Autonomous Robot Racing Competitions

Truly Multivehicle Autonomous Racing Competitions

By HeeChang Moon, Shin Han Kang, Jeongsik Eom, Myun Joong Hwang, Youngmin Kim, Jungha Wang, Beomjun Kim, Taehyung Kim, Taekwon Ga, Jongeun Choi, Wonsang You, Jiyou Shin, Jongsoo Han, Kyeongbeen Park, Hyungpil Moon, Seok-Cheol Kee, Hak-Jin Kim, Yong-Hyun Kim, Kibeom Lee, and Jaeseung Yu

Autonomous car racing competitions have become widespread across many countries and events, taking inspiration from renowned motorsports like Formula 1 and IndyCar, among others. Similar to how traditional motorsports have fueled advancements in automotive technology, researchers and students are now anticipating similar contributions from autonomous car racing competitions. Betz et al. have provided a comprehensive review of the current landscape of autonomous car racing in [1]. These competitions feature a diverse range of robot cars, varying in scale from 1:10 to full-scale (refer to Table 1 for specifics). The size differences also extend to the sensory and computing systems employed. While many of these events are tailored for educational purposes, targeting students from high school to postgraduate programs, it is noted that smaller-scale hardware systems may not necessarily utilize the same sensory and computing systems as their real-scale counterparts. This, however, does not imply that the technical challenges in algorithm and program development are any less demanding than those encountered in full-scale or near full-scale cases. The variations in sensory, computing, and hardware resources due to the size differences may give rise to distinct technical problems not encountered by autonomous cars navigating real-world road environments. Nevertheless, fundamental issues, such as planning, motion control, and real-time recognition with onboard systems are common for both scales.

The Autonomous Robot Racing Competition (ARRC) is a scaled-down (1:2.6) multiple autonomous vehicles racing competition that originated in Korea, commencing with its inaugural event in 2021, followed by two events in 2022 and three in 2023, totaling six events so far. It has evolved into the world's unique and truly multivehicle (involving more than two cars) autonomous racing competition.

Despite significant efforts by research institutions and companies worldwide in the development of autonomous driving technology since the early 2000s, commercial services have not yet been widely implemented. This is attributed to many corner cases that autonomous systems face in various environments in various climates. Consumer or user-centric development is also not well prepared. Additionally, the lack of sufficient infrastructure for autonomous vehicles on the roads and testing in mixed environments with both autonomous and conventional vehicles contribute to the challenges. The absence or diversity of standardized regulations for autonomous vehicles also leads to various development goals among different companies, hindering practical testing and validation. One system may work fine in one specific area of a given city but may not be easily adapted to a new area with different traffic regulations and climate.

To address some of these issues, ARRC was planned in Korea under the leadership of Prof. HeeChang Moon. It commenced with seven teams in 2021 and expanded to a total of 11 autonomous robot racing vehicles by 2023. These vehicles simultaneously start and cover approximately 10 km within a maximum time limit of one hour, conducting research on lane-keeping, lane-changing, overtaking, abrupt stops, and various scenarios that may occur among multiple robot racing vehicles. The priorities of ARRC are to ensure that no serious accidents occur during the race and that all vehicles successfully complete the course. In 2024, preparations are underway for four competitions, including a total of 13 teams with 15 robot racing vehicles. All students and researchers participating in the competition are dedicated to advancing autonomous driving technology for safety and practicality.

Over the course of six events, three distinct racing courses were employed (Figure 1). In all events, time-attack qualifying is conducted for grid-slot distribution. The best lap time in C-track is 2 min and 40 s of Team Halla University. The best lap time in the Sunmoon University circuit is 1 min 49 s of Team RISE (Robotics and Intelligent System Engineering. Initially, the first four main events featured a standing start. However, due to multiple collision accidents occurring during the starting period, a standing start with a handicap of 3 s was implemented in 2023.

TEAMS' EFFORTS

Nine teams have actively contributed to discussions regarding their efforts in

Digital Object Identifier 10.1109/MRA.2024.3354524
Date of current version: 19 March 2024

TABLE 1. Autonomous racing hardware: Overview of different available hardware and racing competitions available for researchers [1], ARRC added.

	F1TEHNTH	EV GRAND PRIX AUTONOMOUS	FORMULA STUDENT DRIVERLESS	INDY AUTONOMOUS CHALLENGE	ROBORACE	AUTONOMOUS ROBOT RACING COMPETITIONS
Vehicle image						
Vehicle type	Small scale 1:10	Reduced scale 1:3	Reduced scale 1:1.5	Real racecar Indy light chassis	Real racecar LMP chassis	Reduced scale 1:2.6
Vehicle parameters	Length: .53 m Width: .28 m Mass: 3.5 kg	Length: 1.5 m Width: 1.4 m Mass: 110 kg	Vehicle parameters are based on the teams design choices	Length: 4.9 m Width: 1.9 m Mass: 750 kg	Length: 4.7 m Width: 2.0 m Mass: 1,200 kg	Length: 1.9 m Width: 1.2 m Mass: 250 kg
Maximum speed	~72 km/h	~100 km/h Depends on components choices	~120 km/h Depends on components choices	~290 km/h	~250 km/h	~25 km/h
Sensor setup	• Monocular camera • Stereo camera • 2D lidar • Indoor GPS	Sensor setup based on team choice: • Monocular camera • Stereo camera • radar • 2D lidar • 3D lidar • (RTK) GPS	Sensor setup based on team choice: • Monocular camera • Stereo camera • radar • 2D lidar • 3D lidar • (RTK) GPS	• 6× Monocular camera • 4× radar • 3× 3D lidar • (RTK) GPS	• 4× Monocular camera • 2× long-range radar • 2× short-range radar • 5× 3D lidar • (RTK) GPS	• Monocular camera • 2× 3D lidar (less than 300,000 points/second) • IMU • (RTK) GPS
Computation unit	Nvidia Jetson Nano Nvidia Jetson NX Nvidia Jetson AGX	Teams choice	Teams choice	Intel Xenon E 2278 GE – 3.30 GHz, 1× Nvidia Quadro RTX 8,000, 64 GB Ram	Nvidia Drive PX2 Speedgoat Mobile McLaren ECU	Teams choice (2× embedded PC)
Software	ROS ROS2 Autoware.Auto	Team's choice	Team's choice	ROS2 Autoware.Auto	Team's choice	Team's choice
Competitions	Several competitions a year, competitions in different countries	One race, United States only	Several competitions a year, competitions in different countries	Two races, United States only	One championship with several races, United States and United Kingdom	Several competitions a year, competitions in different cities, Korea only
Single/ multivehicle race	Multivehicle (two cars)	Single vehicle	Single vehicle	Single vehicle Multivehicle (two cars)	Single vehicle Multivehicle (two cars)	Single vehicle Multivehicle (>eight cars)
Real race competition type	Time trial Head to head racing	Time trial	Acceleration Skidpad Autocross Trackdrive Efficiency Business, cost, design	Time trial Overtacking competition	Time trial (with virtual objects)	Solo-run time attack for qualifying. Head-to-head racing for the final
Virtual race	Yes	No	No	Yes	No	No
Simulation environments	F1TENTH gym F1TENTH simulator SVL simulator	—	Formula student driverless simulator	Ansys simulator SVL simulator	Roborace simulator	—

