Efficient Trajectory Planning for Autonomous Vehicles Using Quadratic Programming with Weak Duality

Dasol Jeong, Seibum B. Choi, *Member, IEEE,*

***Abstract*—Highly autonomous driving technology is expected to improve driving safety and convenience, and collision avoidance technology is essential for fully autonomous driving. Planning a collision-free trajectory that includes velocity and path is one of the most challenging objectives. Optimization-based trajectory planners have been proposed in many previous studies because they offer a high degree of freedom and can handle various situations. However, most previous trajectory planners used nonlinear programming due to the nonlinearity or non-convexity of the optimization problem. These methods come with a high computational load. The trajectory planner requires the real-time ability to cope with dynamically changing environments. This paper focuses on the trajectory planning of autonomous vehicles through quadratic programming (QP), which requires a low computational load. To achieve this, we introduce the longitudinal-lateral decomposition method. In addition, collision-free constraints are expressed as linear constraints through proposed ingenious dual functions. The proposed weak duality optimization problem has a QP form and optimized trajectory and obstacle avoidance timing through only one QP problem. This study verified that the proposed trajectory planner could plan smooth collision-free maneuvers for several driving situations by simulations.**

***Index Terms*—Autonomous vehicle, Model predictive control (MPC), Quadratic programming (QP), Trajectory planning, Weak duality.**

I. Introduction

INTELLIGENT vehicles are attracting attention with advances in sensor and control technology [1]. Advanced autonomous driving technology transfers control to the vehicle and improves driving safety and convenience [2]. These technologies have received considerable attention in academia, industry, and the military over the past few decades. Recent studies from the DARPA Grand and Urban challenge [3] to Google cars [4] and Tesla [5] have shown admirable autonomous driving performance. Nevertheless, some challenges remain to be addressed for a fully autonomous vehicle. One of these challenges is trajectory planning technology, essential for ensuring vehicle stability and safety during autonomous driving [6], [7]. Planning a trajectory that includes the path and velocity for dynamically changing vehicles and environments is a complicated challenge [8]. Trajectory planning techniques need to meet the driving purpose and offer a collision-free path and velocity. Furthermore, real-time implementation is necessary for the commercialization of autonomous vehicles [9].

Trajectory planning is mainly divided into sampling-based approaches and optimization-based approaches. Sampling-based approaches select the best path among finite collision-free path candidates [10]. Cubic splines [11] and quintic polynomials [12] have been used to generate path candidates. Recently, trajectory planning algorithms have been studied to plan the vehicle velocity simultaneously with the path [13]. The sampling-based approach is relatively inexpensive due to the characteristic of predetermining candidate paths. However, since the shape of paths is predetermined, there is a limit to expressing various driving maneuvers. While it may not be a problem for smooth maneuvers such as overtaking, it can pose significant challenges for more complex maneuvers like obstacle avoidance. In other words, these approaches are judged infeasible if there is no collision-free path among the predetermined paths, even if avoidance is possible. Therefore, these sampling-based approaches have limitations in terms of feasibility [14].

The optimization-based approaches have the advantage of having a higher degree of freedom than the sampling-based approach because the path is not predetermined. This advantage allows trajectory planning of various driving maneuvers. These approaches plan a path and velocity that satisfy the constraints and optimize the cost function. The cost function is set to the cost for ride comfort [15] and ego vehicle velocity [16]. Obstacle avoidance has usually been handled through constraints. In previous studies, obstacles were represented by constraints in the shape of circles [17] and ellipses [18]. However, these studies included non-convex cost functions or nonlinear constraints. Due to nonlinearity, nonlinear programming (NP), which has a significant computational load, has been used. The usage of NP has the fatal disadvantage of not guaranteeing the real-time implementation of the optimization-based approach. Therefore, reducing the computational load for real-time trajectory planning is essential [19].

Recently, some studies proposed optimizing trajectory through quadratic programming (QP) rather than NP [20], [21]. QP has a significantly lower computational load than NP. Therefore, it is attracting attention as a technology that can improve real-time implementation ability. One of the challenges is to express the conditions for obstacle avoidance using the QP form. Previous studies expressed obstacles as a quadratic

Dasol Jeong is Senior Research Engineer, Hyundai Motor Company, 150 HyundaiYeonguso-ro, Namyang-eup, Hwaseong-si, Gyeonggi-do, 18280, Republic of Korea (jds5513@hyundai.com)

Seibum B. Choi is Professor at the School of Mechanical, Department of Mechanical Engineering, at the Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro, Yuseong-gu, Daejeon 305-338, Republic of Korea (e-mail: sbchoi@kaist.ac.kr).

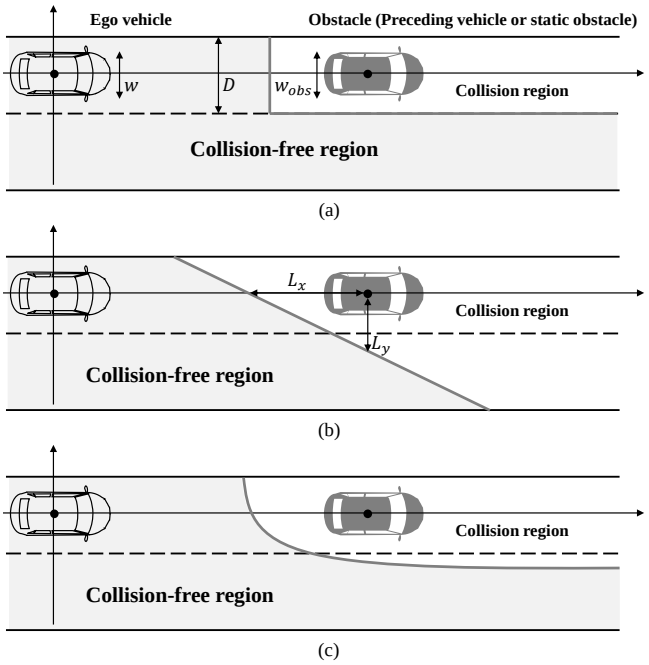

Fig. 1. Collision-free region for obstacle avoidance. (a) Nonlinear constraints. (b) Linear approximation constraints [20]. (c) Proposed constraints.

cost function and linear constraints. In [21], the modified cost function, which increased as the distance to the obstacle was closer, has been proposed for obstacle avoidance. However, the initial modified cost function did not ensure complete obstacle avoidance. Therefore it involved checking, modifying, and iterating until obstacle avoidance. These iterating processes had an enormous computational load than general QP. It also had a limitation in that it could harm the essence of the optimization cost function. In [20], obstacles were expressed as linear constraints, as shown in Fig. 1(b). This expression had advantages in terms of computational load, but linear constraints predetermined by L_x and L_y could cause fatal limitations. First, setting an overly conservative collision-free region could produce an unusual trajectory. Second, several tricks have been used because trajectory planning for overtaking obstacles by changing lanes was impossible with predetermined linear constraints, as shown in Fig. 1(b). Thus, expressing obstacles with linear constraints was an original concept, but several problems needed to be resolved.

We propose a trajectory planning algorithm through QP form. The cost function was not modified, and the obstacles were expressed as linear constraints. The proposed algorithm ensured a high degree of freedom and feasibility, as shown in Fig. 1(c). The longitudinal-lateral decomposition method was utilized for precise trajectory planning while maintaining the QP form [22]. This method separates and analyzes the collision-free region into two domains, time-longitudinal and time-lateral, rather than a longitudinal-lateral domain. In this case, obstacle avoidance timing, which means when to overtake obstacles, is an important parameter. In [23], a hybrid trajectory panning algorithm that analyzes longitudinally by optimization and laterally by sampling was proposed. Since the

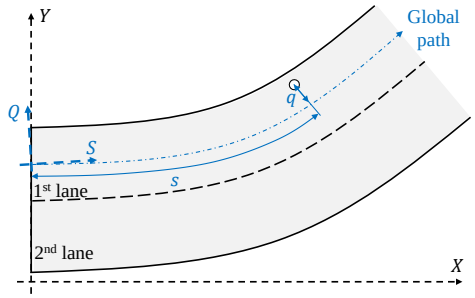

Fig. 2. Transformation into curvilinear coordinates.

lateral candidate path was predetermined, obstacle avoidance timing was also predetermined. However, since multiple longitudinal model predictive control (MPC) for multiple lateral paths was used, the computational load was significant. In [24], an optimization-based approach to both longitudinal and lateral paths was applied. Their algorithm predefined obstacle avoidance timing through several assumptions and applied MPC to each longitudinal and lateral path. However, the obstacle avoidance timing determines the optimality and smoothness of the trajectory. The analysis of this aspect has been inadequate, and the optimization of obstacle avoidance timing has not been addressed. Therefore, optimization of the obstacle avoidance timing in longitudinal and lateral decomposition is a task to be solved.

This study introduces a new optimization variable: information on obstacle avoidance timing. Trajectory planning, including obstacle avoidance timing, has usually been optimized through NP [24] or mixed integer quadratic programming (MIQP) [25] due to its nonlinearity. We propose a method to transform the primal NP problem into a weak duality QP problem. In a dual problem, all cost functions are expressed in quadratic form, and constraints are expressed as a linear combination of optimization variables. We provide the details in Section III. Finally, we optimized the trajectory and obstacle avoidance timing by only one QP. The main contributions of this study are as follows: (1) A trajectory planning algorithm through QP is proposed. (2) Weak duality linear constraints for collision avoidance are proposed. (3) Obstacle avoidance timings are simultaneously optimized with the trajectory. (4) Extension for multiple obstacles and situations.

The rest of this paper is organized as follows. Section II defines the problem and introduces the overall schematic of the proposed trajectory planner. In Section III, we construct the primal optimization problem for trajectory planning and propose weak duality QP. The performance validation results of the simulations are provided in Section IV. The conclusion and future work are presented in Section V.

II. Problem Statement

A. Curvilinear Coordinate

In this study, the station-lateral offset coordinate, denoted as the $S-Q$ coordinate, was employed to describe the surrounding environment and obstacles. This framework facilitates the representation of the ego-vehicle, environment, and obstacles

as a lateral offset and station relative to the global path, as shown in Fig. 2. The transition from the $X-Y$ to the $S-Q$ coordinates has been methodically formulated and leveraged in previous studies of trajectory planning [11], [13].

A noteworthy advantage of adopting the $S-Q$ coordinate system lies in its efficacy in planning trajectories for curved roads. In real life, encountering curved roads is inevitable, characterized by diverse curvatures. Under such circumstances, the $S-Q$ coordinate system emerges as a potent mechanism for effectively capturing the intricacies of curvilinear roads.

