IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 15, 2022, accepted March 20, 2022, date of publication March 30, 2022, date of current version April 6, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3163309

Development of an Adaptive and Weighted Model Predictive Control Algorithm for Autonomous Driving With Disturbance Estimation and Grey Prediction

KWANGSEOK OH[1] AND JAHO SEO[2]
[1]School of ICT, Robotics & Mechanical Engineering, Hankyong National University, Anseong-si 17579, South Korea
[2]Department of Automotive and Mechatronics Engineering, Ontario Tech University, Oshawa, ON L1G 0C5, Canada

Corresponding author: Jaho Seo (jaho.seo@ontariotechu.ca)

ABSTRACT This paper presents an adaptive and weighted model predictive control (MPC) algorithm for autonomous driving with disturbance estimation and prediction. Unexpected and unpredictable disturbances in the real world limit the performance of MPC. To overcome this limitation, this paper proposes adaptive and weighted prediction methods with a sliding mode observer and a weighting function with the grey prediction model. The sliding mode observer is designed for disturbance estimation with finite stability conditions, and the estimated disturbance is predicted using the grey prediction model. Based on the adaptive and weighted prediction method, the length of prediction horizon and cost value of each predicted state are adjusted in real time to eliminate any negative impact on future predicted states. Meanwhile, a variation in the cost value, which is caused by prediction horizon adaptation and weighted prediction, may harm the control performance as it can excessively increase or decrease the model uncertainty. Therefore, an input weighting factor is adapted in the MPC cost function based on an exponential weighting function. The performance of the proposed adaptive control algorithm is evaluated using CarMaker software under longitudinal and lateral autonomous driving scenarios.

INDEX TERMS Autonomous driving, disturbance estimation, grey prediction, model predictive control, sliding mode observer, weighted prediction.

I. INTRODUCTION

Autonomous vehicles represent an important part of the mobility platform for transportation and shipping applications because of their features such as safe control and convenient driving without requiring a human driver. For these features, the target value tracking performance should be ensured by appropriate control technologies that can withstand various conditions, including physical, control input, and state conditions. Relative displacement states, such as lateral and yaw angle, are generally used as target values for the path tracking of autonomous vehicles. The time derivatives of these states can be used as target values according to the controller type and available information. Various control methods are used for target value tracking; these methods include robust and predictive control algorithms are used for sliding mode control and model predictive control (MPC), respectively. Sliding mode control [1] can be designed based on a sliding surface, and various surfaces can be designed to improve the control performance despite the disturbance due to the stability condition in the Lyapunov direct method. Because of its robustness, this technique has been used for the tracking control of vehicles. However, there are some limitations: the controller only considers current states, and chattering may occur due to a discrete injection term of the control input. Accordingly, MPC [2] has also been used for the multi-input multi-output (MIMO) system control of vehicles based on a reasonable prediction model. Because MPC can consider states and input constraints, predictive and realistic optimal control inputs can be computed if the system model can represent the actual dynamics of the system. However, model uncertainty, which can be increased or decreased over the MPC prediction process, is inevitable. This unpredictable uncertainty can also have a negative influence on

The associate editor coordinating the review of this manuscript and approving it for publication was Guangcun Shan.

the performance of MPC. Therefore, a reasonable estimation of current disturbances and a reasonable prediction of future disturbances are essential for maintaining the control stability of MIMO control systems. The following text describes algorithms that have been proposed to improve the performance of adaptive MPC by estimating, predicting, and rejecting disturbances.

Akpan and Hassapis [3] proposed two adaptive MPC algorithms consisting of online process identification and predictive control parts. The two control strategies, nonlinear MPC and generalized predictive control, were designed in the predictive control part. For the process identification, a process model was approximated using a series-parallel neural network structure trained by recursive least squares. An MPC scheme with guaranteed closed-loop asymptotic stability was proposed in [4] for a class of constrained nonlinear time-delay systems with discrete and distributed delays. A locally asymptotically stabilizing controller and Lyapunov–Krasoskii functional of the locally stabilized system were designed and employed for the terminal cost of the MPC. The proposed method was demonstrated by a numerical example. Pourjafari and Mojallali [5] proposed an MPC-based voltage control scheme to overcome voltage instability, which has a close relationship with the adequacy of reactive power and the response under load-tap changers. Zeng *et al.* [6] presented a new multivariable grey prediction model by adding dependent variable lag, linear correction, and random disturbance terms to the traditional grey model. The proposed model was proved theoretically, and three case studies were conducted for performance evaluation. Pedersen *et al.* [7] compared three methods to investigate the superheat performance in a refrigeration system with respect to disturbance rejection: traditional gain scheduled proportional–integral-based controller, predictive functional controller, and predictive functional controller. Han *et al.* [8] proposed a new congestion control method to compensate for the effects of nonlinear disturbance, uncertainty, time-varying delay, and input constraint. The authors designed a state feedback congestion controller by applying the MPC approach, and they analyzed the stability of the closed-loop system using a Lyapunov–Kraovskii functional. Fukushima *et al.* [9] proposed an adaptive MPC algorithm for a class of constrained linear systems. It was designed so that the adaptive control algorithm could estimate system parameters online and produce a control input that would satisfy input and state constraints for possible parameter estimation errors. The robust MPC method using a comparison model was combined with an adaptive parameter estimation method. Li and Shi [10] studied the event-triggered MPC problem for continuous-time nonlinear systems subject to bounded disturbances. Based on the measured error between the system state and its optimal prediction, an event-triggered mechanism was developed and used to design an event-triggered MPC algorithm with the dual-mode approach. Kohler *et al.* [11] presented a tube-based framework for robust adaptive model predictive control (RAMPC) for nonlinear systems subject to parametric uncertainty and additive disturbances with the set-membership estimation method. This study showed that the RAMPC algorithm ensures robust recursive feasibility and robust constraint satisfaction. Hajizadeh *et al.* [12] proposed an adaptive MPC algorithm with dynamic adjustments of constraints and objective function weights based on estimates of the plasma insulin concentration for artificial pancreas systems. The performance of the proposed adaptive MPC algorithm was demonstrated by simulation case studies. Garimella *et al.* [13] addressed the problem of motion planning among obstacles for quadrotor platforms under external disturbances and with model uncertainty. They proposed a novel nonlinear MPC optimization technique to address the aforementioned problem. The technique incorporates specified uncertainties into the planned trajectories. Babaie *et al.* [14] developed a supervised learning MPC to cancel common-mode voltage in a three-phase NPC inverter while tracking the control objectives of capacitor voltage balancing, load current reference tracking, and common-mode voltage suppression. In the study, the weighting factor for a cost function was optimized, and the developed algorithm was evaluated under several experimental and simulation tests. Rosolia *et al.* [15] presented a reference-free learning model predictive controller for linear systems, which can improve performance by learning from previous iterations. Simulation results showed the effectiveness of the control logic. Li *et al.* [16] proposed a latent variable iterative learning MPC method for trajectory tracking in batch processes, where a state–space model was used to express the dynamic characteristics of the internal model. Koller *et al.* [17] presented a learning-based MPC scheme that can provide provable high-probability safety guarantees. For the design of the control algorithm, regularity assumptions on the dynamics were exploited under a Gaussian process prior to constructing provably accurate confidence intervals on predicted trajectories. Based on the results of previous studies, it can be seen that there exist various methods to design robust and adaptive MPC algorithms, including parameter estimation, event-triggered methods, and disturbance compensation. Generally, the methods used for adaptive MPC algorithm design are based on mathematical models or rule-based approaches. However, model- and rule-based design methods have limitations in that they can have a positive influence on control performance under limited conditions considered in the design process and there always exist unpredictable internal and external disturbances. However, because MPC generally has advantages such that the system state and input constraints can be considered and human-like predictive control inputs can be derived if model uncertainty is not relatively large, MPC has been used for various multivariable systems, including vehicle control systems. As vehicle systems become more complex and nonlinear, vehicle system models have the same limitation as above in that there always exist unpredictable internal and external disturbances. To overcome this limitation, several studies have developed diverse advanced MPC algorithms based on stochastic-, prediction-,

and learning-based methods for vehicle system control as follows.

