Optimal Path Planning based on Initial Area Classification for Parallel Parking

by

Kaushik Kumar Gujarathi

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2022 by the
Graduate Supervisory Committee:

Yan Chen, Chair
Sze Zheng Yong
Yi Ren

ARIZONA STATE UNIVERSITY

August 2022

ABSTRACT

The need for autonomous cars has never been more vital, and for a vehicle to be completely autonomous, multiple components must work together, one of which is the capacity to park at the end of a mission. This thesis project aims to design and execute an automated parking assist system (APAS). Traditional Automated parking assist systems (APAS) may not be effective in some constrained urban parking environments because of the parking space dimension. The thesis proposes a novel four-wheel steering (4-WS) vehicle for automated parallel parking to overcome this kind of challenge. Then, benefiting from the maneuverability enabled by the 4WS system, the feasible initial parking area is vastly expanded from those for the conventional 2WS vehicles. In addition, the expanded initial area is divided into four areas where different paths are planned correspondingly. In the proposed novel APAS first, a suitable parking space is identified through ultra-sonic sensors, which are mounted around the vehicle, and then depending upon the vehicle's initial position, various compact and smooth parallel parking paths are generated. An optimization function is built to get the smoothest (i.e., the smallest steering angle change and the shortest path) parallel parking path. With the full utilization of the 4WS system, the proposed path planning algorithm can allow a larger initial parking area that can be easily tracked by the 4WS vehicles. The proposed APAS for 4WS vehicles makes the automatic parking process in restricted spaces efficient. To verify the feasibility and effectiveness of the proposed APAS, a 4WS vehicle prototype is applied for validation through both simulation and experiment results.

ACKNOWLEDGMENTS

This thesis is dedicated to my advisor, Prof. Yan Chen, whose passion for research and teaching has enlightened me and given me a magnificent positive influence on my future career. Without his guidance, I could not finish my thesis and all the other research work in the lab. I would like to thank the committee members, Prof. Sze Zheng Yong and Prof. Yi Ren, for their time and effort in reviewing my thesis.

I would like to thank my parents for their support, both financially and spiritually. And, thank my brother Kartik for always believing in me and supporting me. Finally, I would like thank my friends who were always there for me no matter what and supported me through this journey.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

CHAPTER 1. INTRODUCTION

1. Background and Motivation

Urban area development or smart city development will cause almost 70% of the world population to move into the cities by 2050 [22], leading to increased vehicles on the road. As of 2019, there are 276 million vehicles on the road, leading to severe traffic congestion, less parking space availability, and more accidents. Several approaches were considered to provide adequate solutions for this problem or at least mitigate its impact on the everyday life of a citizen. The construction of new roads is an example of a supply-related solution that is not cost-effective and does not address the issue because adding capacity frequently leads to new demand creation. Similarly, demand-related strategies like increased public transit use, congestion pricing, and flexible work schedules are hard to implement or receive little support from the general public.

One good direction to overcome these situations due to the development of cities is an intelligent system in the vehicle which can assist the driver with various functions to overcome the stress produced. A large variety of such systems collectively are called advanced driver assist systems (ADAS). ADAS are being developed by various parties, including system developers, car manufacturers, and scientists worldwide. The enhancement of vehicles' traffic dynamics, which increases the effectiveness of the network as a whole, is one of the main factors considered when designing and developing ADAS. The reduction of fuel usage and associated emissions, as well as safety enhancements, are other factors. A few of the ADAS features are adaptive cruise control, collision avoidance system, anti-lock braking system, and automatic parking assist systems (APAS).

APAS are one of the most demanding ADAS features because, with the rapid development of cities, parking spaces have become narrow, causing drivers a heavy burden. Drivers must be attentive and careful while parking their vehicles to prevent them from damaging, making it a difficult and stressful job. Automatic parking is a way to relieve stress while also improving the driver's comfort and safety. To be commercialized, autonomous parking must meet the drivers' expectations for a quick, predictable, and low-demand solution to the tires. There were many kinds of research on APAS through which parking efficiency, driver safety, and driverless vehicles can be effectively improved.

2. Literature Review

APAS are generally classified into two types, i.e., fully- automated parking systems and semi-automated parking systems [1], [2]. A fully APAS controls all the aspects of the vehicle like steering, braking, and accelerating, and it does not require human intervention. In semi-APAS, the system is only in-charge of path generation and tracking, and human drivers take charge of all the controls. In both scenarios, path planning is the main problem. Two types of path planning are considered, i.e., global path planning [3] and local path planning [3]. Global path planning in the system has complete knowledge of the environment, which increases computational complexity and affects the real-time application. Local path planning in the system does not know the environment, and sensors are used to collect data from the environment.

Various local path planning strategies have been proposed for APAS in the past decade. Methods based on fuzzy logic [3], [4], [5] or neural networking [6], or artificial intelligence [7] is used for effective and accurate path planning and also handle uncertainty and model inaccuracies as the system learns from human knowledge and techniques. However, human wisdom cannot be trusted for perfect path planning and requires severe computational effort. Methods based on reference functions [10], the Lyapunov function is used to stabilize the vehicle to the desired final position, i.e., orientation and the position of the vehicle, starting from the initial position [9]. This method highly depends on parameters chosen by the function, which makes it difficult to adjust and can lead to incorrect parking maneuvers [11], [12]. This problem can be partly overcome by integrating fuzzy logic [13] in the procedure of coefficient derivation. However, it is challenging to consider obstacles in tiny spaces [13]. Methods are based on two-phase path planning [8]; in the first phase, a collision-free path between two postures is planned without considering the nonholonomic constraints. In the second phase, the path generated is then approximated to the nonholonomic constraints. However, this method is very complicated when applied in real-time parking as it causes a lot of back-and-forth maneuvers.

A method based on Geometric path planning [2], [11], [12] is the most commonly studied and used method because of its simple path representation and low computational efforts. With the initial position and the parking slot dimensions, a geometric path that consists of multiple segments represented in simple geometric equations is generated [2]. This method evaluates the parking environment, then reversely searches for a path from the desired parking slot to the initial configuration [18]. These segments are connected using circular arcs and lines. In [19], a geometric method with two identical circles is designed for one trial parking which is only possible if the parking space is long enough and cannot be used in narrow spaces. In [20], path planning with only the circle with the minimum radius for the second part of the maneuver is retained, which allows vehicles with a different maximal steering angle for the right direction than for the left direction but requires long parking space. Another method was also proposed for narrow spaces, but it requires an N number of trails to park the vehicle. In [21], a geometrical method based on the grid method was proposed, which uses a polyline path on the grid and then fits a Bezier curve using the shortest path length optimization function, and this method is complex. And it should be noted that these methods rarely focus on the scope of the starting position, which is important information to park successfully. It is challenging to construct a workable path for the automated parking system when starting at an illogical location. Additionally, when the starting position is fixed, these approaches can only generate one path corresponding to one vehicle algorithm, which restricts their flexibility and efficiency in creating viable and effective paths.

3. Problem statement

The aforementioned research accelerated the development and popularization of APAS. However, the existing APAS still has some issues and can be further developed for some complicated situations. For instance, the parking spaces in urban areas are becoming restricted due to city development and space limitations. Using the existing APAS, the vehicle may need to complete the parking task by tracking the planned path consisting of multiple back and forth maneuvers [14][18][19], or even is not able to finish the parking task within limited spaces. However, 4WS vehicles enabled by the steer by wire (SbW) technology can effectively resolve such an issue. 4WS vehicles are the new generation of vehicles where both front and rear wheels are steerable. The advantage of 4WS is that the number of controlling actuators is more than the required actuators for the controllability of the system. This increases the number of control inputs while keeping the workspace constant. As a result, the possible path through which the vehicle moves are not unique, which means it provides an optimized cost function [15]. Increasing the control variable of the front and rear wheel allows it to move easily in a cluttered environment or achieve a small turning radius at low speed and high speed with stability [16], [17].

This paper proposes a novel APAS for 4WS vehicles using only low-cost ultrasonic sensors for parallel parking. Benefiting from the intrinsic advantage of the 4WS technology, the proposed APAS can allow a larger vehicle initial area and generate the most straightforward path. In some restricted areas, the proposed APAS is especially effective in handling the parking task with higher vehicle maneuverability and a smoothest path (i.e., minimum steering angle change from arc to another) to generate using optimization functions.

4. Outline

The remainder of this thesis is organized as follows. In chapter 2, the overview of how parallel parking works is explained. In Chapter 3, the initial area classification and the path generation of 4WS vehicles for parallel parking are first proposed. Then in chapter 4, a 4WS reverse pure pursuit path tracking control is designed. In Chapter 5, the implementation of the algorithm in matlab/simulink and the simulation results were discussed. In Chapter 6, experimental setup and the experimental results are presented and discussed, and in Chapter 7, the conclusive remarks are given.

Chapter 2: Overview of Model

This section will give a short overview of the different components of APAS. A visualization is shown below Figure 1. There are four various subsystems in the APAS:

- Empty Space Detection
- Sensors
- Path Planning
- Tracking control

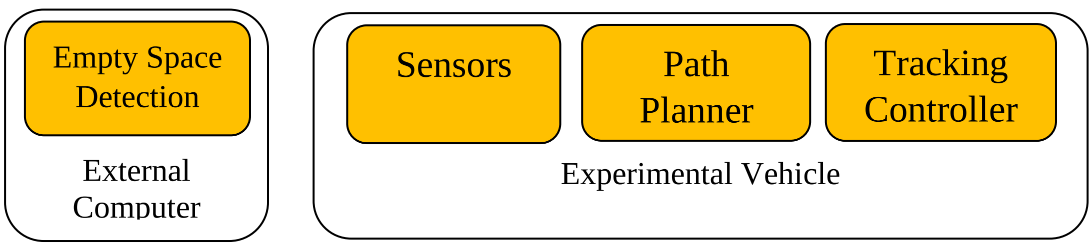

Figure 1. Layout of the process

The Empty Space Detection runs at an external computer connected to the experimental vehicle. The Path Planner, Tracking Controller, and signal processing of the sensor data is located on the experimental vehicle computer.

1. Empty Space detection

The empty space detection was developed in the previous paper for the geometric reverse parking method developed in the lab. The same technology is used in this thesis to detect empty parking space. Generally, the empty space identification can be achieved by four main methods, namely the intelligent transportation system (ITS) [23][24], sensor fusion-based identification system [25]-[27], vision-based identification system [28], and ultrasonic sensor-based identification system [29]-[31]. The ITS provides vehicle guidance to the parking space based on the digitalized parking environment. However, the installation of ITS docking equipment is expensive and not widely available. Unlike the ITS, the sensor fusion-based empty space identification system, such as the around view monitor system (AVM) [27], provides a high accuracy environment detection around the vehicle. However, systems like AVM are usually expensive for common passenger cars.

With a lower cost, the vision-based system can provide high accuracy detection results with advanced image processing technology. However, the accessible camera vision is sensitive to the light and weather in the parking environment. Having no defects of the aforementioned systems, the ultrasonic sensor detection system can be developed with low-cost and high detection accuracy [30]. However, the drawback of using the ultrasonic sensor detection system is that the vehicles or the obstacles are needed at least on one side of the empty parking space. Therefore, the ultrasonic sensor is normally used together with the camera as a completed empty space detection system at a reasonable cost.

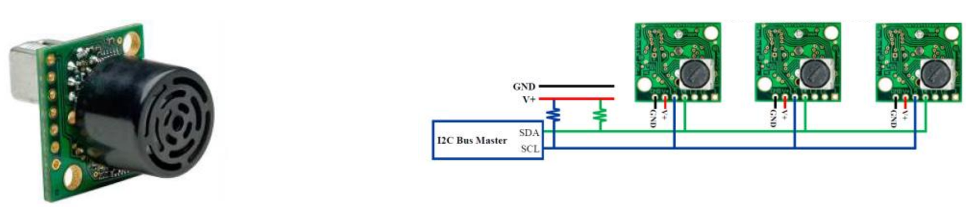

Figure 2. Ultra-sonic sensors

Parameter	Value	Unit
Horizontal FOV (field of view)	60	degree
Vertical FOV	60	degree
Detection range	20-765	cm
Reading frequency	40	Hz
Accuracy	±1	Cm

Table 1. Ultra-sonic Sensor Parameters

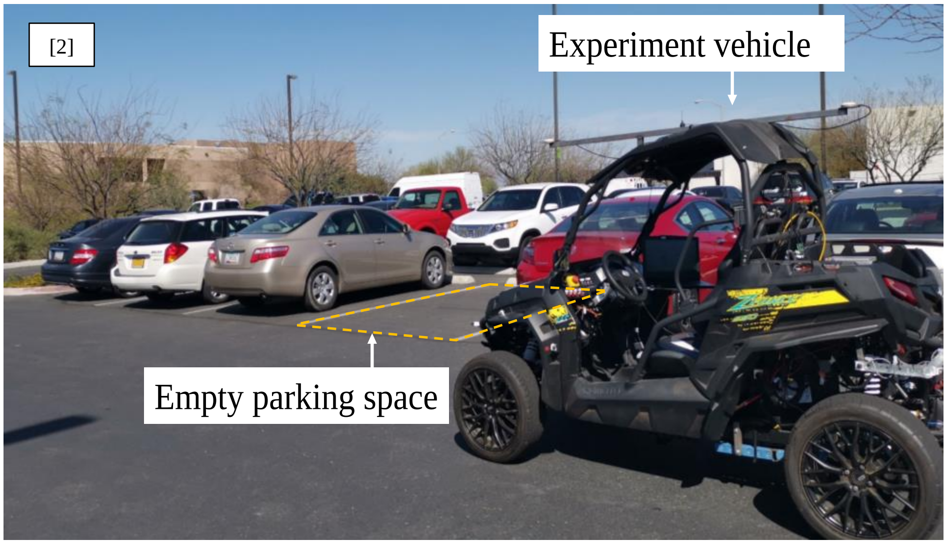

Figure 3. Empty space detection setup

2. Sensors

The sensors used to estimate the state of the experimental vehicle and the surroundings consist of a high-performance Global positioning system (GPS) sensor.

Global positioning and heading are available from the Real Time Kinematic-GPS system (RTK-GPS), also known as Carrier-Phase Enhancement. Additionally, to the receiver, the RTK-GPS requires a base station transmitter that determines its position according to its defined place. The receiver uses the carrier phase from the base station transmitter to determine the receiver's position with centimeter accuracy. This RTK-GPS system provides the heading relative to the north geographical pole with high precision, roll/pitch, acceleration, slip angle, and velocity of the experimental vehicle can also be determined using this system. The position and the heading angle are filtered, and the position is translated to the point of the center of gravity of the experimental vehicle. There won't be any further signal processing needed for the system in this thesis.