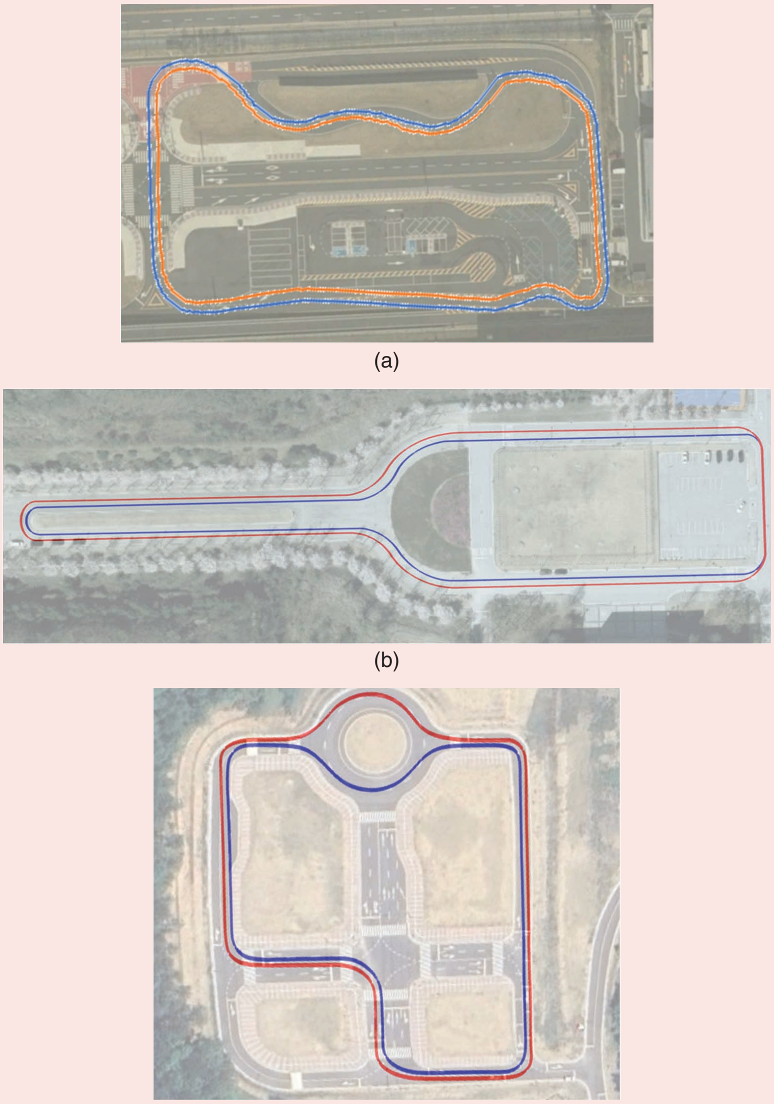
ROS, robot operating system; RTK: real-time kinematics; SVL: system vehicle loop.

implementing and realizing autonomous multiple-car racing competitions. All teams utilize the same robotic platform (ERP-42, detailed in Table 1 and visually represented in Figure 2), albeit with slight variations in sensor setups and computing systems. The software systems employed range from LabVIEW to robot operating systems (ROS), reflecting the diverse skill sets of participating students and their educational objectives. While MORAI provided a commercialized simulation software at one point, it is not an official simulator. Sensory systems that are used in each team are similar in their specifications to what are used in commercial autonomous cars, or their technologies can be easily adaptable to commercial applications. The computing systems, being developed independently by each team, undergo a range of setups and testing within the ARRC framework. See Table 2 for an overview of ARRC teams' systems.

HONGIK UNIVERSITY

The Hongik University autonomous driving team (Team HU) utilizes LabVIEW, a graphical language, as its advantage. LabVIEW, known for its relatively shorter learning curve compared to text-based languages like Python and C++, reduces complexity by allowing simultaneous execution of multiple algorithms in separate files and real-time parameter communication using notifiers. Typically, most teams use text-based languages like Python or C++ for autonomous driving development. However, Team HU opts for LabVIEW, which, despite its perceived disadvantage of being time-consuming due to its graphical nature, is relatively quicker to grasp. Additionally, LabVIEW provides the advantage of executing multiple files simultaneously. LabVIEW also facilitates real-time communication of parameters among different files through notifiers, further reducing code complexity.

The second advantage is the nonutilization of open source tools. In contrast to many teams, especially those using ROS, which heavily relies on open source software, we choose LabVIEW, which has relatively fewer open source resources. This allows us to develop



(a)

(b)

(c)

FIGURE 1. Racing tracks used in ARRC for 2021–2023. (a) The Future Mobility Tech Center in Siheung City was used in 2021. (b) Sunmoon University Circuit was used in 2022 and the second and third events in 2023. (c) C-track within Ochang Campus, Chungbuk National University was used for the first event in 2023.

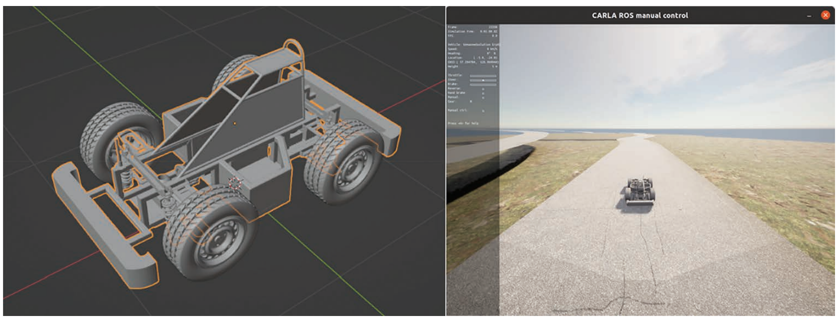


FIGURE 2. The 3D model of ERP42 and CARLA view of the spawned vehicle.