Ripaccioli *et al.* [18] illustrated the use of stochastic MPC for power management in vehicles equipped with advanced hybrid powertrains and proposed several control strategies mainly based on heuristics or rules and tuned certain reference drive cycles. Tsao *et al.* [19] presented a stochastic MPC algorithm that leveraged short-term probabilistic forecasts for dispatching and rebalancing autonomous mobility-on-demand systems. The authors also presented a core stochastic optimization problem in terms of a time-expanded network flow model to design the controller. Seo *et al.* [20] presented a motion planning algorithm for lane change with a combination of probabilistic and deterministic prediction for automated driving under complex driving circumstances. For the design of a vehicle motion planning algorithm for a lane change, a collision probability was defined using a reachable set of uncertainty propagation; furthermore, the lane change risk was monitored using predicted time-to-collision and safety distance to guarantee safety in lane change behavior. Moser *et al.* [21] proposed a stochastic MPC approach to optimize the fuel consumption of vehicles. A real measurement-based conditional linear Gauss model was developed and trained to estimate the probability distribution of the future velocity of the preceding vehicle. He *et al.* [22] proposed a stochastic model predictive controller for air conditioning systems to improve the energy efficiency of electric vehicles. The velocity predictor was adopted based on a Markov chain to realize a sense of future disturbances over the stochastic MPC horizon; three control approaches were compared in terms of electricity consumption, cabin temperature, and comfort fluctuation. Rosolia *et al.* [23] developed a novel learning MPC technique for application to an autonomous racing problem, with the aim of minimizing the time to complete a lap by analyzing data from previous laps to improve performance while satisfying safety requirements. Elsisi *et al.* [24] proposed a hybridization method between discrete-time Laguerre function and MPC to formulate the MPC with few parameters using a recent artificial intelligence technique. Various scenarios were considered to confirm the superiority of the proposed control method. The same author Elsisi [25] suggested an optimal design for the nonlinear MPC based on a newly improved intelligence technique known as modified multitracker optimization algorithm. This modification was carried out using the opposition-based learning and quasi OBL approach. The author in [26] also provided a new design of the adaptive MPC for blade pitch control in wind energy conversion systems using the crow search algorithm. With the help of a new artificial intelligence technique named bat-inspired algorithm, Elsisi *et al.* [27] developed an optimal design of the MPC with superconducting magnetic energy storage (SMES) and capacitive energy storage (CES) devices for load frequency control.

These studies focused on the development of stochastic MPC and learning-based MPC algorithms using past data to enhance control performance by reducing the negative impact of model uncertainty and improving the MPC optimization algorithms.

In our study, an adaptive and weighted MPC algorithm with disturbance estimation and prediction was developed for autonomous driving by the application of the grey prediction method and weighting function. The main contributions of this study can be summarized as follows with problem statements of each issue:

(Problem statement) Uncertainties can be predicted unreasonably in the prediction step of the MPC, which can have a negative impact on control performance.

(Contribution-1) A prediction horizon adaptation algorithm using disturbance estimation and prediction was proposed to improve the MPC performance.

(Contribution-2) A weighted prediction algorithm for the MPC was designed to decrease the negative impact of unreasonable uncertainty prediction on control performance by adjusting weighting values for predicted states.

A state–space model-based MPC algorithm was formulated with physical constraints, such as input magnitude and input change rate limits, to design an adaptive and weighted MPC controller. The lumped disturbance of the error dynamic equation was also estimated using a sliding mode observer with finite stability conditions. The estimated disturbance was used for disturbance prediction using the grey prediction method, and the predicted future disturbances were used to determine the length of prediction horizon. The exponential weighting function was designed with a time constant function to determine weighting values in the cost function. The time constant function was designed based on the change rate of the estimated disturbance. A negative impact from uncertainty can be reduced through adaptive prediction horizon and weighted cost values. The performance of the designed algorithm was evaluated using the CarMaker software under autonomous driving scenarios, such as car-following and lane change scenarios.

The remainder of this paper is organized as follows. In Section 2, the problem formulation is explained, and the proposed adaptive controller is conceptually described. Section 3 describes the adaptive and weighted MPC algorithm, and Section 4 presents the performance evaluation results. Finally, Section 5 presents a summary of the study alongside a limitation analysis and discussion of future works.

II. PROBLEM DESCRIPTION AND OVERALL CONTROLLER CONCEPT

In the real world, control performance may be degraded generally by internal and external disturbances. The current disturbance can be estimated, but there always exists a time delay due to computation time, and a future disturbance is unpredictable. In the prediction step of the MPC formulation, an exact future disturbance is required to compute accurate optimal predictive inputs. However, it is impossible to predict future disturbance using the estimated current disturbance because it cannot represent the actual current disturbance due

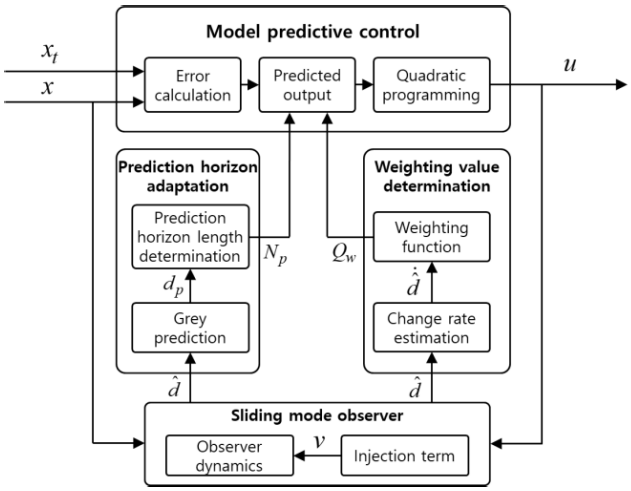

FIGURE 1. Detailed model schematic of the proposed adaptive and weighted MPC.

to time delay, and the future disturbance is unpredictable, as described. This inaccurate current and future disturbance information can have a negative influence on the control performance of MPC because the cost function for MPC will have an inaccurate value due to the wrong disturbance information. In order to overcome the limitations caused by inaccurate future disturbance information, we designed an adaptation algorithm of the MPC prediction horizon using the grey prediction method and a weighting function for cost values of predicted states. For the estimation of the current disturbance, a sliding mode observer was designed using a discrete injection term with finite stability conditions. Figure 1 shows a detailed model schematic of the proposed adaptive and weighted controller.

The MPC block consists of three sub-blocks, constituting error calculation, predicted output, and quadratic programming, as shown in Fig. 1 (detailed model schematic). In the error calculation block, a state error is calculated using the target state x_t and current state x. Based on the calculated state error, the prediction horizon adaptation algorithm was designed such that the predicted outputs are calculated using the adapted prediction horizon N_p and computed weighting value Q_w from the weighting value determination block. The length of prediction horizon for a predicted output can be adjusted based on the threshold approach using the predicted disturbance. The future disturbance is predicted using the grey prediction method, and the current disturbance d is estimated based on a sliding mode observer under finite stability conditions. The estimated disturbance $\hat{d}$ along with the designed weighting function is used to determine the weighting value. The weighting function was designed so that the effectiveness of the predicted states is decreased if the disturbance negatively affects the control performance. Based on the aforementioned functional flow, the control inputs u can be computed based on the MPC algorithm, and the first element of the computed predictive control inputs is selected as the current control input. The next section describes the adaptive and weighted MPC algorithm in detail.

III. ADAPTIVE AND WEIGHTED MODEL PREDICTIVE CONTROL

The proposed MPC algorithm uses two methods of adaptive and weighted prediction to cope with the negative effect on control performance by unpredictable disturbances and lessen the computational burden of the algorithm.

The two methods for the MPC parameters adaptation consist of the parts of 'prediction horizon adaptation' and 'weighting value determination'. For the prediction horizon adaptation, it was designed that the length of the prediction horizon of the MPC is adjusted by the estimation and prediction of disturbance. Particularly, the current disturbance is estimated robustly using a sliding mode observer and the estimated disturbance is predicted using the grey prediction method. Then, the number of prediction steps is determined based on the estimated and predicted disturbances by applying the threshold approach. If there exists a predicted disturbance larger than the predetermined threshold value at the i_{th} prediction step ($1 \leq i \leq N_p$), the prediction step is selected as (i-1) for the adaptation algorithm.

In the second part, adaptation of weighting factors for predicted states, the change rate magnitude of the estimated current disturbance is used to determine the time constant in a weighting function. Based on the exponential function, the weight function was designed so that weighting values are decreased to reduce the negative impact of the predicted disturbances if the change rate magnitude of the estimated disturbance is increased. Both blocks for prediction horizon adaptation and weighting value determination in Fig. 1 are divided into further sub-blocks. First, the prediction horizon adaptation block consists of disturbance prediction and prediction horizon determination blocks. In the disturbance prediction block, a future disturbance can be predicted based on the grey prediction method using the estimated disturbance from the sliding mode observer. The predicted disturbance is used to determine the length of prediction horizon for the predicted output in the MPC based on the threshold approach. Second, the weighting value determination is sub-divided into change rate estimation and weighting function blocks. The estimated disturbance from the sliding mode observer is also used to determine the weighting value for adaptive MPC with the weighting function. For the determination of weighting values, the change rate (time-derivative of the estimated disturbance) is estimated using a linear Kalman filter in the change rate block, which is used to compute the weighting values for adaptive prediction in the weighting function block. The determined prediction horizon and weighting values are used for adaptation of the length of prediction horizon and effectiveness of the future states towards control input derivation. The following sub-section describes the designed MPC algorithm in detail.

