\frac{k.e(t)}{V(t)}\right), & |\varphi(t) + \tan^{-1}\left(\frac{k.e(t)}{\dot{x}}\right)| < \delta(max) \\ \delta(max), & \varphi(t) + \tan^{-1}\left(\frac{k.e(t)}{\dot{x}}\right) \geq \delta(max) \\ -\delta(max), & \varphi(t) + \tan^{-1}\left(\frac{k.e(t)}{\dot{x}}\right) \leq -\delta(max) \end{cases} \tag{12}$$

The algorithm ensures that the steering angle is always within upper and lower limits defined by (δ_{max} and δ_{min}).

B. *Model Predictive Control*

The strategy of the MPC is to use the plant's mathematical model to predict its response over a series of control actions within a prediction horizon, where each control action is a result of the solution of an online optimization problem that aims to minimize the gap (*error*) between the reference and the actual value of each controlled variable (plant-output variable), and through this, the system is driven as much as possible to the desired reference states. In path tracking control applications, it is required to guide the vehicle to follow a predetermined path defined by: longitudinal position, lateral position, and yaw angle of the vehicle (*X, Y and φ*) at every time (*t*) with respect to a global frame of reference (*XOY*). According to the vehicle model obtained in section II, the MPC is designed to have one manipulated variable (*i.e.*, single output) which is the steering angle (δ_f) and two plant-output variables (*i.e.*, two inputs) which are the lateral position (Y) and the yaw angle (φ). Because the control loop runs in a discrete system, the pre-obtained state-space model in Eq. (7-11) is discretized as:

$$X(k+1) = A_k X(k) + B_k u(k) \tag{13}$$

$$Y(k) = C_k X(k) + D_k u(k) \tag{14}$$

where

$$\begin{cases} A_k = e^{A_t T_s} \\ B_k = \int_0^{T_s} e^{A\tau} d\tau \,.\, B_t \\ C_k = C_t \\ D_k = D_t \end{cases} \tag{15}$$

such that A_k, B_k, C_k and D_k are the state-space matrices in the discrete form and T_s is the sampling time.

The basic cost function that is subjected to the online optimization for an MPC with one manipulated variable and two plant-output variables can be expressed as follows:

$$J_k = \sum_{j=1}^{n_y} \sum_{i=1}^{p} \left\{ \frac{w_{i,j}}{s_j} \left[r_j(k) - y_j(k) \right] \right\}^2 \tag{16}$$

such that,

k:	Current control step.	
p:	Prediction horizon.	***steps***
n_y:	Number of plant-output variables.	***2***
$y_j(k)$:	Predicted value of the *j*th plant-output variable at the *i*th step.	***$y_1(k) \equiv Y_{pred.}(k)$*** ***$y_2(k) \equiv \varphi_{pred.}(k)$***
$r_j(k)$:	Reference value of the *j*th plant-output variable at the *i*th step.	***$r_1(k) \equiv Y_{ref.}(k)$*** ***$r_2(k) \equiv \varphi_{ref.}(k)$***
s_j:	Scale factor of *j*th plant output	
$w_{i,j}$:	Weight for *j*th plant output at *i*th step.	***dimensionless***

At each control loop the MPC receives the actual values for the lateral position and the yaw angle (Y and φ) as feedback returned from the vehicle and compares them with the reference values computed from the path interpolation function, then, within the prediction horizon, the MPC attempts to predict ahead p values for the manipulated variable that yields to the optimum path that should minimize the error with the reference values and the controller performs only the first control action of this series and then repeats the same algorithm again within each control loop (time step). The optimization problem is usually a weighted minimization for the cost function, we gave a fixed value for the weight of the lateral position (Y) double that given to the yaw angle (φ) such that $w_y = 2$, while $w_\varphi = 1$. Also, the MPC problem here has a constraint for the manipulated variable (δ_f) where:

$$-\delta_f^{max} \leq u(k) \leq \delta_f^{max} \tag{14}$$

So that hard boundaries are kept for the front wheel angle within the real range of the vehicle.

The QP-KWIK algorithm [25] is used for solving the optimization problem.

