



Automatic Car Parking Based on Petri Net, PFA Algorithm, and Fuzzy System

A. M. Hatami^{1*}, A. Zamani¹

¹ Department of Electrical Engineering, Technical and Vocational University, Tehran, Iran

ARTICLE INFO	ABSTRACT
<p>Article History: Received 12 April 2020 Received in revised form 18 May 2020 Accepted 9 June 2020 Available online 9 June 2020</p>	<p>This paper presents the design and implementation of a hybrid intelligent system aimed at optimizing vehicle parking through efficient path planning. With the increasing demand for intelligent transportation systems, optimizing parking spaces and vehicle movement within parking areas has become a crucial research focus. The proposed system integrates multiple computational techniques to enhance the accuracy and efficiency of automatic parking. The Path Finding Algorithm (PFA) is employed to identify potential parking locations based on predefined constraints and real-time data. Unlike conventional approaches that rely on static flowcharts for parking path planning, this study utilizes Petri nets, which offer a more dynamic and structured framework for modeling alternative parking paths, particularly in global coordinate systems. This method enables adaptive and flexible decision-making in response to varying parking scenarios. To ensure precise maneuverability along the optimized path, a fuzzy logic control system is implemented, allowing the vehicle to adapt its movements in real time based on environmental factors and space constraints. The effectiveness of the proposed system is validated through numerical simulations and experimental studies, demonstrating its capability to improve both parking efficiency and vehicle positioning accuracy. Results indicate that the hybrid integration of PFA, Petri nets, and fuzzy logic significantly enhances the automation and optimization of the vehicle parking process, offering a robust, adaptive, and intelligent parking solution for modern transportation systems.</p>
<p>Keywords: Automatic Parking Systems, Fuzzy Logic Control, Hybrid Intelligent Systems, Petri Nets, Pathfinder Algorithm, Autonomous Vehicles.</p>	

1. INTRODUCTION

The application of artificial intelligence in the automotive industry has become a significant research focus. Many researchers are working on driverless vehicles that can exhibit intelligence similar to humans. Automatic car parking [3, 6, 8, 9, 10, 11, 13] refers to a vehicle's ability to park itself in a designated spot considering the surrounding static or dynamic environment, posing a challenging problem. Parallel parking has been widely discussed in the literature [1, 2, 4, 5, 6, 7], with researchers proposing techniques to address this issue. The vehicle parking problem can be solved using a path planning approach [1, 8, 9]. Recent literature has developed and tested steering control mechanisms based on numerical computations [2, 3, 4, 5, 6, 10], generally providing real-time simulation to achieve

* Corresponding Author: amirhatami195@gmail.com
 Department of Electrical Engineering, Technical and Vocational University, Tehran, Iran



the desired path. These systems take environmental feedback to determine the vehicle's movement and path. Fuzzy logic-based parking control [15] seems promising due to its need for less space and can be applied in environments like streets and markets. Such issues can be highly beneficial in large cities facing traffic problems.

Recently, AI-based control methods have been increasingly employed, including fuzzy logic controllers, neural network controllers, and genetic algorithms. In the problem of automatic parallel parking, various factors need to be considered to extend some technologies developed for mobile robot systems to automatic parking applications. Hwang et al. [16] proposed that Petri nets are suitable for modeling real-time systems, and [17-19] suggested many analytical techniques that allow Petri nets to examine the occurrence of potentially undesirable states.

Previous approaches to parallel parking have modeled the parking problem without considering collision risks. In a real-world scenario, a human driver possesses all the necessary expertise and skills to reduce the likelihood of accidents during vehicle parking. For example, humans can change the path by looking at obstacles near the vehicle.

This paper presents an automatic parallel parking path planning system that can be used in real-world applications. Unlike traditional design methods, the proposed approach mainly focuses on path planning in global coordinates and calculates the optimal vehicle parking path by combining Petri nets and the PFA algorithm. The proposed strategy can be used in parking assistance devices and has the potential for use in vehicles.

This paper is organized as follows: Section 2 describes the vehicle model and path planning methodology. Section 3 details the system used in this study. Simulations for various scenarios are presented in Section 4. Finally, conclusions and findings are presented in Section 5.

