Proceedings of the 2020 4th CAA International Conference on Vehicular Control and Intelligence (CVCI), Hangzhou, China, December 18-20, 2020

SaPosI1-01.8

Path Planning and Fault-tolerant Control Based on Resistance Network for Autonomous Driving

Tenglong Huang, Huihui Pan, Chi Zhang, and Weichao Sun
Research Institute of Intelligent Control and Systems
Harbin Institute of Technology
Harbin, China
huangtenglong@hit.edu.cn, huihuipan@hit.edu.cn, moshang@hit.edu.cn, w.sun@hit.edu.cn

2020 4th CAA International Conference on Vehicular Control and Intelligence (CVCI) | 978-1-7281-8497-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/CVCI51460.2020.9338441

***Abstract*—This paper proposes a path tracking and planning approach based on backstepping, neural network, and resistance network for autonomous driving. The improved diamond resistance network method is presented to avoid the discontinuous curvature of the generated path in this paper. The path tracking controller is designed based on radial basis function neural network and backstepping method, taking the failure of the actuator into account. The stability of the designed controller is proven in this paper. The simulation results show that the designed controller can track the planned path accurately. Meanwhile, it is compared with the classic backstepping controller, which proves the advantages of the designed controller to deal with actuator failures.**

***Index Terms*—resistance network, path tracking, fault tolerant, path planning, radial basis function**

I. Introduction

Traffic accidents can be reduced by the high-level autonomous vehicle because autonomous driving has the potential to avoid collisions for improving safety [1]. In recent decades, a large number of researchers dedicated to the investigation of autonomous driving to promote the development of relevant technologies. There are many research works, such as ABS, VSC [2], attempt to minimize or even avoid accidents through the designed autonomous driving system. Meanwhile, research on the control strategy of autonomous driving has always been one of the core issues in this field. The performance of the control strategy directly affects the autonomous driving performance. Thus, there has been numerous research work on the design of the control strategy. For example, the collision avoidance system (CAS) is for vehicle chassis control, and vehicle stability control (VCS) is for avoiding accidents by the designed control strategy [2].

The three most important parts of an autonomous vehicle are perception, planning, and control. The perception part is mainly utilized to obtain information about the surrounding environment. Accurate environmental information is the prerequisite of reasonable actions. The primary function of the planning part is to generate decision making instructions and to generate a collision-free trajectory from the beginning to the destination. The control part is designed to track the planned behavior and trajectory. The control part directly affects the action of the actuator and is one of the most critical factors affecting the performance of autonomous driving. Therefore, the path planning and control strategy always are the hot spots in this field [3], [4].

This work was supported in part by the China Automobile Industry Innovation and Development Joint Fund under Grant U1964201.

Path planning has been extensively studied, and many classic algorithms have been proposed to solve path planning problems. To mention some results, state lattice is used in [5], can achieve collision avoidance effectively, but the calculation time and resource consumption will increase sharply as the sampling increases. Rapidly-exploring Random Trees (RRT) is very suitable to handle nonholonomic constraints, autonomous vehicles are often regarded as nonholonomic constraint, so RRT is very suitable for autonomous driving scenarios, but RRT needs to be optimized to reduce the calculation time [6]. More detailed information about the planning algorithm can be referred to [7]. A novel planning algorithm based on resistance network is proposed in [8]. This algorithm can generate a collision-free trajectory efficiently and is very suitable for lane changing, so it is used to generate the planned path in this paper. However, the curvature of the lane-changing trajectory generated by this diamond resistance network is discontinuous. This paper proposes a post-optimization method to solve the problem of discontinuous curvature.

There have been many results in the research and optimization of the control strategy for autonomous vehicles. To mention a few, the pure pursuit approach can be implemented easily and be used in real scenarios, but this algorithm is only suitable for low-speed scenarios [9]. LQR approach has attracted widespread attention, this method can track the planned trajectory accurately [10]. But the robustness of the LQR algorithm may need to be further improved. To improve the robustness, sliding mode control (SMC) has been investigated in [11], the H-infinity control has been studied in [12]. But there are some shortcomings to improve, such as the chattering for the SMC [11]. The more detailed applied scenarios and shortcomings of these robust methods can be found in [13]. Due to the high nonlinearity of the vehicle model [14], the backstepping [15] method can avoid cancellations of useful nonlinearities, and the RBF neural network [16] can approximate the nonlinear model effectively. The designed controller in this paper is based on the combination of backstepping and radial basis function neural network (RBF).

