Model Predictive Steering Control with Flexible Lane Crossing Considering Nearby Vehicles

Ryuichi Maki 1)  Isao Okawa 2)  Kenichiro Nonaka 1)

1) Tokyo City University  1-28-1 Tamazutsumi, Setagaya, Tokyo 158-8557, Japan
2) DENSO CORPORATION, W building 17F, 1-8-15 Konan, Minato, Tokyo 108-0075, Japan

Received on March 29, 2020

ABSTRACT: Various lane-keeping assist systems have been studied. To avoid collisions, automated lane crossing is sometimes required. In this paper, we propose model predictive steering control with flexible lane crossing for obstacle avoidance. In the proposed method, collision avoidance is guaranteed by hard constraints, and inappropriate lane boundary crossing is suppressed by soft constraints. Experiments show that the proposed controller generates the optimal trajectory for autonomously crossing a lane boundary while suppressing lane deviation and preventing collisions.

KEY WORDS: electronics and control, autonomous driving, control system / Model Predictive Control, Steering control, Path planning [E1]

1. Introduction

Automated driving systems have been extensively studied to reduce the number of traffic accidents caused by human error. Part of this technology, namely lane-keeping assist systems (LKASs), is already in practical use (1). Previous research on LKASs includes that on lane recognition using a camera, prevention of unexpected lane deviation, and lane tracking (2-4). For safe driving, avoiding collisions with traffic participants (i.e., pedestrians and other vehicles) must be taken into account. Automated lane crossing is sometimes required for this task.

The obstacle avoidance problem has been studied based on model predictive control (MPC). MPC determines the optimal input by minimizing the evaluation function under the given constraints while predicting the future behavior of the ego vehicle using a model. Obstacle avoidance with MPC that utilizes a potential field has been proposed (5-7). However, in general, it is necessary to heuristically adjust a weight parameter to improve the response. MPC-based collision avoidance that utilizes constraints for obstacle avoidance has been proposed to address this problem (8,9). However, these studies adopted nonlinear MPC, making real-time computation difficult.

A previous study linearized nonholonomic systems by utilizing a time-state control form, allowing linear MPC that imposes upper and lower bound constraints to be applied (10). This generates a trajectory for obstacle avoidance in real-time. However, this study considered only hard constraints for collision avoidance. Thus, if the ego vehicle cannot avoid a collision while staying within its lane, obstacle avoidance cannot be realized. In contrast, soft constraints (11) allow the constraints to be relaxed if necessary. With such constraints, lane crossing can be used for collision avoidance.

In this study, we propose a steering control method that allows active lane crossing using MPC with soft constraints. In the proposed method, hard constraints are imposed to prevent collisions and soft constraints are imposed to keep the vehicle in the lane as much as possible. The two scenarios depicted in Fig. 1 are used to investigate whether the proposed method realizes automated lane crossing. In Fig. 1(a), the ego vehicle avoids a collision while staying within its lane. In Fig. 1(b), the ego vehicle crosses the lane boundary to avoid a collision. The proposed method allows active crossing of the lane boundary if the ego vehicle cannot avoid a collision while staying within its lane. The effectiveness of the proposed method was verified through experiments, in which both measurement error and modeling error were considered.

![Fig. 1 Two avoidance scenarios.](image-url)
2. Vehicle Model

Figure 2 depicts the vehicle model for kinematic motion. The reference path is the center of the lane. s is the travel distance along the path, θ is the heading angle with respect to the X-axis, and κ is the instantaneous curvature of the travel path. O is defined as the closest point from the vehicle’s center of gravity (CoG) to the path. z is defined as the distance between the vehicle’s CoG and O. \( \psi \) is the angle between the vehicle heading and lane direction, i.e., \( \psi := \theta - \theta_r \). The vehicle state \( x \) and input \( u \) are defined as

\[
x = \begin{bmatrix} z \\ \kappa - \kappa_r \end{bmatrix},
\]

\[
u = \frac{d}{dt}(\kappa - \kappa_r).
\]

After proper linearization, the state equation is described as a triple integrator system. Its discretized system with the distance \( \Delta s \) is

\[
x[k+1] = Ax[k] + Bu[k],
\]

\[
A := \begin{bmatrix} 1 & \Delta s & \frac{\Delta s^2}{2} \\ 0 & 1 & \frac{\Delta s}{2} \\ 0 & 0 & 1 \end{bmatrix}, \quad B := \frac{\Delta s}{2}.
\]

We can transform (2) into the steering angle of the plant model. The curvature \( \kappa \) is calculated by integrating \( u \) with respect to the travel distance \( s \). Then, the steering angle is calculated from \( \kappa \) by applying Ackermann steering angle geometry.