B. System Dynamics

For optimization-based trajectory planning, we performed a system dynamics analysis. Previous studies used the point mass model, bicycle model, and planar car model as system dynamics [26]. Among them, the point mass model has the advantage of simplifying the vehicle model. Although it is less accurate than other models, it is accurate enough for trajectory planning [27]. Therefore, it has often been used in the previous trajectory planning studies [28], [15]. Likewise, we focused on trajectory planning using QP rather than developing a precise vehicle model. Therefore, the following point mass kinematics were used as vehicle models:

$$\begin{aligned} s(k+1) &= s(k) + v_x(k)\triangle t + a_x(k)\frac{\triangle t^2}{2}, \\ v_x(k+1) &= v_x(k) + a_x(k)\triangle t, \\ q(k+1) &= q(k) + v_y(k)\triangle t + a_y(k)\frac{\triangle t^2}{2}, \\ v_y(k+1) &= v_y(k) + a_y(k)\triangle t \end{aligned} \tag{1}$$

where s and q are the station and lateral offset of the ego vehicle, as shown in Fig. 2. v_x and v_y are the longitudinal and lateral velocity relative to the global path, and a_x and a_y are the longitudinal and lateral acceleration relative to the global path, respectively. $\triangle t$ is the sampling time.

Furthermore, a lagged system was introduced to consider the difference between the point mass models and vehicle models.

$$\begin{aligned} a_x(k+1) &= e^{-\frac{\triangle t}{\tau_{l,x}}}a_x(k) + \left(1 - e^{-\frac{\triangle t}{\tau_{l,x}}}\right)a_{x,d}(k) \\ a_y(k+1) &= e^{-\frac{\triangle t}{\tau_{l,y}}}a_y(k) + \left(1 - e^{-\frac{\triangle t}{\tau_{l,y}}}\right)a_{y,d}(k) \end{aligned} \tag{2}$$

where $a_{x,d}(k)$ and $a_{y,d}(k)$ are desired longitudinal and lateral accelerations. $\tau_{l,x}$ and $\tau_{l,y}$ are time constants of longitudinal and lateral dynamics. In this study, considering the reactivity of articulated vehicles, $\tau_{l,x}$ and $\tau_{l,y}$ were set to $500\,ms$ each. The above model consists of the following state space dynamics:

$$\begin{aligned} &\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \\ &\text{where } \mathbf{x}(k) = \begin{bmatrix} s(k) & v_x(k) & q(k) & v_y(k) & a_x(k) & a_y(k) \end{bmatrix}^T, \\ &\mathbf{u}(k) = \begin{bmatrix} a_{x,d}(k) & a_{y,d}(k) \end{bmatrix}^T, \\ &\mathbf{A} = \begin{bmatrix} 1 & \triangle t & 0 & 0 & \triangle t^2/2 & 0 \\ 0 & 1 & 0 & 0 & \triangle t & 0 \\ 0 & 0 & 1 & \triangle t & 0 & \triangle t^2/2 \\ 0 & 0 & 0 & 1 & 0 & \triangle t \\ 0 & 0 & 0 & 0 & e^{-\frac{\triangle t}{\tau_{l,x}}} & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{-\frac{\triangle t}{\tau_{l,y}}} \end{bmatrix}, \\ &\mathbf{B} = \begin{bmatrix} \mathbf{0}_{4\times 2} \\ 1 - e^{-\frac{\triangle t}{\tau_{l,x}}} & 0 \\ 0 & 1 - e^{-\frac{\triangle t}{\tau_{l,y}}} \end{bmatrix} \end{aligned} \tag{3}$$

The system was a linear time-invariant system. This property had a significant advantage in terms of computational load. For this reason, lagged point mass models have been frequently used in trajectory planning.

Remark 1. *The proposed trajectory planning algorithm used the lagged point-mass model. This model, characterized by its simplicity and linearity, offers advantages in computational efficiency. Furthermore, we considered the lagged effect to address disparities between this model and vehicle dynamics. Nonetheless, this model may exhibit limitations in accurately representing real-world vehicles—these modeling errors, though present, are manageable.*

In recent autonomous vehicle studies, trajectory planning studies have adopted simplified models, while complex vehicle dynamics are considered in the tracking control [29], [30]. Establishing a safety margin during trajectory planning is crucial in these trajectory planning algorithms. The design of the tracking controller is rigorously constrained to ensure it operates within this safety margin, thereby guaranteeing the overall safety of the autonomous vehicle. The proposed algorithm used a safety margin of 50 cm on both sides. This safety margin compensates for modeling errors from using the point-mass model, enhancing the safety of the trajectory planning process.

C. Model Predictive Control

MPC is a well-known optimization algorithm. Future inputs are optimized based on predicted states of MPC. This method can easily consider system constraints and is robust against disturbance and model uncertainty due to the receding horizon method [31]. Several MPCs, such as linear MPC [12], robust MPC [32], and Laguerre MPC [33], have been applied in autonomous driving studies, including trajectory planning. In this study, the trajectory was planned and optimized based on MPC. The general MPC formulation is as in (4). Terminal cost ($\mathbf{P_f}$) is determined through an algebraic Riccati equation [34].

$$\min_{\mathbf{U}} J = \sum_{m=0}^{N_p-1} \|\mathbf{x}(k+m|k) - \mathbf{x_{des}}(k+m)\|_{\mathbf{Q}} \tag{4a}$$

TABLE I
TUNING PARAMETERS OF MPC.

Symbol	Parameter	Value
$\triangle t$	Sampling time	$100\,ms$
$\tau_{l,x}, \tau_{l,y}$	Time constants	$500\,ms$
$\mathbf{Q}$	Weight matrix for errors	$\mathrm{diag}(0, 1, 1, 1, 100, 100)$
$\mathbf{R}$	Weight matrix for input	$\mathrm{diag}(100, 100)$
N_p	Prediction horizon	50
N_c	Control horizon	50

$$+ \sum_{m=0}^{N_c-1} \|\mathbf{u}(k+m)\|_{\mathbf{R}} + \|\mathbf{x}(k+N_p|k)\|_{\mathbf{P_f}}$$

$$\text{s.t.} \quad \mathbf{x}(k+m+1|k) = \mathbf{A}\mathbf{x}(k+m|k) + \mathbf{B}\mathbf{u}(k+m) \tag{4b}$$
$$\mathbf{x}(k|k) = \mathbf{x}(k) \tag{4c}$$
$$\mathbf{x}(k+m|k) \in \mathcal{X}, \qquad m = 1, \cdots, N_p \tag{4d}$$
$$\mathbf{u}(k+m) \in \mathcal{U}, \qquad m = 0, \cdots, N_c - 1 \tag{4e}$$

where $\mathbf{x}(k)$ represents the states observed or estimated at k step, $\mathbf{x}(k+m|k)$ represents the predicted states at $k+m$ step from step k, $\mathbf{u}(k+m)$ represents the future input at time $k+m$ step, and $\mathbf{U}$ represents the optimized input sequence. In addition, $\mathbf{Q}$ and $\mathbf{R}$ are covariance matrices that determine the weights of tracking error, lagged input, and desired input. N_p and N_c are the prediction horizon and control horizon. The tuning parameters are given in Table I. $\bar{\mathcal{U}}$ is the set of input constraints, and $\bar{\mathcal{X}}$ is the set of state constraints. If $\mathcal{U}$ and $\mathcal{X}$ are non-convex or nonlinear, the MPC problem is optimized through NP. Conversely, if $\mathcal{U}$ and $\mathcal{X}$ are linear constraints, the MPC problem is optimized through QP. In other words, for trajectory planning using QP, it is essential to express obstacle avoidance constraints as linear constraints.

The control input is the first element of the solution $\mathbf{U}^*$ to the problem (4) as follows:

$$\mathbf{u}(k) = \mathbf{U}^*(1) \tag{5}$$

The minimum cost function (J_{min}), the optimized cost function, is defined as follows:

$$J_{min} = J\big|_{\mathbf{U}=\mathbf{U}^*} \tag{6}$$

In this study, two MPCs for lane-keeping and lane-change were applied. Both MPCs shared the same cost function. The driving mode with the lower J_{min} was selected in the judgment phase, and J_{min} was actively utilized.

D. Overall Schematic

Fig. 3 shows the strategy of the proposed trajectory planner. For a safe and realistic trajectory, trajectories for two situations were planned: lane-keeping mode and lane-change mode. Afterward, the planner selected the driving mode according to minimum cost functions. In general, the lane-keeping mode is preferred if there are no obstacles or there is a large gap with the preceding vehicle. However, in some situations, it is advantageous to adopt a lane-change mode.

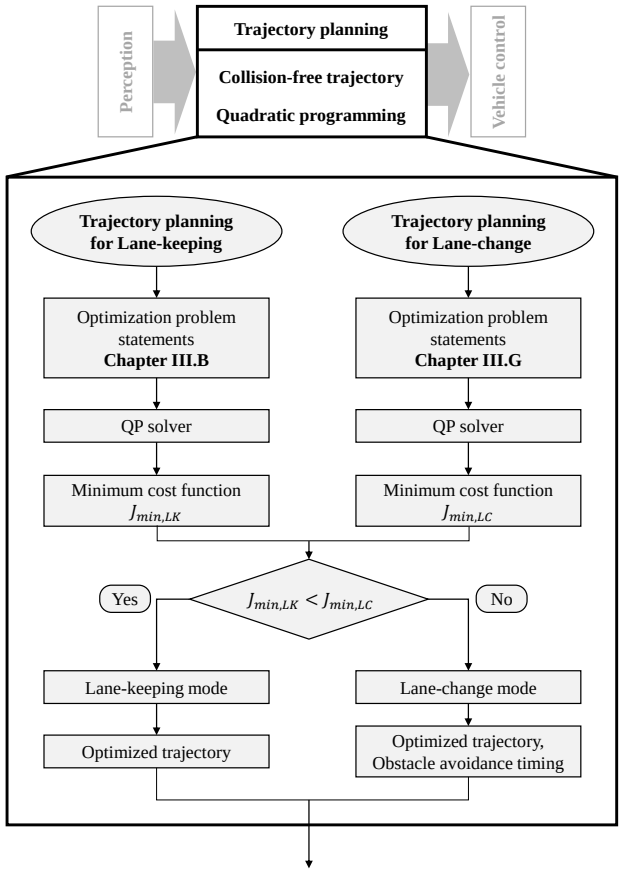

Fig. 3. Overall schematic of proposed trajectory planner.

- Overtaking: If the preceding vehicle is very slow relative to the desired velocity, overtaking may be a strategy to reduce the cost function.
- Static obstacle avoidance: An autonomous vehicle should avoid static obstacles such as construction or lane reduction roads. The autonomous vehicle should overcome the situation by adopting a lane-change mode.
- Emergency obstacle avoidance: An emergency situation in which braking alone cannot avoid a collision. The vehicle can avoid collision through the lane-change mode by planning the path and velocity simultaneously.

The mode change decision for the above situations was based on the minimum cost function of each MPC. Both MPCs shared the same cost function. Therefore, the mode with a low minimum cost function was judged to have a more optimal trajectory. In conclusion, the trajectory planner selected the trajectory with a lower minimum cost function.

III. TRAJECTORY PLANNING USING QUADRATIC PROGRAMMING

Trajectory planning is an essential technology in autonomous driving. This technology is responsible for planning a collision-free trajectory based on the perception sensor and handing it over to the tracking controller. We developed a lane-keeping mode that played an advanced cruise control (ACC)

role and a lane-change mode that avoided obstacles. The lane-keeping mode, which considers only the longitudinal relative distance and velocity to obstacles, has been proposed using linear MPC in many previous studies [35], [36]. Similarly, we designed the lane-keeping mode through linear MPC in this study.

