Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Stochastic Model Predictive Velocity Control for Automobiles Considering Uncertainty of Nearby Vehicles

Kentaro Hashimoto * **Yoshihide Mizushima** * **Koji Shibata** *
Isao Okawa ** **Kenichiro Nonaka** *

* *Graduate School of Integrative Science and Engineering, Mechanics, Tokyo City University, 1-28-1, Tamazutsumi, Setagaya, Tokyo, Japan (e-mail: g1981038@tcu.ac.jp)*
** *DENSO CORPORATION, Japan (e-mail: isao.okawa.j3g@jp.denso.com)*

Abstract: In this paper, we present vehicle velocity control based on stochastic model predictive control applied to an actual automobile near other vehicles with uncertain motion. Modern external sensors can measure the pose and velocity of transportation participants, whose motion can be predicted utilizing a model; however, observation and process noise results in uncertainty. Thus, near other vehicles, the velocity of the ego vehicle should be reduced to account for the variance of nearby vehicle motion while suppressing the reduction of speed. In this paper, we utilize stochastic model predictive control to reduce the expectation of relative velocity with respect to nearby vehicles during passing. We evaluate the proposed control using numerical simulation and an experiment with an automobile developed for self-driving and equipped with GNSS, radar, and LiDAR. The results show that the vehicle velocity is automatically reduced based on the expectation of relative velocity calculated from its probabilistic distribution.

Keywords: Self-driving vehicle, Automobile, Stochastic model predictive control, Velocity control, Kalman filter, Relative velocity, Expectation, Risk prediction

1. INTRODUCTION

Automatic driving assistance systems have been extensively studied. Although automatic driving on highways has been implemented with systems such as adaptive cruise control and lane tracking control, it is difficult to implement in urban areas because nearby traffic participants must be considered. For example, on a narrow two-lane road, skilled drivers avoid oncoming vehicles by decelerating and steering because they predict a possibility of collision (Akagi and Raksincharoensak, 2015a). Such predictive driving reduces the possibility of traffic accidents and makes passengers feel safe. Thus, prediction and reduction of potential risk are important for automatic driving in urban areas.

Obstacle avoidance has been extensively studied. For example, the artificial potential field (APF) method (Khatib, 1990; Park et al., 2001), model predictive control (MPC) (Okawa and Nonaka, 2018), and the combination of APF and MPC (Yoon et al., 2009; Shibata et al., 2018) have been proposed. MPC is a kind of real-time optimal control that optimizes motion behavior on the prediction horizon under various constraints. Thus, MPC is suitable for designing an optimal path that guarantees collision avoidance. For speed control, methods such as APF (Akagi and Raksincharoensak, 2015b), the combination of MPC and APF (Hasegawa et al., 2015), and the combination of MPC and constraints (Li et al., 2011; Mizushima et al., 2019) have been proposed. However, the physical meaning of the APF is obscure and for other methods it is sometimes difficult to obtain the proper control parameter that fits a given situation. The position and velocity of the ego vehicle and those of nearby transportation participants vary with time and thus the probabilistic uncertainty of their dynamics should be considered. Therefore, speed control should quantitatively evaluate the collision risk and account for these probabilistic uncertainties.

An MPC-based stochastic method for the prevention of collisions that utilizes a chance constraint (Moser et al., 2017) and a probabilistic method (Shimizu et al., 2016) for evaluating the risk of collision with a pedestrian running out onto a street have been proposed. To account for velocity uncertainty, stochastic MPC, which minimizes the expectation of the relative velocity to reduce collision risk, has been proposed (Shibata et al., 2019). This method was verified to be suitable for collision avoidance in a narrow corridor and applied to an electric wheelchair moving at low speed. The application of this method to self-driving automobiles, which have far larger inertia, higher speed, and stricter safety requirements, is challenging.