TABLE 2. Overview of ARRC teams' system.

	HONGIK UNIVERSITY	SUNMOON UNIVERSITY	UNIVERSITY OF SEOUL	YONSEI UNIVERSITY	HALLA UNIVERSITY	SUNGKYUNKWAN UNIVERSITY	CHUNGBUK NATIONAL UNIVERSITY	SEOUL NATIONAL UNIVERSITY	GACHON UNIVERSITY
Software	LabVIEW 2023 Q1	LabVIEW 2023 Q1	• ROS melodic • Ubuntu 18.04	• ROS melodic • Ubuntu 18.04	• ROS melodic • Ubuntu 18.04	• ROS noetic • Ubuntu 20.04 • Unreal engine	• ROS noetic • Ubuntu 20.04	• ROS melodic • Ubuntu 18.04	• ROS noetic • Ubuntu 20.04
Control method	• Stanley [8] • PID	Pure pursuit [4]	• Pure pursuit • PID	MPC [6]	• Pure pursuit • PID	• Kanayama [5] • MPCC [7]	Pure pursuit	Pure pursuit	• Pure pursuit • PID
Planning method	Adaptive ROI	Adaptive ROI	Trajectory-based ROI [2]	Lane tracking planner	Adaptive ROI	Frenet-Frame-based path planner [9]	Trajectory-based ROI	Adaptive ROI	• Adaptive ROI • MPC [3]
Localization	(RTK) GPS	(RTK) GPS	(RTK) GPS	(RTK) GPS	(RTK) GPS	(RTK) GPS	(RTK) GPS	(RTK) GPS	(RTK) GPS
Sensors	• Velodyne VLP-16 lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Livox Horizon lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Livox Horizon lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Livox Horizon lidar • SEKONIX SF322x-10X NVIDIA camera • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Ublox-F9P GPS	• Velodyne VLP-16 lidar • Ublox-F9P GPS
Simulator	—	—	—	• Gazebo	• MORAI	• MORAI • CARLA	MORAI	MORAI	MORAI

MPC: model predictive control; MPCC: model predictive contouring control; PID: proportional-integral-derivative.

our own algorithms independently, leading students to a more in-depth study of algorithms. For instance, in control algorithms, we created a novel algorithm by adding the θ value of the Stanley algorithm to the proportional-integral-derivative (PID) algorithm learned in undergraduate studies. For lidar, we employed a variable region-of-interest (ROI) algorithm based on waypoints and speed.

The third advantage lies in our diverse participation and achievements in various competitions. Last year, our team participated in seven competitions, including the first, second, and third Autonomous Driving Race, the second International Autonomous Driving Competition for University Students, the 2023 University Student Creative Mobility Competition, the first Autonomous Driving Software Enhancement Competition, and the aMAP Innovator Championship 1/5. We secured prizes in five of these competitions. This diverse engagement in different platforms and mission-specific algorithm development has allowed us to refine our algorithms through competition application, identify weaknesses, and make necessary improvements. It has been a valuable experience, contributing to our success in these competitions.

SUNMOON UNIVERSITY (TEAM AUTORED)

Team AutoRED is composed of undergraduate students majoring in smart information and communication engineering at Sunmoon University, led by Prof. Won-sang Yu of the Artificial Intelligence Image Processing Lab, aiming to grow as autonomous driving technology experts.

Since its first competition in 2021, Team AutoRED has continuously participated, developing autonomous driving algorithms optimized for both safety and racing. Recently, the team has proven the performance of algorithms for 360° obstacle recognition, obstacle avoidance, and collision detection in competitions. The proposed algorithm is a lightweight solution based on the pure-pursuit path-tracking algorithm, detecting obstacles in a 360° forward direction from lidar and efficiently avoiding them. The team is currently designing and developing a driving algorithm that ensures both safety and speed in complex and risky situations during racing. The consistent research efforts have resulted in multiple awards at competitions.

Team AutoRED consists of students with backgrounds in computer science, information and communication engineering, mechanical engineering, and others. They leverage their strengths in various areas, such as algorithm implementation, vehicle hardware inspection, and vehicle operation, demonstrating a spirit of collaboration. They conduct practical tests by driving the vehicle every week, addressing and analyzing any issues scientifically to find solutions.

HALLA UNIVERSITY (TEAM FLETA-AD)

TEAM INTRODUCTION

The FLETA-AD team is an autonomous robot racing team composed of undergraduate students from the School of Mechanical, Automotive, and Robot Engineering at Halla University. All members are organically linked to roles, such as planning, control, coding, and hardware setting, etc. Given the nature of competitions conducted on the same platform, most participants focus on coding and sensor control. In contrast, this team aims for autonomous driving supported by hardware. Traditionally, the students in the School of Mechanical, Automotive, and Robot Engineering at Halla University are highly motivated to participate in college student's self-made car competitions and domestic car racing. They use a variety of equipment related to automobiles and machinery. They pursue perfect hardware settings through accumulated technology based on experience using equipment, and education through hands-on courses is of great help.

Robot racing is also based on a car. This team sets the principle of strict maintenance within the allowable range and periodically performs tasks, such as wheel alignment, coupling setup, chassis part maintenance, and sensor maintenance to obtain information about the vehicle. All team members participate in this work and the coding work is processed with sufficient understanding of the vehicle conditions.

This team consists of the team leader (senior year undergraduate student), part leaders (junior year undergraduate students), and part time members. The team leader understands all situations and status of the team, and he/she maintains the team to respond immediately in any situation. The part leader trains his/her members in respective fields and conduct field tests. The first- and second-year students are assigned to each part according to their individual abilities and begin with the basic hardware system, complete learning related to their roles, and then proceed with the process of understanding the work of other parts.