A. MODEL PREDICTIVE CONTROL ALGORITHM

The system model used to design the MPC algorithm in state–space form and the system output y defined using output matrix C are described as follows:

$$\dot{x} = Ax + Bu + Fw \tag{1}$$
$$y = Cx \tag{2}$$

where A, B, and F represent the system, input, and disturbance matrices, respectively, x, u, and w represent the system state, input, and disturbance, respectively. As part of the MPC algorithm design, an error (e) dynamic model based on the target state is required and e represents the error between the vehicle's target state and its current state. In the car-following scenario, the clearance error and velocity error between the preceding vehicle and the subject vehicle velocities are considered. The lateral displacement error and yaw angle error are considered for the lane-change scenario. The following equations represent the derived error dynamic model based on (1) and the matrix definitions:

$$\dot{e} = Ae - Bu + \dot{x}_t - Ax_t - Fw \tag{3}$$

The system model in (3) must be discretized for design of the MPC, and the Euler forward method was used for discretization in this study. Equations (4), (5), and (6) represent the discretized error dynamic model of (3), output for error y_e, and matrix definitions, respectively.

$$e_{k+1} = A_D e_k + B_D u_k + F_D w_{D,k} \tag{4}$$
$$y_{e,k} = Ce_k \tag{5}$$
$$A_D = A\Delta t + I_n, B_D = -B\Delta t,$$
$$F_D = \dot{x}_t - Ax_t - Fw, w_{D,k} = \Delta t \tag{6}$$

where A_D, B_D, and F_D represent system, input, and disturbance matrices for the discrete system model, respectively, and Δt and I represent discretization time and the identity matrix, respectively. The n in (6) represents the number of state (x) elements. Based on the error state defined in the derived discrete error dynamics in (4) and the output error from (5), the predicted output for the MPC formulation is derived using matrices A_D, B_D, and F_D as follows. Equation (7) is the predicted output Y_p and (8)–(14) define the matrices in (7).

$$Y_p = G_p e_k + H_p U_p + F_p u_k + M_p W_p + K_p w_{D,k} \tag{7}$$

where u_k is the current optimal control input and U_p is the system control input vector that contains future optimal control inputs.

$$G_p = \begin{bmatrix} CA_D & CA_D^2 & \cdots & CA_D^N \end{bmatrix}^T \tag{8}$$

$$H_p = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ CB_D & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA_D^{N_p-2}B_D & CA_D^{N_p-3}B_D & \cdots & 0 \end{bmatrix} \tag{9}$$

$$U_p = \begin{bmatrix} u_{k+1} & u_{k+2} & \cdots & u_{k+N} \end{bmatrix}^T \tag{10}$$

$$F_p = \begin{bmatrix} CB_D & CA_DB_D & \cdots & CA_D^{N_p-1}B_D \end{bmatrix}^T \tag{11}$$

$$M_p = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ CF_D & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA_D^{N_p-2}F_D & CA_D^{N_p-3}F_D & \cdots & 0 \end{bmatrix} \tag{12}$$

$$W_p = \begin{bmatrix} w_{D,k+1} & w_{D,k+2} & \cdots & w_{D,k+N} \end{bmatrix}^T \tag{13}$$

$$K_p = \begin{bmatrix} CF_D & CA_DF_D & \cdots & CA_D^{N_p-1}F_D \end{bmatrix}^T \tag{14}$$

For the computation of the optimal control inputs, the cost function shown in (15) was defined with weighting matrix $W_x \in \mathbb{R}^{nN_p \times nN_p}$ and constant $W_u \in \mathbb{R}$ for state and control input.

$$J = Y_p^T W_x^T W_x Y_p + W_u U_p^T D^T D U_p \tag{15}$$

where matrix D is a difference matrix consisting of positive and negative 1's. In this study, weighting matrix W_x was designed using the weighting function for weighted prediction. The designed weighting matrix in (15) is given as follows.

$$W_x = \begin{bmatrix} Q_{w,1} & 0 & \cdots & 0 \\ 0 & Q_{w,2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & Q_{w,N_p} \end{bmatrix} \tag{16}$$

As shown in (16), matrix W_x is a square diagonal matrix, and its diagonal elements $Q_{w,i}(i = 1, \cdots, N_p)$ were designed based on the change rate of disturbance for adjusting the control performance effectiveness of the future states in the cost function. Detailed explanations are given in the following sub-sections. Based on the defined cost function in (15) with weighting values and the computed predicted output, the quadratic programming method was used to find the optimal control input with multiple constraints, such as equality, inequality, and boundary constraints. In this study, the inequality constraint and boundary constraint were applied as physical constraints for consideration of change rate limit and magnitude limit of control inputs. The following equations describe the applied inequality constraint and boundary constraint:

$$A_{in}U_p \leq B_{in} \tag{17}$$
$$U_{p,upper} = \begin{bmatrix} u_{\max} & u_{\max} & \cdots & u_{\max} \end{bmatrix}^T$$
$$U_{p,lower} = \begin{bmatrix} u_{\min} & u_{\min} & \cdots & u_{\min} \end{bmatrix}^T \tag{18}$$

where $u_{\max}$ and $u_{\min}$ are maximum and minimum control inputs that a physical actuator can produce in the real world. Matrices A_{in} and B_{in} represent inequality constraints and contain the current and future control inputs, respectively.

The following equations describe A_{in} and B_{in}.

$$A_{in,1} = \begin{bmatrix} I_m & -I_m & 0 & 0 & \cdots \\ 0 & I_m & -I_m & 0 & \cdots \\ 0 & 0 & I_m & -I_m & \cdots \\ 0 & 0 & 0 & I_m & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{19}$$

$$A_{in,2} = -A_{in,1}$$

$$B_{in,1} = \begin{bmatrix} u_k + \Delta u_{\max} & \Delta u_{\max} & \cdots & \Delta u_{\max} \end{bmatrix}^T$$

$$B_{in,1} = \begin{bmatrix} -u_k + \Delta u_{\max} & \Delta u_{\max} & \cdots & \Delta u_{\max} \end{bmatrix}^T \tag{20}$$

$$A_{in} = \begin{bmatrix} A_{in,1} & A_{in,2} \end{bmatrix}^T, B_{in} = \begin{bmatrix} B_{in,1} & B_{in,2} \end{bmatrix}^T \tag{21}$$

where $\Delta u_{\max}$ represents the maximum input difference that an actual control actuator can make, and it takes strictly positive values, and m is the number of control inputs of the system. The MATLAB function for quadratic programming was used, and constraints were also applied to the quadratic programming function using matrices A_{in}, B_{in}, $U_{p,upper}$, and $U_{p,lower}$ designed in MATLAB. For the design of the adaptive and weighted MPC algorithm, the disturbance was estimated based on a sliding mode observer, and adaptation algorithms were developed for adjusting the prediction horizon N_p and weighting values w. The three algorithms mentioned above are described and explained in the following sub-sections. The next sub-section explains the sliding mode observer design under finite stability conditions for disturbance estimation.

B. SLIDING MODE OBSERVER-BASED DISTURBANCE ESTIMATION

In this study, the sliding mode observer was adopted for disturbance estimation under finite stability conditions. The following equations describe the observer dynamics based on (3), the linear transformation matrix for observer design, and observer output y_o, respectively.

$$\dot{\hat{e}} = A\hat{e} - Bu + G_n v \tag{22}$$

$$T = \begin{bmatrix} null(C_o) & C_o \end{bmatrix}^T \tag{23}$$

$$y_o = C_o e \tag{24}$$

where v and G_n represent the injection term and its distribution matrix, respectively. $\hat{e}$ and C_o are the estimated error and output matrix for the sliding mode observer, respectively. The output matrix and distribution matrix are defined as $C_o = \begin{bmatrix} 1 & 1 \end{bmatrix}$ and $G_n = \begin{bmatrix} L & -I \end{bmatrix}^T$. L is the design parameter for observer stability. Based on linear transformation of the error state and observer output, partitioned error equations of error $\tilde{e}$ between current error and estimated error can be derived as follows.

$$\dot{\tilde{e}}_s = A_{11}\tilde{e}_s + A_{12}\tilde{e}_y + Lv \tag{25}$$

$$\dot{\tilde{e}}_y = A_{21}\tilde{e}_s + A_{22}\tilde{e}_y - v \tag{26}$$

where $\tilde{e}_s$ and $\tilde{e}_y$ represent estimation errors for transformed error by the transformation matrix and output error, respectively. v is the injection term designed so that the errors converge to zero. To ensure the stability of errors defined in (25) and (26), we first used the Lyapunov direct method to stabilize the output error under finite stability conditions. The cost function for the Lyapunov method was defined with two conditions as follows.

$$J_o = \frac{1}{2}\tilde{e}_y^2 \tag{27}$$

$$\text{Condition-1} : \lim_{|\tilde{e}_y| \to \infty} J_o = \infty$$

$$\text{Condition-2} : \dot{J}_o \leq -\alpha J_o^{1/2} \tag{28}$$

Condition-1 is the basic condition such that the cost value becomes infinite if the observer output error approaches infinity. Condition-2 represents the finite stability conditions. The time-derivative of the cost function was derived, and the injection term was designed using the sign value of the output error. The following equation is used to obtain the time-derivative of the cost function.

$$\dot{J}_o = \tilde{e}_y\dot{\tilde{e}}_y = \tilde{e}_y\left(A_{21}\tilde{e}_s + A_{22}\tilde{e}_y - v\right) \tag{29}$$

With the assumption that the observer disturbance $A_{21}\tilde{e}_x + A_{22}\tilde{e}_y$ on the right side of (29) is bounded by upper limit L_b, i.e., $\left|A_{21}\tilde{e}_x + A_{22}\tilde{e}_y\right| \leq L_b$ and the injection term is defined by $v = -\rho sign(\tilde{e}_y)$, (29) can be rewritten as follows.