This paper presents a velocity control method that reduces collision risk for automobiles considering the uncertainties in the positions and velocities of nearby vehicles. We validate the proposed method through experiments in which a vehicle avoids a moving obstacle (i.e., another vehicle). The positions of the moving obstacles are observed by sensors and thus contain uncertainty. We estimate the positions and velocities using a Kalman filter, from which the posterior covariance matrix is simultaneously obtained

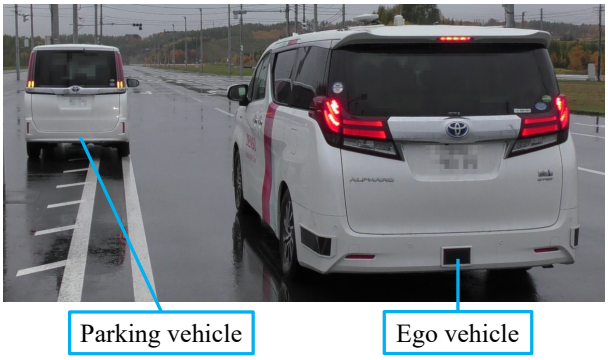


Fig. 1. Ego vehicle passing a parked vehicle.

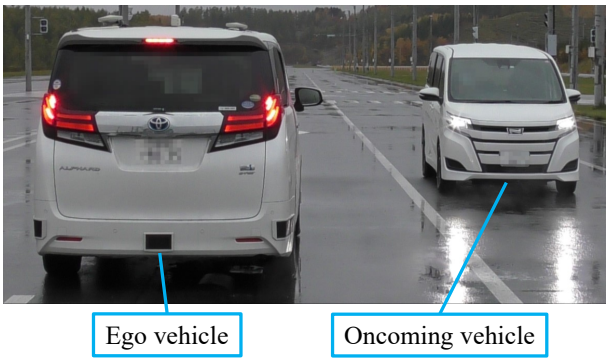


Fig. 2. Ego vehicle passing an oncoming vehicle.

and the expected position and variance of obstacles are predicted. Next, we build a probability density function of the velocity to evaluate the expectation of the relative velocity with respect to the obstacles. MPC minimizes the evaluation function, which comprises the expected relative velocity, the error from the target state, and the amount of input. The proposed MPC generates a jerk input or a velocity input to reduce the risk of collision. We verify the proposed method using simulations where the ego vehicle passes a parking vehicle or an oncoming vehicle. In experiments, we use an experimental self-driving vehicle to validate the real-time feasibility and robustness against sensor measurement error and motion model uncertainty.

2. ASSUMPTIONS AND TEST SITUATIONS

In this paper, we consider the situations depicted in Fig. 1, where the ego vehicle attempts to pass a parked vehicle, and Fig. 2, where the ego vehicle attempts to pass an oncoming vehicle on a narrow road. We assume that the position and orientation of the ego vehicle can be localized, and that nearby objects are observed by sensors with measurement error. We also assume that each obstacle moves at a constant velocity along the road and that its motion is disturbed by random noise with a given variance.

3. CONTROLLED OBJECT

In this section, we describe a controller model and a plant model of the ego vehicle. Because the focus is on velocity control, we only consider the longitudinal direction in the model. The controller model is described as the following triple integrator system with jerk input (Mizushima et al., 2019):

$$\zeta_{k+1} = \begin{bmatrix} 1 & \Delta & \Delta^2/2 \\ 0 & 1 & \Delta \\ 0 & 0 & 1 \end{bmatrix} \zeta_k + \begin{bmatrix} \Delta^3/6 \\ \Delta^2/2 \\ \Delta \end{bmatrix} j_k, \tag{1}$$

where $\zeta := [s\ v\ a]^T$, where s is the travel distance, v is the speed, a is the acceleration, j is the jerk input, and Δ is the control cycle.

The plant model used to represent the actual vehicle in the simulation is a bicycle model (Okajima and Asai, 2004) that considers tire force due to slip and the dynamics of tire rotation speed, described as:

$$f_x := K_s \frac{r\omega - v}{\max(r\omega, v)}, \tag{2}$$

$$m\dot{v} = 4f_x - dv, \tag{3}$$

$$I_w\dot{\omega} = K_m a_{\text{in}} - rf_x, \tag{4}$$

where f_x is the longitudinal tire force, K_s is the longitudinal tire stiffness, r is the tire radius, ω is the tire rotation speed, I_w is the inertia moment, a is the input acceleration, K_m is the torque coefficient for input acceleration, and d is the vehicle resistance coefficient.