In this paper, we improve the resistance network [8] algorithm for path planning and present a path tracking controller. 1) We propose a smoothing method to avoid the discontinuous curvature of the diamond resistance network algorithm. 2) We design a controller based on the radial basis function and backstepping to cope with the highly nonlinear model of the autonomous vehicle. 3) The actuator failures of the autonomous vehicle is considered and compensated to design the adaptive controller.

The remainder of this paper is organized as follows. Section II explains the principle of the vehicle model which is used in this paper and presents the smoothing method of the diamond resistance network. The designed controller to track the planned path is described in section III, and the stability is proved. In section IV, the simulation results are provided to prove the effectiveness and advantage of the presented approach. We summarize this paper in section V.

II. Vehicle Model And Path Planning

The section focus on the vehicle model and the presented method to smooth resistance network for path planning. A method is proposed to avoid the discontinuous curvature for the diamond resistance network [8].

A. Overall Structure

As shown in Fig. 1, this subsection describes the overall structure for autonomous driving.

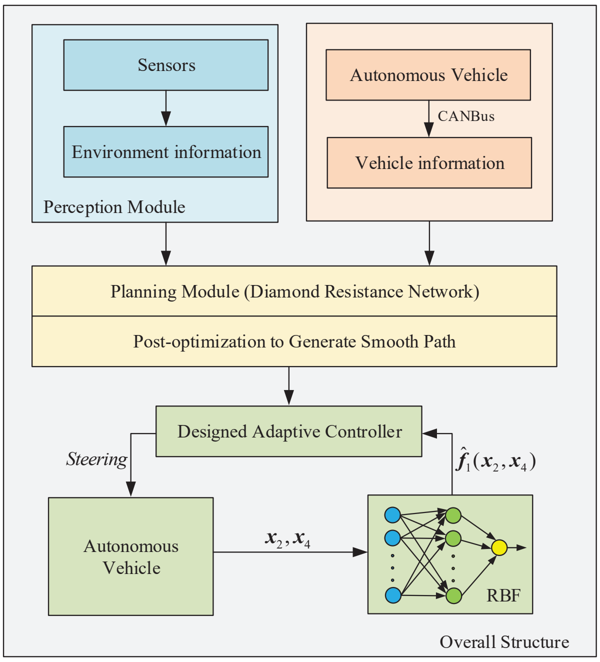

Fig. 1. Overall Structure.

Sensors are used to obtain the environmental information in the perception module. Meanwhile, the vehicle information can be obtained from the CANBus. Combining all the information, the planning module generates the planned path, and the control module is utilized to track the planned path by adjusting the steering.

The planned path is generated based on a diamond resistance network [8] in this paper. RBF neural network is utilized to approximate the model of the autonomous vehicle.

B. Vehicle Model

The generation of the planned trajectory and controller design are based on the bicycle model. The principle of the bicycle model is explained in this subsection. The adaptive controller, which takes the actuator failures into consideration, is designed to track the planned path.

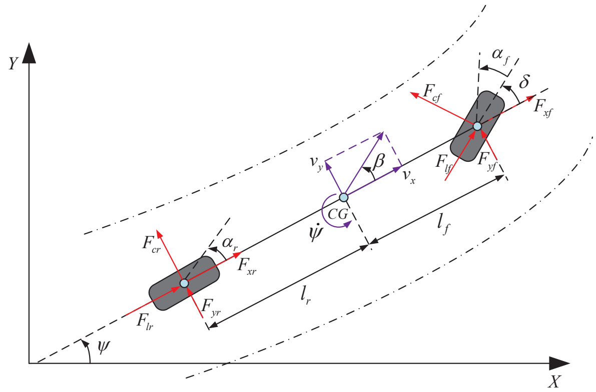

Fig. 2. Bicycle model for the autonomous vehicle.

The bicycle model used in this paper is shown in Fig. 1. A linear parameter varying model (LPV) [17] can be obtained as did. The LPV model of the autonomous vehicle can be expressed as follows:

$$\ddot{y} = -\frac{(C_f + C_r)}{mV_x}\dot{y} - \left(\frac{l_f C_f - l_r C_r}{mV_x} + V_x\right)\dot{\psi} + \frac{C_f}{m}\delta \quad (1)$$

$$\ddot{\psi} = -\frac{l_f C_f - l_r C_r}{I_z V_x}\dot{y} - \frac{l_f^2 C_f + l_r^2 C_r}{I_z V_x}\dot{\psi} + \frac{l_f C_f}{I_z}\delta \quad (2)$$

where the parameters used in the model are listed in Table.I. The value of the parameters are determined according to the DYNA autonomous vehicle [18].