3. Model Predictive Steering Control

In this section, we describe the proposed steering controller based on MPC that minimizes an evaluation function under a given set of constraints while predicting the future behavior of the vehicle model to prevent collisions.

3.1. Constraints

We define constraints for avoiding collisions. The inequality constraints on state \( x[k] \) are given as

\[
x[k] \leq \bar{x}[k] \leq \underline{x}[k],
\]

where " \( \leq \) " means the element-wise inequality of a vector, and \( \bar{x} \) and \( \underline{x} \) are the upper and lower limits of a state, respectively. To prevent excessive input \( u[k] \) and an abrupt change in the heading angle \( \psi[k] \), the inequality constraints on the input and the heading angle are given as

\[
u[k] \leq u[k] \leq \bar{u}[k],
\]

\[
\underline{S}_q[k] \leq \psi[k] - \psi[k-1] \leq \bar{S}_q[k],
\]

where \( \bar{u} \) and \( \bar{S}_q \) are the respective lower limits and \( \underline{u} \) and \( \underline{S}_q \) are the respective upper limits. In addition, to suppress lane crossing as much as possible, the soft constraints on the lateral distance \( z \) using slack variables are given as

\[
x[k] - \underline{z}_s[k] \leq z[k] \leq z_s[k] + \bar{z}_s[k],
\]

where \( \underline{z}_s \) and \( z_s \) are the lower and upper limits for distance, respectively, and \( \bar{z}_s \) and \( \underline{z}_s \) are the slack variables for the lower and upper limits, respectively. These constraints suppress lane deviation and prevent collisions.

3.2. Evaluation function

MPC determines the optimal input \( u \) by minimizing the evaluation function shown in (8).

\[
J = \sum_{k=0}^{H-1} ((x[k] - x_s[k])^T Q(x[k] - x_s[k]) + R u[k])^2 + ((x[H] - x_s[H])^T P(x[H] - x_s[H])
+ \sum_{k=1}^{n} Q_s[k](\bar{z}_s[k]^2 + \underline{z}_s[k]^2),
\]

where \( H \) is the number of horizons, \( P, Q, R, \) and \( Q_s \) are the constant weights, and \( x_s := [x_t \ \theta_t \ \kappa_t]^T \) is a reference trajectory. The trajectory \( x_s \) is described in Section 4.3. The first term suppresses the deviation from the reference trajectory and the magnitude of the input. The second term is the terminal cost for the deviation from the reference trajectory. The third term evaluates the magnitude of the soft constraints. Figure 3 shows the value for the third term along the road. The steering controller based on MPC is formulated as the following optimization problem:

Minimize

with respect to

subject to

(3) ~ (7).
4. Constraints for Obstacle Avoidance

4.1. Constraints for lane

The constraints of lateral distance are depicted in Fig. 4. The blue dashed line indicates the soft constraints that allow crossing of the lane boundary and the red dashed line indicates the hard constraints that restrict the lateral deviation of the ego vehicle.

4.2. Constraints for obstacles

The constraints for obstacles, expressed by trapezoids, are depicted in Fig. 5, where $d_\text{y}$ is the lateral distance from an obstacle, and $l_\text{y}$ and $l_\text{z}$ are the longitudinal distances for safe avoidance. $v_\text{y}$ and $v_\text{z}$ are the velocities of the ego vehicle and an obstacle, respectively. $x_\text{p}$ is the position of a traffic participant, and $x_1$ and $x_2$ are the coordinates of the boundaries between sections. The constraints are set for three sections. Sections 1 and 3 are used to keep a suitable lateral distance from obstacles. Section 2 is designed based on a time margin to avoid obstacles. First, to prevent excessive input of the lateral velocity, we set the lateral velocities $v_\text{y}$ and $v_\text{z}$ to constant values. Section 2 is designed using distances $l_\text{y}$ and $l_\text{z}$, respectively, given as

\begin{align}
l_\text{y} &= (v_\text{y} - v_\text{z}) t_\text{y}, \\
l_\text{z} &= (v_\text{y} - v_\text{z}) t_\text{z},
\end{align}

where $t_\text{y}$ and $t_\text{z}$ are the respective time margins. The distances $l_\text{y}$ and $l_\text{z}$ are modeled as functions of relative velocity. We set the constraints for obstacle avoidance as depicted in Fig. 6. The red dashed line represents the prohibited zone for collision avoidance and the blue dashed line is used to prevent calculation failure and improve response.