The most challenging problem is the lane change mode. Simultaneously optimizing not only the path but also the velocity while avoiding obstacles is still challenging. In previous studies, algorithms were proposed through NP [24] and MIQP [25]. These algorithms had the limitation of having a massive computational load. Our proposed novel obstacle constraint processing algorithm maintains QP form while simultaneously considering longitudinal and lateral trajectories in lane-change mode. For this purpose, we used the longitudinal-lateral decomposition method. We also propose a weak duality problem that converts nonlinear constraints to linear constraints by introducing weakly dual functions. The proposed algorithm optimizes trajectory and obstacle avoidance timing by only one QP.

A. Cost Functions and General Constraints

This section defines the cost function and sets general constraints for vehicle safety. These apply equally to all driving situations. The lane-keeping mode and lane-change mode are proposed by adding constraints on obstacles.

1) Cost Function: A cost function is defined for optimization-based trajectory planning. The cost function is as (4a). The desired states are as follows:

$$\mathbf{x_{des}}(k+m) = \begin{cases} \begin{bmatrix} \cdot & v_{x,des} & 0 & 0 & 0 \end{bmatrix}^T & : \text{Lane-keeping} \\ \begin{bmatrix} \cdot & v_{x,des} & -D & 0 & 0 \end{bmatrix}^T & : \text{Lane-change} \end{cases} \tag{7}$$

where D is the lane width, $v_{x,des}$ is the desired velocity or road velocity limit. The cost function includes the difference with the desired velocity, the lateral position error for the global path, and the lateral velocity. In this study, the cost function for the station was not defined; consequently, the desired station was represented as a null value ($\cdot$). For the lane-keeping mode, the lateral position error is based on lane 1, and the lane-change mode is based on lane 2. In addition, acceleration and desired acceleration were considered by adding cost for a_x, a_y and $\mathbf{u}$. That was a cost function for ride comfort.

Remark 2. *The cost function encompasses a weighted summation of vehicle state values and inputs. The determination of these weights is contingent upon the weight matrix for errors (*$\mathbf{Q}$*) and the weight matrix for inputs (*$\mathbf{R}$*). Tuning these matrices is a crucial task and is imperative to consider the following physical considerations. Regarding the matrix* $\mathbf{Q}$*, its second diagonal element embodies the weight associated with longitudinal tracking, while the third and fourth diagonal elements correspond to the lateral tracking weight. Should expedited acceleration be desired, elevating the value of the second diagonal element is recommended. An enormous value for the third and fourth diagonal elements can facilitate swifter lane changes. The fifth and sixth diagonal elements of* $\mathbf{Q}$ *are equated to those of* $\mathbf{R}$*. Within* $\mathbf{R}$*, the first and second diagonal components signify the weights for longitudinal and lateral accelerations. Analogously, the weighting can be adjusted for the longitudinal and lateral directions by tuning these components. The magnitude of* $\mathbf{Q}$ *and* $\mathbf{R}$ *governs the relationship between tracking error and input effort. A larger* $\mathbf{Q}$ *enables fast acceleration and lane changes, and a larger* $\mathbf{R}$ *enables smooth and comfortable driving. This way, selecting appropriate weight matrices can achieve the desired purpose.*

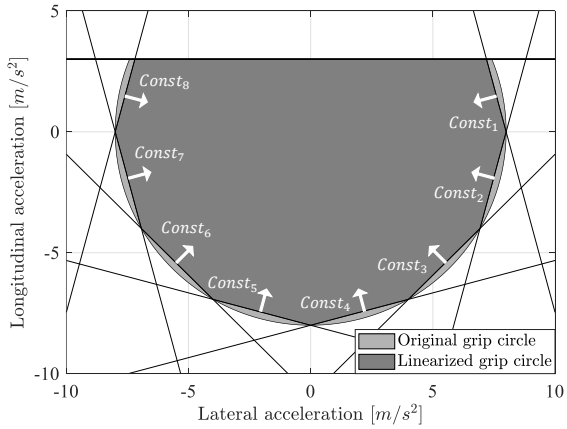

Fig. 4. Grip circle and linearized grip circle.

2) Constraints for Vehicle Stability: Longitudinal and lateral accelerations are limited due to constraints imposed by tire grip and engine power capabilities as follows:

$$\begin{aligned} &a_{x,d}(k+m) \le \overline{a}_x \\ &\sqrt{a_{x,d}(k+m)^2 + a_{y,d}(k+m)^2} \le a_{max}, \\ &\text{for } m = 0, \cdots, N_c - 1 \end{aligned} \tag{8}$$

Where the maximum longitudinal acceleration of the vehicle, denoted as $\overline{a}_x$, is set to $3\,m/s^2$, the combined maximum acceleration, denoted a_{max}, is set to $8\,m/s^2$. The graphical representation of (8) is shown in Fig. 4, where it appears in a light gray shade.

These constraints encompass quadratic components, thereby significantly elevating the computational intricacy. It is noteworthy, however, that the grip circle constraints exhibit convex characteristics. As a result, the expression (8) can be reformulated to following linearized constraints:

$$\begin{aligned} &a_{x,d}(k+m) \le \overline{a}_x \\ &Const_n : \left(\sin\left(\frac{n-1}{6}\pi\right) - \sin\left(\frac{n-2}{6}\pi\right)\right) a_{y,d}(k+m) \\ &\quad + \left(\cos\left(\frac{n-1}{6}\pi\right) - \cos\left(\frac{n-2}{6}\pi\right)\right) a_{x,d}(k+m) \le \frac{a_{max}}{2} \\ &\text{for } n = 1, \cdots, 8, \quad m = 0, \cdots, N_c - 1 \end{aligned} \tag{9}$$

We also set constraints for lanes. We assumed the situation of avoiding obstacles by moving to the right lane. Therefore,

the constraint that the vehicle should be located inside the current and right lanes was set as follows:

$$-\frac{D-w}{2}-D \leq y(k+m) \leq \frac{D-w}{2}, \, m=1,\cdots,N_p \tag{10}$$

where w is the ego vehicle width. We assumed the situation of avoiding obstacles by moving to the right lane. Furthermore, avoiding obstacles by moving to the left lane or to off-road conditions can be easily applied.

B. Lane-keeping Mode

Lane-keeping mode played the role of ACC for the preceding vehicle. This mode did not consider obstacle avoidance in the lateral direction. In other words, through longitudinal trajectory planning, the ego vehicle kept close to the desired velocity while maintaining a certain distance from the preceding vehicle or moving obstacle. The cost function was the same as (4a). The obstacle's future trajectory was predicted by its current position ($s_O(k)$) and its current longitudinal velocity ($v_O(k)$) as follows:

$$s_O(k+m) = s_O(k) + v_O(k)(m\triangle t) \tag{11}$$

The relative distance constraints to the target obstacle are as follows:

$$\begin{aligned} s(k+m) &\leq s_O(k+m) - \overline{s}, \\ \overline{s} &= \overline{s}_0 + v_x(k) t_H \end{aligned} \tag{12}$$

where $\overline{s}_0$ is the standstill distance and t_H is the time headway. The above relative distance formula has been used in several ACC studies [37]. Based on previous studies, $\overline{s}_0 = 5\,m$ and $t_H = 2\,sec$ were set.

In conclusion, the optimization problem for trajectory planning for lane-keeping mode is as follows:

$$\min_{\mathbf{U}} \quad (4a), \quad \text{s.t. } (4b),\ (8),\ (10),\ (12) \tag{13}$$

(13) had a QP form that can be optimized through a general QP solver. The minimum cost function when selecting the lane-keeping mode is as follows:

$$J_{min,LK} = (4a)\big|_{\mathbf{U}=\mathbf{U}^*_{LK}} \tag{14}$$

where $\mathbf{U}^*_{LK}$ is the optimized input sequence of (13). $J_{min,LK}$ was compared with the minimum cost function of lane-change mode to determine whether to change mode.

C. Lane-change Mode: Primal Optimization Problem

Lane-change mode is a situation that includes lateral movement to avoid collision with obstacles. In general driving, lane keeping is preferred, but lane change is essential in situations such as overtaking and obstacle avoidance. This section introduces the primal optimization problem for optimization-based lane change trajectory planning.

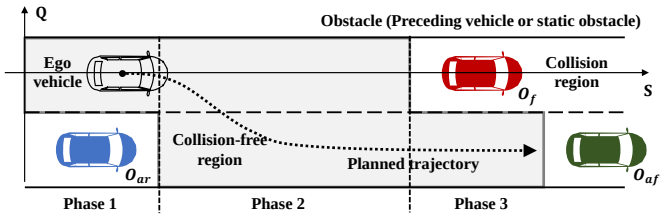

Fig. 5. Collision-free region, collision region, and obstacles.

The collision-free region can be expressed as $\mathcal{CF}$ set, as shown in Fig. 5. Collision-free constraints are expressed as:

$$(s(k+m), q(k+m)) \in \mathcal{CF}, \, m=1,\cdots,N_p \tag{15}$$

where $\mathcal{CF}$ is the set of the collision-free region as shown in Fig. 5. The primal optimization problem for lane-change mode is defined as follows:

$$\min_{\mathbf{U}} \quad (4a), \quad \text{s.t. } (4b),\ (8),\ (10),\ (15) \tag{16}$$

Due to nonlinear collision-free constraints, the primal optimization problem should be optimized via NP [24] or MIQP [25]. NP or MIQP causes excessive computational load. These limitations are a significant weakness in the real-time implementation of autonomous driving technology.

D. Strategy of Proposed Algorithm

This section introduces the concept of converting method trajectory planning in lane-change mode to QP form. The proposed trajectory planning algorithm, expressed in QP form, guaranteed a lower computational load than previous studies such as NP or MIQP. We propose our algorithm on the basis of the longitudinal-lateral decomposition method. Unlike previous studies, this study simultaneously optimized trajectory and obstacle avoidance timing ($N_{OAT,s}, N_{OAT,e}$). For this, we introduced ingenious dual functions and added an optimization variable with information about $N_{OAT,s}, N_{OAT,e}$. Finally, a weak duality QP problem is proposed.

1) Considered obstacles: Distinct from the lane-keeping mode, which solely accounts for the preceding vehicle, the lane-change mode encompasses an all-encompassing assessment of proximate obstacles. As shown in Fig. 5, this study considered various obstacles: front obstacle within the same lane (O_f), rear obstacle in the different lane (O_{ar}), and front obstacle in the different lane (O_{af}). The requisite information for each obstacle is as (17). Furthermore, for algorithmic simplicity, we assumed that obstacles are positioned at the precise center of each respective lane. It can readily be rectified when such is not central alignment through a straightforward adjustment of obstacle width.

$$\begin{aligned} s_{O,(\cdot)} &: \text{Initial stations of obstacles} \\ l_{O,(\cdot)} &: \text{Lengths of obstacles} \\ v_{O,(\cdot)} &: \text{Initial velocities of obstacles} \\ w_{O,(\cdot)} &: \text{Widths of obstacles} \end{aligned} \tag{17}$$

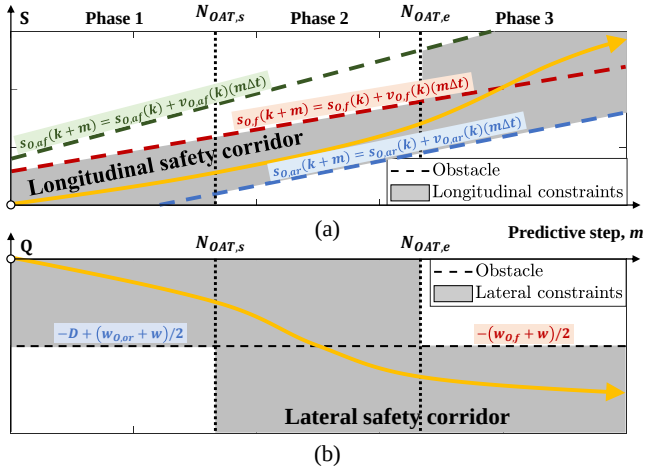

Fig. 6. Collision-free constraints using longitudinal-lateral decomposition method. (a) Longitudinal constraints. (b) Lateral constraints.

