2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Model Predictive Parking Control for Nonholonomic Vehicles using Time-State Control Form

Kentaro Oyama and Kenichiro Nonaka

Abstract— **In this study, a model predictive control (MPC) for nonholonomic vehicles with steering and travel range constraints is proposed and applied to the parking problem. We transformed the nonlinear dynamics into two linear subsystems using time-state control form (TSCF) where we suppress the computational effort of MPC with reduced order dynamics. Both steering and travel range constraints can be considered using 1st order approximation with enough accuracy. The parking problem for nonholonomic vehicles needs a switching movement where the switching point is automatically optimized. The performance of the proposed method is verified through actual vehicle experiments on the parking problem in narrow garage.**

I. Introduction

Automatic control of vehicles has been studied intensively in these years, in order to achieve high efficiency as well as advanced safety. When front steering vehicles run at low speed, so that the side slip angles of tires are negligible, they can be regarded as a kind of nonholonomic system, which cannot be stabilized by continuous control inputs and thus requires discontinuous input, or discontinuous coordinates transformation especially to use the techniques of linear feedback control. While there are several control methods for nonholonomic sytems, time-state control form (TSCF) proposed by Sampei et.al. [1] is known as a useful technique for a kind of nonholonomic systems including front steering vehicles. TSCF transforms nonholonomic dynamics described as differential equations with respect to time into the two subsystems where one subsystem is a differential equation of special state, called *time-state*, with respect to the time, and the other is the differential equation with respect to the *time-state* instead of time. Althogh the original system is not a linearizable one, the transformed subsystem becomes a linearizable one; rich and effective design methods for linear system can be applied. In addition, if TSCF is applied to front steering vehicles so that a position with monotonously increasing or decreasing is taken as a time-state, then the path tracking control is naturally realized. But in the past researches, at the best of our knowledge, physical constraints on steering angle or the width of the road have not been considered explicitly.

Model predictive control (MPC) is also a proven method applied to nonlinear systems for decades. MPC provides an optimal control input satisfying various kinds of constraints. The principal difficulty of MPC is that it is prone to require high computational load. In the last decade, significant advance of algorithms of MPC alleviates the computational load; they invites MPC to the application of mechanical systems with small time constants. For example, a sophisticated special custom solver for MPC [2], and an ingenious continuation algorithm along the time axis [3] has been proposed. Effort on simplification for embedded CPU was also reported [4]. For the vehicle control, MPC has been applied to various problems, including path-tracking control [5][6][7], obstacle avoidance control [8][9], optimal velocity control in path-tracking control [10]. In most cases, MPC is directly applied to nonlinear dynamics; there are concerns that whether the computed solution is optimal under the nonlinear constraints. In addition, the computational load may still large for real-time control in some kind of mechanical systems which requires fast response. To address these issues, for front steering vehicles, we utilized TSCF to divide the original system into reduced order two subsystems and linearized them using coordinates and input transformation [11]; we realized both high-speed computation and optimal control with constraints based on the framework of MPC.

In this paper, for front steering vehicles, we study a parking control with switchback motion. Since it is difficult to determine when and where to conduct the switchback motion, parking control requires skilled driving technique and becomes one of the difficult driving task in narrow road in urban area. There are a number of researches on parking control of vehicles with nonholonomic constraints. Some methods using vehicle model with two wheels is proposed [12][13], but these models are not front steering. For front steering vehicles, there are researches on garage parking control [14][15][16] and parallel parking control [17][18][19]. Motion planning based methods are the classical but promising method to realize parking control which has been evolutionally developed and advanced. However, in these methods, a constraint on steering angle of the vehicle has not been considered explicitly. Recently, a method based on navigation field is applied to parking assistant systems [20]. In addition, to cope with the variation of the environments and uncertainty of dynamics, autonomous feedback control methods [21][22] are proposed. In these studies, the critical problem of parking control is to determine the point of switchback motion, but in [21][22], the switchback point is predetermined; The vehicle is navigated to the point then a feedback regulator is applied to render the vehicle converge into the garage. To determine the switchback point automatically, in this study, we propose a model predictive control including switchback motion for vehicles. First, we apply TSCF to transform the nonlinear dynamics into two

K. Oyama and K. Nonaka are with the Department of Mechanical Systems Engineering, Tokyo City University, 1-28-1, Tamazutsumi, Setagaya, Tokyo, 158–8557, Japan. knonoaka@tcu.ac.jp