4.3. Integration of constraints

In this section, we integrate the constraints described in the previous sections. For this integration, the most restrictive constraints in Figs. 4 and 6 are selected, as shown in Fig. 7. We generate a reference trajectory $z_\text{r}$ for obstacle avoidance from the integrated constraints, as shown in Fig. 8. The reference trajectory is initially set at the center of the soft constraints shown in Fig. 4. The reference trajectory is set at the center of the hard constraints for obstacle avoidance.

5. Experiment

5.1. Purpose

The effectiveness of the proposed method was verified through real-vehicle experiments, where an ego vehicle attempts to avoid a parked vehicle. The parked vehicle position and velocity measured by sensors were fluctuated, thus we utilized the Kalman filter to estimate the position and velocity. We converted the proposed method to a quadratic programming (QP) formulation to utilize an efficient QP solver. We verified that the ego vehicle could deviate from its lane if necessary using two scenarios. In Scenario 1, the ego vehicle avoided an obstacle while staying mostly within its lane, and in Scenario 2, the ego vehicle had to cross the lane boundary, as shown in Fig. 1. The proposed method generates an optimal trajectory that suppresses lane deviation while preventing collisions with traffic participants.

5.2. Automated driving system

Figure 10 depicts an automated driving system for steering automation. The automated driving system comprises a trajectory generator and a tracking controller. The trajectory generator comprises constraints for lane bounds and obstacles and the integration of these constraints and the MPC. First, the trajectory generator sets the constraints using the position of nearby vehicles and the lane geometry. Next, the trajectory generator sets the constraints used in the MPC and the reference trajectory for obstacle avoidance. Then, the MPC calculates the optimal input using the constraints and the reference trajectory. The optimal trajectory is calculated from the optimal input. Finally, the actual vehicle tracks the optimal state calculated by the MPC.
The lane width $5.3 \text{ m}$ was relaxed beyond the lower limit. The trajectory returns inside the lane at $53.2 \text{ s}$, and then the deviation decreases. These results show the effect of the lower soft constraints. The lateral trajectory was suppressed by the lower slack variable term in (8). The other state is calculated, as depicted in Figs. 11(c) and (d), based on the lateral distance term. Thus, the proposed method generates a trajectory that keeps the vehicle in the lane and suppresses lane deviation.

In Fig. 12(a), the reference trajectory crosses the lane boundary in the period $41.0 - 53.2 \text{ s}$. The ego vehicle, which follows the reference trajectory, crosses the lane boundary in the period $44.7 - 54.7 \text{ s}$, while lane deviation is suppressed. Figure 11(b) shows that the lateral distance of the ego vehicle is relaxed beyond the lower limit. The trajectory returns inside the lane at $53.2 \text{ s}$, and then the deviation decreases. These results show the effect of the lower soft constraints. The lateral trajectory was suppressed by the lower slack variable term in (8). The other state is calculated, as depicted in Figs. 11(c) and (d), based on the lateral distance term. Thus, the proposed method generates a trajectory that keeps the vehicle in the lane and suppresses lane deviation.

In Figs. 11(e) and 12(e), estimation of the obstacle still had error applying Kalman filter. As depicted in Figs. 11(a) and 12(a),
the constraints on the parked vehicle were oscillated slowly due to the estimation error as depicted in Fig. 11(e) and Fig. 12(e), which could not be removed by the Kalman filter. However, the steering angle was oscillatory only when the upper limit was relaxed in Scenario 2, as depicted in Fig. 12(d). By conducting another experiment using an input weight $R = 1$, we confirmed that state $x$ was more oscillatory. The oscillation can be suppressed using a larger input weight $R = 40$, as depicted in Figs. 12(c) and 12(d).

The actual state of the steering angle and the heading angle in each scenario tracks the optimal trajectory with a maximum delay of 1 s. It is considered that the delay came from the vehicle inertia and actuator response as well as the communication time, since the control system is separated into the trajectory generator of the MPC and the tracking controller as depicted in Fig. 10.