Figure 4. OxTS GPS system

3. Planning a Path:

The path planner algorithm in this thesis is a geometric method used to generate a path between two points. The path planner comes into action after the empty space detection system when the vehicle stops after finding an empty space to move into the available parking spot. Inputs to the path planner are the coordinates given by the RTK-

GPS and the heading angle of the vehicle. Given the inputs, the path planner computes a path from the vehicle's current position to the desired parking spot using geometrical methods. A path tracking controller is then used to execute the planned maneuver by stabilizing the vehicle to the generated path. The path planner that does not consider the vehicle's size cannot guarantee the vehicle's safety and the safety of the people inside it.

The Optimized cost function used to generate the path from the initial position to the final position will give the smoothest path curvature in terms of steering angle change from the right side to the left side with the shortest path that can be generated for the vehicle to make a parallel parking maneuver. The geometric path planning method is typically represented by a line and arc segments, which compose the shortest path from one position to another on the overall parking path. The combination of the multiple path segments should also follow specific rules, such as minimizing the path length or the total parking time while satisfying the collision-free conditions. The advantage of the geometric path planning method is the simplicity in both path representation and computational effort. Therefore, this method was commonly adopted in many real applications. The output is a possible path with information about the desired position, curvature in each point along the path, and the heading angle.

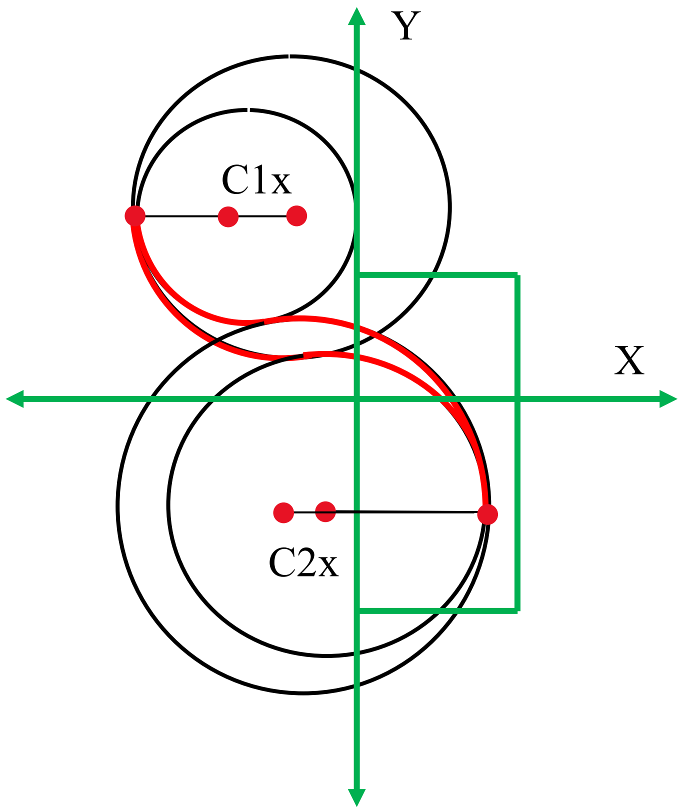

Figure 5. Parallel parking path

Above is an example of the path planning for parallel parking with the geometric method, the path is marked in red, connecting the initial point where the empty space is detected and the final position where the final desired position is. The optimization function finds the smoothest and the shortest path from all the paths.

Chapter 3: Geometric Path Planning

This paper involves the geometrical equations for 2-wheel steering (2WS) and 4-wheel steering (4WS) vehicles. We know for a fact that for a vehicle to make a parallel parking maneuver into the available parking spot, the vehicle has to stop at a particular area which will give the human a definite chance of parking their vehicle in the empty space with no collision. Still, this chance becomes thin when the parking spots are narrow, and there is not enough space to make parallel parking maneuvers. This is where automatic parking comes into the act, which helps humans do this hectic parking maneuver with ease and collision free. With the vehicle using 4-wheel steering (4-WS) for parking its vehicle, there need not be enough space for the parking maneuvers. The vehicle can start the parallel parking maneuvers without getting into a particular spot.

Initial area classification:

The development of a geometrical path planning method for 2-WS and 4-WS is discussed in this paper, which involves different areas from where the vehicle can start its parallel parking maneuvers to reach its desired destination. The path planning algorithm involves the geometrical equations for both 2-WS and 4-WS, and the first step in the parking path planning is the classification of the initial area. Then according to the initial area, the geometric path planning algorithm generates different paths.

As shown in the Figure 6 below, the initial area that the vehicle could possibly stop is classified as four areas based on the following considerations.

1. The outer boundaries of the whole initial area are determined by the lengths and widths of the vehicle, empty space, and road.
2. The points A, B, and C are determined using the minimum turning radius of the vehicle.
3. Points D and E are determined using the maximum steering angles of the vehicle.

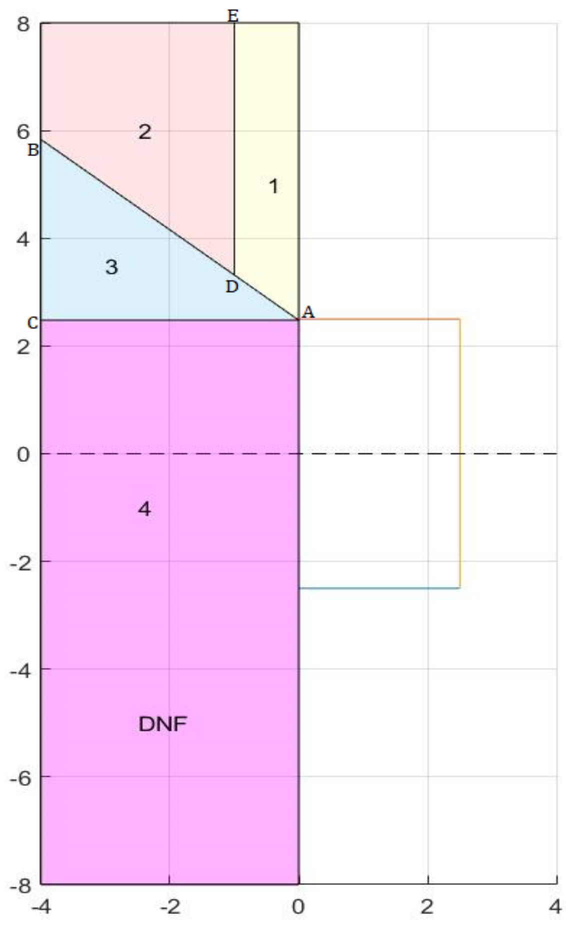

Figure 6. Area classification map

Note: In the above Figure 6 and other similar figures included in this thesis, the vehicle is heading upward (positive Y).

In the above Figure 6, Area 2 denotes where a simple path can be generated and tracked for the front 2-WS vehicle because it is the common area where the vehicle has its initial stop position. Area 1, and 3 denote the areas where a simple path can be generated using a 4-WS. A 4-WS vehicle can also be used for area 2, with even better and simpler path generation and tracking. DNF (Do Not find) is an area where no simple parking paths can be generated.

In this thesis, the experimental vehicle we use to do the experiments can use its front wheel and rear wheel for steering control. The main factor that affects the difference in the initial classified areas between 4-WS and 2-WS is the vehicle's turning radius. The minimum turning radius of 4-WS and 2-WS are calculated below, respectively

The turning radius a vehicle can turn is given by

$$R_{4WS} = \frac{L}{\tan(\delta_f) + \tan(\delta_r)} \qquad (1)$$

$$R_{2WS} = \frac{L}{\tan(\delta_f)} \qquad (2)$$

Were,

L = wheel base length, δ_f = front steering angle, & δ_r = rear steering angle

With the additional rear-wheel steering, the turning radii of the 4-WS vehicles are usually much smaller compared to the 2-WS vehicles. In addition, by steering the front and the rear wheels in the same direction with the same angle, a 4-WS vehicle can achieve the parallel shifting maneuver without changing the vehicle heading angle. This feature in the vehicle dramatically enhances the maneuverability of the 4-WS vehicle in narrow spaces.

With the advantages mentioned above of 4-WS vehicles, when the vehicle stops in area-1, the path planner generates a path that uses this parallel shifting maneuver to bring the vehicle into area-3 and then uses 4-WS to complete the parallel parking maneuver into the desired empty spot.

The path planning algorithm is illustrated in Figure 7, where the final output is a continuous $[X, Y]$ coordinate of the planned path.

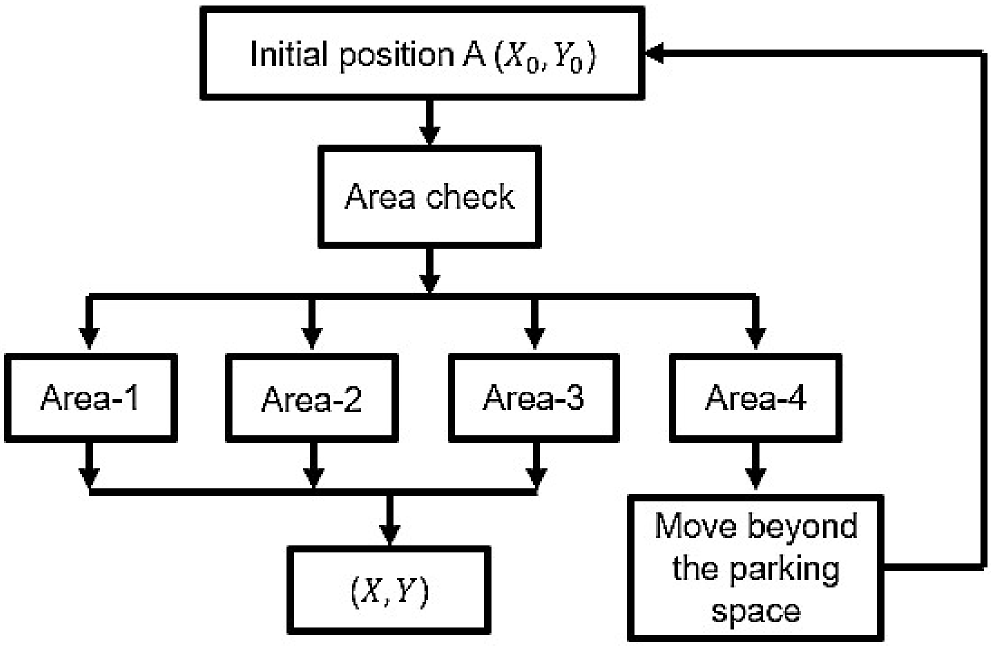

Figure 7. Parallel parking path planning layout

The planned path consists of two arcs whose center points can be given as (C_{1x}, C_{1y}) and (C_{2x}, C_{2y}) respectively. To make parallel parking from the initial position where the vehicle stopped after finding the empty spot to park, the vehicle uses its maximum steering angle and turns the steering wheels to its right side and maneuvers the vehicle to the center of the path, and then turns the steering wheels to a maximum steering angle to the other side and continues its maneuver to put the vehicle in the desired final position.

The algorithm developed in this thesis uses the radius of the circles to plan a path from the initial position to the final position. The assumption we consider in this thesis is that the final position where the vehicle is designed to park is fixed in a way where it avoids collision with the curb and the vehicles that might be parked near the empty space. The advantage of fixing the final parking position is that the center of the arcs used to generate the path for parallel parking can change its position and which gives us a chance to optimize the selected path in a way where the cost function chooses the path that has the smoothest change in steering angle from right side to the left side and a chance to select the shortest path length from the initial position to the final position.

The Figure 8, below illustrates the idea of the geometric path planning method, which is discussed above.

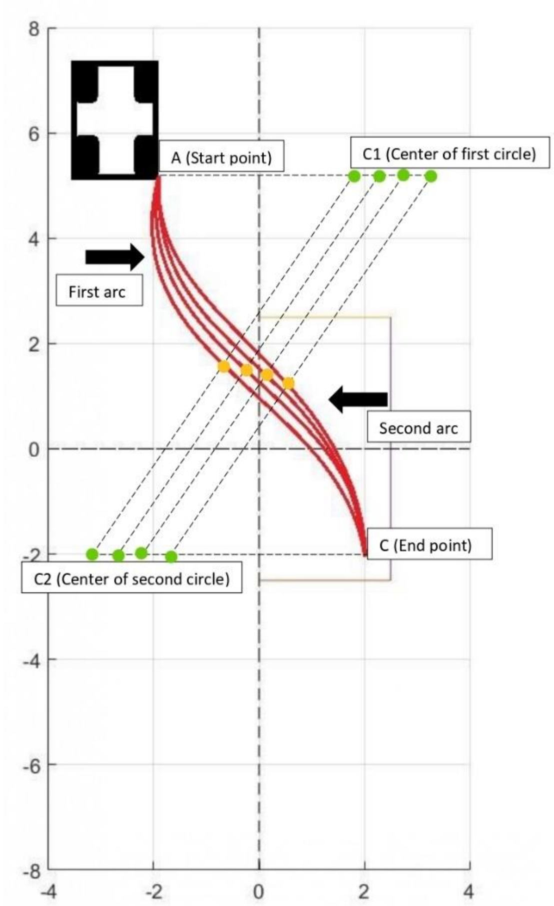

Figure 8. Parallel parking

The experimental vehicle we use in this thesis has 4-WS system, which has OxTS GPS with +-1 cm accuracy and ultra-sonic sensors fixed around the vehicle, which helps identify empty space for vehicle parking. Some basic vehicle parameters used in this APAS and experimental setups are listed in the below table

Symbol	Parameters	Values
m_v	Vehicle mass	1270 kg
g	Gravity constant	9.81 m/s^2
I_z	Yaw inertia	1536.7 kg·m^2
μ	Tire-road friction coefficient	0.85
L	Wheelbase	2.08 m
W_{car}	Width	1.5 m

Table 2. Vehicle Parameters

Path generation:

This thesis involves a geometrical path planning method for parallel parking for 2-WS and 4-WS vehicles. The vehicle uses ultra-sonic sensors located on every side of the vehicle to detect empty space accurately. This method of empty parking space detection is efficient and also cheap. After the vehicle has detected the empty parking space in the parking lot, it comes to a halt. This position becomes the initial position where the vehicle starts its parallel parking process. The initial point is considered the center of gravity of the vehicle. The initial position where the vehicle stops can be given as

$$A = [X_0, Y_0];$$

To end the parallel parking maneuver, the vehicle has to stop at a final desired position, and this final desired position has to be a place where it is far from the collision with other vehicles that are parked beside the selected parking path and should not collide with the curb of the parking spot. To maintain the above conditions, the final desired spot is manually set and assumed to be the final position. The final point is considered at the center of gravity of the vehicle. The final desired point where the vehicle stops its parking maneuver is given as

$$\mathrm{D} = [\mathrm{X}_3, \mathrm{Y}_3] = [2.0, -2.3]; \text{ from the origin}$$

The experimental vehicle used in the thesis has a 4-WS system, and the steering angles of both front and the rear steering wheel have limitations that will limit the turning radii of the vehicle and will affect the parking maneuver. The geometric path planning algorithm developed was designed in a way that it considers its steering angle limitations and plans a path in such a way that the vehicle reaches its destination in the desired manner.