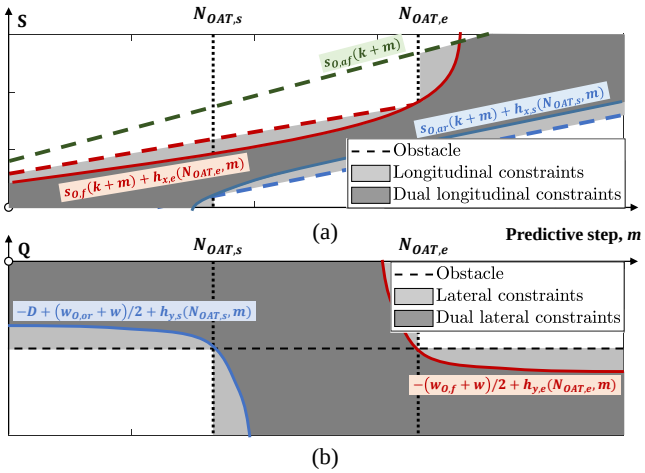

Fig. 7. Collision-free constraints of primal and dual optimization problems. (a) Longitudinal constraints. (b) Lateral constraints.

2) Longitudinal-lateral Decomposition: The longitudinal-lateral decomposition algorithm planned the trajectory in the T-S (time-longitudinal) and T-Q (time-lateral) coordinate systems instead of the S-Q coordinate system [23], [24], as shown in Fig. 6. The light gray regions in Fig. 6 represent collision-free constraints. Longitudinal and lateral constraints are mathematically expressed according to phase as:

- Phase 1: Pre region (Before lane change initiation), $m = 1, \cdots, N_{OAT,s}$

$$\begin{aligned} -\infty \leq & s(k+m) \leq s_{O,f}(k+m), \\ -D + \frac{w + w_{O,or}}{2} \leq & q(k+m) \leq \infty \end{aligned} \tag{18}$$

- Phase 2: Peri region (During lane change execution), $m = N_{OAT,s}, \cdots, N_{OAT,e}$

$$s_{O,ar}(k+m) \leq s(k+m) \leq s_{O,f}(k+m) \tag{19}$$

- Phase 3: Post region (Upon lane change completion), $m = N_{OAT,e}, \cdots, N_p$

$$\begin{aligned} s_{O,ar}(k+m) \leq & s(k+m) \leq s_{O,af}(k+m), \\ -\infty \leq & q(k+m) \leq -\frac{w + w_{O,f}}{2} \end{aligned} \tag{20}$$

Here, $N_{OAT,s}$ $N_{OAT,e}$ are the obstacle avoidance timings. Given $N_{OAT,(\cdot)}$, a collision-free region can be expressed with linear constraints as in (18)-(20). Therefore, determining and optimizing $N_{OAT,(\cdot)}$ is essential to the longitudinal-lateral decomposition method. In previous studies, $N_{OAT,(\cdot)}$ was selected through multiple candidate paths or was used as a fixed value through various assumptions. When using many $N_{OAT,(\cdot)}$ candidates, the computational load was multiplied by that much. Moreover, when using $N_{OAT,(\cdot)}$ as a fixed value, the planned trajectory might not be optimal. In this study, we propose a weak duality QP problem that simultaneously optimizes trajectory and $N_{OAT,(\cdot)}$ and maintains the QP form.

3) Strategy of Proposed Algorithm: Fig. 7 shows the algorithm's strategy in trajectory planning for obstacle avoidance situations. The light gray area represents the collision-free constraints of the primal optimization problem. The dark gray area represents the dual collision-free constraints through the proposed dual functions. Dual constraints are expressed as follows:

$$\begin{aligned} N_{OAT,s} \rightarrow & \begin{cases} s(k+m) \geq s_{O,ar}(k+m) + h_{x,s}(N_{OAT,s}, m) \\ q(k+m) \geq -D + \frac{w_{O,or}+w}{2} + h_{y,s}(N_{OAT,s}, m) \end{cases} \\ N_{OAT,e} \rightarrow & \begin{cases} s(k+m) \leq s_{O,f}(k+m) + h_{x,e}(N_{OAT,e}, m) \\ q(k+m) \leq -\frac{w_{O,f}+w}{2} + h_{y,e}(N_{OAT,e}, m) \end{cases} \\ & s(k+m) \leq s_{O,af}(k+m) \\ & \text{for} m = 1, \cdots, N_p \end{aligned} \tag{21}$$

where $h_{x,(\cdot)}(\cdot,\cdot)$ is the dual longitudinal constraint function, $h_{y,(\cdot)}(\cdot,\cdot)$ is the dual lateral constraint function, and $h_{x,(\cdot)}(\cdot,\cdot)$ and $h_{y,(\cdot)}(\cdot,\cdot)$ change according to $N_{OAT,(\cdot)}$. The introduction of the dual function makes it possible to express continuous constraints rather than conditional constraints. Furthermore, nonlinearity can be eliminated if those functions can be expressed as linear constraints of optimization variables. This study aimed to convert collision-free constraints, which were nonlinear constraints, into linear constraints by introducing $h_{x,(\cdot)}(\cdot,\cdot)$ and $h_{y,(\cdot)}(\cdot,\cdot)$. For the above strategy, proper selection of $h_{x,(\cdot)}(\cdot,\cdot)$ and $h_{y,(\cdot)}(\cdot,\cdot)$ was essential. Next, we analyzed the necessary conditions and proposed the function.

Remark 3. *The dual functions, denoted as $h_{(\cdot),s}$ and $h_{(\cdot),e}$, encompass information corresponding to $N_{OAT,s}$ and $N_{OAT,e}$, and share analogous attributes. The driving scenario predominantly involves the avoidance of a forward obstacle within the same lane (O_f). Consequently, we first analyze the characteristics of $h_{(\cdot),e}$, which is a function for $N_{OAT,e}$, and propose a dual function based on this. After, we extended and proposed the dual constraints $h_{(\cdot),e}$ for the rear obstacle of the other lane.*

E. Weak Dual Constraints for Front Obstacle within the Same Lane (O_f)

1) Necessary Conditions of $h_{(\cdot),e}$: This section analyzes the necessary conditions and proposes dual constraint functions. Dual functions must 1) represent existing constraints well. In other words, the duality should be secured as much as possible. 2) The dual optimization problem should be expressed in QP form. 3) $N_{OAT,e}$ should also be optimized simultaneously.

- Primal constraints representation

A function should represent primal constraints well. First, the ego vehicle should pass through an obstacle when $m = N_{OAT,e}$ as follows:

$$\begin{aligned} h_{x,e}(N_{OAT,e},m)\big|_{m=N_{OAT,e}} &= h_{x,e}(N_{OAT,e},N_{OAT,e}) = 0 \\ h_{x,e}(N_{OAT,e},m)\big|_{m=N_{OAT,e}} &= h_{y,e}(N_{OAT,e},N_{OAT,e}) = 0 \end{aligned} \tag{22}$$

(22) has to satisfy the condition for all $N_{OAT,e}$. Thus, (22) led to the following necessary condition:

$$\begin{aligned} h_{x,e}(N_{OAT,e},m) &= C_x\left(\frac{g_{x,e}(N_{OAT,e})}{g_{x,e}(m)} - 1\right) \\ h_{y,e}(N_{OAT,e},m) &= C_y\left(\frac{g_{y,e}(N_{OAT,e})}{g_{y,e}(m)} - 1\right) \end{aligned} \tag{23}$$

where C_x and C_y are constant tuning parameters. $g_{x,e}(\cdot)$ and $g_{y,e}(\cdot)$ are decomposed functions of $h_{x,e}(\cdot,\cdot)$, $h_{y,e}(\cdot,\cdot)$. The problem of function selection was the same as the selection of C_x, C_y, $g_{x,e}(\cdot)$, and $g_{y,e}(\cdot)$.

In addition, before $N_{OAT,e}$, the ego vehicle cannot overtake an obstacle and does not have to complete lateral avoidance. After $N_{OAT,e}$, the ego vehicle can overtake and require the ego vehicle to complete lateral avoidance. The above conditions are expressed as follows:

$$\begin{aligned} h_{x,e}(N_{OAT,e},m) &\begin{cases} \leq 0 & \text{if } m \leq N_{OAT,e} \\ > 0 & \text{else} \end{cases} \\ h_{y,e}(N_{OAT,e},m) &\begin{cases} \geq 0 & \text{if } m \leq N_{OAT,e} \\ < 0 & \text{else} \end{cases} \end{aligned} \tag{24}$$

- QP form necessary condition

The ultimate goal of proposing a dual optimization problem was to transform the primal optimization problem into a QP form. Therefore, (21) had to be expressed as linear constraints of the optimization variable. Therefore, the following necessary condition was defined.

$$g_{y,e}(N_{OAT,e}) = a \times g_{x,e}(N_{OAT,e}) + b \tag{25}$$

where a and b are coefficients of a linear relationship.

- Adding an optimization variable

In dual QP, $g_{x,e}(N_{OAT,e})$ was selected as an additional optimization variable.

$$\gamma_e \equiv g_{x,e}(N_{OAT,e}) \tag{26}$$

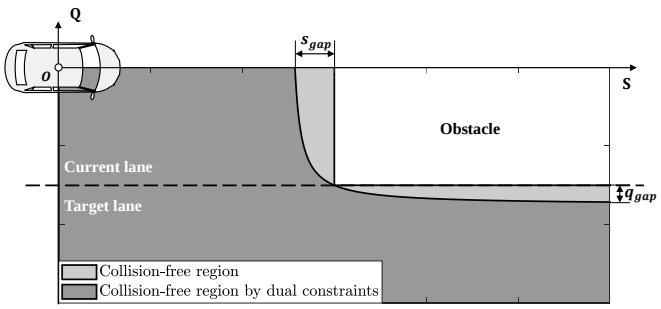

Fig. 8. Collision-free constraints in S-Q coordinate.