RACING STRATEGY

The driving path of vehicles participating in the autonomous robot racing competition is provided in batches. Based on this, each team creates a unique path creation algorithm to participate in the race. Even if the provided coordinate data are normal, it is necessary to understand the concept of the vehicle's path when driving through continuous corners with many variables. When creating a driving route, the task of finding the optimal driving route to minimize lap time is performed. The data logging system can record location and acceleration values using GPS latitude and longitude data and can use these to obtain additional information about the driving line.

The driving route recorded by data logging can be displayed on a map as shown in Figure 3. Using this, it is possible to compare and analyze coded results and create a route that can minimize lap time. Once the track is identified through coordinates, information such as maximum turning speed can be inferred by predicting the radius of the driving path in the linked CAD program, as shown in Figure 4, and lap time can also be predicted accordingly.

Turning speed can be determined by using recorded data to analyze acceleration and speed data in corner sections where many unpredictable events can occur. In addition, the driving speed in the following straight section can be calculated, and the lap time can be predicted accordingly. This makes it possible to create an optimal racing route.

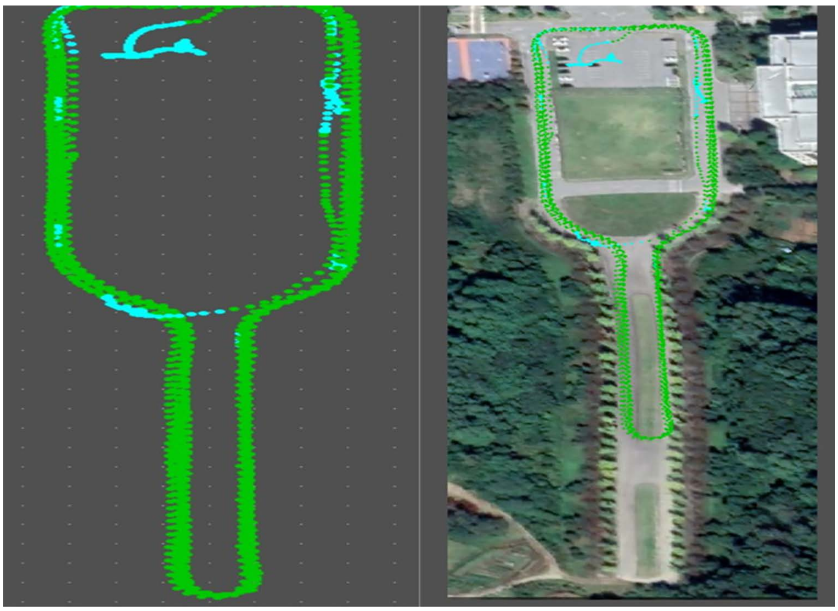


FIGURE 3. Driving route generation through logging and mapping.

UNIVERSITY OF SEOUL (TEAM UOS ROBOTICS)

Team University of Seoul (UOS) ROBOTICS comprises undergraduate students from the Robotics Lab in the

Department of Mechanical and Information Engineering at UOS. Consisting of students from the second to fourth year, the team members have continuously participated in competitions since 2021, striving to develop safe and accurate autonomous driving algorithms. The team focuses on the algorithms for path planning and tracking, obstacle recognition, and avoidance. With a strong teamwork foundation, the UOS ROBOTICS team has completed every race and won the championship at the first competition in 2023.

KEY FEATURES OF THE TEAM

The Department of Mechanical and Information Engineering at UOS offers the mandatory course "Smart Mobility Design," providing theoretical study and hands-on practice with autonomous robot platforms for all students. This course has led to a high interest of students and active participation in our team. Led by the team leader, the members conduct weekly meetings for research and strategy in autonomous driving racing competitions. During these meetings, various algorithms and strategies are reviewed, and newly proposed algorithms undergo thorough reviews by theoretical analysis and source code reviews. The simulation validations are executed before applying them to offline driving experiments, as shown in Figure 5. Results from the meetings are further validated through field tests, emphasizing the improvement of the robot's driving stability. The Robotics Lab provides self-developed educational materials and practical training using a small robot platform to facilitate the quick adaptation of new team members. This systematic training enhances the capabilities of new team members, ensuring the team's sustainability and consistent achievements.

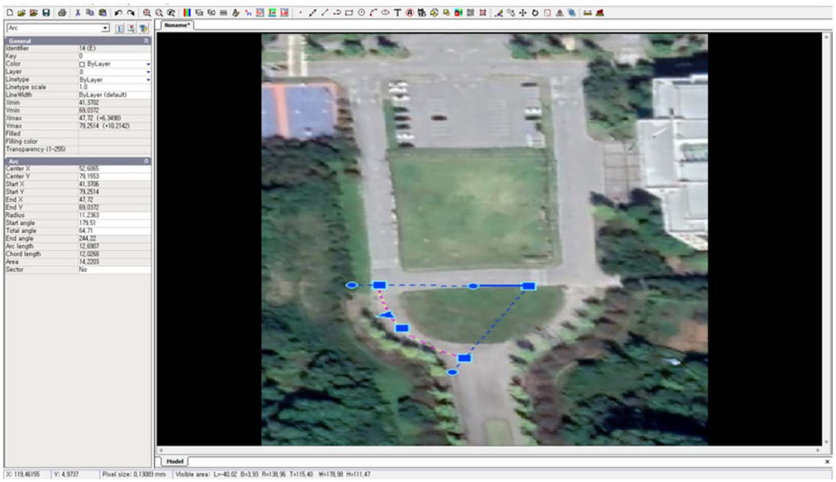


FIGURE 4. Driving line prediction with the CAD program.

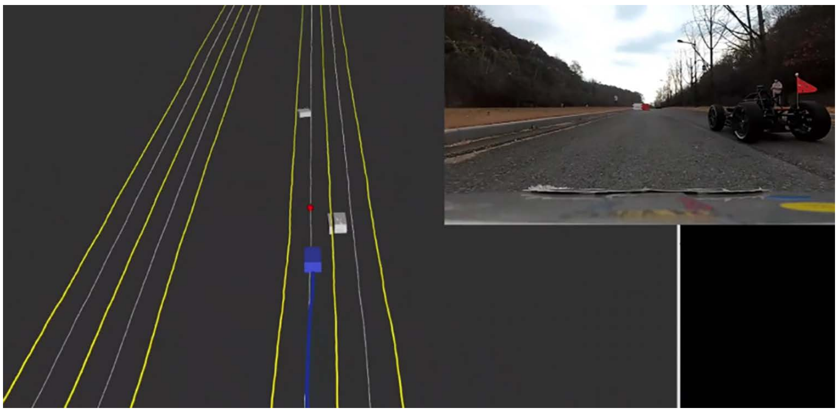


FIGURE 5. Simulation and real driving for obstacle detection and path tracking (red dot: target point at look-ahead distance in pure-pursuit algorithm).