TABLE I

Symbols and Description Of Parameters

Symbol	*Description*	*Value*	*Unit*
$\dot{y}$	Lateral velocity	-	m/s
$\dot{\psi}$	Yaw rate	-	rad/s
δ	Steering angle	-	rad
CG	The center of gravity	-	-
m	Mass of the vehicle	1719	Kg
V_x	Longitudinal velocity	10	m/s
I_z	Yaw moment of inertia	3300	Kg.m^2
C_f	Cornering stiffness of the front tire	170550	N/rad
C_r	Cornering stiffness of the rear tire	137844	N/rad
l_f	Distance between the front tire and CG	1.195	m
l_r	Distance between the rear tire and CG	1.513	m

The model of the autonomous vehicles can be written as follows:

$$\dot{x}_1 = x_2 \quad (3)$$

$$\dot{x}_2 = f_1(x_2, x_4) + g_1\delta \quad (4)$$

$$\dot{x}_3 = x_4 \quad (5)$$

$$\dot{x}_4 = f_2(x_2, x_4) + g_2\delta \quad (6)$$

where $x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T = \begin{bmatrix} y & \dot{y} & \psi & \dot{\psi} \end{bmatrix}^T$, $\boldsymbol{u} = \delta$. y is the the lateral position and ψ denotes yaw angle of the vehicle.

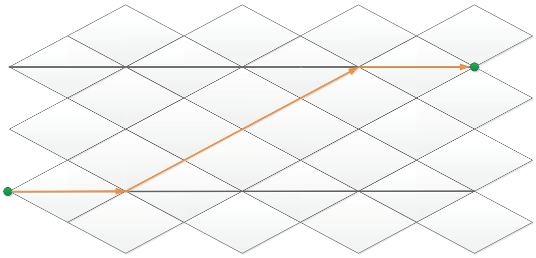

Fig. 3. Planned path by diamond resistance network.

C. Smoothing Method for The Planned Path

This subsection focus on the presented smoothing method for path planning. The diamond resistance network [8] is used to generate the planned path. Meanwhile, a post-optimization method is proposed to avoid the discontinuous curvature of the planned path generated by the diamond resistance network.

The diamond resistance network is shown in Fig. 2, the brown line represents the planned path, and the green dots indicate the start points and destination of the planned path. The curvature of the planned path for changing lanes, which is generated by the diamond resistance network, is discontinuous obviously.

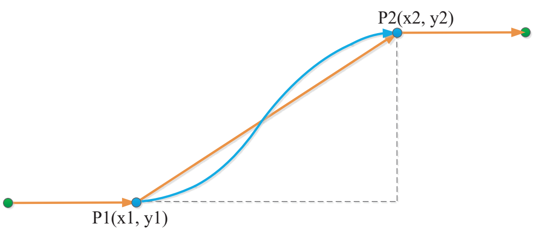

Fig. 4. Smoothing method for diamond resistance network.

To solve this problem, a method is proposed to generate smooth planned path. The post-optimization method is shown in Fig. 3. The planned path is optimized between the blue points $\mathrm{P1}(x1, y1)$ and $\mathrm{P2}(x2, y2)$. The blue path is the smooth path generated by the proposed method. The smooth path can be expressed as follows:

$$Y_p = \begin{cases} y_1 & X_p \in [0, x_1) \\ \frac{S_y}{2} \sin\left[\frac{\pi}{S_x}\left(X_p - \frac{x_1+x_2}{2}\right)\right] + \frac{S_y}{2} & X_p \in [x_1, x_2) \\ y_2 & X_p \in [x_2, 100] \end{cases} \tag{7}$$

where X_p and Y_p denote the longitudinal and lateral position of the planned smooth path, respectively. And $S_y = y_1 + y_2$, $S_x = x_2 - x_1$.

III. Fault-tolerant Controller Design

Because the model of the autonomous vehicle is highly nonlinear [14] and time-varying, the backstepping method [15] can effectively avoid the elimination of useful nonlinear terms, and the RBF neural [16] network can approximate the system nonlinear model efficiently. The designed controller is based on the combination of the backstepping method and RBF neural network in this paper.

Meanwhile, the actuator failures for autonomous vehicles always exist [19]. In this section, the actuator failures of autonomous vehicles are modeled and compensated by the designed controller.