4. MODEL PREDICTIVE CONTROLLER

In this section, to achieve automatic driving at a safe velocity considering the probabilistic uncertainty of nearby vehicles, we propose a velocity controller that takes into account the expectations of the relative velocities of nearby vehicles in the evaluation function. To suppress excessive acceleration, we impose constraints on the state.

4.1 Velocity trajectory generation

We first impose constraints on the vehicle. The constraints comprise the controller model in Eq. (1) and the initial state, defined as follows:

$$\zeta_0 = \zeta. \tag{5}$$

where ζ is the current state. Furthermore, to comply with the velocity limitation and to explicitly consider the limitations of acceleration and braking performance, the constraints for velocity, acceleration, and jerk are represented by

$$\underline{v} \le v_i \le \overline{v}, \tag{6}$$

$$\underline{a} \le a_i \le \overline{a}, \tag{7}$$

$$\underline{j} \le j_i \le \overline{j}. \tag{8}$$

We set the rate limit between the previous control input and the input of the first horizon as follows:

$$\underline{j}_b\delta \le j_0 - j_0^- \le \overline{j}_b\delta, \tag{9}$$

where δ is the step size of prediction, by which we can obtain the jerk input that suppresses sudden changes. To decelerate considering the uncertainty of nearby vehicle movement, the following evaluation function is used:

$$J_{\text{jerk}} = \sum_{i=0}^{H} e_i^{\mathrm{T}} Q e_i + \sum_{i=0}^{H-1} \left\{ j_i^{\mathrm{T}} R j_i + \sum_{n=1}^{N} Q_{\text{o}i,n} m_{i,n}(s_i, z_i, v_i) \right\}, \tag{10}$$

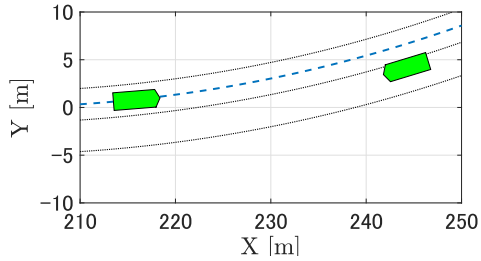


Fig. 3. Simulation scenario: a vehicle near the center line approaches from the opposite direction.

where the error between the actual velocity and the desired velocity is $e_i = v_i - v_r$ and Q, R, and $Q_{o_{i,n}}$ are weights. $m_{i,n}(\cdot)$ is the expected relative velocity between the vehicle and the n-th nearby vehicle on the i-th horizon, which is calculated by integrating the product of relative velocity $v_i - V_{xi,n}$ and probability density function $p(V_{xi,n}|s_i, z_i)$. z_i is the deviation of the vehicle from the target path. Because the vehicle is less likely to collide with other vehicles after passing nearby cars, we set the expectation to zero as follows:

$$m_{i,n}(s_i, z_i, v_i) = \begin{cases} \displaystyle\int_{-\infty}^{+\infty} (v_i - V_{xi,n})\, p(V_{xi,n}|s_i, z_i) \mathrm{d}V_{xi,n}, & (s_i < X_{i,n}) \\ 0. & (s_i \geq X_{i,n}) \end{cases} \tag{11}$$

By assuming that the system follows a Gaussian process, we set $p(V_{xi,n}|s_i, z_i) \sim \mathcal{N}(x_{oi,n}, P^-_{i|k})$, where the covariance matrix is obtained using the following prediction step of the Kalman filter:

$$P^-_{i|k} = A_\mathrm{o} P^-_{i-1|k} A_\mathrm{o}^\mathrm{T} + B_\mathrm{o} \Sigma_n B_\mathrm{o}^\mathrm{T} \; (i = 1, 2, \cdots, H), \tag{12}$$

where A_o and B_o are the system matrices for a nearby vehicle. We assume the model for nearby vehicles to be a double integrator for the position described in Cartesian coordinates with respect to the position and velocity driven by Gaussian noise with the covariance Σ_n imposed on the acceleration channel. The predicted state is calculated by the prediction step of the Kalman filter. To reflect the uncertainty of motion and observation of obstacles, the optimization problem is represented as follows:

$$\begin{aligned} \text{minimize} \quad & J_\mathrm{jerk}, \\ \text{with respect to} \quad & j_i, \\ \text{subject to} \quad & (1),\ (5),\ (6),\ (7),\ (8),\ \text{and}\ (9). \end{aligned}$$