AUTONOMOUS DRIVING ALGORITHMS

Our team implements autonomous driving algorithms using two lidars and two real-time kinematics (RTK)-GPS. The software development environment and language are Ubuntu 18.04, ROS melodic, C++ for recognition, and Python for control. Our team proposed a trajectory-based 3D point-cloud ROI determination method in [2]. Using the proposed method, point-cloud data collected through lidar are optimized for the shape of the driving path to detect obstacles with minimal data for fast processing. Our path-tracking algorithm is based on the pure-pursuit algorithm, setting a variable look-ahead distance and tracking it for optimal performance based on the shape of the driving path. Two GPS devices are used to determine the vehicle's current position and heading angle. The path-tracking and control algorithms, based on pure-pursuit and PID, are employed to enhance path-following performance. When encountering another vehicle while driving, algorithms are implemented to follow or overtake the vehicle according to the situation.

YONSEI UNIVERSITY (TEAM MECH IN AI)

Team Yonsei University uses the ROS integration method for implementing a lane-keeping algorithm for autonomous robot racing vehicles. The lane-keeping algorithm divides the path planning into two steps: generating a global path, following the designated lane, using the lane tracking planner, and creating an effective local path to adapt to environments with obstacles ahead using the model predictive control (MPC) local planner. The lane tracking planner generates the path from the current position to the target point, considering the vehicle's current position, the goal point, and the postprocessed path. It assesses the path's validity, considering the vehicle's outline. Additionally, it interpolates discontinuous paths generated during processes, such as initial path entry or lane changes using the cubic spline curve. The lane tracking planner, compared to grid-based search algorithms like A* or Dijkstra, commonly used in global path planning, restricts the driving path to a predefined route, enabling more stable driving in racing environments. The MPC local planner uses receding-horizon quadratic-form MPC to track the global path generated by the lane tracking planner. MPC, as opposed to methods commonly used in local path planning, such as pure pursuit or the dynamic window approach, can calculate flexible and robust paths for dynamic obstacles in racing environments (Figure 6). The algorithm in this study is modularized and implemented in the ROS navigation stack, and its driving performance is validated by reproducing the actual race vehicle, sensors, and racing environment using the Gazebo simulator. While the proposed algorithm was well implemented on the vehicle, due to hardware issues the record of Yonsei team was not competitive in the last racing game in 2023 using this new approach.

GACHON UNIVERSITY (TEAM GADIS)

The team GADIS of Gachon University is composed of students from the departments of Mechanical Engineering, Future Mobility, and Electronic Engineering. In 2023, the team achieved outstanding success by winning the Best Award and Grand Prize in two competitions, ultimately securing the season's overall championship.

The entire autonomous system is built on the ROS, integrating various algorithms for perception, decision making with driving strategy, and tracking control, similar to conventional autonomous driving systems architecture. As it is a racing competition, the team calculated the maximum achievable speed for each type of track, ensuring the vehicle can navigate all courses stably. After ensuring stable tracking performance at various speeds, the focus shifted to algorithms, considering diverse interactions in multivehicle scenarios. The team designed algorithms to assess risk in multivehicle situations and select a safe optimal path [3]. To validate these algorithms, the team created multivehicle scenarios using simulators and conducted iterative simulation tests. The simulations were performed using MORAI's autonomous driving mobility simulator, replicating real-world vehicle conditions on a digital twin precision map. The multivehicle simulation environment is shown in Figure 7.

Witnessing multiple autonomous vehicles with distinct algorithms racing simultaneously provided a unique experience. Various evaluations were conducted in different situations, but predicting the movements of surrounding vehicles with different algorithms proved challenging and posed a significant difficulty.

SUNGKYUNKWAN UNIVERSITY (TEAM RISE)

Team RISE consists of undergraduate and graduate students from the Robotics and Intelligent System Engineering (RISE) lab of Sungkyunkwan University. Team RISE started racing when ARRC first made and won at 2023 second ARRC, making a new record of lap time. The development environment is built on Ubuntu 20.04, ROS noetic, C++, and Python, and algorithms are tested on the CARLA simulator, the open source unreal-engine–based simulator. The details of the algorithms are described below. The racing video of Team RISE is uploaded at https://youtu.be/4Sx8gnPWXSY?si=BJR0tGaBWfd53j26.

PERCEPTION

Team RISE used a 16-channel Velodyne puck lidar to detect and track obstacles. The data were preprocessed to remove ground (road, curb, sidewalk) parts to extract obstacles standing on the ground. Nonground data are then clustered by binding points with a certain distance to think they are from the same obstacle. By using the information of clusters, the team tracked each obstacle using a Kalman filter, which can estimate the obstacle's state. Finally, the team can get the obstacle's position and velocity that will be used in path planning. Figure 8 shows the image of detection and tracking results during racing.

PATH PLANNER

Team RISE made a path planner based on the Frenet-Frame [9], making it possible to keep lanes and avoid obstacles. Frenet-Frame makes the reference line as *s*– axis and the normal direction of the reference line as *d*– axis, as you can see in Figure 9. The lane we must keep is considered as a reference line and make the candidate paths containing