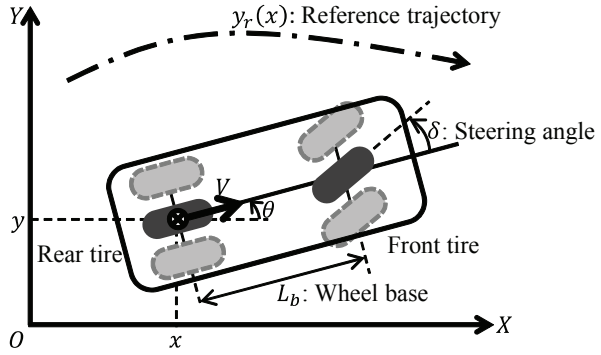

Fig. 1. Single track vehicle model

subsystems comprised of a first order differential equation with respect to the *time-state* which increases or decreases monotonously as the time grows, and a vector first order system with respect to the *time-state*, then divide the dynamics into the forward motion (approach) and backward motion (parking). The performance index function is described as a sum of the cost functions of forward and backward motion when the switching point is included in the horizon. Since the switching point explicitly appears in the performance index function, we can optimize it through repetitive computation.

II. Front Steering Vehicles described by Time-State Control Form[1]

Figure 1 depicts the model of a front steering vehicle represented as a single-track model where both left and right wheels are brought together by neglecting the tire force difference of two wheels. Since we consider the parking control, in which the driving speed is low, we assume that the side-slip angles of the front and rear wheels are zero in this paper. The rear wheel position and orientation of the vehicle are represented as (x,y,θ) on $X-Y$ coordinates. δ is the steering angle of the front wheel, V is the velocity at the point of rear wheel, and L_b represents the wheelbase. y_r represents the reference path to be tracked by the point of rear tire of the vehicle. The kinematic equation of the vehicle is represented by the equation (1)(2)(3):

$$\begin{cases} \dfrac{dx}{dt} = V\cos(\theta) & (1) \\ \dfrac{dy}{dt} = V\sin(\theta) & (2) \\ \dfrac{d\theta}{dt} = \dfrac{V}{L_b}\tan(\delta). & (3) \end{cases}$$

In order to realize the path-tracking control of the vehicle, we utilize the time-state control form, in which the dynamics is represented as a differential equation with respect to the state instead of time. We take x as a *time-state* by assuming that the reference path of the vehicle lies along the x-axis. Dividing (2) and (3) by (1), we get the following time-state control form:

$$\begin{cases} \dfrac{dy}{dx} = \tan(\theta) & (4) \\ \dfrac{d\theta}{dx} = \dfrac{1}{L_b\cos(\theta)}\tan(\delta). & (5) \end{cases}$$

We introduce the nonlinear state transformation (6) and input transformation (7) and (8):

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} x \\ dy/dx \\ y \end{bmatrix} = \begin{bmatrix} x \\ \tan(\theta) \\ y \end{bmatrix} \quad (6)$$

$$\mu_1 = V\cos(\theta) \quad (7)$$

$$\mu_2 = \frac{1}{L_b\cos^3(\theta)}\tan(\delta) \quad (8)$$

It should be noted that this transformation is defined in the range $-\pi/2 < \theta < \pi/2$. Applying (6)(7)(8), the nonlinear state space equation (1)(2)(3) is transformed into the two linear subsystems given by

$$\frac{dz_1}{dt} = \mu_1 \quad (9)$$

$$\frac{d}{dz_1}\begin{bmatrix} z_3 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} z_3 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}\mu_2 \quad (10)$$

where (9) is the differential equation of the time-state z_1 with respect to the time t, and (10) is the differential equation of other states with respect to the time-state z_1. The actual steering angle δ is computed using θ and μ_2 by the following equation transformed from (8):

$$\delta = \tan^{-1}\left(L_b\cos^3(\theta)\mu_2\right). \quad (11)$$

Since the linearized time-state control form (10) is a double integrator system with respect to the time-state x, it is straightforward to realize the path-tracking control if the desired path y_r is described as a function of the time-state. In fact, if we apply

$$\mu_2 = \frac{d^2y_r}{dz_1^2} - \kappa_2\left(z_2 - \frac{dy_r}{dz_1}\right) - \kappa_1(z_3 - y_r) \quad (12)$$

with positive κ_1 and κ_2, then $z_3 \to y_r$ as $z_1 \to \infty$, which is achieved for positive μ_1 in (9). For $z_1 \to -\infty$, $\kappa_1 > 0$ and $\kappa_2 < 0$ is used to ensure stability. In addition, (10) is tractable for model predictive control, because its linearity and reduction of order suppress the computational effort of the real-time model predictive control.