C. *Adaptive MPC*

The state-space matrices in Eq. (9-11) show that the instantaneous values of the vehicle states ($i.e., \dot{x}, \dot{y}\ and\ \varphi$) significantly affect the whole model of the vehicle, and that's why, we study here the effect of feeding an updated plant model to the MPC based on the actual values of the feedback states returned from the vehicle at each control loop, so that the MPC may adapt to these dynamic changes to substitute for the effect of linearization over a wide range of vehicle states. To achieve this, a MATLAB function is developed, at each control loop, this function acquires the current values for the longitudinal speed and yaw angle ($\dot{x}(k)\ and\ \varphi(k)$) from the vehicle's feedback, together with, the previous control action from the controller ($\delta_f(k-1)$), the function then computes the new continuous state-space matrices (A_t, B_t, C_t and D_t) from Eq. (9-11) and discretize them according to Eq. (15) to obtain an updated discrete model for the vehicle and passes it to the MPC to be used in the next control loop.

During simulation, we studied the effect of this function as an adaptive component and compared the results with those of a same MPC that uses constant state-space matrices all over the path.

IV. Simulation and Results

The proposed controller is built in MATLAB and Simulink, and the vehicle is chosen from CarSim simulation software. Fig. 2 shows the overall system block diagram. The AMPC is compared with an MPC that ignores the changing parameters of the vehicle's model and a Stanley lateral controller for some predefined paths that are chosen to represent known maneuvers of driving and testing scenarios which are: double lane change (DLC) [26], S-road (single lane change) and curved-road. Each path is defined as an [n x 3] matrix where the path is discretized into (n) points, each point is given by (X, Y and φ) coordinates in the global frame of reference. An interpolation function is developed to acquire the actual longitudinal position (X) returned from the vehicle at each loop and returns the corresponding lateral position and yaw angle (Y and φ); these values are used as the set points for the controller as shown in Fig. 2. As the vehicle's speed increases, these maneuvers become more challenging to track. In Table III below, the speeds at which the controller is tested are given, together with the vehicle parameters' values.

TABLE III. Values for Vehicle Parameters.

m (kg)	*I_Z (kg.m^2)*	*l_f (m)*	*l_r (m)*	*v_x (m/s)*
1110	*1343*	*1.04*	*1.56*	*10, 15 & 19*

Simulation results shown in Fig. 3-8 represent the vehicle's lateral position (Y) with respect to its longitudinal position (X) in case of each controller plotted together with the reference path's coordinates of each maneuver at 15m/s and 19m/s. Also, Table 4-6 shows the RMS values of both the lateral position and the yaw angle errors for each controller at different speeds for the S-road, curved road and the double lane change maneuvers respectively.

While trying to tune MPC parameters we reached best results at: 1) 0.1s for step time, 2) a prediction horizon of 14 steps, 3) a control horizon of 3 steps and 4) a constraint for the front wheel angle of about $\pm$ 68° which is close to a real car's value.

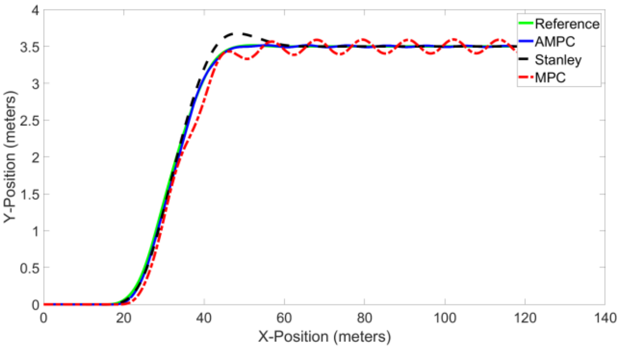

Figure 3. Lateral position for S-Road at 15 m/s.

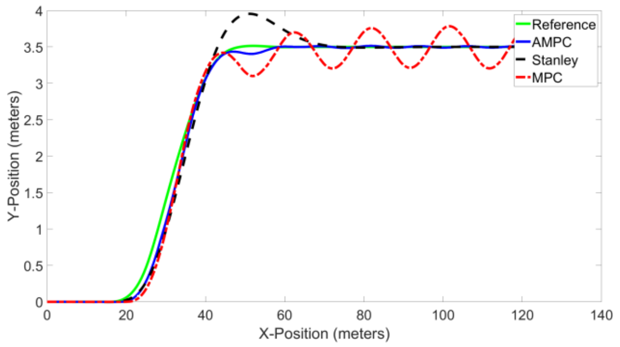

Figure 4. Lateral position for S-Road at 19 m/s.