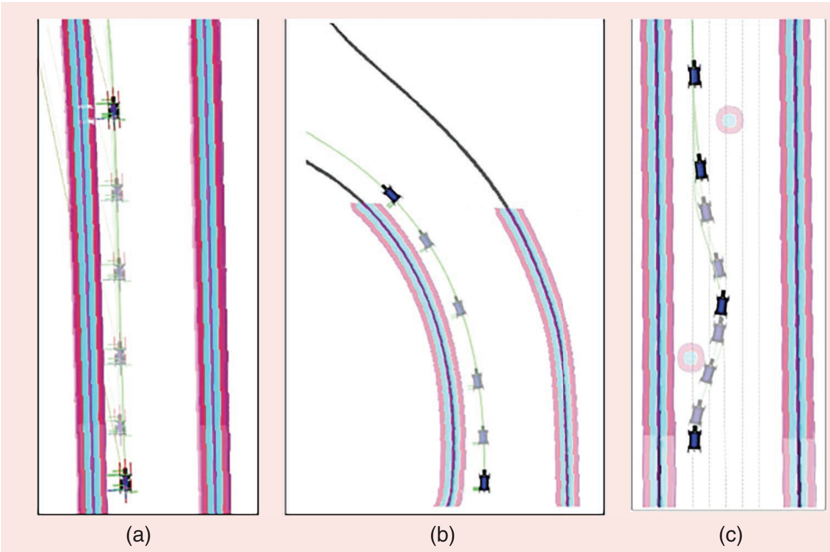
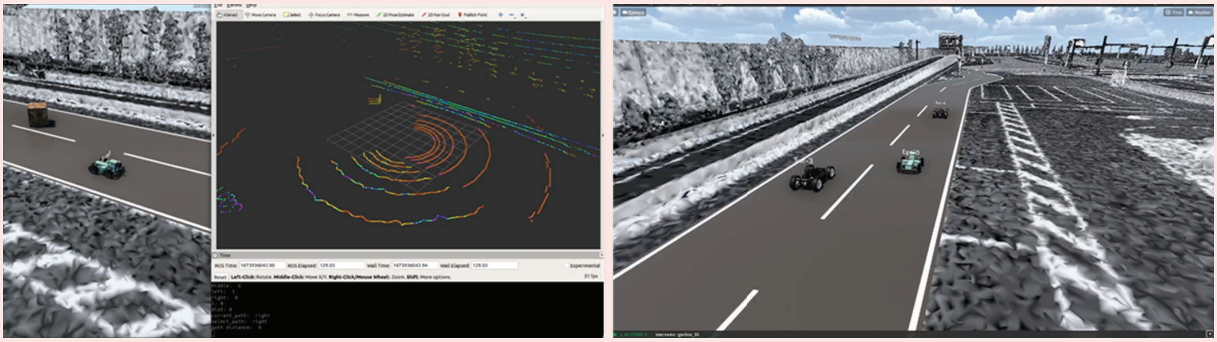


FIGURE 6. (a) Lane-keeping driving on a straight course. (b) Lane-keeping driving on a curved course. (c) Obstacle avoidance with the lane change behavior.



(a) (b)

FIGURE 7. Algorithm verification using MORAI Sim. (a) Lidar-based static obstacle detection. (b) Driving algorithm performance evaluation with multivehicle interaction situation.

different speed and lane. The obstacle's state from perception is used to estimate the obstacle's trajectory and penalize candidate paths that have the danger of collision. There are three costs to evaluate candidate paths: offset cost, safety cost, and speed cost. The offset cost penalizes the path that has offset from the reference line. This cost makes planners keep the lane. The safety cost penalizes paths that are near obstacles. The obstacle's estimated trajectory makes a cost map that distributes cost in the grid. If the candidate's path is close to the obstacle's trajectory, it is considered a dangerous path and allocates high cost. This cost helps the planner to avoid obstacles. The speed cost helps to select the fastest path since this is a racing competition. The best path is selected by finding the path that has the lowest cost. The result is shown in Figure 10.

CONTROLLER

Team RISE has implemented various control algorithms: pure pursuit, Kanayama, and MPC. The algorithm implemented at racing was Kanayama controller, the stability of which is proved and can easily be computed. For the next competition, the team is trying to apply model predictive contouring control (MPCC) [7]. As the team is using a kinematic bicycle model, normal MPC has linearization error to follow the path since the model and path are nonlinear. By using MPCC, error terms can be linearized using path parameter θ and find an optimal solution that minimizes error from the reference path.

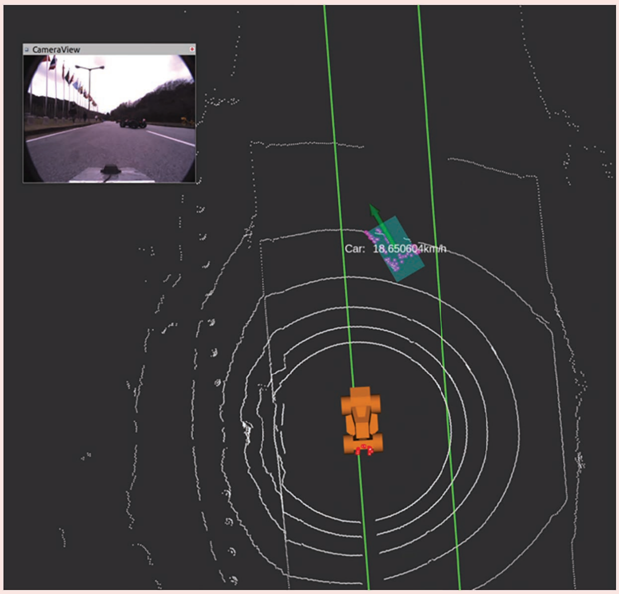


FIGURE 8. The detection and tracking result of the Team RISE algorithm. The white points show raw data, purple show clustered data. The cyan box represents the obstacle and the green arrow visualizes the speed of the obstacle. The orange car is the ego vehicle and the green line is two lanes we should keep.

SIMULATOR

Team RISE made a custom simulator using CARLA (Figure 2). The team made a 3D model of the ERP42 platform and set motor specifications similar to the real world. Also, the map of the racing track is built to test algorithms. The simulator is available to spawn nonplayer character vehicles that can move automatically or control manually. This nonplayer character made a team to test scenarios that can happen in real racing situations and have repetitive experiments rapidly.

CHUNGBUK UNIVERSITY (TEAM CLOTHOID-R)

The Clothoid-R team at Chungbuk University is a newly formed team consisting of undergraduate students from the College of Electronics and Information, and they have the achievement of winning an Excellent Prize in the 2023 first Robot Racing Competition. The team's primary development algorithm focuses on high-speed, high-precision object recognition using a fusion of lidar and camera sensors. Most autonomous driving algorithms undergo initial validation in a simulation environment, followed by development and testing processes through real vehicle tests on the autonomous driving testbed, C-track (http://cbnuscrc.org), owned by Chungbuk University.

The C-track provides an optimal environment for testing algorithms, allowing undergraduate students to utilize autonomous driving platforms in practical courses to test algorithms, setup, and evaluate various scenarios, enabling our team to prepare safely and efficiently for autonomous driving racing competitions.