For the vehicle, there is modeling error, such as that caused by actuator delay, which deteriorates performance. To address this issue, the optimal state obtained in the optimization problem is used as the target state of velocity and acceleration, which is tracked by the actual vehicle controlled by a feedback controller.

5. SIMULATION

5.1 Simulation conditions

In this section, we show the simulation results for velocity control and verify the effectiveness of the proposed method. We assume the situation depicted in Fig. 3, where a vehicle near the center line approaches from the opposite direction. In this simulation, the ego vehicle should decelerate according to the collision risk when passing the oncoming vehicle. The target path is a curved road with a curvature radius of 150 m. The ego vehicle and the oncoming vehicle, whose widths are both 1.85 m, move at 20 km/h in the center of their respective lanes. The oncoming vehicle is 2.35 m from the target path . In this simulation, the lateral gap between the sides of the ego vehicle and the nearby vehicle is 0.5 m. In addition, the ego-vehicle follows the target path using proportional steering control with respect to the lateral deviation and angular error. In the simulation, the predicted state of the oncoming vehicle is transformed from the inertial coordinate system to the coordinate system along the path (Okajima and Asai, 2004). The parameters for the numerical simulation are shown in Table 1. The vehicle parameters are shown in Table 2. The observation noise Σ_n used for the Kalman filter is $\Sigma_n = \mathrm{diag}(0.0752^2, 0.0752^2, 0.1497^2, 0.1497^2)$, and the system noise Σ_w is $\Sigma_w = \mathrm{diag}(7.848^2, 7.848^2)$. The observation noise and the system noise were estimated by analyzing the measurement data obtained from the experimental system.

Table 1. Simulation parameters

Parameter	Value	Parameter	Value
δ	0.1	Δ	0.1
H	30	v_r	5.556
$Q_{o_{i,n}}$	$910 - q_o i$	q_o	0.333
Q	1.0	R	1.0
$\overline{v}$	11.111	$\underline{v}$	1.389
$\overline{a}$	1.962	$\underline{a}$	-1.962
$\overline{j}$	2.0	$\underline{j}$	-2.0
$\overline{j}_b$	5.0	$\underline{j}_b$	-5.0

Table 2. Vehicle parameters

Parameter	Value	Unit	Parameter	Value	Unit
m	1500	kg	I	3429	kg m^2
d	10.0	N/(m/s)	I_w	2.0	kg m^2
r	0.35	m	K_m	500	(Nm)/(m/s^2)
Ks	3394	N/-			

5.2 Simulation results and discussion

Figure 7 shows a bird's-eye view at several moments of the simulation. The green rectangle is the ego vehicle. The blue asterisks and dotted circles are the predicted position and error covariance, respectively. The position and length of the orange arrow is the target state on the prediction horizon. Figure 8 depicts the simulation results of velocity, acceleration, and jerk. The red line represents the vehicle model and optimal input and the blue line represents the state calculated by the plant model in Eqs. (2) (3) and (4). In Fig. 8(a), the right axis indicates the relative distance from the nearby vehicle. When it is zero, the ego vehicle has passed the oncoming vehicle. Furthermore, the dotted line in each graph represents the constraints of the controller.

Figure 7 shows that the ego vehicle decelerated as it approached the oncoming vehicle along the curved road. Thus, deceleration control was realized in the path coordinate system. Figure 8(a) shows that the vehicle deceler-

Fig. 4. Experimental vehicle equipped with high-precision GNSS, LiDAR, radar, and cameras.

ated at around $t = 5\,\mathrm{s}$. Thus, we realized the deceleration of an automobile considering the risk of collision. Because the vehicle dynamics are considered in constraints, velocity and acceleration changed smoothly, as shown in Fig. 8(b). Figure 8(c) confirms that excessive jerk input was suppressed by the constraints. In addition, Fig. 7 depicts that the optimal velocity decreased ahead of the prediction horizon when the ego vehicle approaches the oncoming vehicle, and that the optimal velocity began to increase when the ego vehicle passed the oncoming vehicle. Therefore, the vehicle was decelerated considering the future state and uncertainty of the nearby vehicle.