$$\dot{J}_o = \tilde{e}_y\dot{\tilde{e}}_y \leq -\left|\tilde{e}_y\right|(\rho - L_b) \tag{30}$$

As it can be seen in (30), the magnitude of the injection term ρ should be greater than or equal to the disturbance boundary L_b for asymptotic stability of the output error. To ensure the finite stability of the output error, we applied the second condition described in (28). The second condition can be rewritten based on the separation of variables and the cost function as follows.

$$\dot{J}_o \leq -\frac{\alpha}{\sqrt{2}}\left|\tilde{e}_y\right| \tag{31}$$

The two inequality conditions for the time-derivatives of the cost value in (30) and (31) should be satisfied for observer stability. The magnitude of the injection term ρ can be determined by setting the right-side terms of (30) and (31) as equal. As a result, the injection value v for output error stability can be derived according to (33) using the determined ρ value and the sign of observer output error.

$$\rho = L_b + \frac{\alpha}{\sqrt{2}} \tag{32}$$

$$v = -\left(L_b + \frac{\alpha}{\sqrt{2}}\right) sign\left(\tilde{e}_y\right) \tag{33}$$

Because the injection term can stabilize the observer output error in finite time based on the design process and conditions, the observer output error can converge to zero in finite time. With the assumption that the observer output error converges to zero, the equivalent injection term v_{eq} can be derived based on (25) and (26), which can be rewritten as follows.

$$v_{eq} = A_{21}\tilde{e}_s \tag{34}$$

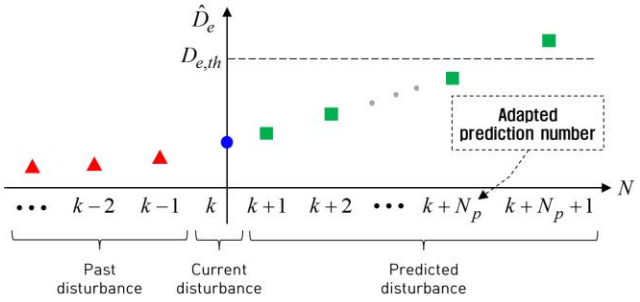

FIGURE 2. Concept of prediction horizon adaptation for MPC algorithm.

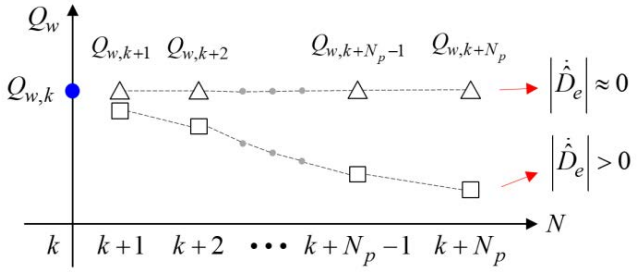

FIGURE 3. Concept of weighting value adaptation algorithm.

$$\dot{\tilde{e}}_s = (A_{11} + LA_{21})\,\tilde{e}_s \tag{35}$$

To stabilize the transformed error state $\tilde{e}_s$ in (35), the design parameter L was designed so that the eigenvalue of the differential equation is negative definite. After convergence of the observer output and transformed state errors, the disturbance term D_e in (3) can be estimated based on the equivalent injection value and its distribution matrix as follows.

$$\hat{D}_e = \dot{x}_t - Ax_t - Fw = G_n v_{eq} \tag{36}$$

The estimated disturbance $\hat{D}_e$ was used for determination of the length of prediction horizon N_p and weighting values Q_w for future states for adaptation of the MPC algorithm. The next sub-section describes the prediction horizon adaptation method of the proposed algorithm.

C. PREDICTION HORIZON ADAPTATION ALGORITHM

In this study, the adaptation algorithm for the length of prediction horizon in MPC was designed using the estimated disturbance from the grey prediction method and the threshold approach. Fig. 2 illustrates the concept of the prediction horizon adaptation algorithm designed in this study, where $D_{e,th}$ represents the disturbance threshold for determination of the length of prediction horizon as a design parameter.

As shown in Fig. 2, the adapted prediction number N_p was designed such that the predicted disturbance is greater than the defined disturbance threshold. For disturbance prediction, the grey prediction method was used with the estimated and saved past N_{past} disturbances. The first-order differential equation for disturbance prediction is

$$\dot{w}_{D_e} + a w_{D_e} = b \tag{37}$$

where a and b are the coefficients of the differential equation. The coefficients were estimated for disturbance prediction using the least squares method. After the discretization of (37) using the same discretization time Δt, matrices M_{D_e} and N_{D_e} based on the linear equation of least squares can be derived as follows.

$$M_{D_e}\begin{bmatrix} a & b \end{bmatrix}^T = N_{D_e} \tag{38}$$

$$M_{D_e} = \begin{bmatrix} \hat{D}_{e,k} & \cdots & \hat{D}_{e,k-N_{past}+1} \\ -1 & \cdots & -1 \end{bmatrix}^T \tag{39}$$

$$N_{D_e} = \begin{bmatrix} \left(D_{e,k-1} - D_{e,k}\right)/\Delta t \\ \vdots \\ \left(D_{e,k-N_{past}+1} - D_{e,k-N_{past}+2}\right)/\Delta t \end{bmatrix} \tag{40}$$

Using matrices M_{D_e} and N_{D_e}, we can estimate the coefficients a and b as follows.

$$\begin{bmatrix} a & b \end{bmatrix}^T = \left(M_{D_e}^T M_{D_e}\right)^{-1} M_{D_e}^T N_{D_e} \tag{41}$$

Based on the estimated coefficients and discretized equation of the differential equation, the future disturbances can be predicted by considering the changing tendency of the saved past disturbances. Based on the adapted prediction horizon with the threshold approach using the predicted disturbances, the length of prediction horizon in the MPC for predicted output derivation can be determined adaptively in real time. It is designed such that the weighting factor W_u for future input difference minimization is tuned according to the adapted prediction horizon N for determining a proper weighting ratio in the cost value. The tuning rule was designed such that the weighting factor decreased as the prediction horizon length decreased. The weighting factors for tuning were determined empirically. The development of a weighting factor adaptation algorithm is considered as a topic of future work for improving control performance. Through the implementation of the adapted prediction horizon with the designed threshold approach, negative effects on control performance from disturbances can be reduced. The next sub-section describes the proposed weighting value adaptation algorithm using the estimated disturbance.

D. WEIGHTING VALUE ADAPTATION ALGORITHM

In this study, the weighting value Q_w adaptation algorithm was designed for adjusting the impact of predicted states in the cost function on the control performance using the estimated disturbance. Fig. 3 illustrates the concept of the proposed weighting value adaptation algorithm, where $Q_{w,k}$ is the current weighting value determined as an initial parameter for the control algorithm. It is designed such that the predicted weighting values $Q_{w,k+1}, \cdots, Q_{w,k+N_p}$ decrease exponentially if the magnitude of the estimated disturbance is greater than zero, based on the function representing the

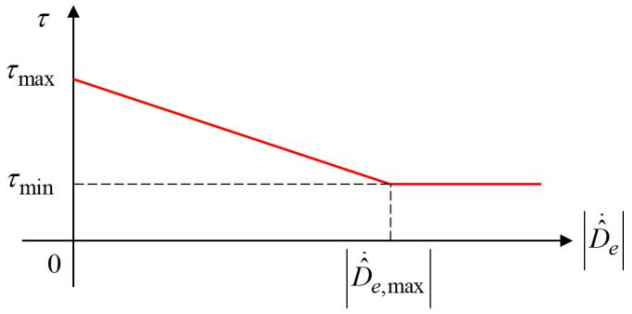

FIGURE 4. Relationship between time constant and magnitude of disturbance change rate.

relationship between time constant τ and the magnitude of the estimated disturbance $\left|\dot{\hat{D}}_e\right|$.

The following equation is the weighting function based on the time constant:

$$Q_{w,i} = \sqrt{C_Q e^{-\frac{1}{\tau}(\Delta t \times i)}}, \quad (i = 1, 2, \cdots, N_p) \tag{42}$$

where C_Q is the proportional gain for determination of the initial magnitude of the weighting value. The weighting value varies with the time constant in the range of $(0, 1]$. In this study, it was designed that the time constant is determined using the estimated disturbance to reduce its negative impact on the control performance. The time constant value decreases to reduce the weighting value if the magnitude of the disturbance change rate is increased as the prediction horizon length increases. Fig. 4 shows the relationship between the time constant and magnitude of the disturbance change rate, where $\tau_{\min}$, $\tau_{\max}$, and $\left|\dot{\hat{D}}_{e,\max}\right|$ represent the minimum and maximum time constants and maximum magnitude of the disturbance change rate as design parameters for the time constant function design, respectively.