2. PROBLEM DESCRIPTION

Automatic parking is defined as the vehicle's independent movement from the traffic line into the parking space for parallel, perpendicular, or diagonal parking. The goal of automatic parking is to increase the safety and convenience of drivers and passengers during driving. Automatic parking is achieved by coordinating the steering angle and speed, considering real-time conditions to ensure collision-free vehicle movement in the available space.

2.1. Vehicle Modeling

Consider the kinematic model of the vehicle shown in Figure 1. In this vehicle, the rear wheels are parallel to the vehicle body and are allowed to rotate but not slip. This ensures that the rear wheels are always tangent to the direction of the vehicle's movement. The vehicle's parameters are defined in Table 1. The rear wheels are always tangent to the vehicle's direction. Under normal conditions, the vehicle's speed is about 5 km/h. Therefore, it can be assumed that non-slip conditions exist, and the rear wheel speed in the vertical direction must be zero. This ensures the following equations hold:

$$\begin{cases} \dot{x}_r = v \cos \theta \cos \phi \\ \dot{y}_r = v \sin \theta \cos \phi \\ \dot{\theta} = v \frac{\sin \phi}{l} \end{cases} \quad (1)$$

These equations are used to determine the vehicle's next state (backward movement) when the current state and control input are known.

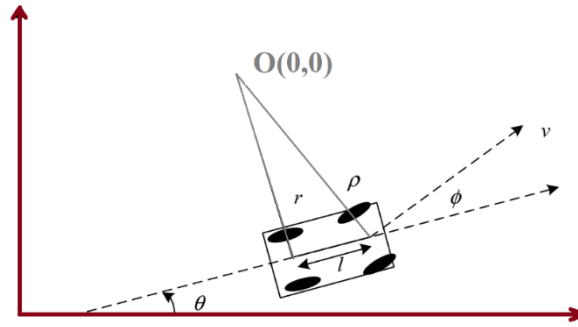


Fig. 1. Kinematic model of the vehicle

2.2. Reference Path for Automatic Parking

Providing a logical reference path for automatic parking is crucial to ensure the vehicle successfully performs the parking operation, which in this study is parallel parking. A method based on the derivatives of fifth-order polynomials [20] is adopted to implement a smooth path as the reference path.

2.3. Required Parking Space

Although this project considers the parking environment as realistic, not every path can be chosen. To prevent collisions or accidents, the vehicle must be positioned within a suitable area before performing parking maneuvers. Therefore, the primary task is to determine the appropriate area for parking using other constraints. The first constraint is the maximum curvature of the path, defined by the following relation:

$$k(x) = \frac{y''}{[1+(y')^2]^{3/2}} \leq k_{max} \quad (2)$$

Another constraint is the distance between the vehicle and an obstacle or poin (x_{ob}, y_{ob}) , as shown in Figure 2. Considering the vehicle dimensions specified in Figure 3, the values (x_{rr}, y_{rr}) and (x_{fr}, y_{fr}) can be calculated from the reference point (x_{ref}, y_{ref}) generated by the reference path.

$$\begin{cases} x_{rr} = (x_{ref} - L_2 \cos \theta) + \left(\frac{W}{2}\right) \sin \theta \\ y_{rr} = (y_{ref} - L_2 \sin \theta) - \left(\frac{W}{2}\right) \cos \theta \end{cases} \quad (3)$$

$$\begin{cases} x_{fi} = [x_{ref} + (L_1 + l) \cos \theta] + \left(\frac{W}{2}\right) \sin \theta \\ y_{fr} = [y_{ref} + (L_1 + l) \sin \theta] - \left(\frac{W}{2}\right) \cos \theta \end{cases} \quad (4)$$

If $y_{rr} \leq y_{ob}$ and $x_{rr} \geq x_{ob}$, the rear right corner of the vehicle will collide with the obstacle. Similarly, if $x_{fr} \geq x_{ob}$ and $y_{fr} \leq y_{ob}$, the front right corner of the vehicle will collide with the obstacle. By adhering to these constraints, the appropriate parking area can be determined.