The Clothoid-R team is composed of a team leader, section leaders for each part, and team members. The fourth-year undergraduate team leader establishes overall development goals for the team and oversees task distribution according to detailed plans. The experienced third-year undergraduate section leaders supervise and educate team members, progressing through the validation and feedback process for development algorithms through real-vehicle tests based on assigned tasks. Team members, composed of first- and second-year undergraduates, acquire essential knowledge about vehicles, learn about the overall team process, and build their skills while actively participating in development.

The technical sections are divided into control, lidar perception, camera perception, and position-recognition technologies, integrating efficiently to complete autonomous driving. Particularly, the team emphasizes object recognition technology to achieve safe autonomous driving. A sensor fusion method was developed through precise sensor calibration, aligning the coordinate systems of lidar and camera, despite their existence in different coordinate spaces. As a result, the camera can detect the presence of collision-risk vehicles in the front and their 2D spatial positions, while lidar's point-cloud information accurately calculates the relative distance to the risk vehicles.

SEOUL NATIONAL UNIVERSITY (TEAM RAPID TRACTOR)

The Seoul National University autonomous driving team, led by Prof. Hak-Jin Kim, was composed of students from the Department of Biosystems Engineering at the College of Agriculture and Life Sciences. This team achieved the grand prize in the first competition held at Seoul National University's Siheung Campus in 2021. Based on an experience in autonomous agricultural machinery research, Team Rapid Tractor focused on achieving precise and safe driving rather than emphasizing speed. The high-level controller consisted of a global navigation satellite system (GNSS)-based autonomous navigation system and a lidar-based collision avoidance system. The GNSS-based system was developed in C/C++ and utilized RTK-GNSS to accurately follow the predefined waypoints in racing roads. In principle, on a curved road, the system

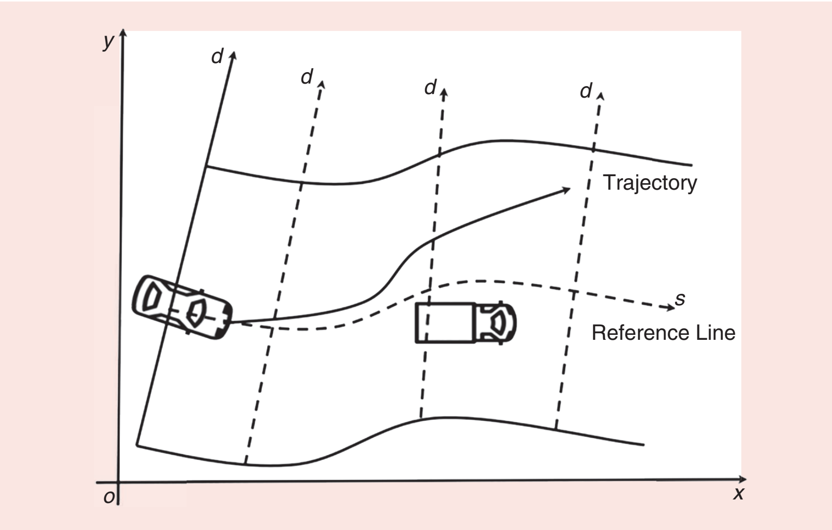


FIGURE 9. The comparison of Frenet-Frame and Cartesian frame.

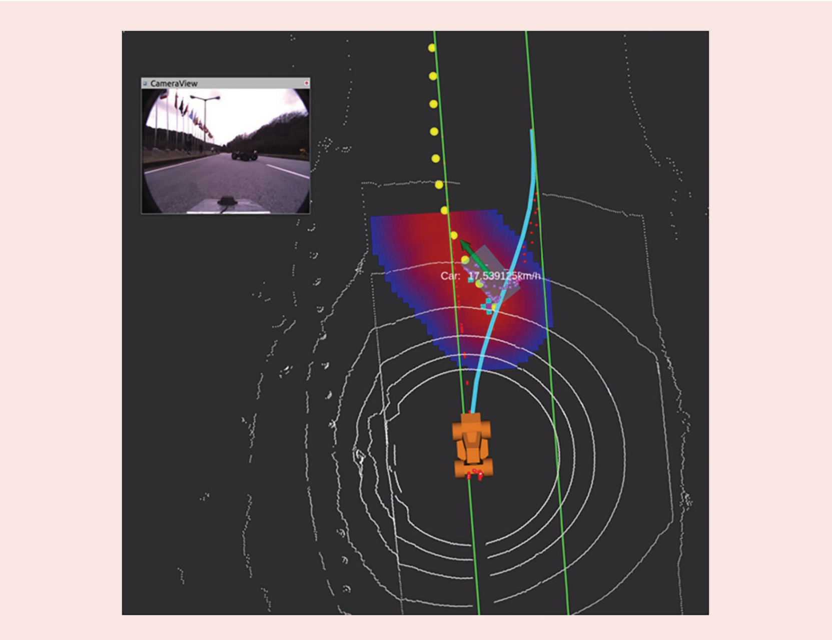


FIGURE 10. The result of the planner. The red and blue area is the cost map made by yellow points, which are the estimated trajectory of the obstacle. The red points show the candidate paths and the cyan line is the selected path that has the lowest cost.