6. EXPERIMENT

6.1 Experimental system

In this section, we show the results of an experiment conducted using an experimental vehicle to verify the effectiveness of the proposed method. In the simulation, a controller with jerk input was used; however, the calculation time is limited for a real-time implementation, and thus it is necessary to reduce the computational load of the controller. We built a controller with velocity input for the experiment based on the proposed method. We took the optimization variable to be v_i and set the controller model to be a single integrator with velocity input $s_{k+1} = s_k + v\Delta$, instead of Eq. (1). The constraints are the same as those for Eqs. (5),(6), but Eqs. (7),(8),(9) are removed. We added the rate constraint for the velocity $\underline{S}_x\Delta \leq v_i - v_{i-1} \leq \overline{S}_x\Delta$, and the rate constraint between the current input and the input of the previous control cycle $\underline{S}_v\delta \leq v_i - v_i^- \leq \overline{S}_v\delta$. The evaluation function is Eq. (10) without the jerk term j_i. The experimental vehicle tracks the target velocity and acceleration; however, we need to calculate the target acceleration because only optimal velocity can be determined. The target acceleration is calculated by numerically differentiating the target velocity and calculating the moving average in the past four control cycles.

In the experiment, we used the experimental vehicle shown in Fig. 4. The position of the vehicle was measured using high-precision GNSS with a measurement error of several centimeters. The velocity and acceleration of the vehicle were measured using a vehicle speed sensor and an acceleration sensor, respectively. The position of the oncoming vehicle was measured by radar, LiDAR, and a camera mounted on the vehicle. Estimation and control were performed every 100 ms using an onboard computer. We used a sequential quadratic programming algorithm for optimization. The code of this solver was generated using the MATLAB function fmincon and a C code generator.

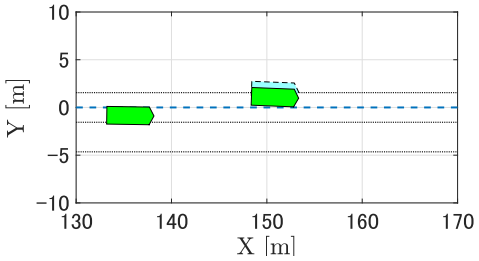


Fig. 5. Experimental situation 1: ego vehicle approaches parked vehicle near the ego lane.

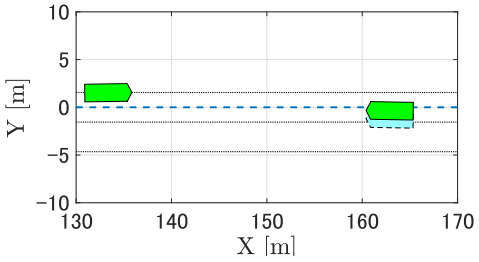


Fig. 6. Experimental situation 2: a vehicle near the center line approaches from the opposite direction.

Table 3. Experimental parameters

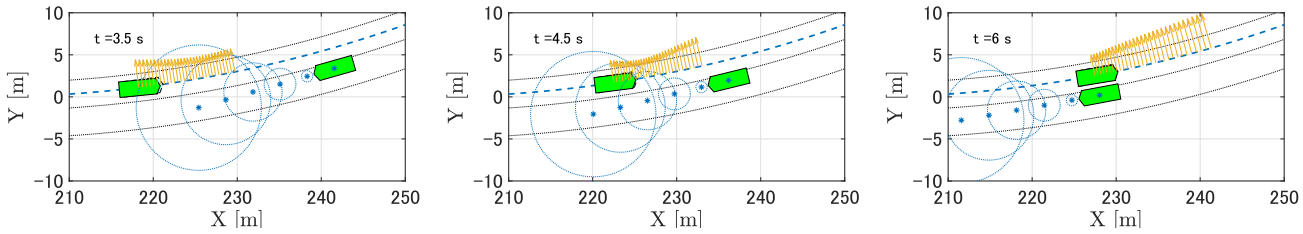
Parameter	Value	Parameter	Value
δ	0.1	Δ	0.1
H	30	v_r	5.556
$Q_{oi,n}$	$310 - q_o i$	q_o	0.333
R	0.8	O_y	0.85
$\overline{v}$	11.111	$\underline{v}$	0
$\overline{S_v}$	1.177	$\underline{S_v}$	-1.472
$\overline{S_x}$	2.943	$\underline{S_x}$	-2.943