TABLE IV. RMS VALUES FOR LATERAL AND YAW ANGLE ERRORS FOR S-ROAD

Controller	***AMPC***		***MPC***		***Stanley***	
RMS value	e_y (m)	e_φ (deg)	e_y (m)	e_φ (deg)	e_y (m)	e_φ (deg)
10 m/s	0.054	1.2	0.085	1.33	0.03	1.06
15 m/s	0.066	1.02	0.11	2.23	0.08	1.24
19 m/s	0.118	1.57	0.2	6.44	0.2	2

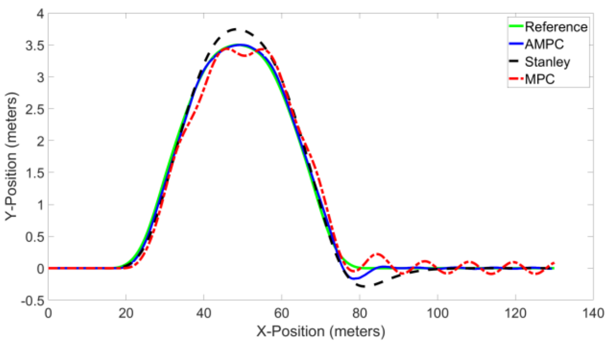

Figure 5. Lateral position for curved road at 15 m/s.

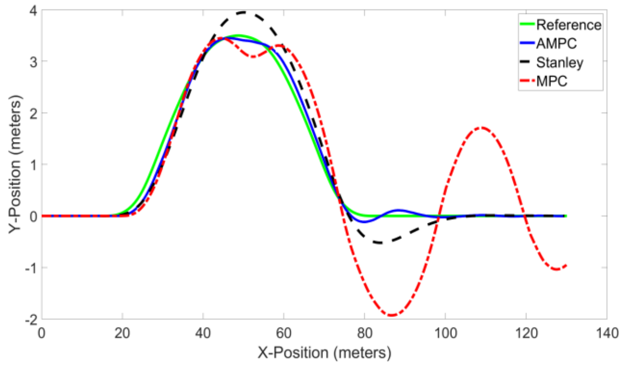

Figure 6. Lateral position for Curved Road at 19 m/s.

TABLE V. RMS VALUES FOR LATERAL AND YAW ANGLE ERRORS FOR CURVED ROAD.

Controller	***AMPC***		***MPC***		***Stanley***	
RMS value	e_y (m)	e_φ (deg)	e_y (m)	e_φ (deg)	e_y (m)	e_φ (deg)
10 m/s	0.08	1.86	0.12	1.92	0.05	1.5
15 m/s	0.1	1.88	0.15	2.89	0.16	1.2
19 m/s	0.16	2.47	0.89	10.7	0.32	2.74

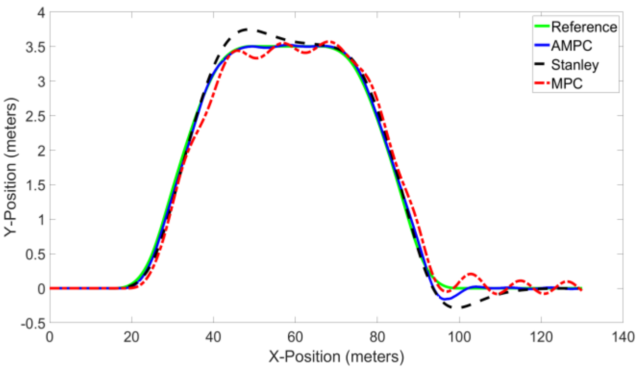

Figure 7. Lateral position for double lane change at 15 m/s.