7. Conclusion

In this paper, we proposed a method for obstacle avoidance with automated lane crossing that guarantees a safe lateral distance. The controller generates an optimal trajectory that can cross the lane boundary autonomously while suppressing deviation using soft constraints and preventing collisions with traffic participants using hard constraints. The effectiveness of the proposed method was verified through experiments. The proposed method suppresses lane deviation if the ego vehicle is able to avoid collisions while staying within its lane. Moreover, the proposed method allows lane crossing if it is required to avoid a collision. In future work, we will take measurement error into account to increase safety in real environments. We also need to verify the effectiveness of the proposed method at the higher speed.

Acknowledgements

This paper is written based on a proceeding presented at JSAE 2015 Annual Congress.

This research was conducted as a collaboration between Tokyo City University and DENSO CORPORATION. We are grateful to Mr. Watanabe, Mr. Hori, and Dr. Shimizu for their cooperation in the experiment.
References

(1) M. Aeberhard, S. Rauch, M. Bahram, G. Tanzmeister, J. Thomas, Y. Pilat, F. Homm, W. Huber, and N. Kaempchen : Experience, results and lessons learned from automated driving on Germany’s highways. IEEE Intelligent Transportation Systems Magazine, Vol. 7, No. 1, pp. 42–57, (2015).

(2) D. C. Andrade, F. Bueno, F. R. Franco, R. A. Silva, J. H. Z. Neme, E. Margraf, W. T. Ono, F. A. Farinelli, A. M. Tuset, S. Okida, M. M. D. Santos, A. Ventura, S. Carvalho, and R. d. S. Amaral : A novel strategy for road lane detection and tracking based on a vehicle’s forward monocular camera. IEEE Transactions on Intelligent Transportation Systems, Vol. 20, No. 4, pp. 1497–1507, (2019).

(3) J. Lee, J. Choi, K. Yi, M. Shin, and B. Ko : Lane-keeping assistance control algorithm using differential braking to prevent unintended lane departures. Control Engineering Practice, Vol. 23, pp. 1 – 13, (2014).

(4) H. Okajima, and T. Asai : Path-following control based on trajectory differences. IFAC Proceedings Volumes, Vol. 37, No. 11, pp. 701 – 706, 2004. 10th IFAC/IFORS/IMACS/IFIP Symposium on Large Scale Systems 2004: Theory and Applications, Osaka, Japan, 26-28, (2004).

(5) Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli : A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles. Vehicle System Dynamics, Vol. 52, No. 6, pp. 802–823, (2014).

(6) Y. Raschkipour, I. Fadakar, and A. Khajepour : Autonomous driving motion planning with obstacles prioritization using lexicographic optimization. Control Engineering Practice, Vol. 77, pp. 235 – 246, (2018).

(7) Z. Elmi, and M. O’ Efe : Path planning using model predictive controller based on potential field for autonomous vehicles. In IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society, pp. 2613–2618, (2018).

(8) B. Alrifaee, J. Maczijewski, and D. Abel : Sequential convex programming mpc for dynamic vehicle collision avoidance. In 2017 IEEE Conference on Control Technology and Applications (CCTA), pp. 2202–2207, (2017).

(9) D. Ioan, S. Olaru, I. Prodan, F. Stoican, and S. Niculescu : Parametrized hyperplane arrangements for control design with collision avoidance constraints. In 2019 IEEE 15th International Conference on Control and Automation (ICCA), pp. 1591–1596, (2019).
(10) K. Oyama, K. Sekiguchi, and K. Nonaka: Path-following control for front-steering vehicles based on time-state control form using travel distance as a virtual time-axis. Transactions of the Society of Instrument and Control Engineers, Vol. 50, No. 10, pp. 746–754, (2014). (in Japanese).

(11) I. Okawa, K. Nonaka: Quasi-optimal trajectory generation for vehicle obstacle avoidance with state box-constraint. The 4th Multi-symposium on Control Systems, (2017). (in Japanese).

(12) M. N. Zeilinger, M. Morari, and C. N. Jones: Soft constrained model predictive control with robust stability guarantees. IEEE Transactions on Automatic Control, Vol. 59, No. 5, pp. 1190–1202, (2014).