A. Radial Basis Function Neural Network

In this subsection, we briefly introduce the basic principles of the RBF neural network. The radial basis function(RBF) neural network is proposed in [20]. As depicted in Fig. 4, the RBF neural network consists of three layers: input layer, hidden layer, and output layer.

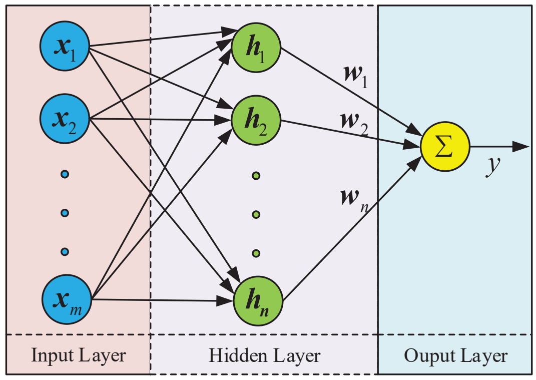

Fig. 5. The structure of RBF neural network.

As shown in Fig. 4, $x = [x_1, x_2, ..., x_m]^T$ is the input vector, h_j denotes the output of the j-th neuron in the hidden layer, $w = [w_1, \cdots, w_n]^T$ is weight of each neuron for RBF. The output of j-th neuron can be described by:

$$h_j = \exp\left(-\frac{\|\boldsymbol{x} - c_j\|^2}{2b_j^2}\right) \tag{8}$$

where $c = [c_{ij}] = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{bmatrix}$, $\boldsymbol{b} = [b_1, \cdots, b_n]^T$ denote the center and width of the j-th neuron, respectively.

The output of the RBF neural network can be generated by:

$$y = w^{\mathrm{T}} h = w_1 h_1 + w_2 h_2 + \cdots + w_{\mathrm{n}} h_{\mathrm{n}} \tag{9}$$

B. Fault-tolerant Adaptive Controller Design

In this subsection, an adaptive fault-tolerant controller is proposed based on the RBF and backstepping method. The actuator fault is modeled, the system model is approximated by RBF, and the adaptive controller is presented based on the Lyapunov method to ensure the stability of the designed controller.

The model of the actuator failure can be described by:

$$\delta_f = pv + d \tag{10}$$

where δ_f is the actual output of the actuator, v denotes the output of the designed controller, $p \in (0, 1)$ is the failure fator, d is the bias failure.

$$\dot{x}_2 = f_1(x_2, x_4) + g_1 pv + g_1 d \tag{11}$$

Defining $\theta_f = g_1 p$, $h_f = g_1 d$, then:

$$\dot{x}_2 = f_1(x_2, x_4) + \theta_f v + h_f \tag{12}$$

Considering $e_1 = x_1 - x_{d_1}$, $e_2 = x_2 - \alpha$, x_{d_1} is the state to track. Defining $\tilde{\theta}_f = \hat{\theta}_f - \theta_f$, $\tilde{h}_f = \hat{h}_f - h_f$, $\tilde{w} = \hat{w} - w^*$, the θ_f and h_f are the actual value, w^* is the weight of RBF neural network to the actual model of autonomous vehicle.

The designed Lyapunov function for the presented controller can be described as:

$$V=\frac{1}{2}e_1^2+\frac{1}{2}e_2^2+\frac{1}{2}r_1\tilde{\theta}_f^T\tilde{\theta}_f+\frac{1}{2}r_2\tilde{h}_f^T\tilde{h}_f+\frac{1}{2}r_3\tilde{w}^T\tilde{w} \quad (13)$$

where r_1, r_2, and r_3 are positive real number. Therefore, $V>0$ holds.

$$\dot{V}=e_1\cdot\dot{e}_1+e_2\cdot\dot{e}_2+r_1\tilde{\theta}_f^T\dot{\tilde{\theta}}_f+r_2\tilde{h}_f^T\dot{\tilde{h}}_f+r_3\tilde{w}^T\dot{\tilde{w}} \quad (14)$$

Considering $\alpha=-k_1e_1+\dot{x}_{d_1}$, the designed apative law can be given by:

$$\dot{\hat{\theta}}_f=\frac{1}{r_1}e_2v \quad (15)$$

$$\dot{\hat{h}}_f=\frac{1}{r_2}e_2 \quad (16)$$

$$\dot{\hat{w}}=\frac{1}{r_3}e_2h(x) \quad (17)$$

The corresponding control law can be described by:

$$v=\frac{1}{\hat{\theta}}\left[-k_2e_2-\hat{w}h(x)-\hat{h}-k_1\dot{e}_1+\ddot{x}_{d_1}\right] \quad (18)$$

According to the designed adaptive law and control law:

$$\dot{V}=-k_1e_1^2-k_2e_2^2 \quad (19)$$

where k_1 and k_2 are positive real number, $\dot{V}<0$ holds. According to the designed adaptive law and control law, the stability of the controller is guaranteed.