Table 1. Definitions of Variables Used in This Study

(x_f, y_f)	Front Wheel Center Position
(x_r, y_r)	Rear Wheel Center Position
φ	Wheel Steering Angle
θ	Vehicle Body to X-Axis Angle
l	Wheelbase
O	Center of Curvature
ρ	Distance from Point O to Front Wheel Midpoint
r	Radius of Curvature
v	Front Wheel Speed
κ	Path Curvature

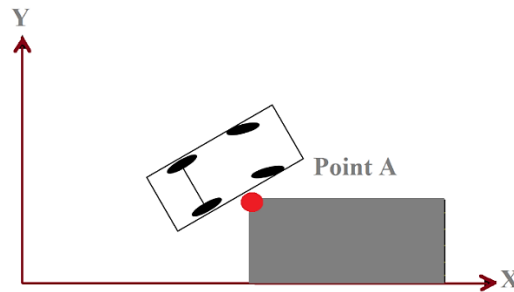


Fig. 2. Vehicle Parking Procedure to Avoid Obstacles

2.4. Petri Nets

For novice drivers, determining the appropriate parking space is usually challenging. In most cases, drivers make decisions and park instinctively. Nevertheless, the parking process is a discrete action that involves moving forward and backward. Thus, different actions are necessary to find an appropriate path and an optimal solution. Petri nets are recognized as a potential tool for collision avoidance and path control. This study employs a novel method combining Petri nets and the PFA algorithm to solve the problem. The Petri net flowchart is illustrated in Figure 3, where the entire parking space is divided into nine sub-regions P1-P9.

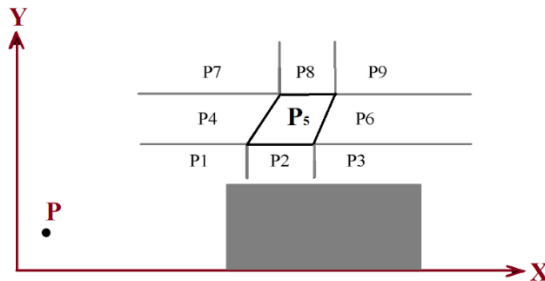


Fig. 3. Description of Paths Taken for Parking the Vehicle

2.5. PFA Optimization Algorithm

The PFA (Path Finding Algorithm) is an efficient algorithm for solving complex optimization problems, recently introduced by Yapici and Cetinkaya in 2019. This algorithm mimics the hierarchical command of creatures in finding a target. In this algorithm, there is a leader that the rest of the members follow.

Each member in this algorithm has a position in a 2D, 3D, or d-dimensional space. If a member of the population is in a promising position at any moment, that member is selected as the group leader, known as the Pathfinder (PF). The position vector of the PF is updated according to Equation (5).

$$x(t + \Delta t) = x(t) \cdot n + f_i + f_p + \varepsilon \tag{5}$$

$$x_p(t + \Delta t) = x_p(t) + \Delta x + A \tag{6}$$

Considering the relationship where t is time, n is a unit vector, f_i is the pairwise interaction effect for neighbors, and f_p is the global force dependent on global optimum or PF position, and ε is the vibration vector. $x_p(t)$ is the position vector of PF, Δx is the distance that PF considers for moving from one point to another, and A is the oscillation rate vector. Given practical issues and the practical use of the previously stated equations in optimization problems, relations 7 to 10 are provided.

$$x_i^{K+1} = x_i^K + R_1 \cdot (x_j^K - x_i^K) + R_2 \cdot (x_p^K - x_i^K) + \varepsilon \tag{7}$$

$$R_1 = \alpha r_1, R_2 = \beta r_2 \tag{8}$$

$$\varepsilon = \left(1 - \frac{K}{K_{max}}\right) \cdot u_1 \cdot D_{ij} \quad , D_{ij} = \|x_i - x_j\| \tag{9}$$

$$x_p^{K+1} = x_p^K + 2r_3 \cdot (x_p^K - x_i^{K-1}) + A \tag{10}$$

In these equations, K is the current iteration number of the algorithm, x_i^K is the position vector of the i -th member, x_j^K is the position vector of the j -th member, R_1 and R_2 are random vectors generated according to relation 11, and ε ensures random and multidirectional movements for oscillation, generated according to relation 12. Here, r_1 and r_2 provide random movement and are chosen as random numbers with a uniform distribution in the range $[0,1]$. The coefficient α indicates the intensity of the displacement of each member towards its neighbor, and β is the absorption coefficient that adjusts the random distance to maintain population impurity with the leader. These two parameters are randomly selected in the range $[1,2]$ throughout the algorithm iterations. u_1 is a random vector in the range $[-1,1]$, and D_{ij} is the distance between two group members. K_{max} is the maximum number of algorithm iterations. The position of PF is updated using relation 13, where r_3 is a random vector with a uniform distribution in the range $[0,1]$, and A ensures random and multidirectional movements for the members, calculated according to relation 16.

$$A = u_2 \cdot e^{\frac{-2K}{K_{max}}} \tag{11}$$

Here, u_2 is a random vector with a uniform distribution in the range $[-1,1]$.