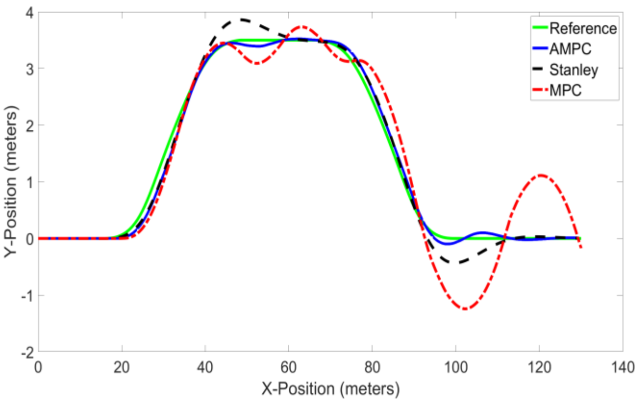

Figure 8. Lateral position for double lane change at 15 m/s.

TABLE VI. RMS VALUES FOR LATERAL AND YAW ANGLE ERRORS FOR DLC.

Controller	***AMPC***		***MPC***		***Stanley***	
RMS value	e_y (m)	e_φ (deg)	e_y (m)	e_φ (deg)	e_y (m)	e_φ (deg)
10 m/s	0.08	1.86	0.12	1.92	0.05	1.52
15 m/s	0.1	1.85	0.15	2.68	0.15	2.06
19 m/s	0.16	2.35	9.47	0.53	0.2	2.59

V. Conclusion

In this paper we made the MPC adaptive to the changing dynamics of the vehicle's model accompanied with the instantaneous states that are fed back from it, this is achieved by updating the plant model in each control loop. The results show that the proposed approach is capable of tracking sharp maneuvers at relatively high speeds (up to ≈ *68 km/hr.*) when compared to an MPC which uses a fixed plant model all over the path and it surpasses the Stanley controller specially at speeds higher than 10 m/s.

The effect of the proposed controller is shown through CarSim-MATLAB/Simulink co-simulations where the controller is implemented in MATLAB/Simulink and the plant used is a vehicle chosen from CarSim.

The real implementation of this controller in a car to replace human steering is a future interest. We would like also to extend our research so that a model-learning layer is added so that the MPC learns the model parameters without declaring them manually and to use runtime changing MPC parameters.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Sherif Hammad, Shady A. Maged and John M. Guirguis conducted the research; Shady A. Maged analyzed the data; and John M. Guirguis wrote the paper; all authors had approved the final version.

References

[1] Amer, Noor Hafizah, et al. "Modelling and control strategies in path tracking control for autonomous ground vehicles: A review of state of the art and challenges," *Journal of Intelligent & Robotic Systems*, vol. 86, no. 2, 2017, pp. 225-254.

[2] M. G. Plessen, "Trajectory planning of automated vehicles in tube-like road segments," in *Proc. 20th Int. Conf. Intell. Transp. Syst.*, Oct. 2017, pp. 1-6.

[3] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1135-1145, Apr. 2016.

[4] L. Li, D. Wen, N. N. Zheng, and L. C. Shen, "Cognitive cars: A new frontier for ADAS research," *IEEE Trans. Intell. Transp. Syst*., vol. 13, no. 1, pp. 395_407, Mar. 2012.

[5] P Raja, S Pugazhenthi, "Optimal path planning of mobile robots: A review," *International Journal of Physical Sciences,* vol. 7, no. 9, pp. 1314-1320, 2012.

[6] D. Q. Zhu and Y. Ming-zhong, "Survey on technology of mobile robot path planning," *Control and Decision*, vol. 25, no. 7, 2010, pp. 961-967.

[7] S. Moveh, M. Hussein, and M. B. Mohamad, "A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle," *International Journal of Computer Applications*, vol. 135, no. 1, pp. 35-38, 2016.

[8] T. Sebastian, et al. "Stanley: The robot that won the DARPA grand challenge," *Journal of field Robotics,* vol. 23, no. 9, pp. 661-692, 2006.

[9] B. Ying and D. L. Wang, "Fundamentals of fuzzy logic control—fuzzy sets, fuzzy rules and defuzzifications," *Advanced Fuzzy Logic Technologies in Industrial Applications*, Springer, London, 2006, pp. 17-36.

[10] C. E. Garcia, D. M. Prett, M. Morari, "Model predictive control: Theory and practice—A survey," *Automatica*, vol. 25, no. 3, pp. 335-348, 1989.