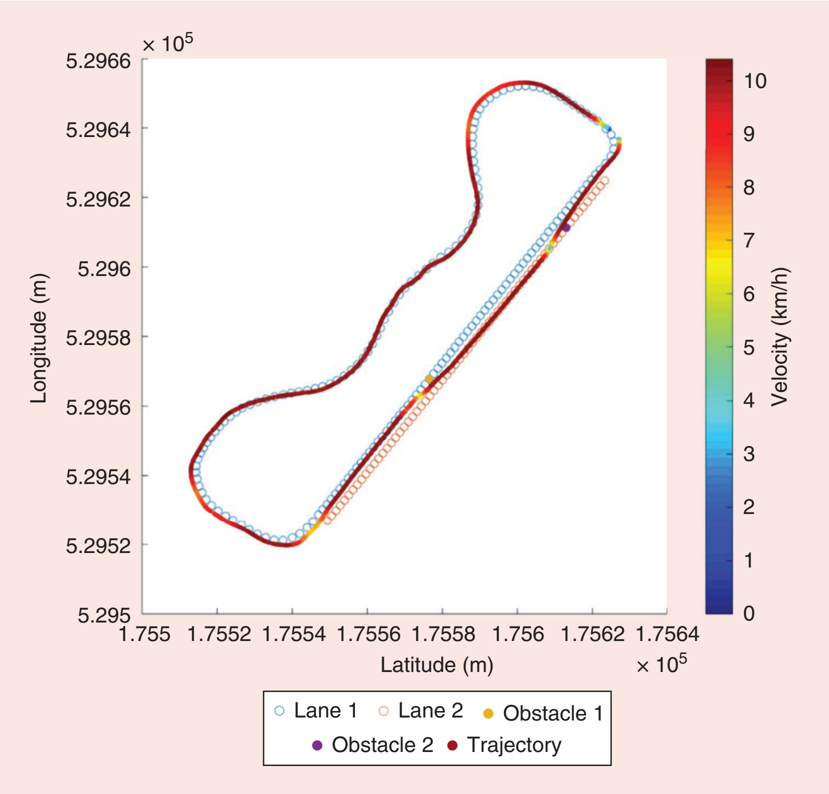


FIGURE 11. Locus of team rapid tractor that followed the racking track with several obstacles and two lanes.

adjusts the speed based on the curvature of the road to enable safe turning. Additionally, a path-tracking algorithm that combines the pure-pursuit algorithm with a look-ahead distance was applied to adaptively change path-tracking parameters based on various experimental results. The lidar-based system was developed in Python. It measured the distance to obstacles in the forward direction and decelerated accordingly, thereby enabling the vehicle to stop and change lanes in a satisfactory manner. The system could identify obstacles by clustering the point-cloud data based on density obtained with the lidar sensor, and it could enhance the accuracy by adjusting the ROI variably according to the curvature of the racing road. The lane-change method was implemented by determining obstacle-free lanes and following waypoints on those lanes after the obstacle detection. Both C/C++ and Python were utilized in the development process, and ROS was used to accommodate various languages. Figure 11 shows the locus the rapid tractor team vehicle traveled at different speeds following the racing track with several obstacles and two lanes.

AUTHORS

HeeChang Moon, Hongik University, 04066 Seoul, South Korea. E-mail: hcmoon@hongik.ac.kr.

Shin Han Kang, Halla University, 220-712 Gangwon-do, South Korea. E-mail: shkang@halla.ac.kr.

Jeongsik Eom, Halla University, 220-712 Gangwon-do, South Korea.

Myun Joong Hwang, University of Seoul, 02504 Seoul, South Korea. E-mail mjhwang@uos.ac.kr.

Youngmin Kim, University of Seoul, 02504 Seoul, South Korea.

Jungha Wang, Yonsei University, 03722 Seoul, South Korea.

Beomjun Kim, Yonsei University, 03722 Seoul, South Korea.

Taehyung Kim, Yonsei University, 03722 Seoul, South Korea.

Taekwon Ga, Yonsei University, 03722 Seoul, South Korea.

Jongeun Choi, Yonsei University, 03722 Seoul, South Korea. E-mail: jongeunchoi@yonsei.ac.kr.

Wonsang You, Sunmoon University, 336-708 Chungcheongnam-do, South Korea. E-mail: wyou@sunmoon.ac.kr.

Jiyou Shin, Sungkyunkwan University, 03063 Seoul, South Korea.

Jongsoo Han, Sungkyunkwan University, 03063 Seoul, South Korea.

Kyeongbeen Park, Sungkyunkwan University, 03063 Seoul, South Korea.

Hyungpil Moon, Sungkyunkwan University, 03063 Seoul, South Korea. E-mail: hyungpil@skku.edu.

Seok-Cheol Kee, Chungbuk National University, 28644 Chungbuk, South Korea. E-mail: sckee@chungbuk.ac.kr.

Hak-Jin Kim, Seoul National University, 08826 Seoul, South Korea. E-mail: kimhj69@snu.ac.kr.

Yong-Hyun Kim, Seoul National University, 08826 Seoul, South Korea.

Kibeom Lee, Gachon University, 13120 Gyeonggi-do, South Korea. E-mail: kibeom.lee@gachon.ac.kr.

Jaeseung Yu, Gachon University, 13120 Gyeonggi-do, South Korea.

REFERENCES

[1] J. Betz et al., “Autonomous vehicles on the edge: A survey on autonomous vehicle racing,” *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 458–488, Jun. 2022, doi: 10.1109/OJITS.2022.3181510.

[2] J. H. Park, Y. E. Lim, J. H. Choi, and M. J. Hwang, “Trajectory-based 3D point cloud ROI determination methods for autonomous mobile robot,” *IEEE Access*, vol. 11, pp. 8504–8522, Jan. 2023, doi: 10.1109/ACCESS.2023.3238824.

[3] J. Jeon, K. Lee, and D. Kum, “Overtaking decision and trajectory planning in highway via hierarchical architecture of conditional state machine and chance constrained model predictive control,” *Robot. Auton. Syst.*, vol. 151, May 2022, Art. no. 10414, doi: 10.1016/j.robot.2021.104014.

[4] R. C. Coulter et al., *Implementation of the Pure Pursuit Path Tracking Algorithm*. Pittsburgh, PA, USA: Carnegie Mellon Univ., The Robotics Inst., 1992.

[5] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi, “A stable tracking control method for an autonomous mobile robot,” in *Proc. IEEE Int. Conf. Robot. Autom.*, Cincinnati, OH, USA, 1990, vol. 1, pp. 384–389, doi: 10.1109/ROBOT.1990.126006.

[6] B. Kouvaritakis and M. Cannon, *Model Predictive Control*, vol. 38. Cham, Switzerland: Springer International Publishing, 2016.

[7] D. Lam, C. Manzie, and M. Good, “Model predictive contouring control,” in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, Atlanta, GA, USA, 2010, pp. 6137–6142, doi: 10.1109/CDC.2010.5717042.

[8] S. Thrun et al., “Stanley: The robot that won the DARPA Grand Challenge,” *J. Field Robot.*, vol. 23, no. 9, pp. 661–692, 2006, doi: 10.1002/rob.20147.

[9] J. Huang, Z. He, Y. Arakawa, and B. Dawton, “Trajectory planning in Frenet frame via multi-objective optimization,” *IEEE Access*, vol. 11, pp. 70,764–70,777, Jul. 2023, doi: 10.1109/ACCESS.2023.3294713.