2.6. Optimal Path Planning with the PFA Algorithm

In this section, the PFA algorithm is used to determine an optimal parking path. In the proposed method, an additional factor, the number of gear shifts, is considered. Generally, a park requiring more than two gear shifts is deemed inefficient.

Distance: In the vehicle parking process, the relative position of the parking spot to the vehicle is a very important factor. Therefore, attention to forward and backward movements is crucial when measuring distance.

Curvature: The curvature of a vehicle directly affects the steering angle of the wheels. The maximum value usually does not exceed 45 degrees. High curvature for a moving vehicle can cause physical discomfort or even dizziness in passengers.

Number of Gear Shifts: Generally, considering the parking process, this method requires more than just steering the vehicle to the desired parking spot. The number of gear shifts must be considered as another important factor. More gear shifts increase the overall distance. Given the previously identified performance indicators, a suitable function for the optimization problem can be designed.

$$\text{Cost Function} = \frac{1}{W_1 \frac{D}{\|D\|_{\max}} + W_2 \frac{Ng}{Ng_{\max}} + W_3 \frac{\sum_{i=1}^n \phi_i}{\|\sum_{i=1}^n \phi_i\|_{\max}}} \quad (12)$$

In the above relation, w_i are scalar weighting functions, D is the total distance traveled, Ng is the number of gear shifts, and ϕ_i is the steering angle. The optimal parking path design for the vehicle shown in Figure 4 is summarized in the following steps. The initial position (x_0, y_0) is randomly selected.

Step 1: Randomly search for point (x_p, y_p) in the P5 area.

Step 2: Apply these points and the initial position to the fifth-order equation and ensure that the curvature k does not exceed the allowable maximum value. If this condition is met, the vehicle is allowed to move, and path 1 is generated. If not, straight movement of the vehicle is impossible. Thus, the vehicle must be directed to another point that can satisfy the maximum curvature condition.

Step 3: Apply (x_p, y_p) and (x_G, y_G) to compute the path.

Step 4: Calculate the number of gear shifts. For example, the vehicle moves backward, forward, and then backward, resulting in three gear shifts.

Step 5: Calculate the change in ϕ .

Step 6: Calculate path conformity and select better chromosomes.

Step 7: If termination conditions or an optimal solution are reached, the process ends. Otherwise, repeat steps 2-7.

3. DESIGNING A FUZZY LOGIC SYSTEM

Assume the target position is defined by coordinates (x_G, y_G) , the current vehicle position by (x, y) , the vehicle direction θ_c , and the horizontal angle between the current and target coordinates θ_G . Therefore, given this target, we have:

$$\theta_G = \tan^{-1} \left(\frac{y_G - y}{x_G - x} \right) \quad (13)$$

$$d_e = \sqrt{(x_G - x)^2 + (y_G - y)^2} \quad (14)$$

$$u_1 = \theta_G - \theta_c \quad (15)$$

$$u_2 = \theta_f - \theta_G \quad (16)$$

In the above relations, d_e is the distance between the vehicle and the target. In this study, a two-stage fuzzy logic system is proposed, as shown in Figure 7, to control the front wheels' steering angle for parking. The input variables of the first control stage are u_1 and u_2 . The output variable is θ_e . The proposed fuzzy system rules are listed in Tables 2 and 3.

θ_e and d_e are defined as input variables for the second stage of the fuzzy logic system. d_e ranges from $[0, 100]$ meters, and θ_e ranges from $[-180, 180]$ degrees. The output variable is the steering angle ϕ . Considering the vehicle's actual status, the range of ϕ is $[-45, 45]$ degrees. The goal is for d_e , u_1 and u_2 to all converge to zero as $t \rightarrow \infty$.