The optimization variable, γ_e, contained information about $N_{OAT,e}$. Later, $N_{OAT,e}$ was calculated as the inverse of $g_{x,e}(N_{OAT,e})$, so $g_{x,e}(N_{OAT,e})$ should satisfy the following conditions:

$$g_{x,e}(N_{OAT,e}) \text{ is one-to-one function on } N_{OAT,e} \in [1, N_p] \tag{27}$$

Applying (21) - (27), the dual longitudinal and lateral constraints are expressed as follows:

$$\begin{aligned} &s(k+m) \leq s_{O,f}(k+m) + C_x\left(\frac{\gamma_e}{g_{x,e}(m)} - 1\right), \\ &q(k+m) \leq -\frac{w + w_{O,f}}{2} + C_y\left(\frac{a\gamma_e + b}{g_{y,e}(m)} - 1\right), \\ &m = 1, \cdots, N_p \end{aligned} \tag{28}$$

2) Proposing Dual Constraint Functions: In this study, the following function was proposed as a dual function. The proposed function satisfies all the conditions of (21) - (27).

$$g_{x,e}(m) = \cos^2\left(\frac{\pi}{2}\frac{m}{N_p+1}\right),\ g_{y,e}(m) = \sin^2\left(\frac{\pi}{2}\frac{m}{N_p+1}\right) \tag{29}$$

3) Parameter Tuning and Time Shifting: After proposing dual functions, the parameters and components constituting the function are tuned.

- Parameter tuning

C_x and C_y are tuning parameters, and their physical meaning is shown in Fig. 8, which shows the dual longitudinal and lateral constraints in station-lateral offset (S-Q) coordinates. s_{gap} and q_{gap} represent the difference between primal constraints and dual constraints. The maximum values of s_{gap} and q_{gap} are as follows:

$$|s_{gap}| \leq |C_x|, |q_{gap}| \leq |C_y| \tag{30}$$

As above, C_x and C_y determine the upper limit of s_{gap} and q_{gap}. In this study, C_x was set according to the current ego vehicle velocity, and C_y was set $25\,\%$ of the width of the lane width as follows:

$$C_x = v_x(k) \times 1.3\,sec, \quad C_y = 1\,m \tag{31}$$

- Time shifting

A time-shifting method was introduced to improve the algorithm's stability, feasibility, and trajectory continuity as follows:

$$\begin{aligned} g_{x,e,TS}(m,\tau_e) &= \cos^2\left(\frac{\pi}{2}\frac{m+\tau_e}{N_p+1}\right) \\ g_{y,e,TS}(m,\tau_e) &= \sin^2\left(\frac{\pi}{2}\frac{m+\tau_e}{N_p+1}\right) \end{aligned} \tag{32}$$

where τ_e is the time-shifting factor for O_f. For selecting τ_e, the following is assumed.

Assumption 1. *The obstacle avoidance timing optimized at the k step ($N_{OAT,e}(k)$) is smaller than the obstacle avoidance timing optimized at the $k-1$ step ($N_{OAT,e}(k-1)$).*

This assumption was physically reasonable because the remaining time to avoid obstacles decreased over the step. It also led to two conditions. 1) In the k step, collision-free constraints over $m > N_{OAT,e}(k-1)$ did not need to be considered. 2) By a necessary condition of (27), $N_{OAT,e}(k)+\tau \le N_p$. If $\tau_e = N_p - N_{OAT,e}(k-1)$ was set, (27) was satisfied. Finally, dual collision-free constraints with time-shifting applied are defined as follows:

$$\begin{aligned} &s(k+m) \le s_{O,f}(k+m) + C_x\left(\frac{\gamma_{e,TS}}{g_{x,e,TS}(m,\tau_e)} - 1\right), \\ &q(k+m) \le -\frac{w+w_{O,f}}{2} + C_y\left(\frac{1-\gamma_{e,TS}}{g_{y,e,TS}(m,\tau_e)} - 1\right), \\ &m = 1,\cdots,N_{OAT,e}(k-1) \\ &\text{where } \gamma_{e,TS} = g_{x,e,TS}(N_{OAT,e},\tau), \tau = N_p - N_{OAT,e}(k-1) \end{aligned} \tag{33}$$

F. Weak Dual Constraints for Rear Obstacle in Different Lane (O_{ar})

The weakly dual collision-free constraints initially established for the front obstacle within the same lane, denoted as (O_f), can be readily expanded to encompass the rear obstacle in the different lane (O_{ar}). A pivotal observation is that the weak duality constraints about O_f and O_{ar} exhibit notable similarities.

This correlation is effectively depicted in Fig. 7. Within this visual representation, the red lines represent the weak duality constraints associated with O_f, while the blue lines correspond to O_{ar}. Upon scrutinizing the shapes delineated in Fig. 7(a) and (b), the symmetry between the blue and red lines becomes evident. Hence, the dual functions for O_{ar} can be proposed by applying origin symmetrization to the dual functions of O_f.

The weak dual functions for O_{ar} can be expressed as (34). At this time, there is a difference in the sign because of the origin symmetry.

$$\begin{aligned} h_{x,s}(N_{OAT,s},m) &= -C_x\left(\frac{g_{x,s}(N_{OAT,s})}{g_{x,s}(m)} - 1\right) \\ h_{y,s}(N_{OAT,s},m) &= -C_y\left(\frac{g_{y,s}(N_{OAT,s})}{g_{y,s}(m)} - 1\right) \end{aligned} \tag{34}$$

The function $g_{(\cdot),s}$ is proposed utilizing trigonometric functions, akin to the approach applied in O_f. Also, the following was proposed reflecting the shape of the origin symmetric type.

$$g_{x,s}(m) = \sin^2\left(\frac{\pi}{2}\frac{m}{N_p+1}\right),\ g_{y,s}(m) = \cos^2\left(\frac{\pi}{2}\frac{m}{N_p+1}\right) \tag{35}$$

The proposed function $g_{(\cdot),s}$ satisfies all previously established necessary conditions. Subsequently, employing the time-shifting technique, an extension of the weak duality function was proposed.

$$\begin{aligned} g_{x,s,TS}(m,\tau_s) &= \sin^2\left(\frac{\pi}{2}\frac{m+\tau_s}{N_p+1}\right) \\ g_{y,s,TS}(m,\tau_s) &= \cos^2\left(\frac{\pi}{2}\frac{m+\tau_s}{N_p+1}\right) \end{aligned} \tag{36}$$

where τ_s is the time-shifting factor for O_{or}. Similar to the treatment of O_f, the selection of τ_s adheres to the following assumption. As a result, τ_s was set as $\tau_s = N_p - N_{OAT,s}(k-1)$.

Assumption 2. *The obstacle avoidance timing optimized at the k step ($N_{OAT,s}(k)$) is smaller than the obstacle avoidance timing optimized at the $k-1$ step ($N_{OAT,s}(k-1)$).*

Finally, the weak dual constraints for the rear obstacle in the different lanes were extended as (37). These are the solid blue line constraints in Fig 7.

$$\begin{aligned} &s(k+m) \ge s_{O,ar}(k+m) - C_x\left(\frac{\gamma_{s,TS}}{g_{x,s,TS}(m,\tau_s)} - 1\right), \\ &q(k+m) \ge -D + \frac{w+w_{O,ar}}{2} - C_y\left(\frac{1-\gamma_{s,TS}}{g_{y,s,TS}(m,\tau_s)} - 1\right), \\ &m = 1,\cdots,N_{OAT,s}(k-1) \\ &\text{where } \gamma_{s,TS} = g_{x,s,TS}(N_{OAT,s},\tau), \tau = N_p - N_{OAT,s}(k-1) \end{aligned} \tag{37}$$

G. Lane-change Mode: Proposed Weak Duality QP Problem

The weak duality QP problem was proposed as follows:

$$\begin{aligned} \min_{\mathbf{U},\gamma_{e,TS},\gamma_{s,TS}}\ & J = \sum_{m=0}^{N_p-1} \|\mathbf{x}(k+m|k) - \mathbf{x_{des}}(k+m)\|_{\mathbf{Q}} \\ &+ \sum_{m=0}^{N_c-1} \|\mathbf{u}(k+m)\|_{\mathbf{R}} + \|\mathbf{x}(k+N_p|k)\|_{\mathbf{P_f}} \\ &+ \omega_{\gamma_e}\gamma_{e,TS}^2 + \omega_{\gamma_s}\gamma_{s,TS}^2 \\ &\text{s.t. } (4b),\ (9),\ (10),\ (33),(37) \end{aligned} \tag{38}$$

where ω_{γ_e}, ω_{γ_s} are the weight of $\gamma_{e,TS}$, $\gamma_{s,TS}$, set to a sufficiently small number. It had a negligible effect on the primal cost function. (38) is expressed as a quadratic cost function and linear constraints for the optimization variables ($\mathbf{U}, \gamma_{e,TS}, \gamma_{s,TS}$), which means that the dual optimization problem has QP form, not NP form. Finally, (38) was optimized through a general QP solver to find the optimal $\mathbf{U}^*_{LC}$, $\gamma^*_{e,TS}$, and $\gamma^*_{s,TS}$. Here, ($\gamma^*_{e,TS}, N_{OAT,e}$) and ($\gamma^*_{s,TS}, N_{OAT,s}$)

had a one-to-one correspondence as in (27). Thus, the optimized $N_{OAT,e}$, $N_{OAT,s}$ was expressed as follows:

$$N_{OAT,e}(k) = N_{OAT,e}(k-1) - \frac{2(N_p+1)}{\pi}\arccos\left({\gamma^*_{e,TS}}^{0.5}\right)$$
$$N_{OAT,s}(k) = N_{OAT,s}(k-1) - \frac{2(N_p+1)}{\pi}\arcsin\left({\gamma^*_{s,TS}}^{0.5}\right) \tag{39}$$

The minimum cost function of the lane-change mode is as follows:

$$J_{min,LC} = (4a)\big|_{\mathbf{U}=\mathbf{U}^*_{LC}} \tag{40}$$

Finally, by comparing (14) and (40), the driving mode was selected among lane-keeping mode and lane-change mode.

Remark 4. *Dual QP problems always showed conservative results compared to primal optimization problems. As a result, the following relation was established:*

$$J_{min,LC} \geq J_{min,LC,primal} \tag{41}$$

where $J_{min,LC,primal}$ is the minimum cost function of the primal optimization problem as (16). (41) meant that the proposed dual QP problem was a weakly dual problem.

The pseudo-code of the overall algorithm is summarized in Algorithm 1.