6.2 Experimental conditions

To verify that the vehicle decelerated according to the collision risk, we considered two cases, namely passing a parked vehicle (Situation 1) depicted in Fig. 5, and passing an oncoming vehicle (Situation 2) depicted in Fig. 6, respectively. The target path was a straight road. In both situations, the vehicle moved along the center of the target path at a speed of 20 km/h. In Situation 1, another vehicle was parked on the side of the road 2.35 m to the left of the target path. In Situation 2, another vehicle approached from the opposite direction 2.35 m to the right of the target path at a speed of 20 km/h. The distance between the sides of the vehicles was 0.5 m in both situations. In the experiment, for safety, an additional virtual offset O_y was added to the estimated position of the nearby vehicle. In other words, the experiment was conducted assuming that the vehicle was closer than the actual nearby vehicle position. The experimental parameters are shown in Table 3. The observation noise and system noise used for estimation are the same as those used in the simulation.

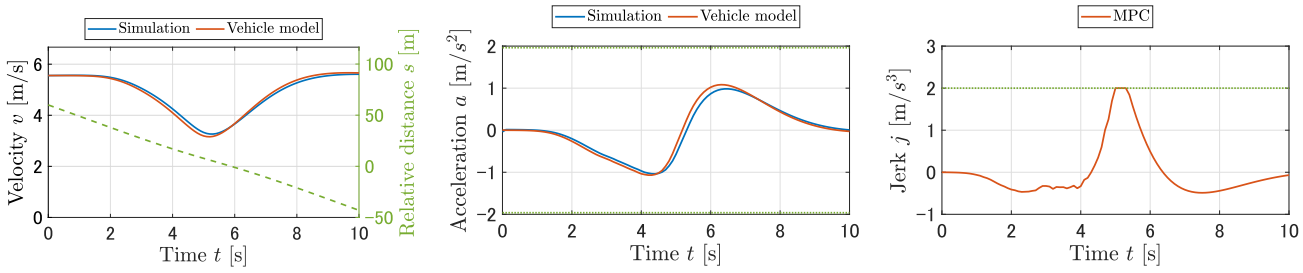
6.3 Experiment results and discussion

Figure 9 shows a bird's-eye view of several moments in Situation 2. The predicted positions of the nearby vehicle are displayed with offset O_y in the left-right direction. The original position of the nearby vehicle is indicated by a light blue broken line. Figures 10 and 11 show the



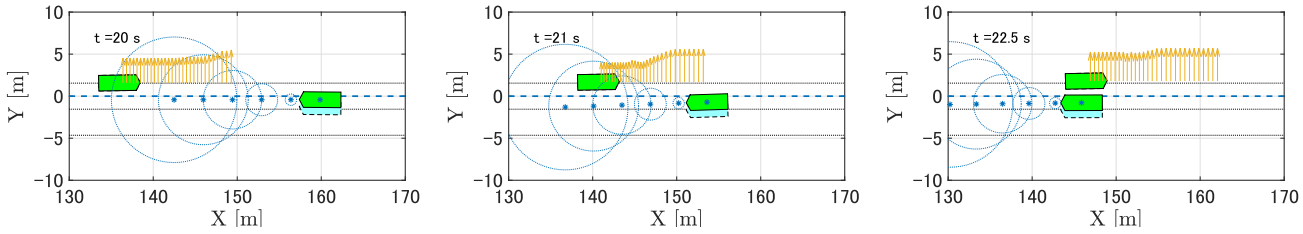
(a) Approaching oncoming vehicle (b) Near deceleration peak (c) Passing the oncoming vehicle

Fig. 7. Bird's-eye view of simulation results with jerk input model.