The maximum steering angle of the vehicle is

$$\begin{aligned} -\delta_{\mathrm{fmax}} \le \delta_{\mathrm{f}} \le \delta_{\mathrm{fmax}} \\ -\delta_{\mathrm{rmax}} \le \delta_{\mathrm{r}} \le \delta_{\mathrm{rmax}} \end{aligned} \tag{3}$$

The $\delta_{\mathrm{f\,max}}$ and $\delta_{\mathrm{r\,max}}$ The experimental vehicle in this thesis is identified as 40 degrees and 30 degrees, respectively. In addition, the steering rate is also bounded by the maximum steering rate of 5 deg/s with the consideration of the steering actuation system characteristics. And the turning radii of both steering wheel systems can be given as

$$\mathrm{R}_{4\mathrm{WS}} = \frac{\mathrm{L}}{\tan(\delta_{\mathrm{f}}) + \tan(\delta_{\mathrm{r}})} \tag{4}$$

$$R_{2WS} = \frac{L}{\tan(\delta_f)} \quad (5)$$

For a vehicle to parallelly park, there needs to be two arcs tangent to each other and which are formed by the vehicle taking a maximum steering angle to its right side and then transitioning to the maximum steering angle to the other side in the middle of the path and this makes the vehicle maneuver into the parking spot using parallel parking method. The algorithm generates two circles tangent to which other connecting the initial point and the final desired point, making it a path for the parallel parking maneuver. The circles can have varying radii, allowing them to generate a path from any given initial position.

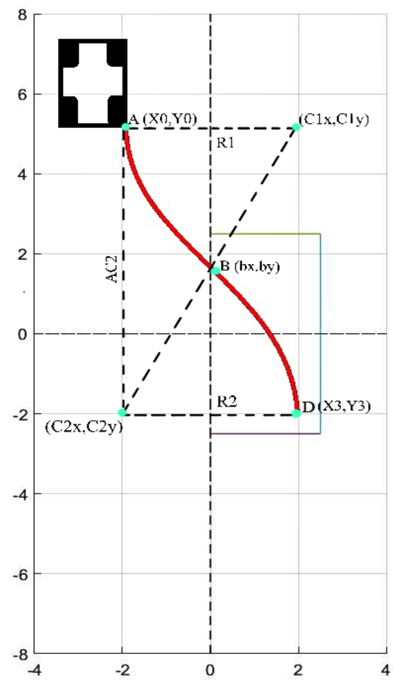

Figure 9. Geometric relationship

The algorithm developed in this thesis uses a varying center of the second circle in a horizontal X-axis to make a circle with a radius that connects the final position point. So, we can say that the C_{2x} has multiple points, which will lead to having multiple C_{1x} points which mean there will be numerous circles tangent to each other. Generally, the radius of the second circle can be defined by the following equation.

The radius of the second circle can be given as

$$R_2 = X_3 - C_{2x} \quad (6)$$

X_3 = the X coordinate of the final position.

C_{2x} = the X coordinate of the center of the second circle.

As discussed above, we know that the generated path is a function of C_{2x} which means the path change with the change in C_{2x} values. C_{2x} is a function of the maximum steering radius:

$$f(R) \leq C_{2x} \leq f(-R)$$

$$X_3 - R \leq C_{2x} \leq -R \quad (7)$$

So, the center point of the second circle can be given as

$$C_2 = [C_{2x}, C_{2y}];$$

Now that we know how to find the center of the second circle, we can find the center of the first circle. To find out the radius of the first circle, consider triangle AC_1C_2. The distance from point A to the center of the second circle can be calculated as

$$dAC_2 = \sqrt{(X_0 - C_{2x})^2 + \left(Y_0 - C_{2y}\right)^2} \qquad (8)$$

because the C_{2x} can vary in the above (5) range, there are two conditions in the function where $< X_0 < C_{2x}$ & $X_0 > C_{2x}$, so the angle between R_1 and AC_2 can be given as

$$\alpha = \begin{cases} \sin^{-1}\left(\frac{Y_0 - C_{2y}}{dAC_2}\right); & X_0 < C_{2x} \\ \pi - \sin^{-1}\left(\frac{Y_0 - C_{2y}}{dAC_2}\right); & X_0 > C_{2x} \end{cases} \qquad (9)$$

given the length of the other two sides and the magnitude of the angle between them, we can find the radius of the first circle R_1 using the cosine rule.

The radius of the first circle can be found using

$$R_1 = \begin{cases} \frac{dAC_2^2 - R_2^2}{2R_2 + 2dAC_2\cos(\alpha)}; & X_0 < C_{2x}, X_0 > C_{2x} \\ \frac{dAC_2^2 - R_2^2}{2R_2}; & X_0 = C_{2x} \end{cases} \qquad (10)$$

Now that we found the equation for the radius of the first circle, we can see the center of the first circle from the above Figure 9, which shows us where the first circle is; according to the Figure, the center of the first circle can be given as

$$C_{1x} = X_0 + R_1;$$

$$C_{1y} = Y_0;$$

$$C_1 = [C_{1x}, C_{1y}]; \quad (11)$$

From the above equations, we can have two circles tangent to each other in a way that forms a path that leads the vehicle into the parking space. As C_{2x} varies in its function, there will be multiple paths that lead the vehicle to the parking space.

Optimization Cost Function:

From the above-discussed equations, there will be many paths generated for every C_{2x} in the range that will be determined by (5). All these paths for every C_{2x} Will lead the vehicle to the empty parking spot, but only one of these paths will have the smoothest transition from the right steering angle to the left steering angle, which will help reduce the effects on the tire wearing and will park the vehicle most efficiently. And all other paths may have a steering angle that is unachievable by the vehicle or does not have a smooth trajectory. So, we propose a cost function that will lead the vehicle to its final parking space with minimum steering angle change and thus have a smooth trajectory.

The cost function can be given as

$$f(C_{2x}) = |\delta_1| + |\delta_2| \quad (12)$$

δ_1 = steering angle of the first curve

δ_2 = steering angle of the second curve

The steering angle of the vehicle during the first circle can be given as

$$\delta_1 = \tan^{-1}\left(\frac{L}{R_1}\right) \quad (13)$$

The steering angle of the vehicle during the second circle can be given by

$$\delta_2 = \tan^{-1}\left(\frac{L}{R_2}\right) \tag{14}$$

When we use the above cost function, the path planner generates a path from the initial stop position of the vehicle to the final desired position of the vehicle in a way where it has the smoothest steering angle change when compared to all other paths that were generated by the path planner which might lead to discomfort for the passengers traveling in the vehicle and also led to the excessive tire wearing situation which is not cost-efficient.

After seeing the above algorithm, we also proposed a cost function to get the minimum path length as a function of the radius of the circles, which will lead it to get the shortest path from every path that has been generated for every C_{2x} in the path planning algorithm. This will lead the vehicle to travel the path in the shortest time when compared to all other paths, but it is found that every path that the path planner is generating C_{2x} has the same path length leading the vehicle into the parking spot. This theory is proved in the lemma below, and it shows that the path length is independent of C_{2x}.

<u>Lemma</u>: The arc length of the path for every path in the constrained C_{2x} will always be independent of C_{2x} i.e., the path length does not change with C_{2x}.

$$L_{arc} = \beta(R_1 + R_2) \tag{15}$$

$\beta =$ Turning angle of the vehicle;

$R_1 =$ radius of the first circle;

$R_2 =$ radius of the second circle;

The turning angle of the vehicle can be given as the angle between AC_2 & C_1C_2 and can be calculated by

$$\beta = \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right) \tag{16}$$

Now, to prove the lemma

$$L_{arc} = \beta(R_1 + R_2)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\left(\frac{dAC_2^2 - R_2^2}{2R_2 + 2dAC_2\cos(\alpha)}\right) + (X_3 - C_{2x})\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\left(\frac{\left(\sqrt{(X_0 - C_{2x})^2 + \left(Y_0 - C_{2y}\right)^2}\right)^2 - (X_3 - C_{2x})^2}{2(X_3 - C_{2x}) + 2\left(\sqrt{(X_0 - C_{2x})^2 + \left(Y_0 - C_{2y}\right)^2}\right)\cos\left(\sin^{-1}\left(\frac{Y_0 - C_{2y}}{dAC_2}\right)\right)}\right) + (X_3 - C_{2x})\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\frac{X_0^2 + {C_{2x}}^2 - 2C_{2x}X_0 + Y_0^2 + {C_{2y}}^2 - 2C_{2y}Y_0 - X_3^2 - {C_{2x}}^2 + 2C_{2x}X_3}{2\sqrt{(C_{2x} - X_0)^2 + \left(C_{2y} - Y_0\right)^2}\left(\sqrt{1 - \left(\frac{C_{2y} - Y_0}{\sqrt{(C_{2x} - X_0)^2 + \left(C_{2y} - Y_0\right)^2}}\right)^2}\right) + 2(X_3 - C_{2x})} + (X_3 - C_{2x})\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\frac{X_0^2 + Y_0^2 + {C_{2y}}^2 - X_3^2 - 2C_{2x}X_0 - 2C_{2y}Y_0 + 2C_{2x}X_3}{2\sqrt{(C_{2x} - X_0)^2 + (C_{2y} - Y_0)^2}\left(\frac{C_{2x} - X_0}{\sqrt{(C_{2x} - X_0)^2 + (C_{2y} - Y_0)^2}}\right) + 2(X_3 - C_{2x})} + (X_3 - C_{2x})\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\frac{X_0^2 + Y_0^2 + {C_{2y}}^2 - X_3^2 - 2C_{2x}X_0 - 2C_{2y}Y_0 + 2C_{2x}X_3}{2(C_{2x} - X_0) + 2(X_3 - C_{2x})} + (X_3 - C_{2x})\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\frac{X_0^2 + Y_0^2 + {C_{2y}}^2 - X_3^2 - 2C_{2x}X_0 - 2C_{2y}Y_0 + 2C_{2x}X_3}{2X_3 - 2X_0} + (X_3 - C_{2x})\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{R_1 + R_2}\right)\left(\frac{X_0^2 + Y_0^2 + {C_{2y}}^2 - X_3^2 - 2C_{2x}X_0 - 2C_{2y}Y_0 + 2C_{2x}X_3 + 2X_3^2 - 2X_3X_0 - 2C_{2x}X_3 + 2C_{2x}X_0}{2X_3 - 2X_0}\right)$$

$$\Rightarrow \sin^{-1}\left(\frac{Y_0 - C_{2y}}{\left(\frac{X_0^2 + Y_0^2 + {C_{2y}}^2 - X_3^2 - 2C_{2y}Y_0 + 2X_3^2 - 2X_3X_0}{2X_3 - 2X_0}\right)}\right)\left(\frac{X_0^2 + Y_0^2 + {C_{2y}}^2 - X_3^2 - 2C_{2y}Y_0 + 2X_3^2 - 2X_3X_0}{2X_3 - 2X_0}\right) \quad (17)$$

The above equation shows that there is no C_{2x} variable in it, which means the theory that we assumed is correct, and we can say that the path length for every path generated by every C_{2x} in its range are the same, and we can conclude that the path length of the path generated is the best path the path planner can generate. Therefore, we can say that the total path length for a path in the constrained C_{2x} is always equal.

The planned path consists of two circles whose centers

$C_1 = [C_{1x}, C_{1y}]$ is defined by

$$X_{C1} = C_{1x} + R_1 \cos(\theta)\,; \qquad (18)$$

$$Y_{C1} = C_{1y} + R_1 \sin(\theta)\,; \qquad (19)$$

and $C_2 = [C_{2x}, C_{2y}]$ is defined by

$$X_{C2} = C_{2x} + R_2 \cos(\theta)\,; \qquad (20)$$

$$Y_{C2} = C_{2y} + R_2 \sin(\theta)\,; \qquad (21)$$

where R_1 and R_2 are the radius of the two circles, and θ is the vehicle's global heading angle.

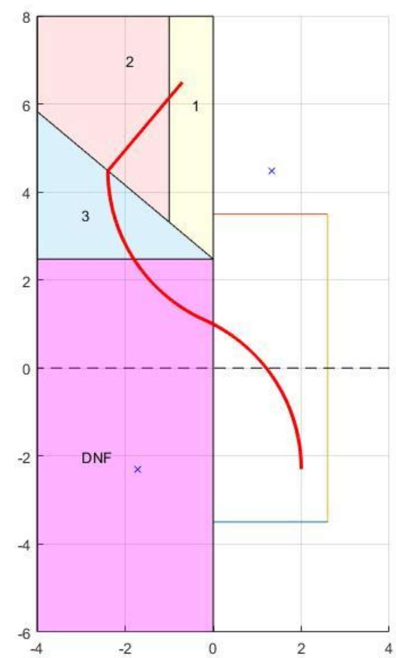

Figure 10. Path planning for Area-1

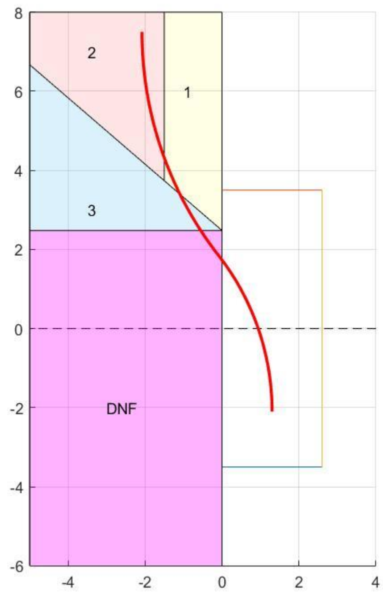

Figure 11. Path planning for Area-2

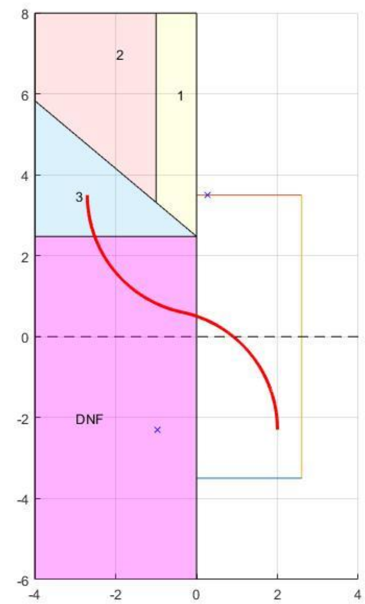

Figure 12. Path planning for Area-3

The above-shown Figures are the results of the geometric path planner. Each Figure shows the vehicle in different defined areas in the parking space, which leads the geometric path planner to design the path according to the vehicle's initial position. Area-1&3 is where the vehicle's initial role is vertically close to the parking space or horizontally relative to the parking space. In these cases, the path planner plans the path to use the 4-WS feature of the experimental vehicle, which gives us an advantage in parking the vehicle in a narrow space. And the case where the vehicle's initial position is in area-2 is the most common scenario in generally parallel parking. The path planner generates a simple path that uses a 2-WS system to complete its parking maneuver.