The mathematical expression of the relationship function described in Fig. 4 is as follows:

$$\tau_k = \frac{\tau_{\min} - \tau_{\max}}{\left|\dot{\hat{D}}_{e,\max}\right|}\left|\dot{\hat{D}}_{e,k}\right| + \tau_{\max} \tag{43}$$

$$if \ \left|\dot{\hat{D}}_{e,k}\right| = 0 \rightarrow \tau_k = \tau_{\max} \tag{44}$$

$$if \ \left|\dot{\hat{D}}_{e,k}\right| \geq \left|\dot{\hat{D}}_{e,\max}\right| \rightarrow \tau_k = \tau_{\min} \tag{45}$$

The computed time constant is used to compute the weighting value based on (42). The number of computed weighting values is equal to N_p. Consequently, the computed weighting value Q_w is used as an element of W_x in the cost function (15) that can be transformed into the quadratic form to solve an optimization problem. The following equation is the quadratic form of the cost function.

$$J = \frac{1}{2}U_p^T A U_p + B U_p \tag{46}$$

Equation (46) should be strictly convex to successfully solve the optimization problem with constraints [28]. The matrices A and B in Eq. (46) include weighting matrices and matrices for the predicted output Y_p. All the eigenvalues of the matrix A should be positive for the strictly convex condition. To achieve this, weighting values were designed to vary in the positive region. The robust analysis is planned for future work because the weighting value adaptation may influence the control performance. The next section describes the performance evaluation results of the designed adaptive MPC algorithm using the CarMaker software.

TABLE 1. Specifications of the vehicle used for evaluation.

Mass	1,600 kg
Wheel base	2.975 m
Wheel tread	1.63 m
Rotational inertia (z-axis)	2333.6 kgm^2
Cornering stiffness - front (approximated)	73,563 Ns/rad
Cornering stiffness - rear (approximated)	140,740 Ns/rad

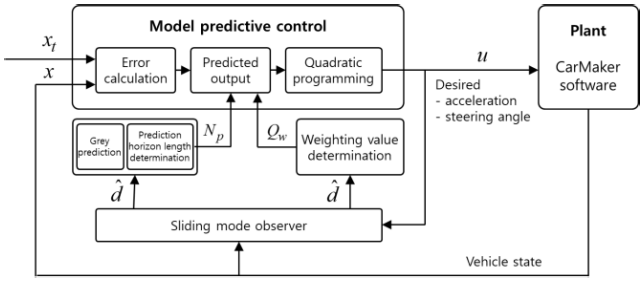

FIGURE 5. Model schematic of the control algorithm used for performance evaluation.

IV. PERFORMANCE EVALUATION

To evaluate the performance of the adaptive and weighted MPC algorithm, a lane change scenario with constant-velocity driving and a car-following scenario were considered. The adaptive MPC algorithm was designed in the MATLAB/Simulink environment, and the CarMaker software was used together with MATLAB/Simulink for performance evaluation. Table 1 presents the specifications of the vehicle used for evaluation.

Figure 5 shows a model schematic for performance evaluation of the control algorithm.

Based on the designed adaptive and weighted MPC algorithm, the control inputs, such as desired acceleration and steering angle, were computed adaptively for path and velocity tracking of the vehicle. To compute the control inputs, we derived the longitudinal and lateral error dynamic models and used them for the MPC design based on the kinematic relationship vehicle state and target state, respectively.

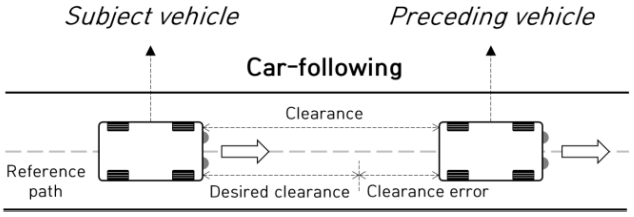

FIGURE 6. Car-following scenario for performance evaluation.

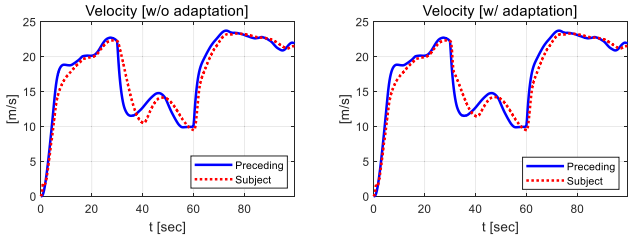

FIGURE 7. Results: velocity [preceding vehicle and subject vehicle].

The following sections describe the performance evaluation results of the longitudinal and lateral autonomous driving.

A. CAR-FOLLOWING SCENARIO

The car-following scenario used for performance evaluation was designed such that the subject vehicle tracked the preceding vehicle velocity and the desired clearance was achieved. In the design, the desired clearance was computed by multiplying the subject vehicle velocity and time headway. The error dynamics used for the MPC formulation were based on a kinematic model that represents the kinematic relationship between positions of the subject and preceding vehicles [28]. Fig. 6 and (47) show the considered car-following scenario and kinematic model-based error dynamics used for evaluation.

$$\begin{bmatrix} \dot{e}_{1,L} \\ \dot{e}_{2,L} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e_{1,L} \\ e_{2,L} \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} a_s + \begin{bmatrix} 0 \\ 1 \end{bmatrix} a_p \quad (47)$$

where $e_{1,L}$ and $e_{2,L}$ represent the clearance error and relative velocity, respectively. a_s and a_p are longitudinal accelerations of the subject vehicle and preceding vehicle, respectively. In this study, the preceding vehicle acceleration was considered as a disturbance in the kinematic model of (47) and was estimated using the designed sliding mode observer. Table 2 lists the parameters used for performance evaluation of the car-following scenario. As a main tuning parameter in the table, the threshold, computation period, physical constraints, and output matrix can influence control performance.

Figs. 7–17 show the performance evaluation results of the proposed adaptive and weighted MPC algorithm. The legend "w/o adaptation" represents the results when the fixed weighting factor and prediction horizon were applied for the adaptive MPC algorithm.

As shown in Fig. 7, the subject vehicle was able to track the preceding vehicle's velocity using adaptive control both with and without adaptation with similar tracking performance. Furthermore, there is a difference in tracking error performance after deceleration begins after 30 s. Fig. 8 shows the

TABLE 2. Control parameters used for performance evaluation [Car-following scenario].

Control parameters for car-following scenario	
Threshold for prediction horizon adaptation	5
Maximum prediction horizon	15
Computation period	0.1 s
Discretization time	0.1 s
Minimum/maximum time constant	100 s / 0.1 s
Maximum disturbance change rate	20
Output matrix [longitudinal]	$C_{longitudinal} = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$
Input limits [upper/lower]	5/-6
Input change rate limit	15
Proportional gain for weighting value determination	10
Observer parameters for car-following scenario	
Output matrix	$C_o = \begin{bmatrix} 1 & 1 \end{bmatrix}$
Time constant for equivalent injection	0.1 s

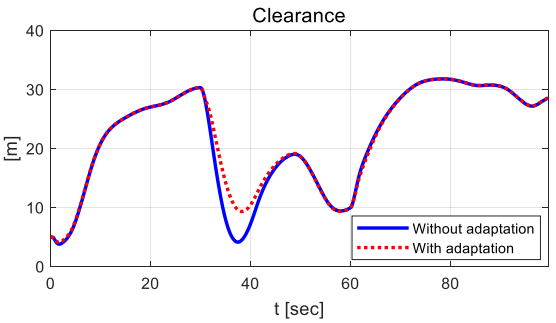

FIGURE 8. Results: clearance [w/o and w/ adaptation].

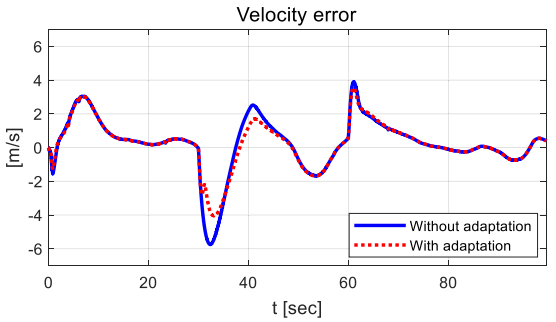

FIGURE 9. Results: velocity error [w/o and w/ adaptation].

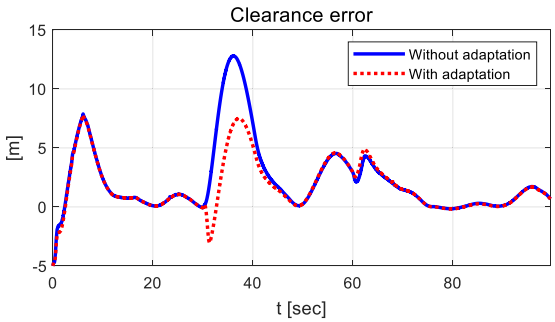

FIGURE 10. Results: clearance error [w/o and w/ adaptation].

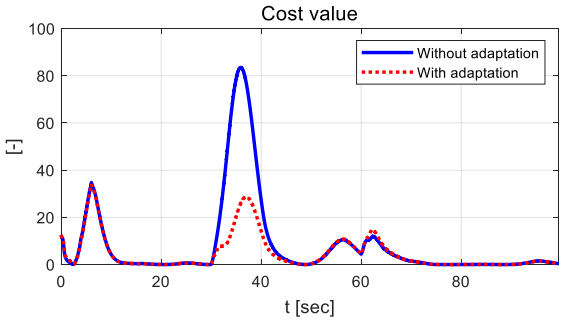

FIGURE 11. Results: cost value [w/o and w/ adaptation].