Algorithm 1 Algorithm for proposed trajectory planning algorithm

Input: $\mathbf{x}(k)$, $N_{OAT,(\cdot)}(k-1)$, $s_{O,(\cdot)}(k)$, $v_{O,(\cdot)}(k)$.
Output: Driving mode, Optimized trajectory, $N_{OAT,(\cdot)}(k)$.
Initialisation : $\mathbf{Q}, \mathbf{R}, \omega_{\gamma_e}, \omega_{\gamma_s} > 0$, $N_p, N_c \in \mathbf{Z}$, $N_{OAT,e}(0) = N_p$, $N_{OAT,s}(0) = N_p$, constraints, tuning parameters.
compute $\mathbf{U}^*_{LK}$, $J_{min,LK}$ using (13)
compute $\mathbf{U}^*_{LC}$, $N_{OAT,(\cdot)}(k)$, $J_{min,LC}$ using (38), (39)
if $J_{min,LK} \leq J_{min,LC}$ **then**
  Driving mode ← Lane-keeping
  Optimized trajectory ← $\mathbf{U}^*_{LK}$
  $N_{OAT,e}(k) \leftarrow N_p$
  $N_{OAT,s}(k) \leftarrow N_p$
else
  Driving mode ← Lane-change
  Optimized trajectory ← $\mathbf{U}^*_{LC}$
  $N_{OAT,e}(k) \leftarrow N_{OAT,e}(k)$
  $N_{OAT,s}(k) \leftarrow N_{OAT,s}(k)$
end if

Remark 5. *The variables $N_{OAT,e}$ and $N_{OAT,s}$ are crucial within trajectory planning, necessitating adjusting suitable initial values and continuous updating. The proposed algorithm used a time-shifting technique for robustness and performance. For this purpose, Assumptions 1 and 2 assumed that $N_{OAT,(\cdot)}(k)$ is less than $N_{OAT,(\cdot)}(k-1)$. Therefore, the initial and update values of $N_{OAT,(\cdot)}$ are set to the largest prediction horizon, N_p. These initial conditions and updates of $N_{OAT,(\cdot)}$ are outlined in Algorithm 1.*

Remark 6. *This study proposed a lane-change mode to address the change from lane 1 to lane 2. The proposed algorithm can be extended to scenarios changing from lane 2 to lane 1. Because the case of changing from lane 2 to lane 1 coincides with the up-down symmetry for the situation of changing from lane 1 to lane 2, it is easily extended by changing the coordinates as follows:*

$$\mathbf{x}(k) \leftarrow \mathbf{x}'(k), \mathbf{u}(k) \leftarrow \mathbf{u}'(k)$$
where
$$\mathbf{x}'(k) = \begin{bmatrix} s(k) & v_x(k) & -D-q(k) & -v_y(k) & a_x(k) & -a_y(k) \end{bmatrix}^T,$$
$$\mathbf{u}'(k) = \begin{bmatrix} a_{x,d}(k) & -a_{y,d}(k) \end{bmatrix}^T \tag{42}$$

Consequently, using ($\mathbf{x}'(k), \mathbf{u}'(k)$) instead of ($\mathbf{x}(k), \mathbf{u}(k)$) as states and inputs can plan lane-change trajectory from lane 2 to lane 1. As such, the proposed algorithm is scalable in many ways.

IV. SIMULATION RESULTS

A. Simulation Environments

The performance of the proposed trajectory planning algorithm was verified using simulations. The simulated environment encompasses obstacles, including vehicles and construction sites. The proposed algorithm plans the trajectory by selecting either lane-keeping mode or lane-change mode according to the situation. Simulation situations include avoidance of the construction site considering the surrounding environment, overtaking, and preceding vehicle following. The ego vehicle and obstacle information are shown in Table II.

The simulation assumes the controller can introduce an additional error of $0.5\,m$. Modeling errors between the lagged-point mass model and actual vehicles can cause this error. The trajectory was planned conservatively by $0.5\,m$ in the lateral direction to ensure robustness that always avoids obstacles despite modeling errors. Furthermore, situations involving lane departure occurring at the vehicle's vertex, according to the vehicle's heading angle, can also be effectively averted. Also, the *'quadprog'* function of MATLAB was used as a QP solver.

B. Simulation results

Fig. 9 shows the simulation result applying the proposed algorithm. Fig. 9(a) shows the trajectories of the ego vehicle and obstacles. The ego vehicle is visually represented as a blue rectangle, while a solid blue line denotes its trajectory. Other square elements represent obstacles. Obstacle 1 represents a static obstacle, such as construction sites or lane reduction areas. Obstacles 2-5 represent the surrounding vehicles as dynamic obstacles. As a result of the simulation, appropriately to the surrounding situation, the proposed algorithm plans a trajectory that includes obstacle avoidance, lane change, and preceding vehicle following.

Fig. 9(b) shows the ego vehicle's longitudinal velocity and acceleration. Longitudinal velocity planning is included in the proposed algorithm. The proposed algorithm plans the deceleration and acceleration profiles appropriate for the purpose.

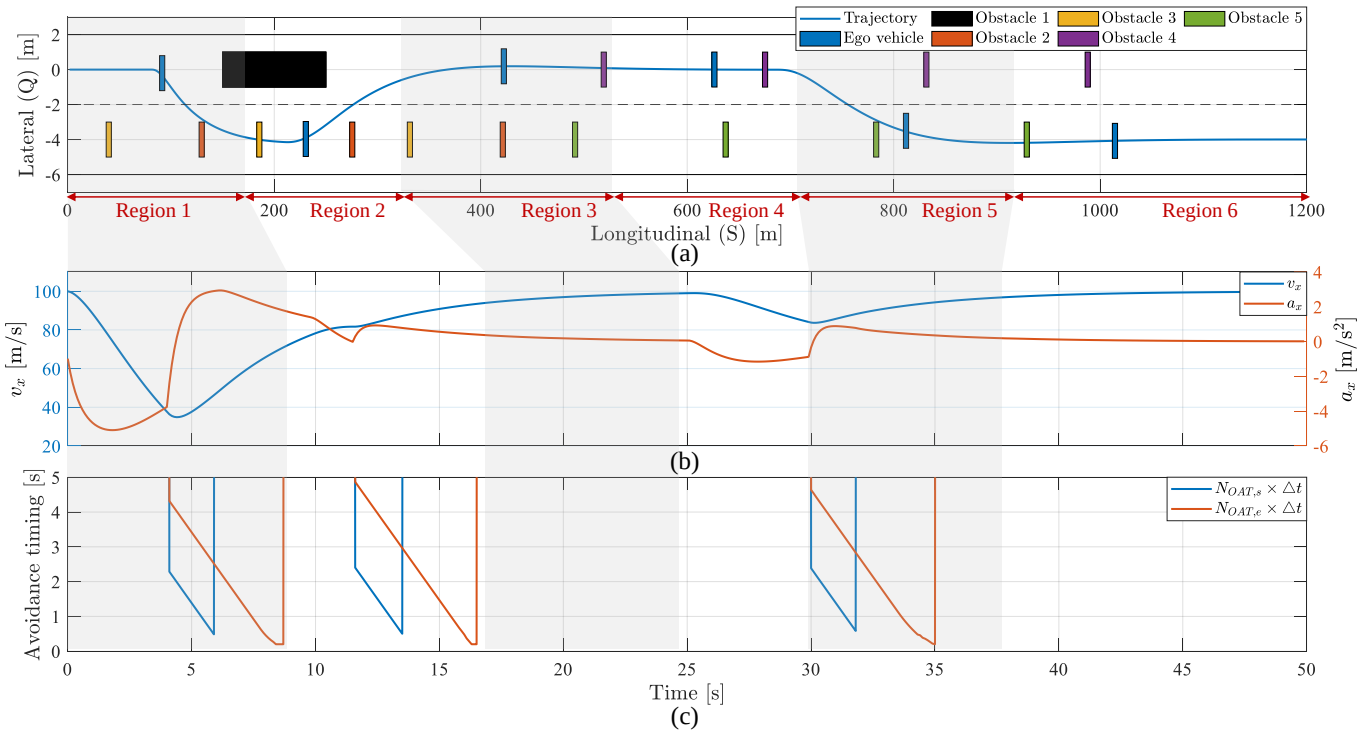

Fig. 9. Simulation results of the proposed trajectory planning algorithm. (a) Trajectory of the ego vehicle. (b) Longitudinal velocity and acceleration. (c) Obstacle avoidance timings ($N_{OAT,s}, N_{OAT,e}$).

TABLE II
EGO VEHICLE AND OBSTACLE INFORMATION.

Symbol	Parameter	Ego vehicle	Symbol	Parameter	Obstacle 1	Obstacle 2	Obstacle 3	Obstacle 4	Obstacle 5
w	Ego vehicle width	$2\,m$	$w_{O,(\cdot)}$	Obstacle width	$2\,m$	$2\,m$	$2\,m$	$2\,m$	$2\,m$
E	Error boundary	$0.5\,m$	$l_{O,(\cdot)}$	Obstacle length	$100\,m$	$5\,m$	$5\,m$	$2\,m$	$2\,m$
$v_{x,des}$	Desired velocity	$100\,km/h$	$s_{O,(\cdot)}(0)$	Initial $s_{O,(\cdot)}$	$150\,m$	$-60\,m$	$30\,m$	$150\,m$	$100\,m$
$v_x(0)$	Initial v_x	$100\,km/h$	$v_{O,(\cdot)}(0)$	Initial $v_{O,(\cdot)}$	$0\,km/h$	$70\,km/h$	$70\,km/h$	$75\,km/h$	$70\,km/h$
D	Lane width	$4\,m$		Lane	Lane 1	Lane 2	Lane 2	Lane 1	Lane 2

Fig. 9(c) shows obstacle avoidance timings. The solid blue line is $N_{OAT,s}$, representing the remaining time until the start of the lane change. Moreover, the solid red line is $N_{OAT,e}$, representing the time remaining until the end of the lane shift.

The proposed algorithm optimizes the trajectory, velocity, and obstacle avoidance timings using only one quadratic programming. Subsequently, an in-depth analysis of each region follows. These regions are visually represented in Fig. 9, divided into six regions. In Fig. 9(a)-(c), the same shading indicates the same region.

- Region 1.

Region 1 showcases avoiding static obstacle situations such as construction sites while considering surrounding vehicles. Initially, It was judged that the ego vehicle could not avoid the construction site by overtaking obstacle 2. Consequently, the ego vehicle slows down first and sends obstacle 2 first. It can be confirmed by the velocity and acceleration data within a 4-second window, as shown in Fig. 9(b). After that, recognizing the advantage of lane change over coming to a stop, the ego vehicle undergoes lane change and positions itself between obstacle 2 and obstacle 3.

In this region, obstacles 1, 2, and 3 take on the role of O_f, O_af, and O_ar, respectively. The proposed weak duality constraints can express the collision-free region for each obstacle. In conclusion, it was verified that the proposed algorithm can plan path and velocity profiles that do not collide with O_f, O_af, and O_ar during lane change.

Fig. 9(c) shows the optimized obstacle avoidance timings. Upon examining the results at 4 seconds, approximately $N_{OAT,s}$ is 2 seconds, while $N_{OAT,e}$ is 4 seconds. It implies that lane change initiation commences 2 seconds after the specified point, with the completed 4 seconds after. These values converge to zero as time progresses and the lane change is executed.

- Region 2.

Region 2 shows a scenario that overtakes a relatively slower preceding vehicle. Before 12 seconds, the ego vehicle follows the preceding vehicle. However, the proposed algorithm concludes that a lane change to regain velocity is advantageous. Consequently, a lane change maneuver is executed. After lane change, the ego vehicle regains velocity as shown in Fig. 9(b). Fig. 9(c) shows the obstacle avoidance timings, representing the initiation and completion timing of the lane change.

- Region 3.

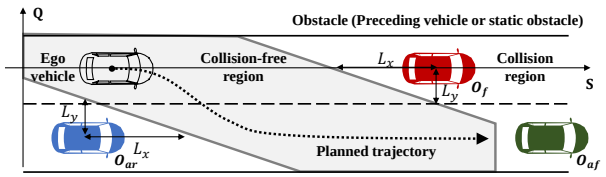

Fig. 10. Previous trajectory planning algorithm using quadratic programming [20].