Chapter 4: Path Tracking Control

Path tracking control used in this thesis is a pure pursuit controller. This controller is a geometrical tracking control method. Pure pursuit is a tracking algorithm calculates the curvature that will move a vehicle from its current position to some goal position. The whole point of the algorithm is to choose a goal position some distance ahead of the vehicle on the path. The name pure pursuit comes from the analogy we use to describe the method. We tend to think of the vehicle as chasing a point on the path some distance ahead - it is pursuing that moving point. That analogy is often used to compare this method to how humans drive. We tend to look some distance before the car and head toward that spot. This look-ahead distance changes as we drive to reflect the twist of the road and vision occlusions.

Path tracking Via a pure pursuit controller

As a 4WS vehicle, its steering kinematics and dynamics differ from that of a front-wheel steering vehicle. So, the path tracking method for the 4WS carrier vehicle should be modified and improved. A pure pursuit algorithm is a geometrical method generally used in front-wheel steering control. It determines a circular arc along which the vehicle drives to the desired path from its current position. The arc is the desired trajectory of the rear axle center of the front wheel steering vehicle. However, in order to apply the pure pursuit algorithm to the 4WS vehicle, we assume the center of mass of the 4WS vehicle lies in the desired trajectory.

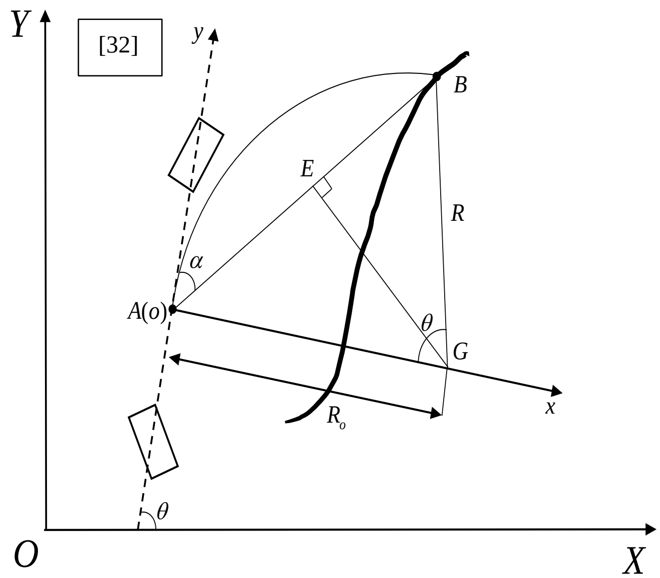

Figure 13. Pure pursuit relationship

A schematic diagram of the path tracking algorithm for the 4WS vehicle is shown in Figure 13. In the world coordinate system (X-O-Y) and vehicle coordinate system (x-o-y), point o represents the mass center of the vehicle, assuming it lies in the center of the carrier vehicle and coincides with point o; The thick solid line represents a virtual digital path, which is generated by fitting curves from the desired GPS road points; Point B represents the preview point determined by the preview distance. (X_0, Y_0) is the coordinate of o in the word coordinate system, (oB) is the coordinate of B in the word coordinate system, and (b_x, b_y) is the coordinate of B in the vehicle coordinate system; θ is the angle between vehicle heading and positive direction of X axle; Ro is the instantaneous steering radius of o; G is the center of arc oB; α is the angle of vehicle heading change from o to B, and θ =2 α. Transform the coordinate of preview point B in word coordinate system to the vehicle coordinate system by

$$\begin{cases} x_b = (X_b - X_o)\sin\left(\theta - \arctan\frac{Y_b - Y_o}{X_b - X_o}\right) \\ y_b = (Y_b - Y_o)\sin\left(\theta - \arctan\frac{Y_b - Y_o}{X_b - X_o}\right) \end{cases} \qquad (22)$$

And R_0 can be calculated by

$$R_0 = \frac{\sqrt{{y_b}^2 + {x_b}^2}}{2\sin\left(arctan\frac{y_b}{x_b}\right)} \qquad (23)$$

Considering $R_{4WS} = R_0$, The desired steering wheel angles can be obtained as

$$\delta_f = \delta_r = \arctan\frac{L\sin\left(\arctan\frac{y_b}{x_b}\right)}{\sqrt{{y_b}^2 + {x_b}^2}} \qquad (24)$$

Where L denotes the length of the vehicle wheelbase, in the 4WS pure pursuit algorithm, the look-ahead distance is the main tuning parameter since the vehicle considers the target point as the next point to reach from the current location. The changing of look-ahead distance can affect the results of path tracking. A small look ahead distance makes the vehicle quickly move towards the path. A larger look ahead distance can be chosen to reduce the oscillations along the path.

Since the vehicle speed during parking is normally low, simply let $\delta_f = \delta_r$, the turning radius R in (3) becomes

The turning radius R can be given as

$$R_{4WS} = \frac{L}{2}\cot\delta_f \qquad (25)$$

For 4WS parallel path tracking

$$\delta_f = \arctan\frac{L\sin\left(\arctan\frac{y_b}{x_b}\right)}{\sqrt{{y_b}^2+{x_b}^2}} \qquad \delta_r = -\arctan\frac{L\sin\left(\arctan\frac{y_b}{x_b}\right)}{\sqrt{{y_b}^2+{x_b}^2}} \tag{26}$$

For parallel path tracking with 2WS, let $\delta_r = 0$

$$\delta_f = \arctan\frac{L\sin\left(\arctan\frac{y_b}{x_b}\right)}{\sqrt{{y_b}^2+{x_b}^2}} \qquad \delta_r = 0 \tag{27}$$

Velocity profile generation:

Our goal is to make the vehicle well follow the generated path. In order to do this, we need to build reference time trajectories $X_{ref}(t)$), $Y_{ref}(t)$, functions of the longitudinal velocity $V_{ref}(t)$ and the steering angle $\delta_{ref}(t)$. The parallel parking method in this paper will have the reference velocity as a trapezoidal shape at each maneuver [19]. The length of each segment can be calculated by using the desired acceleration (a_{des}) for the vehicle to reach the maximum velocity (V_{max}) we can generate the reference velocity. The velocity profile is essential for safer and more comfortable driving, and applying constraints on the speed smoothness is also promoted.

First, the time needed to reach V_{max} is given by

$$t_1 = \frac{V_{max}}{a_{des}} \tag{28}$$

Then, the distance over time t_1

$$d_1 = \frac{a_{des} * {t_1}^2}{2} \tag{29}$$

The time during which vehicle stays at constant velocity

$$t_2 = \frac{d-2*d_1}{V_{max}} \qquad (30)$$

d is the distance of the segment of the geometric path

Kinematic model of the vehicle:

First, the kinematic model of 4WS vehicles is formulated as follows [27],

$$\dot{X} = V_x\, cos(\theta + \beta) \qquad (31)$$

$$\dot{Y} = V_x\, sin(\theta + \beta) \qquad (32)$$

$$\dot{\theta} = \frac{V_x \cos\beta\left(\tan\delta_f + \tan\delta_r\right)}{l_f + l_r} \qquad (33)$$

where X , Y , and θ are the vehicle global longitudinal position, lateral position, and heading angle, respectively. The side slip angle is defined as

$$\beta = arctan\frac{l_f \tan\delta_r + l_r \tan\delta_f}{l_f + l_r} \qquad (34)$$

where δ_f , δ_r , l_f and l_r are the front and rear wheel steering angle and the front and rear wheel base, respectively. The leftward steering angle is defined as the positive direction for both front and rear wheels. Practically, due to mechanical structure limitation, the front and rear wheel steering angles are normally bounded as

$$-\delta_{fmax} \le \delta_f \le \delta_{fmax}$$
$$-\delta_{rmax} \le \delta_r \le \delta_{rmax}$$

The δ_{fmax} and δ_{rmax} of the experimental vehicle in this paper is identified as 40 degrees and 30 degrees, respectively. In addition, the steering rate is also bounded by the maximum steering rate (5 deg/s) with the consideration of the steering actuation system characteristics.

Note: All the Figures shown below have a green area near the path generated. That green area is the total paths that are generated in the range of C_{2x}.

Chapter 5: Implementation and Simulation results

In this chapter, simulation using MATLAB/Simulink with a kinematic vehicle model for the simple parallel parking algorithm will be presented.

Test Environment:

This thesis has always aimed to provide an APAS that is actually running in a car. Prior to testing the system on the full-scale test platform, a significant amount of effort was invested throughout development making sure it functioned in simulation scenarios. To establish the necessary behavior before tests in a real car were deemed to be sufficiently safe and profitable, this was done in a basic and advanced simulation environment. In contrast to the other two, the simple simulation environment just uses the necessary parts of the control system. Simulink is used to implement the entire system in MATLAB.

Simulation Results:

After the empty parking lot is identified by the ultrasonic sensors and the identification algorithm, the simulation starts at the initial location where the vehicle fully stops. To demonstrate the effectiveness of the proposed APAS for 4WS vehicles, the initial location in area 1 is taken as an example. Assume the same paths are planned, and the path tracking performance of 2WS and 4WS vehicles are compared. In Figure 14 given the planned path starting in area 1, the vehicle with only front 2WS is not able to track the planned path well because when the 2WS vehicle tracks the path from area 1, the planned path collides with the other parking spots, and the vehicle will collide with other parked vehicles. From that point of view, starting in area 1 with a conventional 2WS vehicle will not be able to finish the automatic parking safely and successfully.

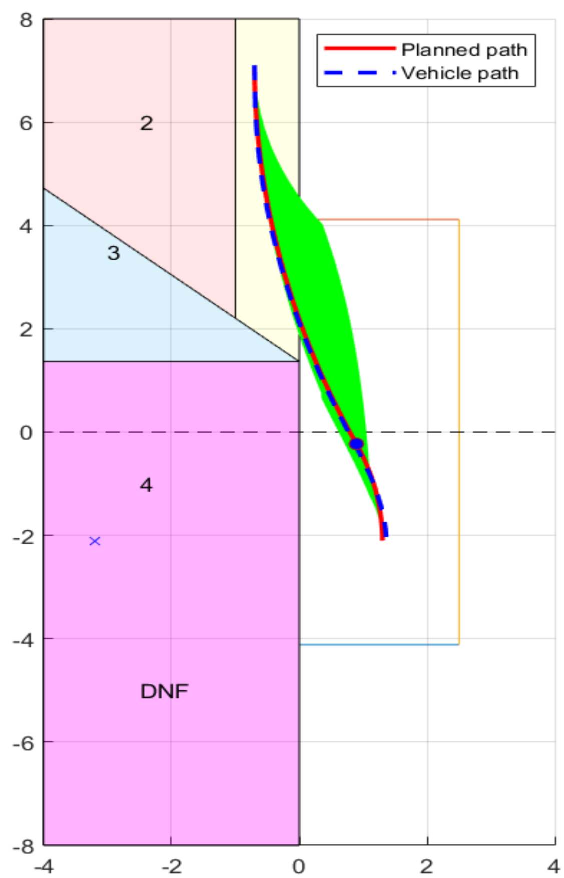

Figure 14. Simulation for vehicle starting in Area-1 using 2-WS

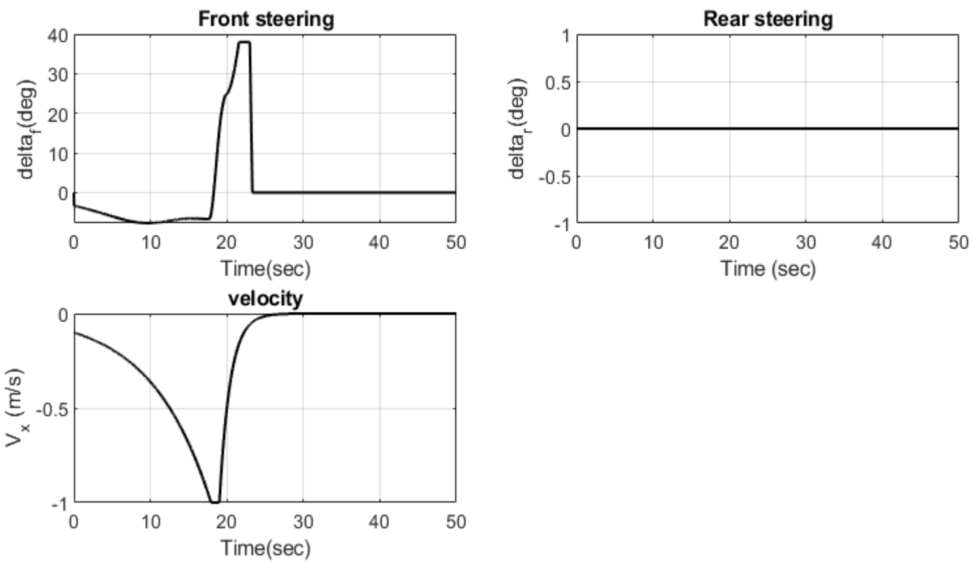

Figure 15. Steering angle and velocity

However, for the 4WS vehicle, the parking task can be finished successfully. In Figure 16, the vehicle first moves to area 3 by parallel back maneuver. Then the vehicle tracks the parallel parking path from area 3 and finishes the parking task. The vehicle successfully controlled between the dash lines in the parking space without collision risks. In the Figure 17, the front and rear wheels turn to the right to finish the parallel straight back maneuver. Then, after the vehicle enters area 3, the rear wheel turns to the left. After the vehicle comes to the tangent point, the wheel turn in the opposite direction to the first circle steering position, which makes the other half of the arc and tracks the vehicle into the destination position, With the opposite front and rear wheel steering direction, the vehicle travels with minimum cornering radius, which allows the vehicle to track the planned path accurately.