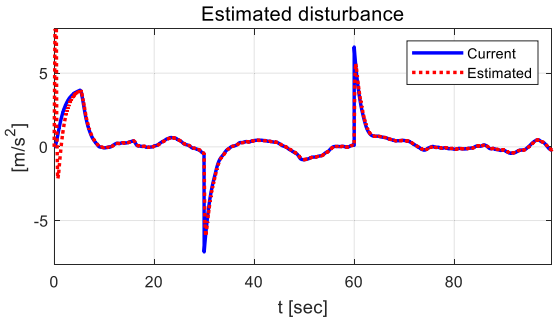

FIGURE 12. Results: preceding vehicle acceleration [current and estimated values].

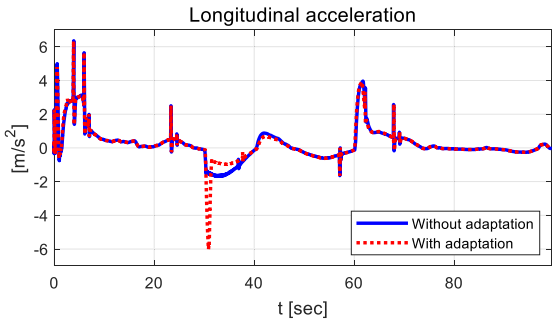

FIGURE 13. Results: subject vehicle acceleration [w/o and w/ adaptation].

clearance between the preceding vehicle and subject vehicle. Figs. 9 and 10 show the control errors of velocity and clearance. The magnitude of control errors with adaptation was less than that without adaptation around 35 s because

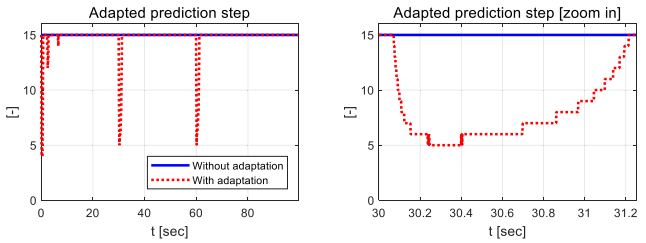

FIGURE 14. Results: adaptive prediction horizon [w/o and w/ adaptation].

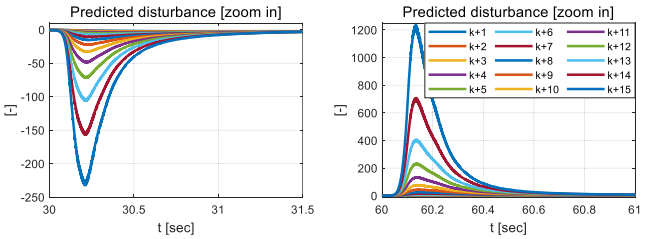

FIGURE 15. Results: predicted disturbances [w/ adaptation].

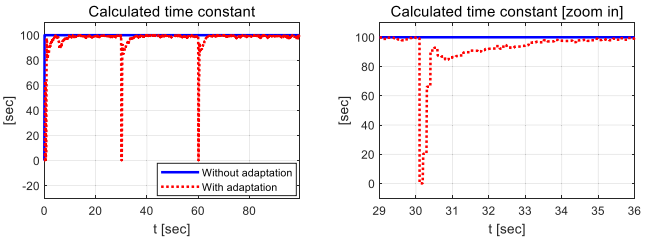

FIGURE 16. Results: calculated time constant [w/o and w/ adaptation].

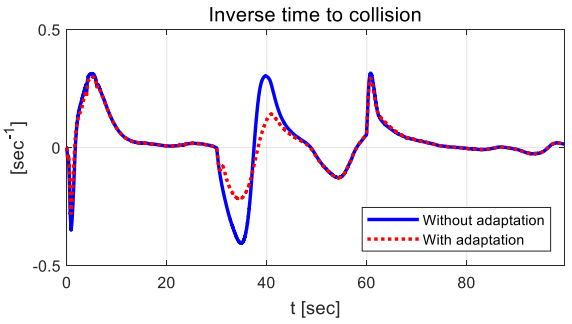

FIGURE 17. Results: inverse time to collision [w/o and w/ adaptation].

the prediction horizon and weighting values were adapted by the designed adaptation algorithms. Fig. 11 shows the cost value as the sum of squared control errors. The maximum cost value without adaptation was less than that with adaptation, with a difference of approximately 50 between the results. The estimated disturbance and current disturbance can be seen in Fig. 12, which shows that the disturbance could be reasonably estimated by the designed sliding mode observer with finite stability conditions. Longitudinal acceleration values are plotted in Fig. 13. The controller with adaptation used a relatively higher acceleration value after 30 s than that without adaptation by using weighting factors

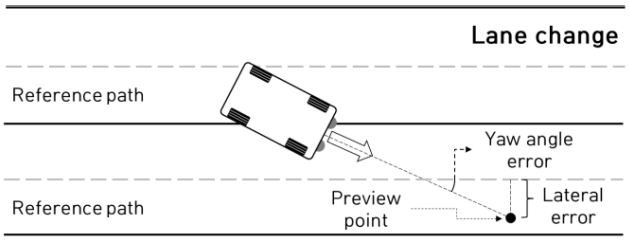

FIGURE 18. Lane change scenario for performance evaluation.

and prediction horizon adaptation. Figs. 14 and 15 show the adapted prediction horizon and predicted disturbance ($k+1$, $\cdots$, $k+N_p$), respectively. In Fig. 15, it is shown that the predicted disturbances decrease and increase exponentially around 30.2 s and 60.15 s, respectively based on the first-order differential equation-based grey prediction method. The length of prediction horizon began to decrease after 30 s and 60 s because the predicted disturbances exceeded the respective designed threshold values. The calculated time constant to determine the weighting factor is shown in Fig. 16, where the graph shape is similar to that of the acceleration graph in Fig. 13. The computed time constant respectively decreased after 30 s and 60 s as the estimated acceleration changed significantly. As the length of prediction horizon decreased and the increased change rate of the estimated disturbance decreased the time constant, the designed weighting function caused a decrease in the weighting values for future states. Consequentially, a relatively large acceleration control input can be derived from these adaptation results. In this study, the inverse time to a collision was computed using the relative velocity and clearance, and the results of the two cases shown in Fig. 17 indicate that the absolute magnitude of the inverse time to collision with adaptation was less than that without adaptation. The next sub-section describes the performance evaluation results under a lane change scenario with constant velocity driving.

B. LANE CHANGE SCENARIO WITH CONSTANT VELOCITY DRIVING

The lane change scenario was designed with constant velocity driving for performance evaluation of the control algorithm. Reference paths were used for straight driving and lane change. The error dynamics used for the lane change scenario were based on the kinematic relationship between the mass center position of the subject vehicle and the reference path [29]. Fig. 18 and (48) describe the lane change scenarios and error dynamics model used for the performance evaluation.

$$\begin{bmatrix} \dot{e}_y \\ \dot{e}_\psi \end{bmatrix} = \begin{bmatrix} 0 & v_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e_y \\ e_\psi \end{bmatrix} + \begin{bmatrix} 0 \\ v_x/L \end{bmatrix} \delta + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \dot{\psi}_d \quad (48)$$

where e_y and e_ψ represent the lateral error and yaw angle error, respectively. v_x and L are the longitudinal velocity of the subject vehicle and length between the front axle and rear axle of the vehicle, respectively. δ and $\dot{\psi}_d$ are the front steering angle and desired yaw rate for path tracking, respectively. Because the reference path is a straight line, the desired yaw rate was assumed to be zero, and the disturbance (i.e., system model uncertainty of 48)) was estimated by applying a sliding mode observer. Table 3 lists the parameters used for performance evaluation of the lane change scenario.

TABLE 3. Control parameters used for performance evaluation [Lane change scenario].

Control parameters for lane change scenario	
Threshold for prediction horizon adaptation	0.2
Maximum prediction horizon	15
Computation period	0.1 s
Discretization time	0.1 s
Minimum/maximum time constant	100 s/0.1 s
Maximum disturbance change rate	20
Output matrix [lateral]	$C_{lateral} = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}$
Input limits [upper/lower]	40/-40 deg
Input change rate limit	21.18 deg/s
Proportional gain for weighting value determination	10
Observer parameters for lane change scenario	
Output matrix	$C_o = \begin{bmatrix} 1 & 1 \end{bmatrix}$
Time constant for equivalent injection	0.1 s

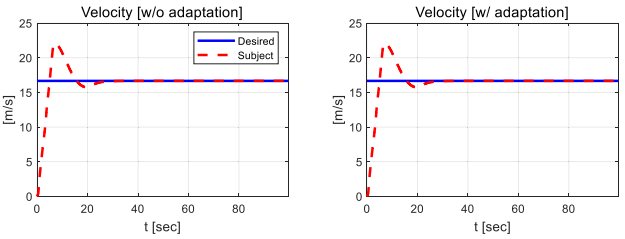

FIGURE 19. Results: velocity [desired velocity and subject vehicle velocity].