[11] M. Manfred and J. H. Lee, "Model predictive control: past, present and future," *Computers & Chemical Engineering*, vol. 23, no. 4-5 pp.: 667-682.

[12] 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, May 2007.

[13] J. E. Normey-Rico, I. Alcalá, J. Gómez-Ortega, and E. F. Camacho, "Mobile robot path tracking using a robust PID controller," *Control Eng. Pract.*, vol. 9, no. 11, pp. 1209-1214, 2001.

[14] Amer, N. Hafizah, et al. "Adaptive modified Stanley controller with fuzzy supervisory system for trajectory tracking of an autonomous armoured vehicle," *Robotics and Autonomous Systems*, vol. 105, pp. 94-111, 2018.

[15] J. E. Naranjo, C. Gonzalez, R. Garcia, and T. D. Pedro, "Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 438-450, Sep. 2008.

[16] Pereira, G. Collares, et al. "Nonlinear curvature modeling for MPC of autonomous vehicles," in *Proc. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2020.

[17] J. Jie, et al. "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 952-964, 2016.

[18] F. Alberto and V. Santos, "Short-term path planning with multiple moving obstacle avoidance based on adaptive MPC," in *Proc. 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2019.

[19] H. Y. Wang, et al. "Path tracking control for autonomous vehicles based on an improved MPC," *IEEE Access*, vol. 7, pp. 161064-161073, 2019.

[20] M. Fatemeh, E. Frisk, and L. Nielsen, "Distributed cooperative MPC for autonomous driving in different traffic scenarios," *IEEE Transactions on Intelligent Vehicles* , vol. 6, no. 2, pp. 299-309, 2020.

[21] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, et al. "Autonomous automobile trajectory tracking for off-road driving," in *Proc. American Control Conference*, New York, NY, USA, 9–13 July 2007, pp. 2296–2301.

[22] S. Xu and H. Peng, "Design, analysis, and experiments of preview path tracking control for autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[23] J. A. Cabrera, J. J. Castillo, J. Pérez, J. M. Velasco, A. J. Guerra, and P. Hernández, "A procedure for determining tire-road friction characteristics using a modi_cation of the magic formula based on experimental results," *Sensors*, vol. 18, p. 896, Mar. 2018.

[24] S. Cheng, L. Li, H. Guo, Z. G. Chen, and P. Song, "Longitudinal collision avoidance and lateral stability adaptive control system based on MPC of autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[25] C. Schmid and L. T. Biegler, "Quadratic programming methods for reduced Hessian SQP," *Computers & Chemical Engineering*, vol. 18, no. 9, pp. 817–32, September 1994.

[26] ISO 3888-2 defines the double-lane change maneuver to test the obstacle avoidance performance of a vehicle. In the test, the driver: Accelerates until vehicle hits a target velocity.

John M. Guirguis received his B.Sc. degree in Mechatronics Engineering for Ain Shams University, Cairo, Egypt in 2017. He worked as research and teaching assistant in the university since 2018 and till now. In 2018, he started as a M.Sc. student in Mechatronics department at Ain Shams University. His research interests include novel and intelligent control strategies, autonomous vehicles, system dynamics, robotics, and automation.

Prof. Sherif Hammad is founder and CEO at GARRAIO; an embedded automotive software supplier to OEM and Tier 1. Hammad is a professor of computer and systems and was the head of Mechatronics Engineering department and the dean of the school of Engineering at Ain Shams University. He served as the minister of Scientific Research of Egypt. Hammad is May University president academic consultant. Construction of new university in Egypt represents an important challenge and achievement for him.

Shady A. Maged received his B.Sc. and M.Sc. degrees in Mechatronics from Ain Shams University, Cairo, Egypt in 2010 and 2013 respectively. He worked as a research and lecturer assistant in the university since 2010 till 2014. In February 2014, he joined the Egypt-Japan University for Science and Technology as a PhD student. He stayed in 2016 as a visiting researcher in Namerikawa-Laboratory, Keio University. He received his PhD in Mechatronics and Robotics Engineering in 2017. In 2017 he joined the Mechatronics Engineering department at Ain Shams University till now. His research interests are the advanced model-based and intelligent control systems, system dynamics, robotics, soft robotics, and automation.