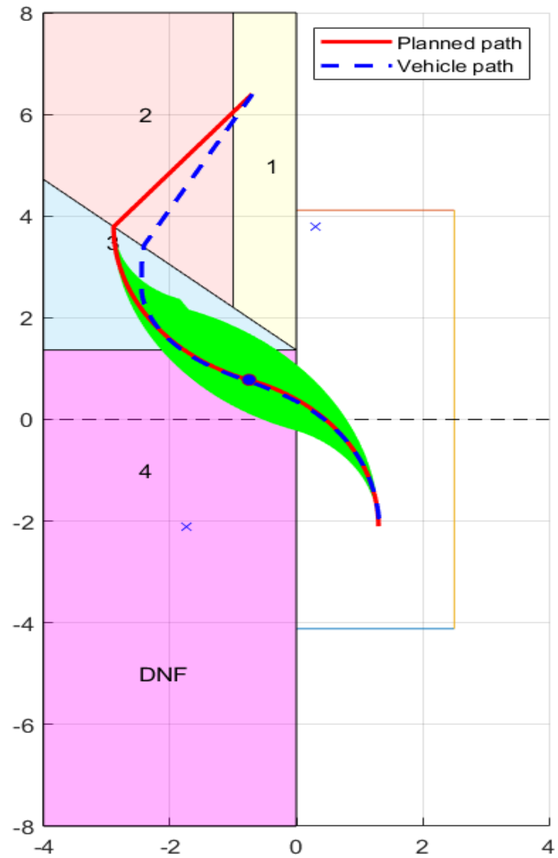

Figure 16. Simulation for vehicle starting in Area-1 using 4-WS

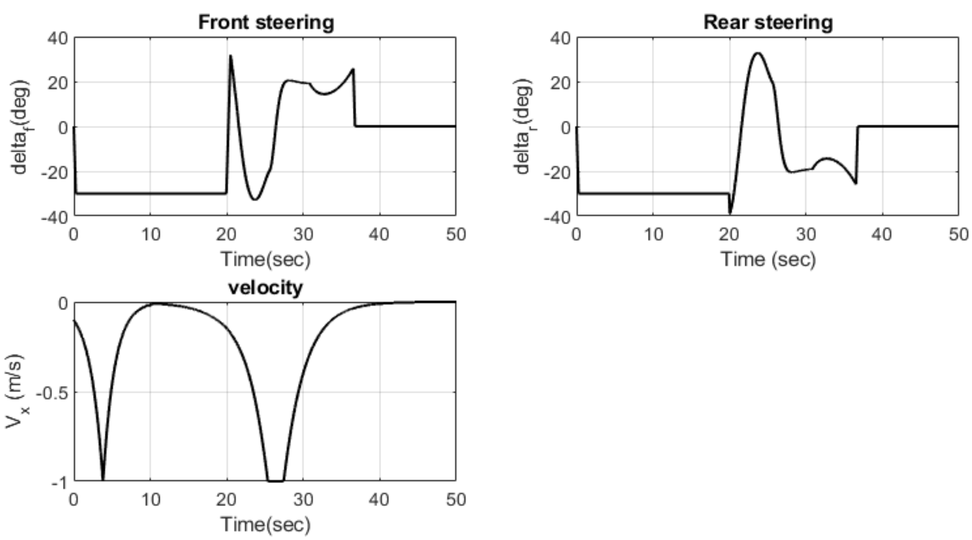

Figure 17. Steering angle and velocity

Since area 2 is a common area from where a vehicle can park without any collision with other vehicles around the parking spot, 2WS vehicle mode is used to make the parallel parking maneuver from the initial position after the vehicle stops after finding the vehicle to the desired position into the parking spot. In Figure 18, The path generation algorithm generates a path into the parking spot. In Figure. 19, the front steering wheels turn right follow the path curve till the tangent point, and then the front steering wheels turn in the opposite direction to complete the reminder path curve into the parking spot. The rear steering wheels throughout the path tracking remain at a constant 0 degrees steering angle.

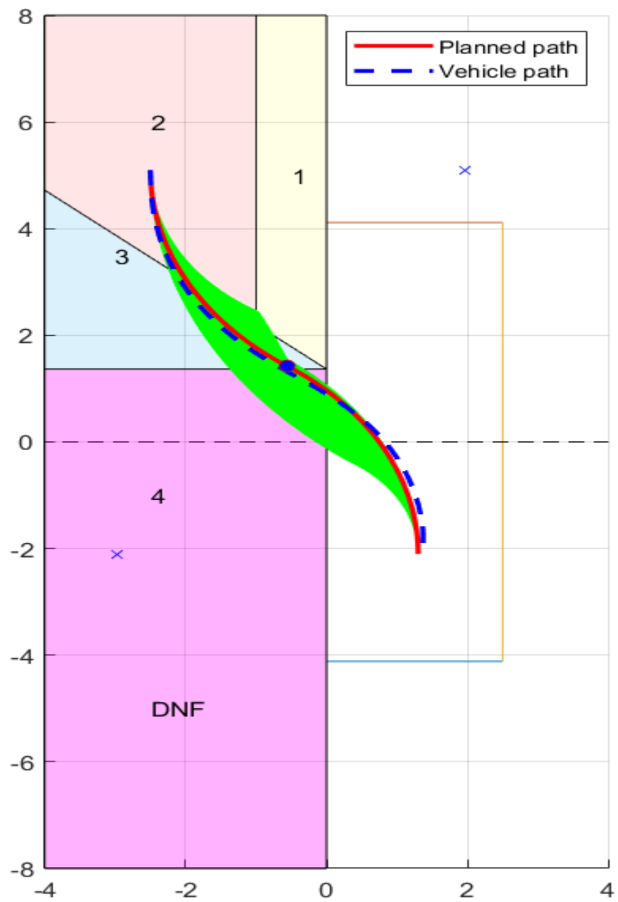

Figure 18. Simulation for vehicle starting in Area-2 using 2-WS

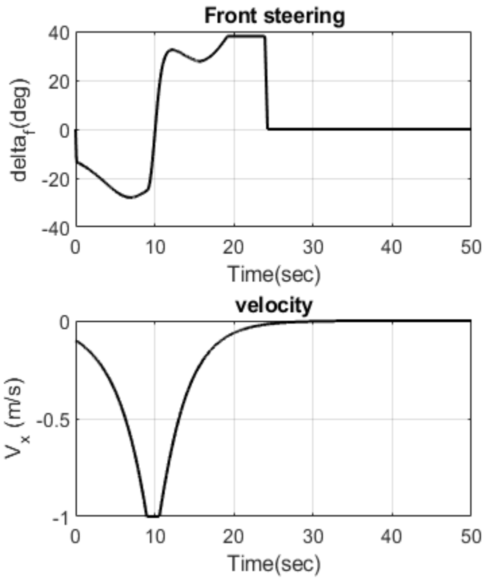

Figure 19. Steering angles and velocity

In Figure 20, the path generation algorithm generates a path for the vehicle in area 3. After the empty parking lot is identified by the ultrasonic sensors and the identification algorithm, the simulation starts at the initial location where the vehicle entirely stops. Assume the same paths are planned, and the path tracking performance of 2WS and 4WS vehicles are compared. In Figure 21 given the planned path starting in area 3, the vehicle with only front 2WS is not able to track the planned path well because when the 2WS vehicle tracks the path from area 3, the planned path collides with the curb and cannot complete the parallel parking maneuver. From that point of view, starting in area 3 with a conventional 2WS vehicle will not be able to finish the automatic parking safely and successfully.

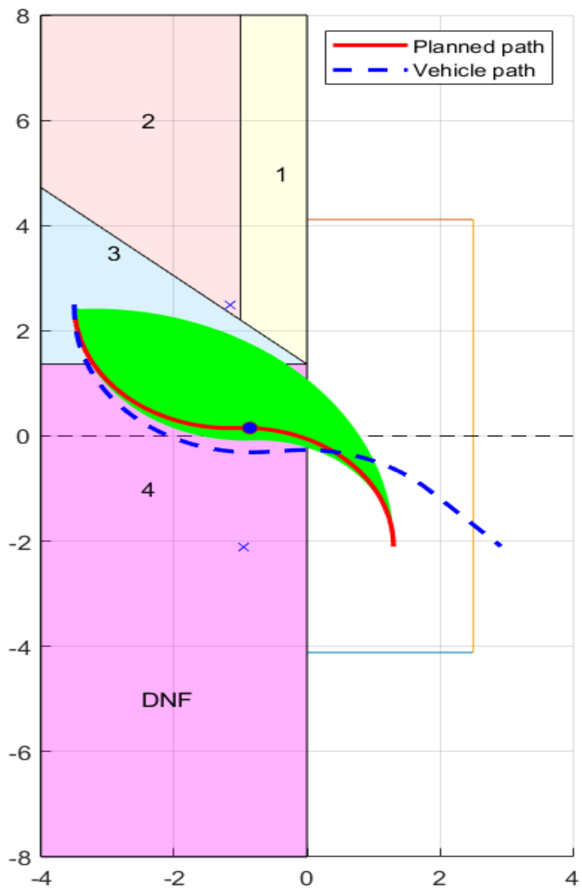

Figure 20. Simulation for vehicle starting in Area-3 using 2-WS

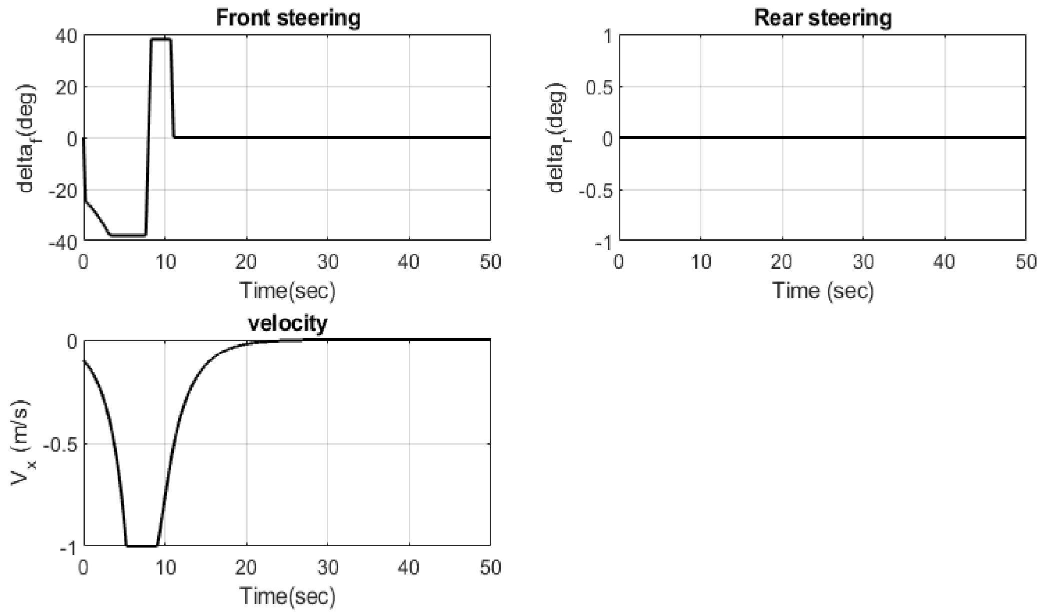

Figure 21. Steering angles and velocity

However, for the 4WS vehicle, the parking task can be finished successfully. In Figure 22, the vehicle tracks the parallel parking path from area 3 and finishes the parking task. The vehicle successfully controlled between the dash lines in the parking space without collision risks. In Figure 23, the rear wheel starts to turn to the left. After the vehicle comes to the tangent point, the wheel turn in the opposite direction to the first circle steering position, which makes the other half of the arc and tracks the vehicle into the destination position, With the opposite front and rear wheel steering direction, the vehicle travels with minimum cornering radius, which allows the vehicle to track the planned path accurately.

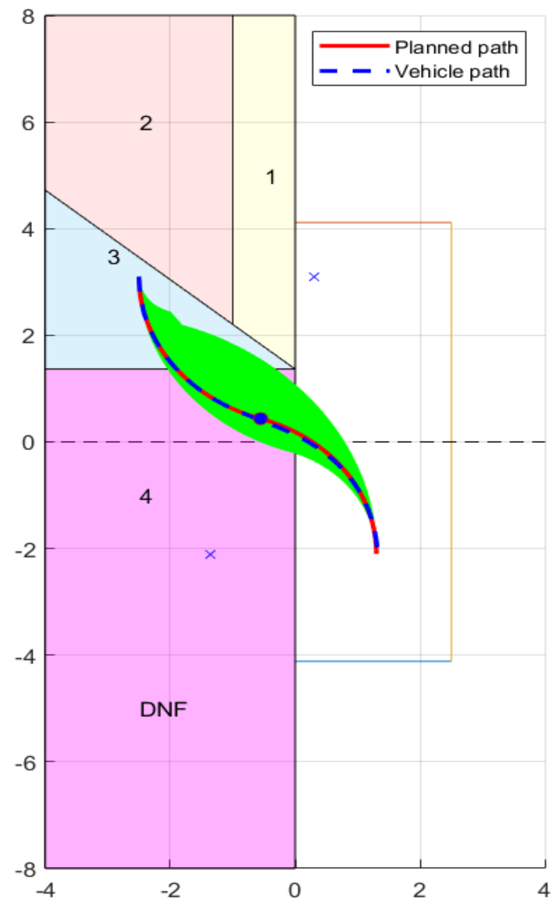

Figure 22. Simulation for vehicle starting in Area-3 using 4-WS

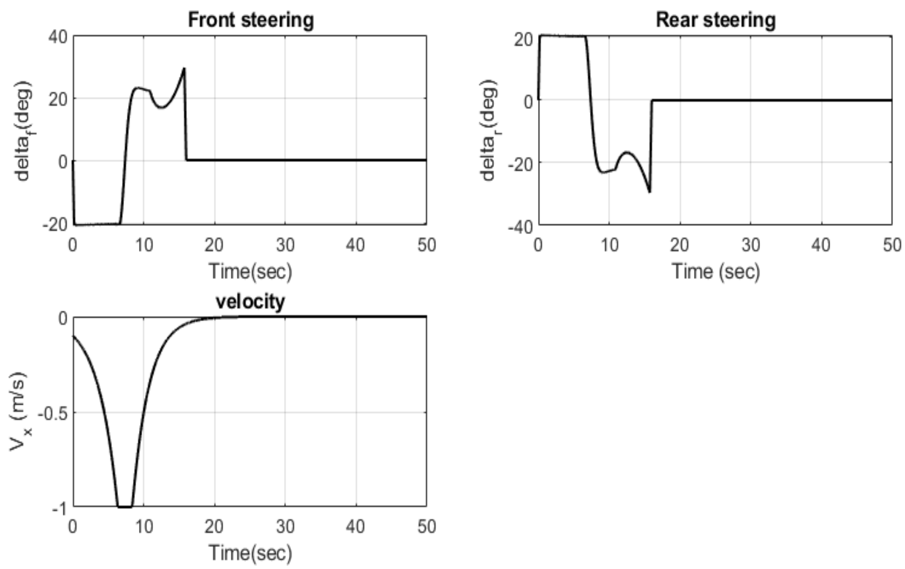

Figure 23. Steering angles and velocity

All the results above shown in the simulation are optimized in such a way that the path is selected from the range of C_{2x}. In Figure 24, an example of a path generation algorithm with a random selection of C_{2x} is shown. We can see the vehicle tracking in Figure 25, where the vehicle's rear steering angle crosses the limit of 30 degrees and reaches 60 degrees which are impossible by any 4WS vehicle or a 2WS. This shows the effectiveness of the optimization function on the path generation algorithm, which chooses the smoothest steering angle change between two arcs.