IV. Simulation Results

In this section, simulation results are provided to prove that the designed controller can track the planned path accurately and cope with actuator failures.

To prove the advantages of the proposed algorithm, the classic backstepping method is used to design the controller for comparison. The planned path is set to $y(t)=4*sin(t)$. The red line and black lines are the planned path and actual path of the classic backstepping, respectively. The red and black dotted lines are the planned and actual path of the proposed RBF controller. The turquoise and magenta lines represent the tracking error of the RBF and backstepping, respectively.

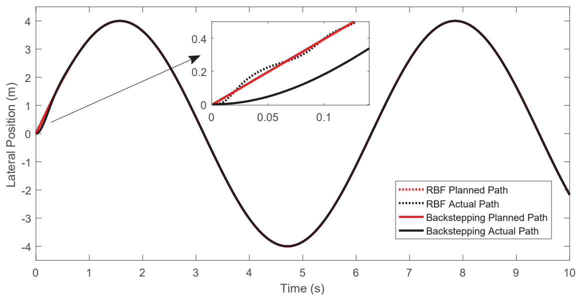

Fig. 6. The comparison of tracking performance without fault.

The tracking performance of classic backstepping and the proposed controller based on RBF are illustrated in Fig. 6. And the tracking error of RBF and classic backstepping is shown in Fig. 7. Both controllers can track the planned path accurately without actuator failures.

To verify the ability of the presented controller to handle actuator faults, failure faults $p=0.7$ and bias faults $d=1$ are added in $t=3$ s. The comparison of tracking performance with fault is shown in Fig. 8. As depicted in Fig. 8, the presented controller can cope with the actuator failures effectively. To illustrate the tracking performance more intuitively, the tracking error is shown in Fig. 9.

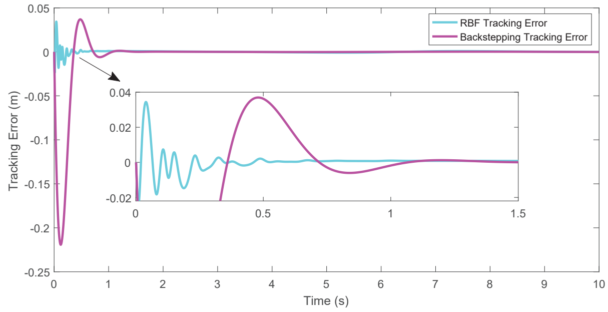

Fig. 7. The comparison of tracking error without fault.

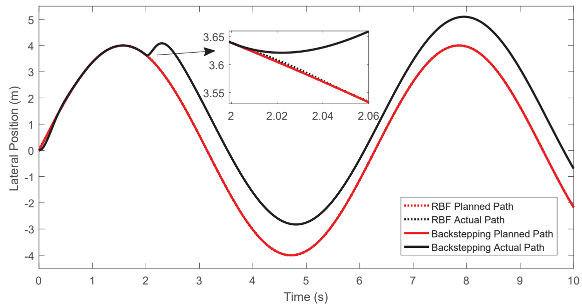

Fig. 8. The comparison of tracking performance with fault.

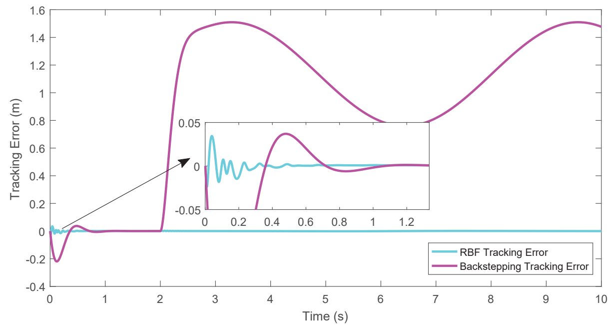

Fig. 9. The comparison of tracking error with fault.

The diamond resistance network is used to generate the planned path, then the generated trajectory is smoothed using the proposed post-optimization method. The smooth path obtained is tracked with the proposed controller, and the tracking performance is shown in Fig. 10. The proposed controller can track the planned path accurately. Meanwhile, we add failure faults $p=0.7$ and bias faults $d=1$ at $t=3$s, the designed controller adjusts quickly and tracks the planned path accurately.