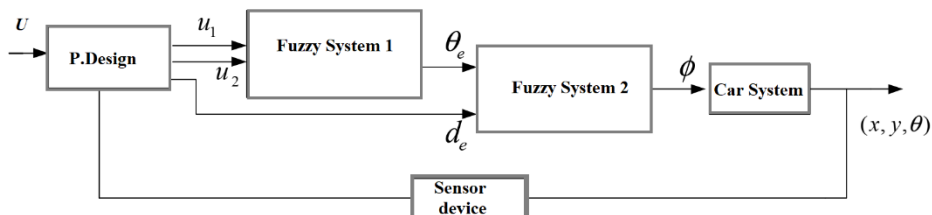


Fig. 4. Block Diagram of a Two-Stage Fuzzy Logic for Vehicle Parking

Table 2. Fuzzy Rules for Steering Angle Adjustment

θ_e		u_1				
		NB	NS	ZE	PS	PB
u_2	NB	NM	NM	NM	NB	NB
	NM	NS	NM	NM	NM	NB
	NS	PM	PS	NS	NM	NM
	ZE	PM	PS	AE	NS	NM
	PS	PB	PS	PS	NS	NM
	PM	PB	PB	PM	PM	PS
	PB	PB	PB	PB	PB	PM

Table 3. Fuzzy Rules for Movement Adjustment

φ		d_e				
		VS	S	M	B	VB
θ_e	NB	PM	PM	PM	PB	PB
	NS	PS	PS	PM	PM	PB
	ZE	PS	ZE	ZE	ZE	ZE
	PS	NS	NS	NM	NM	NB
	PM	NM	NM	NM	NB	NB
	PB	NM	NM	NM	NB	NB

4. SIMULATION RESULTS

The simulation results presented here are based on a vehicle with a length of 4.825 meters, a width of 1.82 meters, $l = 2.755$ meters, and a maximum $\varphi = 45$. The parking space has dimensions of 6.5 meters by 2.5 meters. In the first scenario, the vehicle has an initial position of (20,30,-50°) and a target point of (50,70,60°). The resulting paths from the simulation optimized by Figure 8 are shown below.

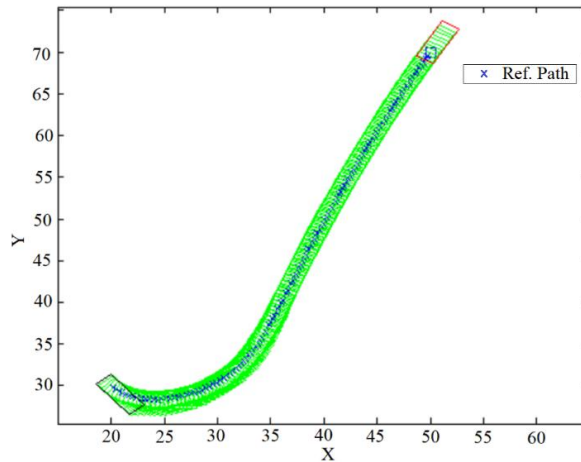


Fig. 5. Path Traveled by the Vehicle

5. CONCLUSION

In this paper, a comprehensive automatic parking mechanism for vehicles is introduced and used to solve the parallel parking problem. The solution involves a global path planner based on Petri networks, the PFA algorithm, and a fuzzy logic system for path tracking. The advantage of the proposed method is that it creates an efficient parking path. The simulation results demonstrate the effectiveness of the algorithms used.

Transparency Statement

The data supporting this study are available upon reasonable request to the corresponding author, subject to ethical and confidentiality considerations.

Acknowledgments

We would like to express our gratitude to all individuals who contributed to this project.

Declaration of Interest

The authors declare that they have no competing interests.

Funding

This research received no specific grant from any funding agency, commercial, or not-for-profit sectors.