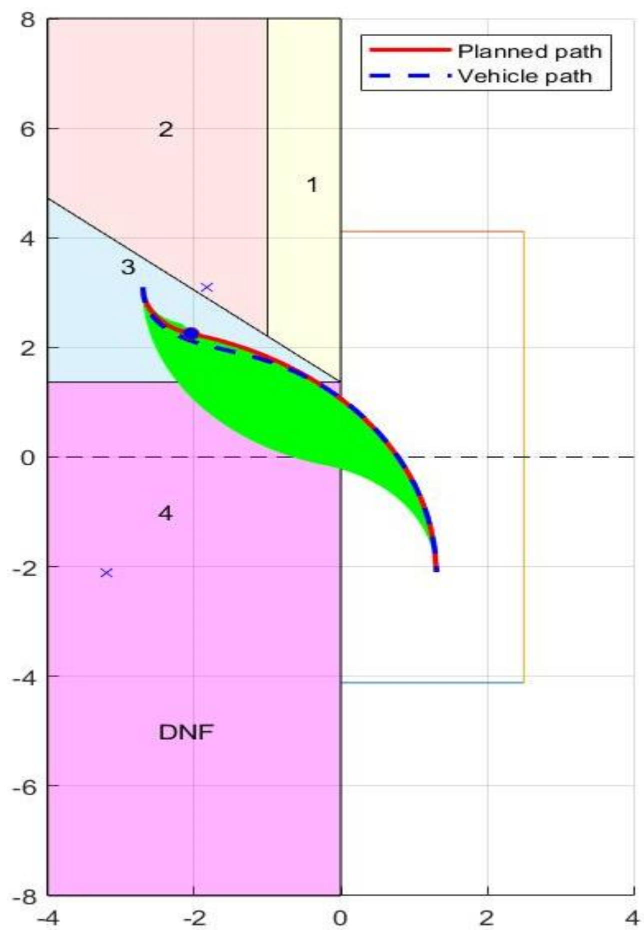

Figure 24. Simulation for vehicle starting in Area-3 using different C2x

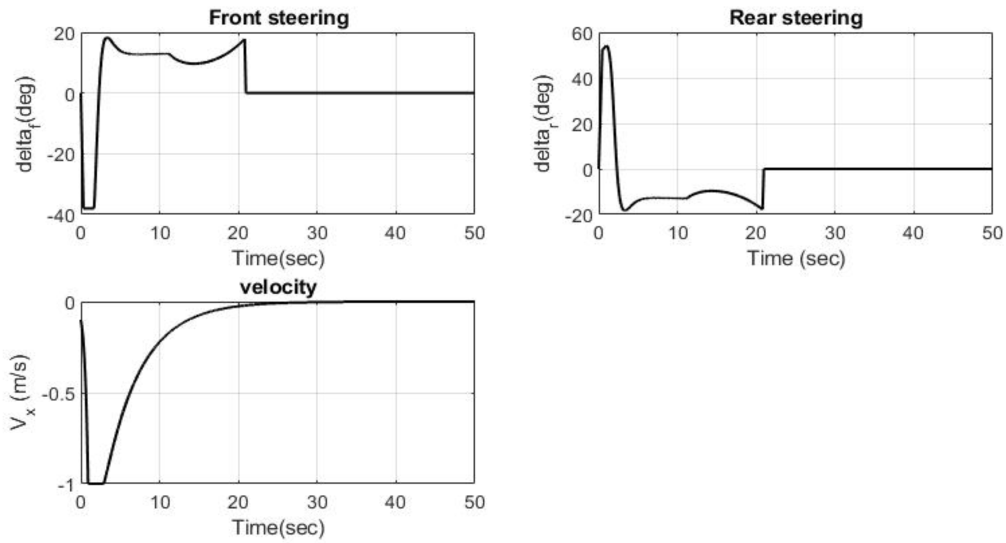

Figure 25. Steering angles and velocity

Chapter 6: Experimental setup and results

In this chapter, the implementation of the parallel parking algorithm into the experimental vehicle using the onboard processor dSPACE® MicroAutoBox II is presented.

Experimental Setup:

To test the algorithm developed for the Automatic parallel parking for both 2-WS & 4-WS, an Autonomous vehicle developed in the DSCL lab at ASU [33]-[35] was used. This Autonomous vehicle is installed with ultra-sonic sensors to identify empty parking spots, High-end INS and differential GPS with dual antennas for GPS coordinates of the vehicle, dSPACE® MicroAutoBox II controller to control all the aspects of the vehicle like acceleration, steering angles of the vehicle while autonomous driving and has two independent front & rear steering motors and four independent in-wheel motors to control the vehicles steering with independent motors for a steer-by-wire (sbw) mechanism.

Figure 26. dSPACE® MicroAutoBox II

The Calibration:

Before starting the experiments, the calibration has to be done first because there are some deviations from the wheels steering angles or the steering rate. The experiment's performance is affected by the condition of the floor, the component of the vehicle model, the program, and other factors.

The first calibration is the vehicle steering angles, the steering angles of the wheels are compared to the real-time measurements, and then the camber, caster, and toe angles are adjusted accordingly. The steering angles play an essential role in autonomous parking maneuvers. The other calibrations are the steering rate of the wheels turning. The wheel turning rate should be adjusted according to the required rate (i.e., 5 degrees/sec) to track the generated path accurately.

Experimental process:

The program we used here is the same as in the simulation part. But the outputs of the program are the real movements of the vehicle.

Figure 27. Autonomous Vehicle

First, the RTI plant model is integrated with the lower-level controller that controls the actuators and the path generation and pure pursuit controller built for the vehicle's steering control for parallel parking maneuvers. The path generation and pure pursuit controller is the controller developed as a part of this thesis. Alienware 32GB ram laptop is used to run the dSPACE control desk. The computer with the control desk is connected to the dSPACE® MicroAutoBox II controller with ethernet, and the plant model RTI controller is built into the dSPACE® MicroAutoBox II controller through the MATLAB/Simulink. After building the RTI controller into the dSPACE® MicroAutoBox II controller, we set up the control desk with the required measuring parameters.

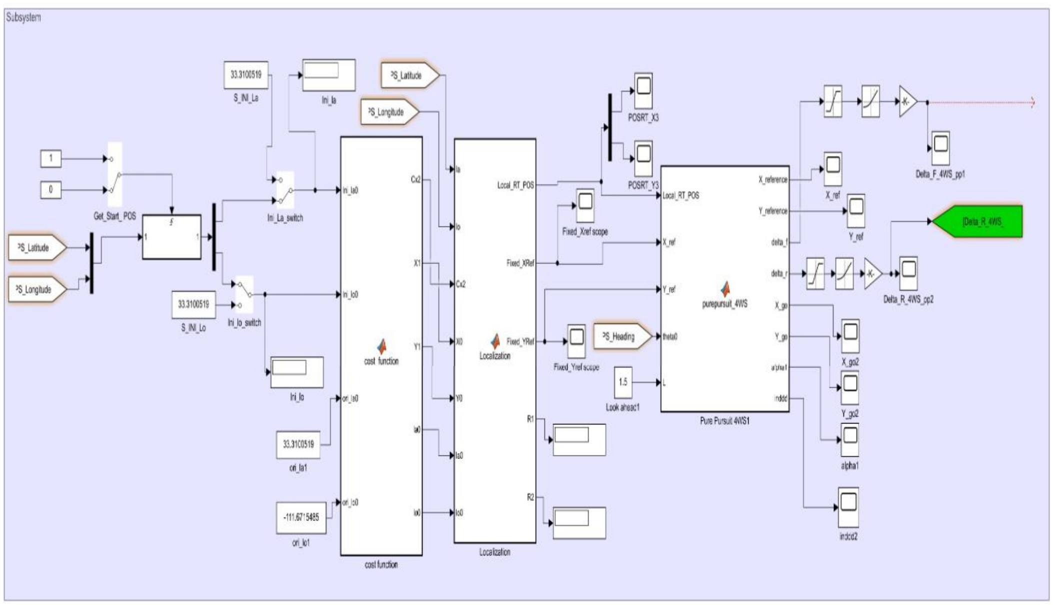

Figure 28. Path generation and pure pursuit controller RTI design

The OxTS GPS base station is set up that provides RTK corrections to one or more differential enabled GNSS receivers via a radio modem. The vehicle is taken near the parking spot, and all the controller actuators are turned on. The vehicle is driven around for a bit to warm up the GPS on the vehicle. After all the setup, the empty space was identified, and the vehicle stop position in the local coordinate was fed to the controller, which generates a path from the area to the desired endpoint. This generated path is then fed to the improved pure pursuit controller that generates steering angles to track the path with respect to its look ahead distance. The steering angles are fed to the actuators in the plant model's lower-level controller, and the dSPACE® MicroAutoBox II controller sends the commands to the steering actuators of the vehicle. By following the steering command from the dSPACE® MicroAutoBox II controller, the vehicle will be able to track the generated path to make a parallel parking maneuver into the parking spot.

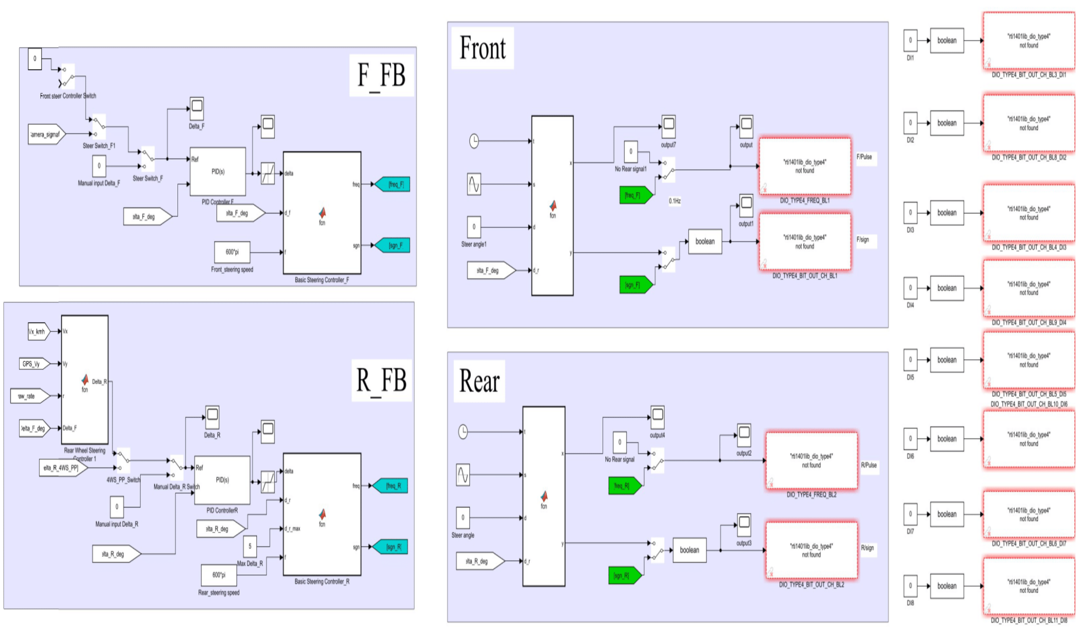

Figure 29. Steering actuator RTI design

Experimental results:

To verify the presented simulation results, using the developed controller for vehicle prototype, a real parking experiment was conducted with the aforementioned empty space identification. The realizations of identification and control algorithm are achieved using the onboard processor dSPACE® MicroAutoBox II. Some basic vehicle parameters used in APAS and experiment setups are listed in Table.

Parameter	Value	Unit
Length (L)	2.08	m
Width (Wcar)	1.5	m
Parking space length	8	m
Parking space width	27	m
Look ahead distance	1.5	m

Table 3. Experimental setup Parameters

Below attached are the results of the autonomous vehicle making a parallel path tracking via geometrically planned path. The results may be affected due the designed pure pursuit controller which is a very old controller and it highly depends on the GPS coordinates, which means a small error in the GPS coordinates can lead to inaccurate tracking control. The unpredictable conditions in the real time scenarios can also have affected the tracking control.

To better demonstrate and compare with the simulation results, the case of the initial location in area 2 with 2WS vehicle is selected. After the empty space was identified, the vehicle stop position in the local coordinate was (-1.08m, 8.36m). The vehicle speed during the automatic parking process was constantly controlled at 0.5 m/s. As shown in Figure 30, based on the path planning algorithm steering angle was planned by the onboard processor. As shown in the Figure, all the path segments were approximately tracked and the vehicle path was successfully controlled into the parking lot, indicating that the vehicle did not have collision risks with the adjacent vehicles.

As shown in Figure 30, during the tracking process, guided by the 2WS pure pursuit controller, the front wheel first turned to the right side to track the first arc. After the vehicle reached the tangent point the wheels are turned to the opposite direction to track the second arc. By tracking these maneuvers, the vehicle finally enters the parking spot and parked at the desired position. Compared with the simulation results, the parking speed and steering rate were limited at smaller values for the safety concern. Therefore, the parking time of the experiment is longer than that in the simulation.