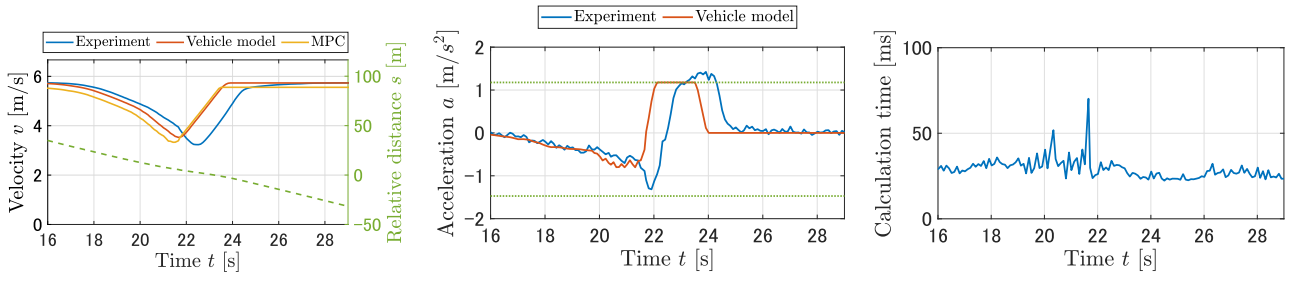
(a) Velocity (b) Acceleration (c) Jerk

Fig. 8. Simulation results with jerk input model.



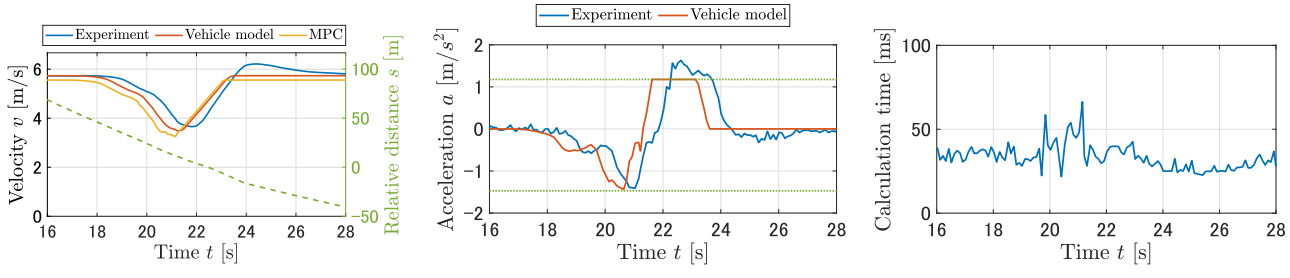
(a) Approaching oncoming vehicle (b) Near deceleration peak (c) Passing oncoming vehicle

Fig. 9. Bird's-eye view of experimental results with oncoming vehicle (Situation 2).



(a) Velocity (b) Acceleration (c) Calculation time

Fig. 10. Experimental results with parked vehicle (Situation 1).



(a) Velocity (b) Acceleration (c) Calculation time

Fig. 11. Experimental results with oncoming vehicle (Situation 2).

experimental results for the velocity, acceleration, and calculation time in Situation 1 and Situation 2, respectively. The red, blue, and yellow lines respectively represent the vehicle model, actual vehicle state, and optimal input. In Figs. 10(a) and 11(a), the right axis shows the relative distance with respect to the nearby vehicle. The blue line represents the actual vehicle state and the yellow line represents the optimal input. The experimental results for Situation 2 are similar to those in the simulation although the tracking delay is larger in the experiment.