Region 3 shows a velocity recovery owing to the substantial distance between the ego vehicle and the preceding car. A desired velocity was set to $100\,kph$ in this simulation. Therefore, the velocity profile was planned that the ego vehicle's velocity approach $100\,kph$ through smooth acceleration.

- Region 4.

Region 4 exemplifies a scenario of the preceding vehicle following. In this situation, there is a vehicle represented by obstacle 5 in the next lane. Also, since the distance between obstacle 4 and obstacle 5 is short, it is impossible to change lanes between them. Therefore, the lane-keeping mode is automatically selected. As a result, it follows while maintaining the distance from the preceding vehicle through deceleration.

- Region 5.

Region 5 illustrates avoiding obstacle 4 and overtaking obstacle 5 through lane change. Unlike region 4, in region 5, the distance between obstacle 4 and obstacle 5 widens, and the proposed algorithm determines that lane change is possible. As a result, a trajectory of accelerating velocity and changing lanes was planned. At this time, obstacle 4 acts as O_f, and obstacle 5 acts as O_{ar}. In conclusion, the proposed algorithm can plan a collision-free trajectory considering the front obstacle within the same lane and the rear obstacle within the right lane.

- Region 6.

In this region, no obstacles lie ahead of the ego vehicle. In this case, it accelerates while maintaining the lane and recovers the speed to the desired velocity.

In this section, we verified the trajectory planning performance through simulation. The proposed algorithm optimized longitudinal and lateral acceleration simultaneously. Therefore, it was possible to perform general lane-change and obstacle avoidance situations. In addition, the proposed algorithm had a QP form consisting of a quadratic cost function and linear constraints. This QP form has the advantage of drastically reducing the computational load while using the optimization-based trajectory planning method.

C. Comparison with Previous Trajectory Planning Algorithm using Quadratic Programming

A recent study has proposed a trajectory planning algorithm using QP [20]. This algorithm is shown in Fig. 10, wherein collision-free constraints are described as linear functions within the S-Q coordinate. This approach straightforwardly represents collision-free regions through linear constraints.

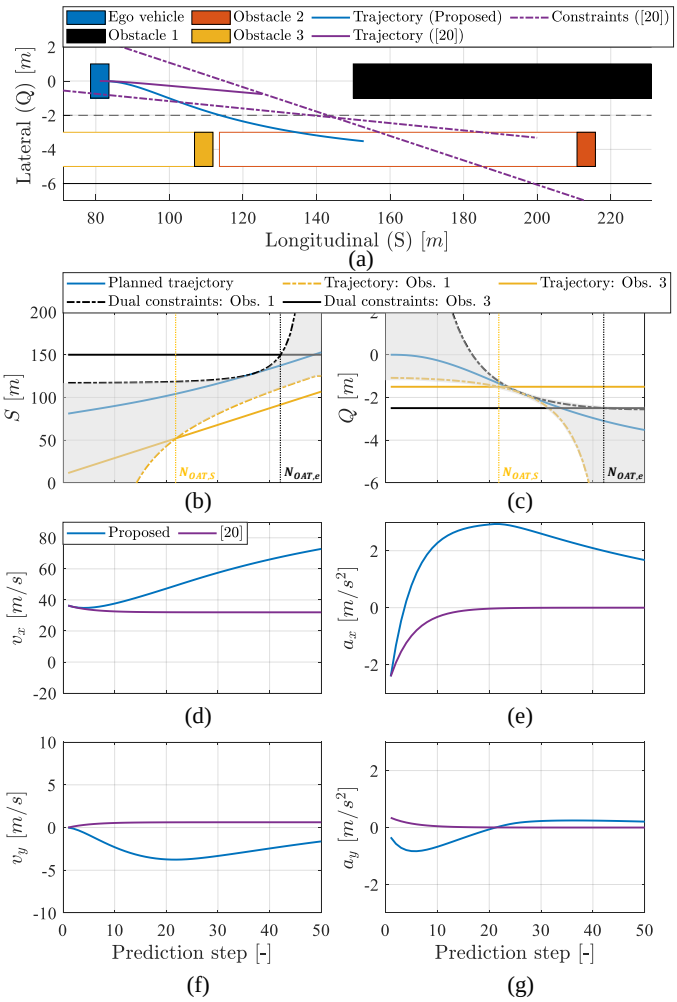

Fig. 11. Comparison with proposed algorithm and [20]. (a) Planned trajectories and collision-free constraints. (b) Longitudinal constraints. (c) Lateral constraints. (d) Longitudinal velocity. (e) Longitudinal acceleration. (f) Lateral velocity. (g) Lateral acceleration.

However, this trajectory planning algorithm is notably conservative. In this section, we conduct a comparative analysis between the proposed and the previous trajectory planning algorithms. For comparison, L_x and L_y are set as follows:

$$L_x = v_x(k) \times 1.3\,sec, \quad L_y = \frac{w_{O,(\cdot)}}{2} + 1\,m \tag{43}$$

To conduct a comparative analysis, trajectory planning outcomes were evaluated at 4 seconds of Fig. 9. This scenario encompasses considerations for neighboring lane vehicles and avoidance of the same lane obstacle, as shown in Fig. 11. Fig. 11(a) shows the planned trajectories of each algorithm. Each square denotes the ego vehicle and obstacles; corresponding details are shown in Table II.

The white rectangle in Fig. 11(a) represents obstacles movement during 5 seconds. The blue and purple solid lines provide each algorithm's planned trajectory. Also, the purple dash-single dotted line represents the linear collision-free constraints proposed in [20]. Previous trajectory planning algorithms using QP typically define collision-free regions using

S-Q coordinates, as shown in Fig. 11(a). Consequently, the previous algorithm expresses collision-free constraints linearly but is very conservative. Therefore, lane change is impossible; the only possible trajectory is deceleration.

In contrast, the proposed algorithm changed lanes while adeptly evading obstacles, positioning the vehicle between obstacles 2 and 3. In conclusion, the proposed trajectory planning algorithm achieves high freedom despite. Further details regarding the collision-free constraints of the proposed algorithm can be found in Fig. 11(b)-(c).

Fig. 11(b)-(c) shows the longitudinal and lateral collision-free constraints, applying a longitudinal-lateral decomposition approach. The trajectories of obstacles 1 and 3 are depicted by black and yellow solid lines, respectively. Notably, the proposed algorithm simultaneously optimizes collision avoidance timings ($N_{OAT,s}, N_{OAT,e}$) with trajectory. By applying optimized timings, the dual collision-free constraints and collision-free regions are represented as dash-single dotted lines and grey areas. Finally, planned trajectories are shown as solid blue lines. The planned trajectory in S-Q coordinates is depicted by the blue line in Fig. 11(a). Like this, the proposed algorithm enables effective trajectory planning.

Fig. 11(d)-(e) shows the longitudinal acceleration and velocity profiles over the prediction horizon. The proposed algorithm plans acceleration and obstacle avoidance, while the previous algorithm decelerate. Lateral components are shown in Fig. 11(f)-(g), wherein the proposed algorithm excels in executing a smooth lane change.

Despite employing linear collision-free constraints, the proposed algorithm guarantees a high degree of freedom. This characteristic makes it realistic and avoids unnecessary braking compared to previous algorithms. In conclusion, the proposed algorithm can plan a realistic and advanced trajectory compared to previous algorithms.

D. Minimum required distance for obstacle avoidance of various algorithms

In this section, the characteristics of the proposed algorithm are analyzed through performance comparison with previous algorithms. The performance evaluation was based on the minimum required distance for obstacle avoidance under various velocities. A scenario is avoiding collision with a static obstacle located in the same lane, as shown in Fig. 12. Information such as lane width and obstacle width is shown in Fig. 12. It analyzes the minimum distance at which an obstacle can be avoided by maneuvers such as deceleration or steering for several algorithms. In other words, we analyze the marginal performance of each algorithm. Each algorithm's marginal performance evaluation results are presented in Fig. 13. Next, we analyze the strengths and weaknesses of each of the previous algorithms.

- Sampling-based method (5th-order)

The sampling-based method employs either a cubic spline (3rd-order) [11] or a quintic polynomial (5th-order) [12] to define the trajectory. Among these, the trajectory planning algorithm based on the quintic polynomial was chosen as the comparison group.

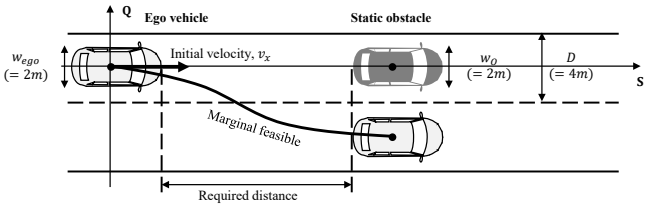

Fig. 12. Marginal performance analysis scenario.

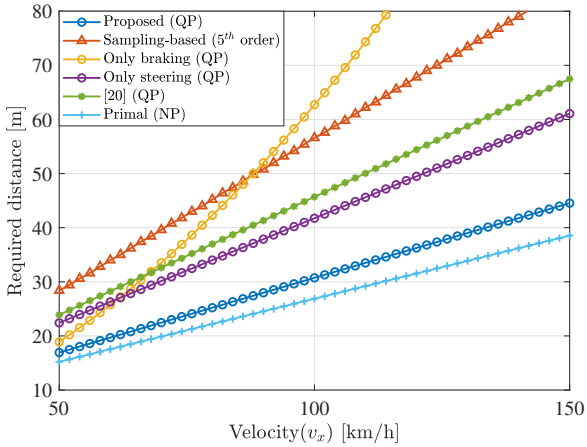

Fig. 13. Marginal performance of each algorithm: Required distance for obstacle avoidance under various velocities.

The red line in Fig. 13 represents the required distance for obstacle avoidance under various velocities. While the sampling-based method offers the advantage of a predetermined trajectory and lower computational load, it can be observed that the required avoidance distance for each velocity is considerably distant. Consequently, while the sampling-based method demonstrates satisfactory performance under normal driving conditions, it could not be suitable for more severe driving scenarios.

- Optimization-based, quadratic programming (Only steering or braking)

These have a similar level of computational load to the proposed algorithm. Therefore, through comparison with them, we compared the performance of algorithms with a similar level of computational load.

The results of analyzing the required distance under various velocities using only steering or braking are shown in Fig. 13. The yellow line represents the result obtained only through braking, while the purple line corresponds to the result obtained only through steering. Obstacle avoidance through braking demonstrated a shorter avoidance distance than avoidance through steering, particularly at low velocities. However, as the velocity increased, avoidance through steering was safer. Nonetheless, both trajectory planning approaches relying solely on braking or steering necessitated longer avoidance distances than the proposed algorithm. Consequently, among various algorithms with similar computational load, the proposed algorithm is the optimal choice to ensure the safest trajectory planning.

- Optimization-based, quadratic programming [20]

The result of the algorithm proposed in [20] is shown in the green line. The algorithm is based on optimization and shows better marginal performance than the sampling-based algorithm. However, this is a very conservative trajectory planning. Therefore, it shows worse performance than obstacle avoidance through steering only. In conclusion, the previous algorithm performs well in normal driving situations but has limitations in marginal driving due to its inherently conservative nature. In contrast, the proposed algorithm shows high freedom and good marginal performance.