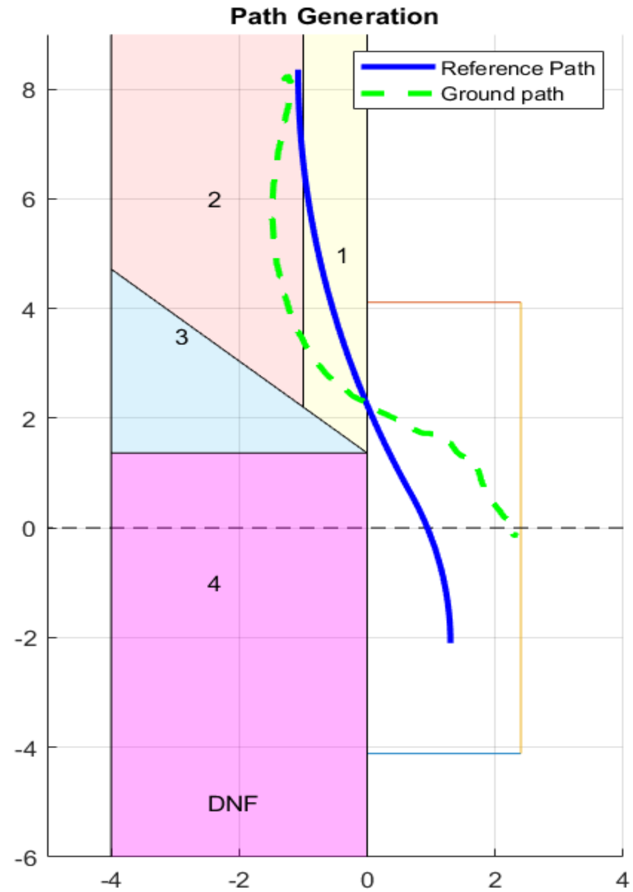

Figure 30. Experimental results of 2-WS in Area-2

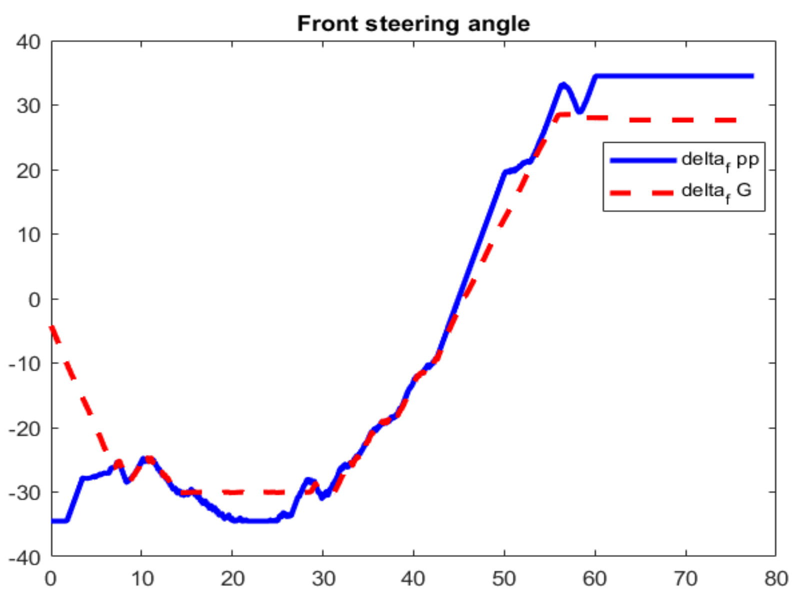

Figure 31. Front steering angle vs time

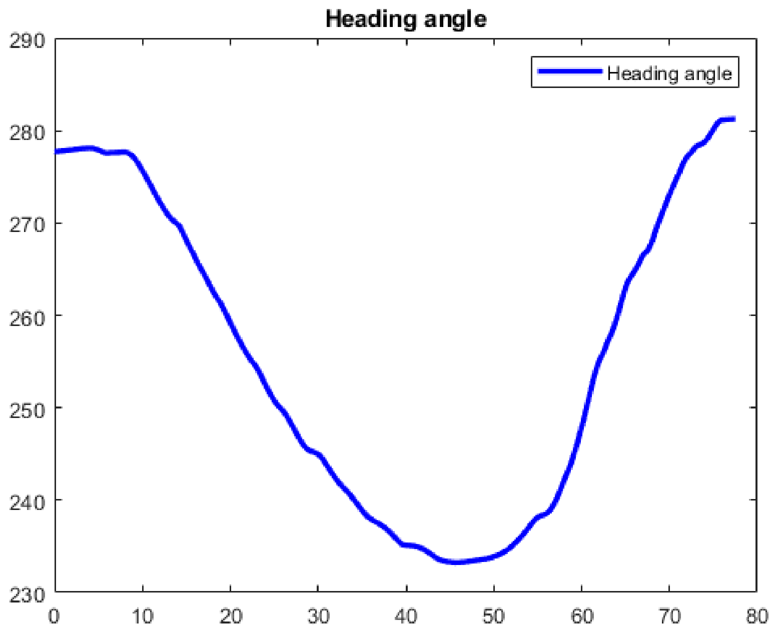

Figure 32. Heading angle vs time

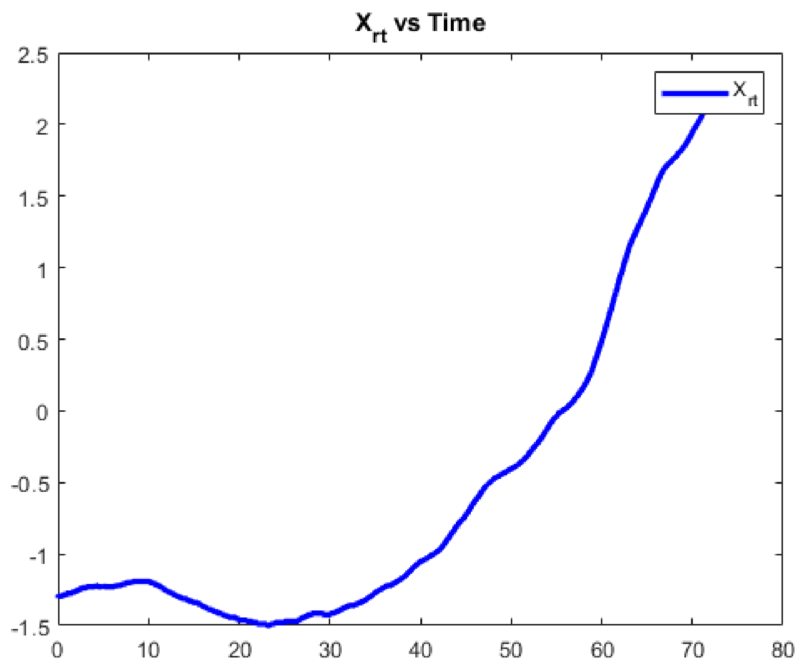

Figure 33. X- coordinate vs time

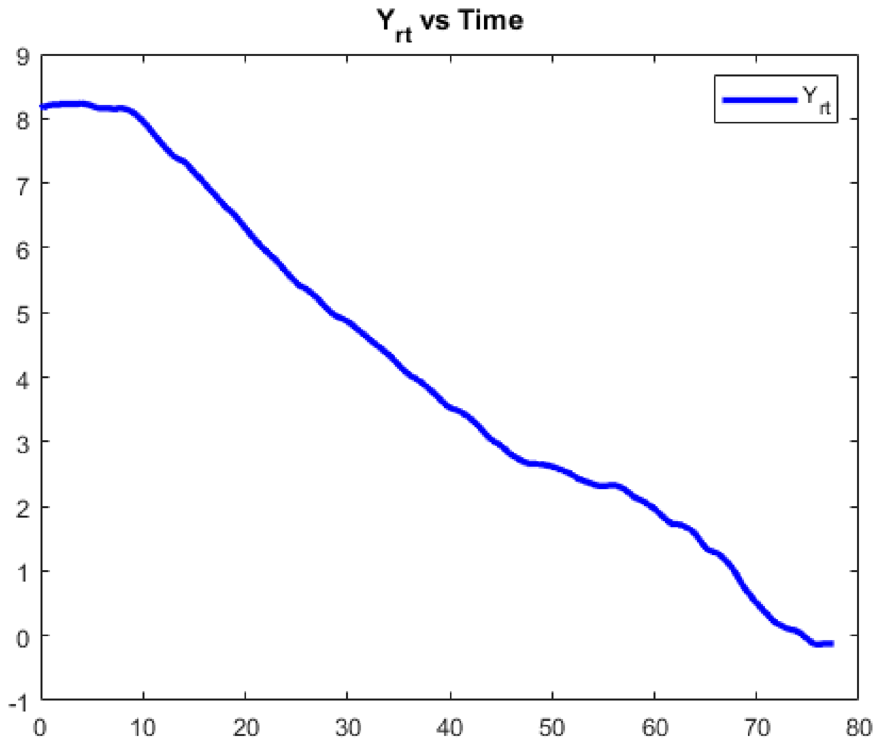

Figure 34. Y- coordinate vs time

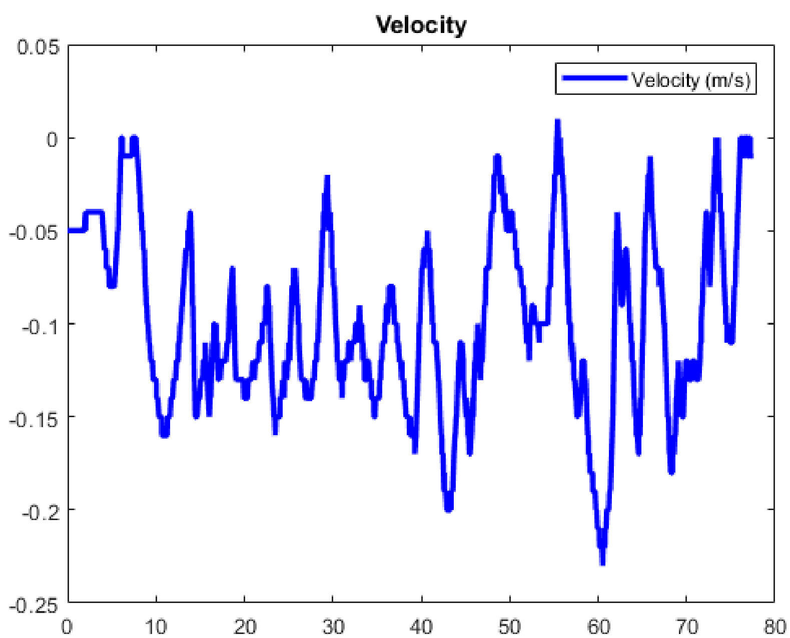

Figure 35. Velocity vs time

In other case shown in Figure 36, after the empty space detection the vehicle stop position in the local coordinate was (-1.49m, 9.59m). As shown in Figure 37, during the tracking process, guided by the 2WS pure pursuit controller, the front wheel first turned to the right side to track the first arc. After the vehicle reached the tangent point the wheels are turned to the opposite direction to track the second arc. By tracking these maneuvers, the vehicle finally enters the parking spot and parked at the desired position. Compared with the simulation results, the parking speed and steering rate were limited at smaller values for the safety concern. Therefore, the parking time of the experiment is longer than that in the simulation.