Figs. 19–27 show the performance evaluation results of the adaptive and weighted MPC algorithm proposed in this study for the given scenario.

The left and right subfigures of Fig. 19 illustrate the velocity tracking performance without and with adaptation, respectively. As a consequence of applying the same longitudinal controller for both cases (without and with adaptation) of constant velocity tracking, as well as relatively small steering inputs, similar tracking performance is observed and both cases achieve reasonable tracking performance with steady-state tracking after 30 s. There is also a relatively high overshoot at near 10 s for both cases because a constant

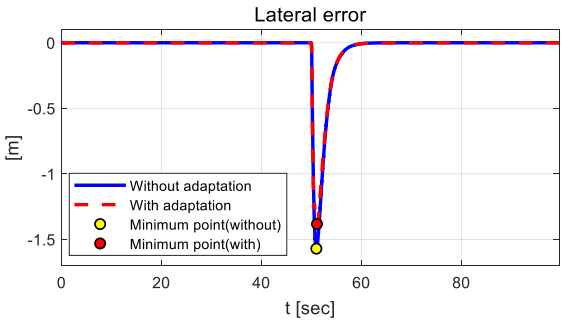

FIGURE 20. Results: lateral error [w/o and w/ adaptation].

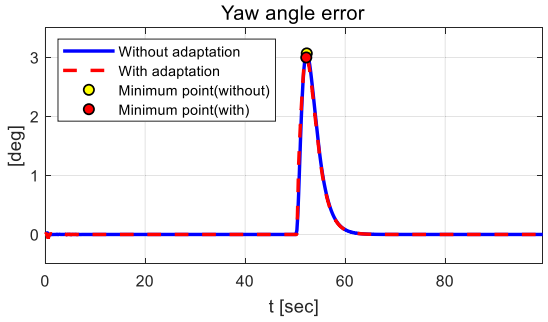

FIGURE 21. Results: yaw angle error [w/o and w/ adaptation].

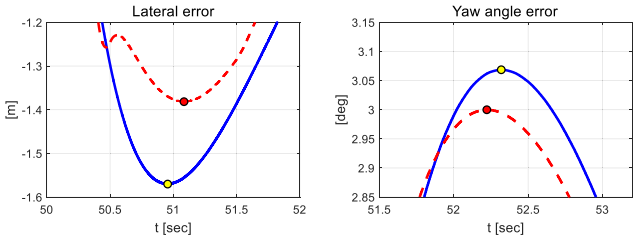

FIGURE 22. Results: lateral and yaw angle errors [zoom in].

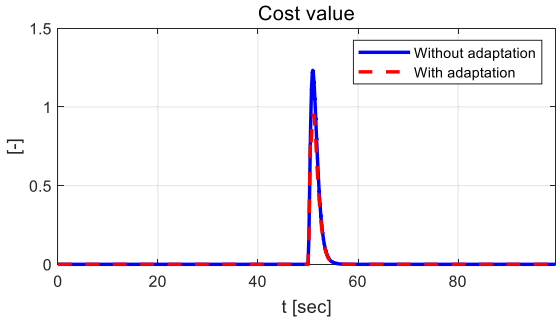

FIGURE 23. Results: cost value [w/o and w/ adaptation].

desired velocity was applied continuously from beginning to end during the performance evaluation (like a step reference input). The lateral and yaw angle errors as a tracking control error can be seen in Figs. 20 and 21, respectively. It can be seen that the maximum magnitude values of lateral and yaw angle errors with adaptation were less than without adaptation. There is a lateral error difference of approximately

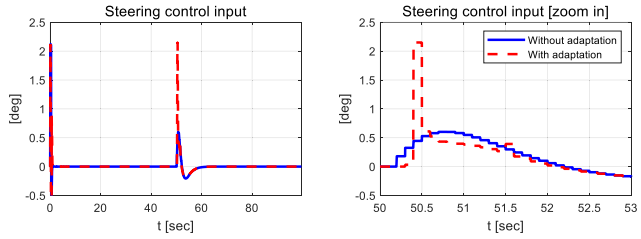

FIGURE 24. Results: steering control input [w/o and w/ adaptation].

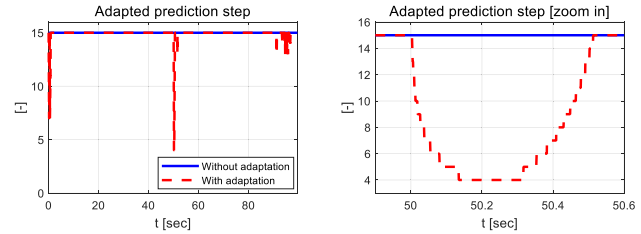

FIGURE 25. Results: adaptive prediction horizon [w/o and w/ adaptation].

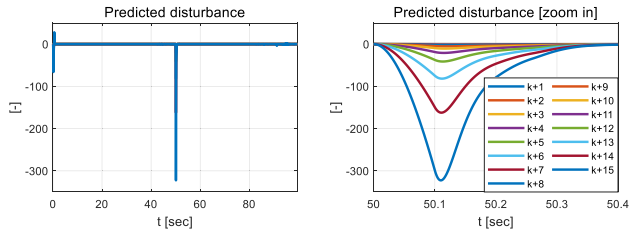

FIGURE 26. Results: predicted disturbances [w/ adaptation].

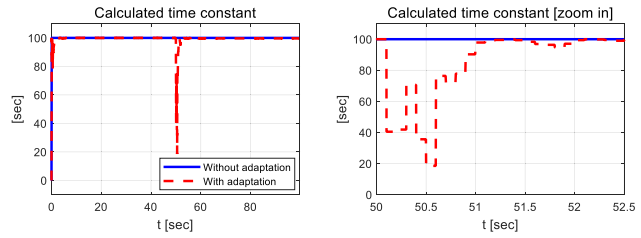

FIGURE 27. Results: calculated time constant [w/o and w/ adaptation].

0.3 m and a yaw angle error difference of approximately 0.05 deg between the two cases. The cost value for quantitative analysis of control performance was computed by summing the squared lateral and yaw angle errors, as shown in Fig. 23. There is a cost value difference of approximately 0.2 between the two cases (i.e., with and without adaptation). The overall steering control input as well as the zoomed-in steering control input, showing a maximum difference of approximately 1.5 degrees, are shown in Fig. 24. It can be seen that the steering control inputs with adaptation were relatively larger and faster than without adaptation, as the weighting factor and prediction horizon were adjusted by the designed algorithms. Also, the steering control inputs with adaptation made the control errors converge to zero faster

than without adaptation. Figs. 25 and 26 show the adapted prediction horizon and predicted disturbances ($k+1, \cdots, k+N_p$) by the first-order differential equation-based grey prediction, respectively. The predicted disturbances in Fig 26 are shown to be decreasing exponentially around 50.1 s. The length of prediction horizon began to decrease after 50 s because the predicted disturbances exceeded the designed threshold value. In this study, upper and lower limits of the prediction horizon were not considered to guarantee the MPC stability with respect to changing prediction horizons. Thus, the derivation of upper and lower limits of prediction horizon based on stability analysis [30], [31] is planned for future work.

The calculated time constant for the weighting factor determination is shown in Fig. 27, which decreased after 50 s as the change in estimated disturbance was significant. It can be seen that the weighting value on future states decreased by using the weighting function. This is due to the fact that the length of prediction horizon decreased, and a reduction in time constant was caused by a higher change rate of the estimated disturbance. We can also see that by using a relatively large steering control input, the maximum absolute values of lateral and yaw angle errors with adaptation decreased more than without adaptation.

V. CONCLUSION

This paper proposed an adaptive and weighted MPC algorithm for autonomous vehicles based on disturbance estimation with a sliding mode observer and the grey prediction method. The sliding mode observer was designed under finite stability conditions for disturbance estimation of the system, and the estimated disturbance was predicted based on the grey prediction model. The predicted disturbance was used to determine the length of prediction horizon in the MPC, with the threshold approach for prediction horizon adaptation. Some limitations exist in determining a threshold value for the adaptive prediction horizon in MPC and the size of a window that contains past disturbances for disturbance prediction using the grey prediction method. Therefore, the development of an online self-tuning algorithm to determine the threshold value and size will be considered as future work so that the proposed control algorithm can respond to various autonomous driving conditions. The change rate of the estimated disturbance was also used for the determination of weighting values to reduce the negative impact of unreasonable future disturbance predictions on the control performance. The weighting values were computed using the designed exponential weighting function. The performance evaluation of the proposed adaptive and weighting MPC algorithm was conducted using the CarMaker software under longitudinal and lateral autonomous driving scenarios. The evaluation results from the proposed adaptive and weighted MPC algorithm show more reasonable control performance in terms of control errors in transient response than the conventional MPC algorithm without adaptation. However, there are some limitations to the method for determining a weighting value for the control input difference. Therefore, the adaptation algorithm for the weighting value in the cost function will be studied in the future, which can improve the control performance by mitigating the negative impact of unexpected disturbances on performance. The derivation of upper and lower limits for adapted prediction horizons can be also considered a future extension for ensuring MPC stability when prediction horizons change in real time. Furthermore, data-driven parameter adaptation methods without any rules that rely on a recently developed artificial intelligence technology are feasible for future research. Finally, the proposed algorithms will be experimentally validated with real-world vehicles by incorporating the aforementioned approaches for future studies.