REFERENCES

- [1] Millard-Ball, A. (2019). The autonomous vehicle parking problem. *Transport Policy*, 75, 99-108. <https://doi.org/10.1016/j.tranpol.2019.01.003>
- [2] Lam, A. Y., James, J. Q., Hou, Y., & Li, V. O. (2017). Coordinated autonomous vehicle parking for vehicle-to-grid services: Formulation and distributed algorithm. *IEEE Transactions on Smart Grid*, 9(5), 4356-4366. <https://doi.org/10.1109/TSG.2017.2655299>
- [3] Li, B., & Shao, Z. (2015). A unified motion planning method for parking an autonomous vehicle in the presence of irregularly placed obstacles. *Knowledge-Based Systems*, 86, 11-20. <https://doi.org/10.1016/j.knosys.2015.04.016>
- [4] Zhang, X., Liu, W., Waller, S. T., & Yin, Y. (2019). Modelling and managing the integrated morning-evening commuting and parking patterns under the fully autonomous vehicle environment. *Transportation Research Part B: Methodological*, 128, 380-407. <https://doi.org/10.1016/j.trb.2019.08.010>
- [5] Yan, H., Kockelman, K. M., & Gurusurthy, K. M. (2020). Shared autonomous vehicle fleet performance: Impacts of trip densities and parking limitations. *Transportation Research Part D: Transport and Environment*, 89, 102577. <https://doi.org/10.1016/j.trd.2020.102577>
- [6] Litman, T. (2020). Autonomous vehicle implementation predictions: Implications for transport planning.
- [7] Thomas, D., & Kooor, B. C. (2018). A genetic algorithm approach to autonomous smart vehicle parking system. *Procedia Computer Science*, 125, 68-76. <https://doi.org/10.1016/j.procs.2017.12.011>
- [8] Nourinejad, M., Bahrami, S., & Roorda, M. J. (2018). Designing parking facilities for autonomous vehicles. *Transportation Research Part B: Methodological*, 109, 110-127. <https://doi.org/10.1016/j.trb.2017.12.017>
- [9] Wiseman, Y. (2017). Remote parking for autonomous vehicles. *International Journal of Hybrid Information Technology*, 10(1), 313-324. <https://doi.org/10.14257/ijhit.2017.10.1.27>
- [10] Lam, A. Y., James, J. Q., Hou, Y., & Li, V. O. (2016). Coordinated autonomous vehicle parking for vehicle-to-grid services. In *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)* (pp. 284-289). IEEE. <https://doi.org/10.1109/SmartGridComm.2016.7778775>
- [11] Min, K. W., & Choi, J. D. (2015). Design and implementation of an intelligent vehicle system for autonomous valet parking service. In *2015 10th Asian Control Conference (ASCC)* (pp. 1-6). IEEE.
- [12] Klemm, S., Essinger, M., Oberländer, J., Zofka, M. R., Kuhnt, F., Weber, M., ... & Zöllner, J. M. (2016).

Autonomous multi-story navigation for valet parking. In 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC) (pp. 1126-1133). IEEE. <https://doi.org/10.1109/ITSC.2016.7795698>

- [13] Chirca, M., Chapuis, R., & Lenain, R. (2015). Autonomous valet parking system architecture. In 2015 IEEE 18th International Conference on Intelligent Transportation Systems (pp. 2619-2624). IEEE. <https://doi.org/10.1109/ITSC.2015.421>
- [14] Han, S. J., & Choi, J. (2015). Parking space recognition for autonomous valet parking using height and salient-line probability maps. *ETRI Journal*, 37(6), 1220-1230. <https://doi.org/10.4218/etrij.15.0114.0112>
- [15] Notomista, G., & Botsch, M. (2015). Maneuver segmentation for autonomous parking based on ensemble learning. In 2015 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8). IEEE. <https://doi.org/10.1109/IJCNN.2015.7280546>
- [16] Notomista, G., & Botsch, M. (2017). A machine learning approach for the segmentation of driving maneuvers and its application in autonomous parking. *Journal of Artificial Intelligence and Soft Computing Research*, 7. <https://doi.org/10.1515/jaiscr-2017-0017>
- [17] Broggi, A., Cardarelli, E., Cattani, S., Medici, P., & Sabbatelli, M. (2014). Vehicle detection for autonomous parking using a soft-cascade AdaBoost classifier. In 2014 IEEE Intelligent Vehicles Symposium Proceedings (pp. 912-917). IEEE. <https://doi.org/10.1109/IVS.2014.6856490>
- [18] Ni, J., Lin, X., & Shen, X. (2019). Toward privacy-preserving valet parking in autonomous driving era. *IEEE Transactions on Vehicular Technology*, 68(3), 2893-2905. <https://doi.org/10.1109/TVT.2019.2894720>
- [19] Bischoff, J., Maciejewski, M., Schlenther, T., & Nagel, K. (2018). Autonomous vehicles and their impact on parking search. *IEEE Intelligent Transportation Systems Magazine*, 11(4), 19-27. <https://doi.org/10.1109/MITS.2018.2876566>