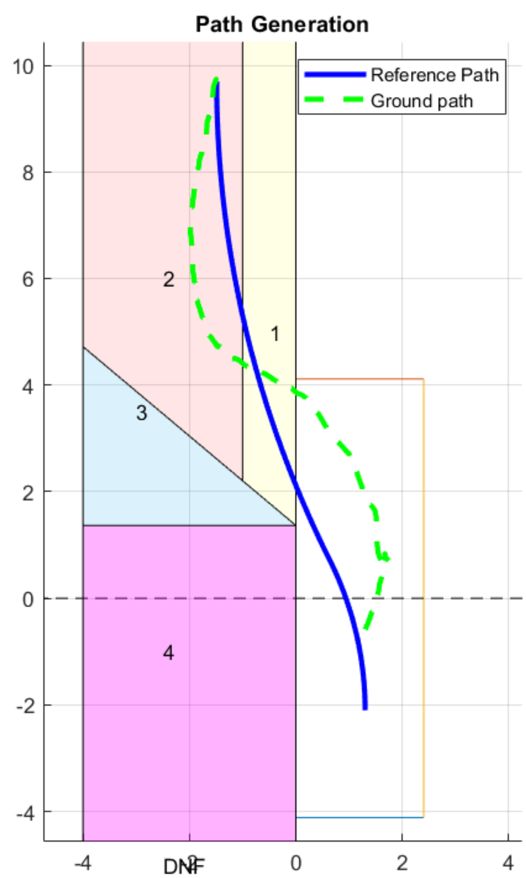

Figure 36. Experimental results of 2-WS in Area-2

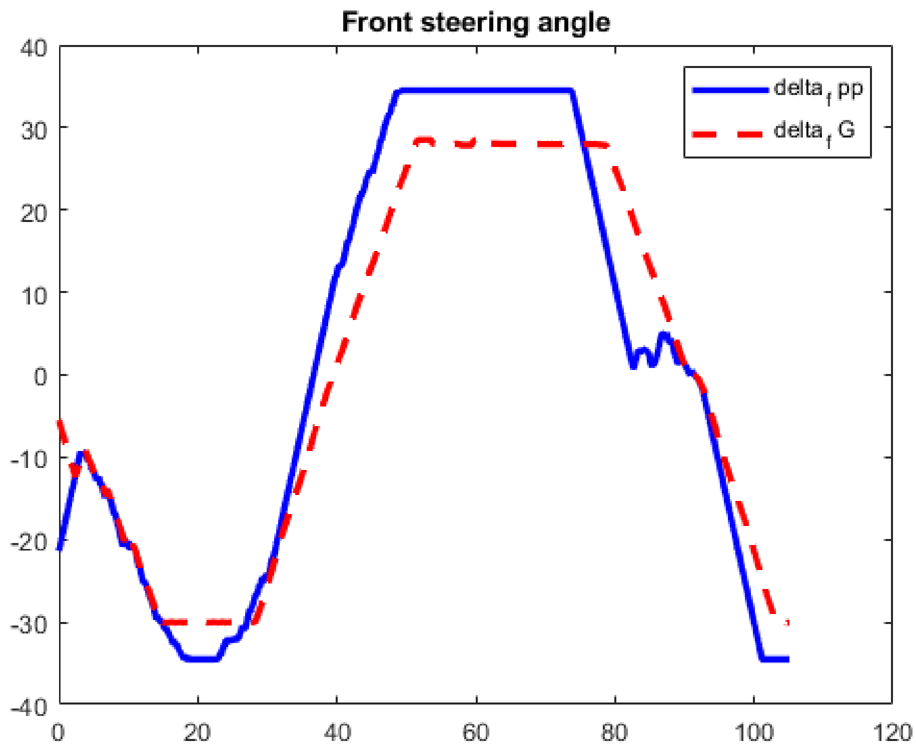

Figure 37. Front steering angle vs time

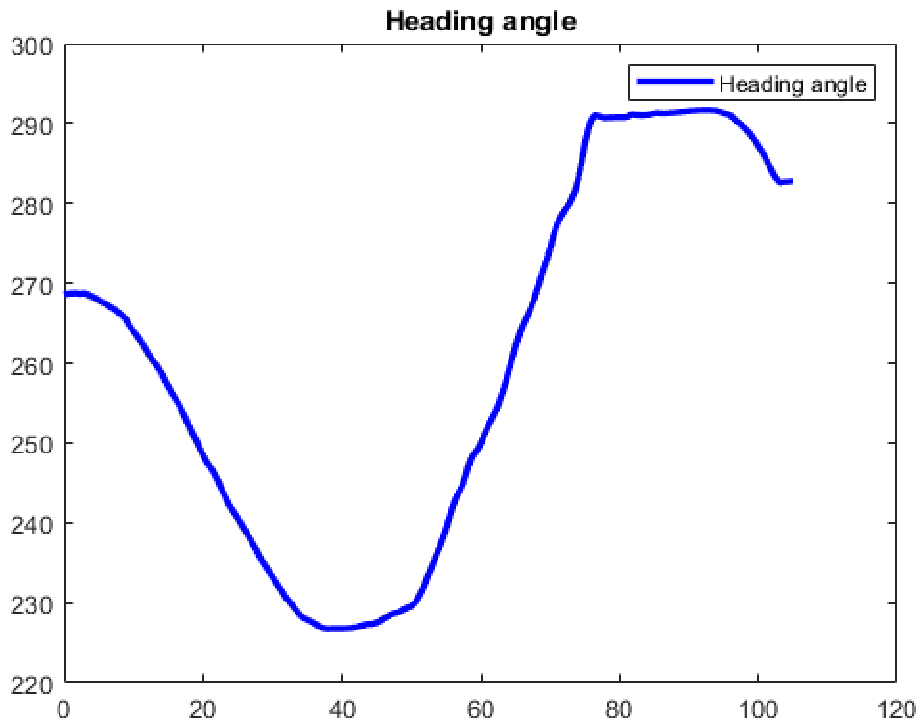

Figure 38. Heading angle vs time

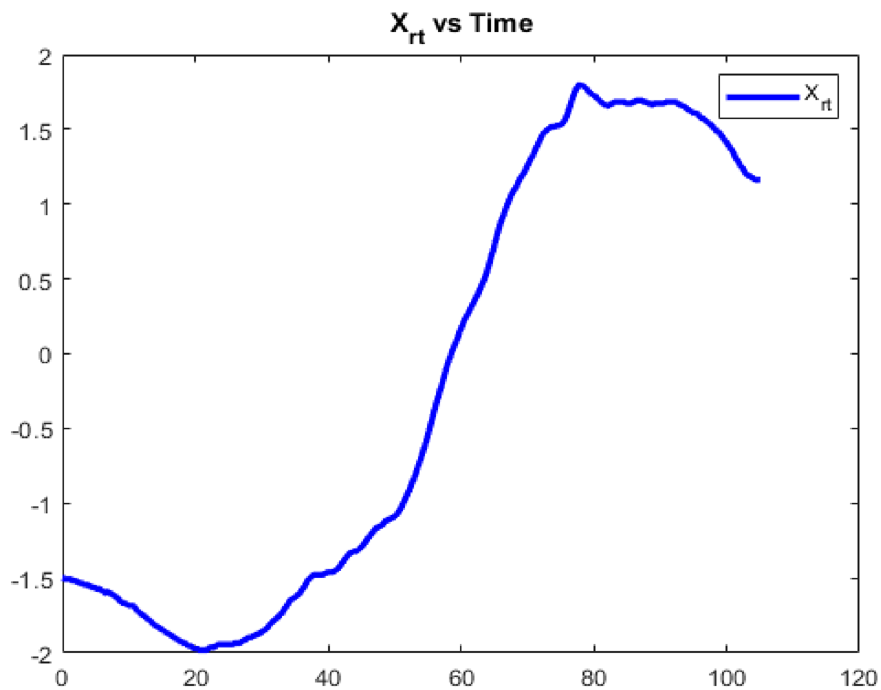

Figure 39. X-coordinate vs time

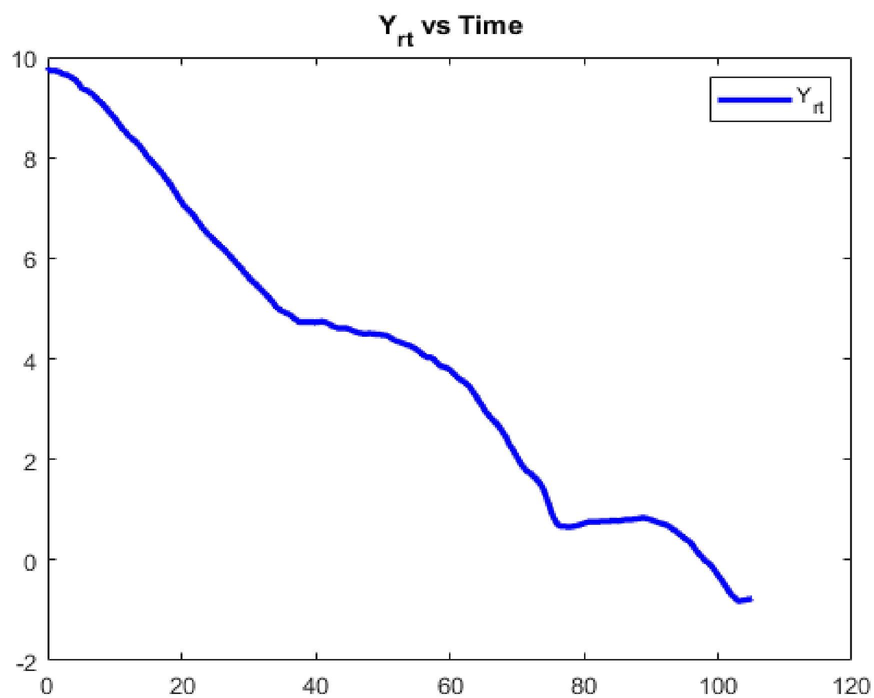

Figure 40. Y-coordinate vs time

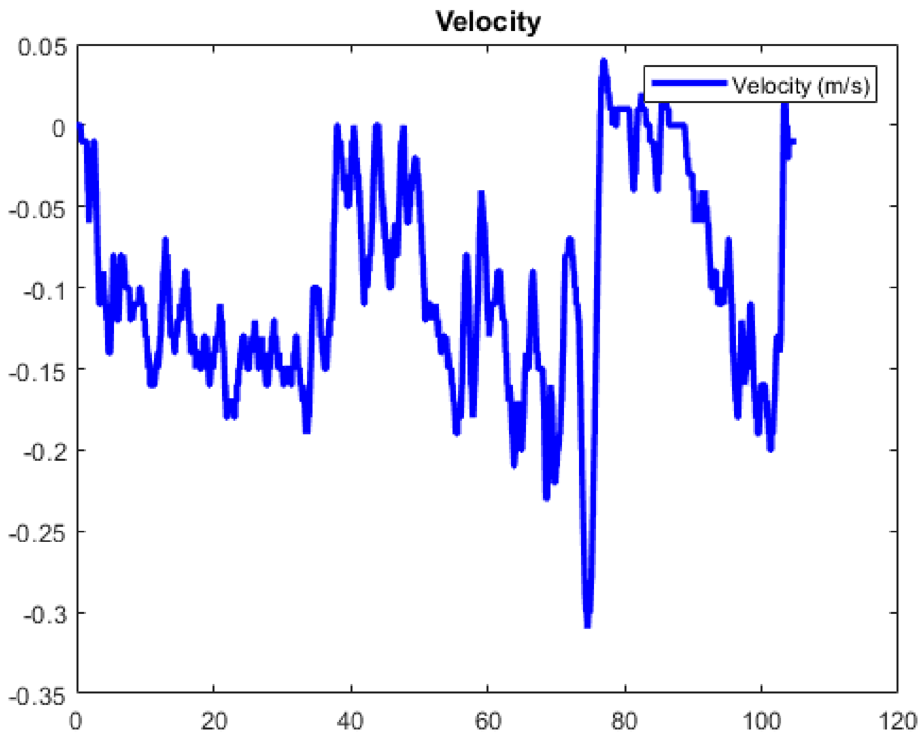

Figure 41. Velocity vs time

Chapter 7: Conclusion

In this thesis, we proposed an automatic parallel parking approach which improves the possibility of parking the vehicle into tight parking spaces. So, APAS for a 4WS vehicle for automatic parallel parking maneuver is proposed. Then, benefited from the maneuverability enabled by the 4WS system, the feasible initial parking area is largely expanded from those for the conventional 2WS vehicles. In addition, the expanded initial area is divided into 4 areas where different paths are planned correspondingly. Then an algorithm for a vehicle to perform automatic parallel parking via 4WS and 2WS is derived with the center of the second circle varying in a range. Then an optimization function which gives a smoothest steering angle change from first arc to the second arc is proposed and a lemma to prove that every path with different center of second circle has the same path length is derived. Both simulation and experimental results are shown to demonstrate the feasibility and effectiveness of the proposed APAS for both 2WS and 4WS vehicles.

REFERENCES

[1] K. Cheng, Y. Zhang, and H. Chen. June, “Planning and control for a fully-automatic parallel parking assist system in narrow parking spaces,” in IEEE Intelligent Vehicles Symposium (IV), pp. 1440-1445, 2013.

[2] Huang, Y., Chidambareswaran, T., Chang, C., Chen, Y., “Two segments (plus) path planning of automatic parking assist system for 4WS vehicles”, ASME 2019 Dynamic Systems and Control Conference, DSCC 2019 Volume 1

[3] Majdi M, Deldar M, Barzamini R, et al. AGV path planning in unknown environment using fuzzy inference systems. In: Proceedings of the 2006 1st IEEE international conference on e-learning in industrial electronics, 18–20 December 2006, Hammamet, Tunisia, pp. 64–67.

[4] Y. Zhao and E. G. Collins, “Robust automatic parallel parking in tight spaces via fuzzy logic,” Robot Autonomous System, vol. 51, no. 2-3, pp. 111-127, 2005.

[5] Flores. C.D.P., Gutiérrez. M.Á.H., Palomares. R.A., “Fuzzy logic approach to autonomous car parking using MATLAB,” Proceedings - 15th International Conference on Electronics, Communications and Computers, CONIELECOMP 2005Volume 2005, 2005, Article number 1488583, Pages 335-340

[6] Moon. J.; Bae. I.; Kim. S.; “Automatic parking controller with a twin artificial neural network architecture”. Math. Probl. Eng. 2019, 2019, 1–18

[7] C. K. Lee, C. L. Lin, and B. M. Shiu, “Autonomous vehicle parking using hybrid artificial intelligent approach,” Journal of Intelligent and Robotic Systems, vol. 56, no. 3, pp. 319-343, 2009.

[8] J. P. Laumond, P. E. Jacobs, M. Taix, and R. M. Murray, “A motion planner for nonholonomic mobile robots,” IEEE Transactions on Robotics and Automation, vol. 10, no. 5, pp.577-593, 1994.

[9] Lee.S., Kim.M., Youm.Y., Chung.W., “Control of a car-like mobile robot for parking problem”, Proceedings of the 1999 lEEE International Conference on Robotics & Automation Detroit, Michigan May 1999

[10] I. E. Paromtchik, Ph. Gamier, C. Laugier, “Autonomous maneuvers of a nonholonomic vehicle”, proc. of the Int. Symp. on Advanced Vehicle Control, 1998, p.485-491

[11] H. Vorobieva, S. Glaser, N. Minoiu-Enache, and S. Mammar, “Geometric path planning for automatic parallel parking in tiny spots,” IFAC Proc., vol. 45, no. 24, pp. 36–42, 2012.

[12] Vorobieva, H.; Glaser, S.; Minoiu-Enache, N.; Mammar, S. "Automatic parallel parking in tiny spots: Path Planning and Control. IEEE Trans. Intell. Transp. Syst. 2015, 16, 396–410

[13] Y. Yi, Z. Lu, Q. Xin, L. Jinzhou, L. Yijin, and W. Jianhang, "Smooth path planning for autonomous parking system," in IEEE Intelligent Vehicles Symposium (IV), pp. 167-173, 2017,

[14] T. Fraichard and A. Scheuer, "From reeds and shepp's to continuous curvature paths," IEEE Transactions on Robotics, vol. 20, no. 6, pp. 1025–1035, 2004.

[15] H. Tourajizadeh, M. Sarvari, S. Ordoo, "Modeling and optimal contol of 4 wheel steering vehicle using LQR and its comparison with 2 wheel steering vehicle" proceedings of the 6th RSI International Conference on Robotics and Mechatronics, IcRoM 20181 March 2019, Article number 8657562, Pages 106-113

[16] M. Yalcin, S. Yesiloglu, M. Dal, and H. Temeltas, "Maneuvering strategies for four-wheel drive, four-wheel steer mobile robots using curvatures based on weingarten-maps," in IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference, Nov 2006, pp. 4148– 4152

[17] Fnadi, M., Du, W., Gomes Da Silva, R., Plumet, F., Benamar, F. "Local obstacle-skirting path planning for a fast bi-steerable rover using bézier curves", IEEE International Conference on Robotics and Automation May 2020, Article number 9197563, Pages 10242-10248

[18] Petrov, P., Georgieva, V., "Geometric path planning and tracking control with bounded steering angle for the parking problem of automatic vehicles", AIP Conference Proceedings Volume 2048, 10 December 2018, Article number 060017

[19] S. CHOI, C. Boussard, B. d'Andréa-Novel, 'Easy path planning and robust control for automatic parallel parking', Proceedings of the 18th World Congress The International Federation of Automatic Control Milano (Italy)

[20] H. Vorobieva, S. Glaser, N. Minoiu-Enache, and S. Mammar, "Geometric path planning for automatic parallel parking in tiny spots," IFAC Proceedings Volumes, vol. 45, no. 24, pp. 36–42, 2012.

[21] Guo Liang Han, 'Automatic parking path planning based on ant colony optimization and the grid method', Journal of Sensors Volume 2021, Article ID 8592558

[22] https://resources.investormanagementservices.com/how-urban-development-impacts-parking-supply-and demandutm_sourceresourcesutm_mediumblogutm_campaignparking_supply_demand/

[23] O. Kotb, Y. C. Shen, and Y. Huang, "Smart parking guidance, monitoring and reservations: a review," IEEE Intelligent Transportation Systems Magazine, vol.9, no.2, pp. 6-16, 2017.

[24] T. Lin, H. Rivano, and F. Le Mouël, "A survey of smart parking solutions," IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 12, pp. 3229-3253, 2017.

[25] H. G. Jung, Y. H. Cho, P. J. Yoon, and J. Kim, "Scanning laser radar-based target position designation for parking aid system," IEEE Transactions on Intelligent Transportation Systems, vol. 9, no. 3, pp. 406-424, 2008.

[26] B. Lee, Y. Wei, and I. Y. Guo, "Automatic parking of self-driving car based on lidar," International Archives of the photogrammetry, remote sensing and spatial information sciences, vol. 42, 2017.

[27] J. K. Suhr, and H.G. Jung, "Sensor fusion-based vacant parking slot detection and tracking," IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 1, pp. 21-36, 2014.

[28] C. Wang, H. Zhang, M. Yang, X. Wang, L. Ye, and C. Guo, "Automatic parking based on a bird's eye view vision system," Advances in Mechanical Engineering, vol. 6, 2014.

[29] P. Degerman, J. Pohl, and M. Sethson, "Ultrasonic sensor modeling for automatic parallel parking systems in passenger cars," SAE Technical Paper, 2007.

[30] G. Gibbs, H. Jia, and I. Madani, "Obstacle detection with ultrasonic sensors and signal analysis metrics," Transportation Research Procedia, vol. 28, pp. 173-182, 2017.

[31] H. Satonaka, M. Okuda, S. Hayasaka, T. Endo, Y. Tanaka, and T. Yoshida, "Development of parking space detection using an ultrasonic sensor," in Proceedings of the 13th ITS World Congress, pp. 8-12, 2006.

[32] M. Quan, Y. Zhai, Y. Jiang, Y. J. Sun, D. L. Xu, and J. W. Gong, "An improved pure pursuit algorithm for four-wheel-steering autonomous driving vehicle," Applied Mechanics and Materials, vol. 511, pp. 958-962, 2011

[33] Yiwen Huang, Fengchen Wang, Ao Li, Yue Shi, and Yan Chen, "Development and performance enhancement of an over-actuated autonomous ground vehicle," IEEE/ASME Transactions on Mechatronics, Vol. 26, Iss. 1, pp. 33-44, 2021

[34] Fengchen Wang and Yan Chen, "Dynamics and control of a novel active yaw stabilizer to enhance vehicle lateral motion stability," ASME Journal of Dynamic Systems, Measurement and Control, Vol.140, No. 8, 081007 (9 pages), 2018

[35] Yue Shi, Yan Chen, and Bingxi Jia, "Local trajectory planning for autonomous trucks in collision avoidance maneuvers with rollover prevention," Proceedings of the 2019 American Control Conference (Invited Paper), pp. 3981-3986, 2019.