V. Conclusion

This paper proposes a methodology to avoid the discontinuous curvature of the diamond resistance network. The generated planned path is smooth with the proposed method. A controller based on RBF is presented to approximate the nonlinear model of the autonomous vehicle. Meanwhile, the failure fault and bias fault of the actuator is modeled and

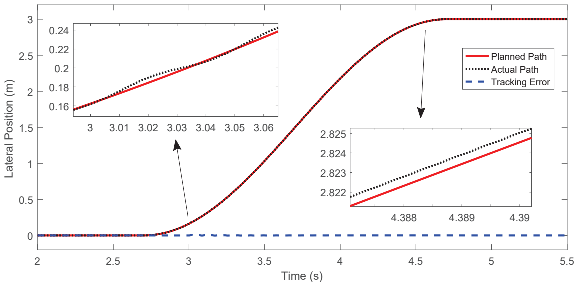

Fig. 10. The tracking performance with fault for LaneChange.

compensated. The simulation results illustrate the effectiveness and advantage of the presented method.

The future work will consider the sensor failures further, and investigate the method to detect and isolate the sensor failures and actuator failures. The improved planning algorithm and the designed controller will be further verified in CarSim or PreScan, and then try to deploy the algorithm on the actual vehicle.

REFERENCES

[1] S. Singh, "Critical reasons for crashes investigated in the national motor vehicle crash causation survey," Tech. Rep., 2015.

[2] S. Chang and T. J. Gordon, "A flexible hierarchical model-based control methodology for vehicle active safety systems," *Vehicle System Dynamics*, vol. 46, no. S1, pp. 63–75, 2008.

[3] A. Khodayari, A. Ghaffari, S. Ameli, and J. Flahatgar, "A historical review on lateral and longitudinal control of autonomous vehicle motions," in *2010 International Conference on Mechanical and Electrical Technology*. IEEE, 2010, pp. 421–429.

[4] H. Chen, X.-m. Wang, and Y. Li, "A survey of autonomous control for uav," in *2009 International Conference on Artificial Intelligence and Computational Intelligence*, vol. 2. IEEE, 2009, pp. 267–271.

[5] M. Pivtoraiko, R. A. Knepper, and A. Kelly, "Differentially constrained mobile robot motion planning in state lattices," *Journal of Field Robotics*, vol. 26, no. 3, pp. 308–333, 2009.

[6] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[7] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[8] Y. Huang, H. Ding, Y. Zhang, H. Wang, D. Cao, N. Xu, and C. Hu, "A motion planning and tracking framework for autonomous vehicles based on artificial potential field elaborated resistance network approach," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 2, pp. 1376–1386, 2019.

[9] N. H. Amer, H. Zamzuri, K. Hudha, and Z. A. Kadir, "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, pp. 225–254, 2017.

[10] C. Hu, R. Wang, F. Yan, and N. Chen, "Should the desired heading in path following of autonomous vehicles be the tangent direction of the desired path?" *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3084–3094, 2015.

[11] M. Aliyu, *Nonlinear H-infinity control, Hamiltonian systems and Hamilton-Jacobi equations*. CRC Press, 2017.

[12] A. T. Azar and Q. Zhu, *Advances and applications in sliding mode control systems*. Springer, 2015.

[13] G. Tagne, R. Talj, and A. Charara, "Design and comparison of robust nonlinear controllers for the lateral dynamics of intelligent vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 796–809, 2015.

[14] G. Reina and A. Messina, "Vehicle dynamics estimation via augmented extended kalman filtering," *Measurement*, vol. 133, pp. 383–395, 2019.

[15] S. Mascolo and G. Grassi, "Controlling chaos via backstepping design," *Physical Review E*, vol. 56, no. 5, p. 6166, 1997.

[16] S. Seshagiri and H. K. Khalil, "Output feedback control of nonlinear systems using rbf neural networks," *IEEE Transactions on Neural Networks*, vol. 11, no. 1, pp. 69–79, 2000.

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

[18] G. Tagne, R. Talj, and A. Charara, "Higher-order sliding mode control for lateral dynamics of autonomous vehicles, with experimental validation," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 678–683.

[19] J. T. Spooner and K. M. Passino, "Fault-tolerant control for automated highway systems," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 3, pp. 770–785, 1997.

[20] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," Royal Signals and Radar Establishment Malvern (United Kingdom), Tech. Rep., 1988.