The experimental results show that the vehicle was able to pass along the side of the nearby vehicle while decelerating. A target velocity similar to that in the simulation was generated and tracked by the real vehicle. The generated acceleration satisfied the constraints. Furthermore, Figs. 10(c) and 11(c) show that the calculation time was within the range of a control cycle (100 ms), indicating that real-time control was achieved. The above results indicate that the proposed method achieved deceleration control despite limited computation time and the presence of sensor noise. The bird's-eye view results show that the vehicle was about 1 m away from the target path. This occurred because the reference coordinate system was shifted due to GNSS errors. There was a delay of about 1 s in the response of the actual vehicle velocity with respect to the target velocity. In addition, the minimum velocity of the actual vehicle was about 0.5 m/s faster than the minimum target velocity. The delay and error result from the inertia of the actual vehicle and the tracking controller used in the actual vehicle. Although they were not taken into account in the model, the experimental vehicle successfully decelerated, which indicates the robustness of the proposed method.

7. CONCLUSION

In this paper, a velocity controller that reduces the risk of collision considering the uncertainty of nearby vehicle motion was proposed. We quantified the risk of collision using the expectation of relative velocity with respect to nearby vehicles. Then, we realized velocity control considering collision risk as model predictive control that calculates the expectation of relative velocity. In addition, a velocity controller using a jerk input model was built. The effectiveness of the proposed method was verified by a numerical simulation and an experiment with an actual vehicle. Real-time velocity control was achieved in the experiment. In future work, we will conduct experiments based on a controller model with a jerk input model on a curved road with steering control. We will also consider various kinds of transportation participants, including pedestrians and bicycles, with different motion models.

REFERENCES

Akagi, Y. and Raksincharoensak, P. (2015a). An analysis of an elderly driver behaviour in urban intersections based on a risk potential model. In *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, 001627–001632.

Akagi, Y. and Raksincharoensak, P. (2015b). Stochastic driver speed control behavior modeling in urban intersections using risk potential-based motion planning framework. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, 368–373.

Hasegawa, T., Raksincharoensak, P., Yamasaki, A., Mouri, H., and Nagai, M. (2015). Study on autonomous driving intelligence system by using optimal control considering risk potential. *Transactions of Society of Automotive Engineers of Japan*, 46(2), 497–502.

Khatib, O. (1990). *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*, 396–404. Springer New York, New York, NY.

Li, S., Li, K., Rajamani, R., and Wang, J. (2011). Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology*, 19(3), 556–566.

Mizushima, Y., Okawa, I., and Nonaka, K. (2019). Model predictive control for autonomous vehicles with speed profile shaping. *10th IFAC Symposium on Intelligent Autonomous Vehicles IAV 2019*, 52(8), 31 – 36.

Moser, D., del Re, L., and Jones, S. (2017). A risk constrained control approach for adaptive cruise control. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 578–583.

Okajima, H. and Asai, T. (2004). Path-following control based on trajectory differences. *10th IFAC/IFORS/IMACS/IFIP Symposium on Large Scale Systems 2004: Theory and Applications, Osaka, Japan, 26-28 July, 2004*, 37(11), 701 – 706.

Okawa, I. and Nonaka, K. (2018). Comparative evaluation of a trajectory generator for obstacle avoidance guaranteeing computational upper cost. *International Journal of Automotive Engineering*, 9(2), 39–47. doi: 10.20485/jsaeijae.9.2_39.

Park, M.G., Jeon, J.H., and Lee, M.C. (2001). Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing. In *2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No.01TH8570)*, volume 3, 1530–1535.

Shibata, K., Nonaka, K., and Sekiguchi, K. (2019). Model predictive obstacle avoidance control suppressing expectation of relative velocity against obstacles. *2019 IEEE Conference on Control Technology and Applications (CCTA)*.

Shibata, K., Shibata, N., Nonaka, K., and Sekiguchi, K. (2018). Model predictive obstacle avoidance control for vehicles with automatic velocity suppression using artificial potential field. *6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018*, 51(20), 313 – 318.

Shimizu, T., Raksincharoensak, P., and Okuwa, M. (2016). Potential risk quantification based on collision velocity resulted from activation of autonomous emergency braking system. *Transactions of Society of Automotive Engineers of Japan*, 47(6), 1411–1416.

Yoon, Y., Shin, J., Kim, H.J., Park, Y., and Sastry, S. (2009). Model-predictive active steering and obstacle avoidance for autonomous ground vehicles. *Control Engineering Practice*, 17(7), 741 – 750.