- Optimization-based, nonlinear programming (Primal optimization problem)

The primal optimization problem can be optimized through nonlinear programming. In contrast, the proposed algorithm can be optimized through QP, significantly reducing computational effort. However, in terms of performance, it yields conservative results and may incur slight performance degradation for marginal performance. Therefore, we compared the marginal performance with the primal optimization problem (16).

The cyan line represents the results of the primal optimization problem optimized through nonlinear programming. Due to the weak-duality property of the proposed algorithm, it can be observed that the required distance of the proposed algorithm is slightly longer compared to the primal optimization problem. Specifically, at a speed of $100\,kph$, this difference amounts to approximately $4\,m$. However, it is essential to note that the primal optimization problem requires computational load that is roughly 50 times greater than the proposed algorithm's. This critical limitation renders real-time implementation infeasible for the primal optimization problem. In conclusion, considering both performance and computational load, the proposed algorithm emerges as the optimal choice.

V. Conclusion and Future Work

We proposed an optimization-based trajectory planning algorithm with QP form. The critical issue with the optimization-based trajectory planner was that it was challenging to implement in real time due to the high computational load. To address this, we proposed a trajectory planner through QP, which has a small computational load, rather than through the commonly used NP. We employed the longitudinal-lateral decomposition method to analyze the primal optimization problem. Furthermore, we introduced a new function defined as a dual function for converting NP to QP form. The necessary conditions and properties of the dual function were analyzed. Based on these conditions, we proposed a dual function. Finally, the optimization problem of trajectory planning was transformed into a weak duality QP problem. The proposed algorithm optimized the trajectory and obstacle avoidance timing by solving only one QP. Furthermore, the proposed algorithm was extended to multi-obstacle, reverse lane change. We demonstrated the performance of the proposed algorithm through simulations, which verified that the proposed algorithm could plan the trajectory for general and emergency driving. In conclusion, the proposed algorithm exhibits superior computational load-to-performance compared to previous research methods.

The core technology of this study lies in the proposal of the dual function. We devised a dual function based on trigonometric functions. Suggesting a new dual function, which satisfies all the necessary conditions and improves duality, will enhance the performance of the trajectory planner. Ultimately, it will be necessary to validate the performance of the autonomous driving integrated algorithm integrated with the perception and control technology and implement the vehicle.

Acknowledgment

This research was partly supported by the BK21 FOUR Program of the National Research Foundation Korea(NRF) grant funded by the Ministry of Education(MOE)., the Technology Innovation Program (20014983, Development of autonomous chassis platform for a modular vehicle) funded By the Ministry of Trade, Industry & Energy(MOTIE, Korea)., and the Autonomous Driving Technology Development Innovation Program (20018181, Development of Lv. 4+ autonomous driving vehicle platform based on point-to-point driving to logistic center for heavy trucks) funded by the Ministry of Trade, Industry & Energy(MOTIE, Korea) and Korea Evaluation Institute of Industrial Technology(KEIT)

References

[1] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intelligent transportation systems magazine*, vol. 6, no. 4, pp. 6–22, 2014.

[2] M. Hasenjäger and H. Wersing, "Personalization in advanced driver assistance systems and autonomous vehicles: A review," in *2017 ieee 20th international conference on intelligent transportation systems (itsc)*. IEEE, 2017, pp. 1–7.

[3] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: autonomous vehicles in city traffic*. springer, 2009, vol. 56.

[4] E. R. Teoh and D. G. Kidd, "Rage against the machine? google's self-driving cars versus human drivers," *Journal of safety research*, vol. 63, pp. 57–60, 2017.

[5] S. Ingle and M. Phute, "Tesla autopilot: semi autonomous driving, an uptick for future autonomy," *International Research Journal of Engineering and Technology*, vol. 3, no. 9, pp. 369–372, 2016.

[6] S. Dixit, U. Montanaro, S. Fallah, M. Dianati, D. Oxtoby, T. Mizutani, and A. Mouzakitis, "Trajectory planning for autonomous high-speed overtaking using mpc with terminal set constraints," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1061–1068.

[7] S. Dixit, S. Fallah, U. Montanaro, M. Dianati, A. Stevens, F. Mccullough, and A. Mouzakitis, "Trajectory planning and tracking for autonomous overtaking: State-of-the-art and future prospects," *Annual Reviews in Control*, vol. 45, pp. 76–86, 2018.

[8] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on intelligent transportation systems*, vol. 17, no. 4, pp. 1135–1145, 2015.

[9] L. Liu, S. Lu, R. Zhong, B. Wu, Y. Yao, Q. Zhang, and W. Shi, "Computing systems for autonomous driving: State of the art and challenges," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6469–6486, 2020.

[10] K. Chu, M. Lee, and M. Sunwoo, "Local path planning for off-road autonomous driving with avoidance of static obstacles," *IEEE transactions on intelligent transportation systems*, vol. 13, no. 4, pp. 1599–1616, 2012.

[11] Y. Zhang, H. Chen, S. L. Waslander, J. Gong, G. Xiong, T. Yang, and K. Liu, "Hybrid trajectory planning for autonomous driving in highly constrained environments," *IEEE Access*, vol. 6, pp. 32 800–32 819, 2018.

[12] H. Lee and S. Choi, "Development of collision avoidance system in slippery road conditions," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[13] X. Hu, L. Chen, B. Tang, D. Cao, and H. He, "Dynamic path planning for autonomous driving on various roads with avoidance of static and moving obstacles," *Mechanical systems and signal processing*, vol. 100, pp. 482–500, 2018.

[14] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359, 2012.

[15] Y. Wang, Z. Liu, Z. Zuo, Z. Li, L. Wang, and X. Luo, "Trajectory planning and safety assessment of autonomous vehicles based on motion prediction and model predictive control," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8546–8556, 2019.

[16] J. Nilsson, M. Brännström, E. Coelingh, and J. Fredriksson, "Lane change maneuvers for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1087–1096, 2016.

[17] B. Gutjahr, L. Gröll, and M. Werling, "Lateral vehicle trajectory optimization using constrained linear time-varying mpc," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1586–1595, 2016.

[18] H. Febbo, P. Jayakumar, J. L. Stein, and T. Ersal, "Real-time trajectory planning for automated vehicle safety and performance in dynamic environments," *Journal of Autonomous Vehicles and Systems*, vol. 1, no. 4, 2021.

[19] S. Li, Z. Li, Z. Yu, B. Zhang, and N. Zhang, "Dynamic trajectory planning and tracking for autonomous vehicle with obstacle avoidance based on model predictive control," *Ieee Access*, vol. 7, pp. 132 074–132 086, 2019.

[20] S. Dixit, U. Montanaro, M. Dianati, D. Oxtoby, T. Mizutani, A. Mouzakitis, and S. Fallah, "Trajectory planning for autonomous high-speed overtaking in structured environments using robust mpc," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2310–2323, 2019.

[21] Z. Wang, G. Li, H. Jiang, Q. Chen, and H. Zhang, "Collision-free navigation of autonomous vehicles using convex quadratic programming-based model predictive control," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 3, pp. 1103–1113, 2018.

[22] T. Fraichard and C. Laugier, "Path-velocity decomposition revisited and applied to dynamic trajectory planning," in *[1993] Proceedings IEEE International Conference on Robotics and Automation*. IEEE, 1993, pp. 40–45.

[23] W. Lim, S. Lee, M. Sunwoo, and K. Jo, "Hybrid trajectory planning for autonomous driving in on-road dynamic scenarios," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 1, pp. 341–355, 2019.

[24] J. Nilsson, M. Brännström, J. Fredriksson, and E. Coelingh, "Longitudinal and lateral control for automated yielding maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 5, pp. 1404–1414, 2016.

[25] X. Qian, F. Altché, P. Bender, C. Stiller, and A. de La Fortelle, "Optimal trajectory planning for autonomous driving integrating logical constraints: An miqp perspective," in *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)*. IEEE, 2016, pp. 205–210.

[26] N. H. Amer, H. Zamzuri, K. Hudha, and Z. A. Kadir, "Modelling and control strategies in path tracking control for autonomous ground vehicles: a review of state of the art and challenges," *Journal of intelligent & robotic systems*, vol. 86, no. 2, pp. 225–254, 2017.

[27] M. Jalalmaab, B. Fidan, S. Jeon, and P. Falcone, "Model predictive path planning with time-varying safety constraints for highway autonomous driving," in *2015 International Conference on Advanced Robotics (ICAR)*. IEEE, 2015, pp. 213–217.

[28] X. Zhang, A. Liniger, and F. Borrelli, "Optimization-based collision avoidance," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 972–983, 2020.

[29] V. A. Laurense and J. C. Gerdes, "Long-horizon vehicle motion planning and control through serially cascaded model complexity," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 1, pp. 166–179, 2021.

[30] Y. Huang, H. Wang, A. Khajepour, H. Ding, K. Yuan, and Y. Qin, "A novel local motion planning framework for autonomous vehicles based on resistance network and model predictive control," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 55–66, 2019.

[31] L. Wang, *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media, 2009.

[32] S. Mata, A. Zubizarreta, and C. Pinto, "Robust tube-based model predictive control for lateral path tracking," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 4, pp. 569–577, 2019.

[33] D. Jeong and S. B. Choi, "Tracking control based on model predictive control using laguerre functions with pole optimization," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[34] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*. Nob Hill Publishing Madison, WI, 2017, vol. 2.

[35] C. Chen, J. Guo, C. Guo, C. Chen, Y. Zhang, and J. Wang, "Adaptive cruise control for cut-in scenarios based on model predictive control algorithm," *Applied Sciences*, vol. 11, no. 11, p. 5293, 2021.

[36] Y. He, B. Ciuffo, Q. Zhou, M. Makridis, K. Mattas, J. Li, Z. Li, F. Yan, and H. Xu, "Adaptive cruise control strategies implemented on experimental vehicles: A review," *IFAC-PapersOnLine*, vol. 52, no. 5, pp. 21–27, 2019.

[37] A. Abdullahi and S. Akkaya, "Adaptive cruise control: A model reference adaptive control approach," in *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, 2020, pp. 904–908.

Dasol Jeong received the B.S. degree in mechanical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2017, the M.S. in mechanical engineering from KAIST, in 2019, and the Ph.D. degree in mechanical engineering from KAIST, in 2023. Since 2023, he has been the senior research engineer of the Hyundai Motor Company. His research interests include vehicle dynamics and control, intelligent tires, path planning, model predictive control, and control theory.

Seibum B. Choi (M'09) received the B.S. in mechanical engineering from Seoul National University, Seoul, Korea, the M.S. in mechanical engineering from KAIST, Deajeon, Korea, and the Ph.D. in control from the University of California, Berkeley, CA, USA, in 1993. From 1993 to 1997, he was involved in the development of automated vehicle control systems at the Institute of Transportation Studies, University of California. Through 2006, he was with TRW, Livonia, MI, USA, where he was involved in the development of advanced vehicle control systems. Since 2006, he has been faculty in the Mechanical Engineering Department, KAIST, Korea. His current research interests include fuel-saving technology, vehicle dynamics and control, and active safety systems. Prof. Choi is a Member of the American Society of Mechanical Engineers, the Society of Automotive Engineers, and the Korean Society of Automotive Engineers.