As a result of demonstrating the successful performance of our proposed adaptive and predictive control algorithm, it is expected to be widely applied to various systems, such as autonomous vehicles and robots.

REFERENCES

[1] L. Qiao and W. Zhang, "Trajectory tracking control of AUVs via adaptive fast nonsingular integral terminal sliding mode control," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 1248–1258, Feb. 2020.

[2] A. Altan and R. Hacioğlu, "Model predictive control of three-axis gimbal system mounted on UAV for real-time target tracking under external disturbances," *Mech. Syst. Signal Process.*, vol. 138, Apr. 2020, Art. no. 106548.

[3] V. A. Akpan and G. D. Hassapis, "Nonlinear model identification and adaptive model predictive control using neural networks," *ISA Trans.*, vol. 50, no. 2, pp. 177–194, Apr. 2011.

[4] R. Mahboobi Esfanjani and S. K. Y. Nikravesh, "Stabilizing model predictive control for constrained nonlinear distributed delay systems," *ISA Trans.*, vol. 50, no. 2, pp. 201–206, Apr. 2011.

[5] E. Pourjafari and H. Mojallali, "Predictive control for voltage collapse avoidance using a modified discrete multi-valued PSO algorithm," *ISA Trans.*, vol. 50, no. 2, pp. 195–200, Apr. 2011.

[6] B. Zeng, H. Duan, and Y. Zhou, "A new multivariable grey prediction model with structure compatibility," *Appl. Math. Model.*, vol. 75, pp. 385–397, Nov. 2019.

[7] T. S. Pedersen, K. M. Nielsen, J. Hindsborg, P. Reichwald, K. Vinther, and R. Izadi-Zamanabadi, "Predictive functional control of superheat in a refrigeration system using a neural network model," *IFAC-Papers Line*, vol. 50, no. 1, pp. 43–48, Jul. 2017.

[8] C. Han, M. Li, Y. Jing, L. Liu, Z. Pang, and D. Sun, "Nonlinear model predictive congestion control for networks," *IFAC-Papers Line*, vol. 50, no. 1, pp. 552–557, Jul. 2017.

[9] H. Fukushima, T.-H. Kim, and T. Sugie, "Adaptive model predictive control for a class of constrained linear systems based on the comparison model," *Automatica*, vol. 43, no. 2, pp. 301–308, Feb. 2007.

[10] H. Li and Y. Shi, "Event-triggered robust model predictive control of continuous-time nonlinear systems," *Automatica*, vol. 50, no. 5, pp. 1507–1513, May 2014.

[11] J. Kohler, P. Kotting, R. Soloperto, F. Allogower, and M. Müller, "A robust adaptive model predictive control framework for nonlinear uncertain systems," *Int. J. Robust Nonlin. Control*, vol. 4, pp. 1–25, Oct. 2020.

[12] I. Hajizadeh, M. Rashid, and A. Cinar, "Plasma-insulin-cognizant adaptive model predictive control for artificial pancreas systems," *J. Process Control*, vol. 77, pp. 97–113, May 2019.

[13] G. Garimella, M. Sheckells, and M. Kobilarov, "Robust obstacle avoidance for aerial platforms using adaptive model predictive control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 5876–5882.

[14] M. Babaie, M. Sharifzadeh, M. Mehrasa, G. Chouinard, and K. Al-Haddad, "Supervised learning model predictive control trained by ABC algorithm for common-mode voltage suppression in NPC inverter," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 9, no. 3, pp. 3446–3456, Jun. 2021.

[15] U. Rosolia and F. Borrelli, "Learning model predictive control for iterative tasks: A computationally efficient approach for linear system," *IFAC-Papers Line*, vol. 50, no. 1, pp. 3142–3147, 2017.
[16] X. Li, Z. Zhao, and F. Liu, "Latent variable iterative learning model predictive control for multivariable control of batch processes," *J. Process Control*, vol. 94, pp. 1–11, Oct. 2020.
[17] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *Proc. IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 6059–6066.
[18] G. Ripaccioli, D. Bernardini, S. Di Cairano, A. Bemporad, and I. V. Kolmanovsky, "A stochastic model predictive control approach for series hybrid electric vehicle power management," in *Proc. Amer. Control Conf.*, Baltimore, MD, USA, Jun. 2010, pp. 5844–5849.
[19] M. Tsao, R. Iglesias, and M. Pavone, "Stochastic model predictive control for autonomous mobility on demand," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 3941–3948.
[20] J. Suh, H. Chae, and K. Yi, "Stochastic model-predictive control for lane change decision of automated driving vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 4771–4782, Jun. 2018.
[21] D. Moser, R. Schmied, H. Waschl, and L. del Re, "Flexible spacing adaptive cruise control using stochastic model predictive control," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 1, pp. 114–127, Jan. 2018.
[22] H. He, H. Jia, C. Sun, and F. Sun, "Stochastic model predictive control of air conditioning system for electric vehicles: Sensitivity study, comparison, and improvement," *IEEE Trans. Ind. Informat.*, vol. 14, no. 9, pp. 4179–4189, Sep. 2018.
[23] U. Rosolia, A. Carvalho, and F. Borrelli, "Autonomous racing using learning model predictive control," in *Proc. Amer. Control Conf. (ACC)*, May 2017, pp. 5115–5120.
[24] M. Elsisi and M. A. Ebrahim, "Optimal design of low computational burden model predictive control based on SSDA towards autonomous vehicle under vision dynamics," *Int. J. Intell. Syst.*, vol. 36, no. 11, pp. 6968–6987, Nov. 2021.
[25] M. Elsisi, "Optimal design of nonlinear model predictive controller based on new modified multitracker optimization algorithm," *Int. J. Intell. Syst.*, vol. 35, no. 11, pp. 1875–1878, 2021.
[26] M. Elsisi, "New design of adaptive model predictive control for energy conversion system with wind torque effect," *J. Cleaner Prod.*, vol. 240, Oct. 2019, Art. no. 118265.
[27] M. Elsisi, M. Soliman, M. A. S. Aboelela, and W. Mansour, "Improving the grid frequency by optimal design of model predictive control with energy storage devices," *Optim. Control Appl. Methods*, vol. 39, no. 1, pp. 263–280, Jan. 2018.
[28] S. Park, K. Oh, Y. Jeong, and K. Yi, "Model predictive control-based fault detection and reconstruction algorithm for longitudinal control of autonomous driving vehicle using multi-sliding mode observer," *Microsyst. Technol.*, vol. 26, no. 1, pp. 239–264, Jan. 2020.
[29] K. Oh and K. Yi, "A predictive driver model with physical constraints for closed loop simulation of vehicle-driver system," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 3126–3131.
[30] Z. Sun, L. Dai, K. Liu, D. V. Dimarogonas, and Y. Xia, "Robust self-triggered MPC with adaptive prediction horizon for perturbed nonlinear systems," *IEEE Trans. Autom. Control*, vol. 64, no. 11, pp. 4780–4787, Nov. 2019.
[31] Y. Song, Z. Wang, L. Zou, and S. Liu, "Endec-decoder-based N-step model predictive control: Detectability, stability and optimization," *Automatica*, vol. 135, Jan. 2022, Art. no. 109961.

KWANGSEOK OH received the B.S. degree in mechanical engineering from Hanyang University, Seoul, in 2009, and the M.S. and Ph.D. degrees in mechanical and aerospace engineering from Seoul National University, Seoul, in 2013 and 2016, respectively. From 2016 to 2017, he was an Assistant Professor wit the Automotive Engineering Department, Honam University. Since 2017, he has been an Assistant Professor with the Mechanical Engineering Department, Hankyong National University, Anseong-si, South Korea. His research interests include fail-safe systems for autonomous driving, and adaptive and predictive control.

JAHO SEO received the B.S. degree in agricultural machinery and process engineering from Seoul National University, Seoul, South Korea, in 1999, the M.E. degree in mechanical engineering from the University of Quebec (Ecole de Technologie Superieure), Montreal, QC, Canada, in 2006, and the Ph.D. degree in mechanical engineering from the University of Waterloo, Waterloo, ON, Canada, in 2011. He was with the Department of Mechanical and Mechatronics Engineering, University of Waterloo, as a Postdoctoral Fellow, in 2011, the Department of System Reliability, Korea Institute of Machinery & Materials (KIMM), as a Senior Researcher, during 2012–2016, and the Department of Biosystems Machinery Engineering, Chungnam National University, South Korea, as an Assistant Professor, during 2016–2017. Since 2017, he has been an Assistant Professor at the Department of Automotive and Mechatronics Engineering, Ontario Tech University, where he has been involved in research on the development of autonomous control systems for intelligent mobile machines.