III. Model Predictive Control based on TSCF[11]

A. Performance index and constraints

Model predictive control is a kind of feedback control technique in which an optimization problem from the current time till the finite time future is solved at every control step. Since the computational effort tends to be large, it is important to make the calculation simple. In this study, the dynamics of the front steering vehicle with three dimensional state is divided into two reduced order subsystems comprised of the first order dynamics (9) of $z_1 = x$ with respect to time t and the second order dynamics (10) of the position y with

respect to the time-state x. In the parking control in this study, the vehicle usually moves slowly; we assume that the sign of $\dot{x}$ can be controlled arbitrarily by μ_1. To control z_3 and z_2 using μ_2, we apply the model predictive control in order to render the vehicle run through the optimal path while keeping the constraints on the steering limitation and passage width. In what follows, we discuss the MPC to design μ_2 for (10). Since (10) is the continuous and linear dynamics, it can be transformed into the following discretized model:

$$\zeta[k+1] = \begin{bmatrix} 1 & \Delta \\ 0 & 1 \end{bmatrix} \zeta[k] + \begin{bmatrix} \frac{\Delta^2}{2} \\ \Delta \end{bmatrix} \mu_2[k] \tag{13}$$

where Δ is the step with respect to time-state $z_1 = x$. Note that Δ is positive for the forward motion and is negative for the backward motion. The performance index for path tracking control is defined by (14)(15)(16):

$$J_t(\zeta, \mu_2, z_1, H) = \phi(\zeta[H]) + \sum_{k=0}^{H-1} L(\zeta[k], \mu_2[k], k) \tag{14}$$

$$\phi(\zeta[H]) = \tilde{\zeta}[H]^T Q_{fin} \tilde{\zeta}[H] \tag{15}$$

$$L(\zeta[k], \mu_2[k], k) = \tilde{\zeta}[k]^T Q \tilde{\zeta}[k] + R\mu_2[k]^2 \tag{16}$$

where (15) is the terminal cost, (16) is the stage cost, and H is the number of steps for which the performance index is evaluated. It is noted that Δ is distance of movement along x instead of time. $\tilde{\zeta}$ is the deviation from the desired state $\zeta_r := y_r$ defined by (17):

$$\tilde{\zeta} = \zeta - \zeta_r(z_1). \tag{17}$$

In addition, the constraints on passage width is represented by (18), and steering limitation is given by (19):

$$\underline{z}_3(z_1) \leq z_3 \leq \overline{z}_3(z_1) \tag{18}$$

$$\underline{\delta} \leq \delta \leq \overline{\delta} \tag{19}$$

where $\underline{\delta}$ and $\overline{\delta}$ are the lower and upper limitation of the steering angle. Similarly, $\underline{z}_3$ and $\overline{z}_3$ are the lower and upper bound of the passage width which varies according to the geometry of walls.

B. Linearization of constraints

To consider the steering constraint (19) for (11), let us define (20):

$$\Gamma(z_2, \mu_2) = L_b \cos^3\left(\tan^{-1}(z_2)\right) \mu_2. \tag{20}$$

Thus (19) is obtained from (11):

$$\tan(\underline{\delta}) \leq \Gamma(z_2, \mu_2) \leq \tan\left(\overline{\delta}\right). \tag{21}$$

Since (20) is the nonlinear function of z_2, (21) becomes nonlinear constraints. In order to solve the optimization problem through the sequential quadratic programming (SQP) which we use in this paper, we introduce the first order Taylor series expansion of (21) at z_2 and μ_2 for each z_1, as given by (22):

$$\begin{aligned} \hat{\Gamma}(z_2, \mu_2) &:= \Gamma(\hat{z}_2[k], \hat{\mu}_2[k]) \\ &\quad + \frac{\partial \Gamma}{\partial z_2}(z_2 - \hat{z}_2[k]) + \frac{\partial \Gamma}{\partial \mu_2}(\mu_2 - \hat{\mu}_2[k]) \end{aligned} \tag{22}$$

$$\frac{\partial \Gamma}{\partial z_2} = -3L_b \cos^2(\theta[k]) \sin(\theta[k]) \frac{\mu_2}{1 + \hat{z}_2[k]^2}$$

$$\frac{\partial \Gamma}{\partial \mu_2} = L_b \cos^3(\theta[k])$$

$$\theta[k] = \tan^{-1}(\hat{z}_2[k]),$$

where $\hat{\mu}_2[k]$ is input of previous sampling and $\hat{z}_2$ is predicted state using input of previous sampling and initial current state, respectively. The constraint related to the steering angle δ is now approximated by the following affine inequalities:

$$\tan(\underline{\delta}) \leq \hat{\Gamma}(z_2, \mu_2) \leq \tan\left(\overline{\delta}\right). \tag{23}$$

Since the dynamics of z_2 and z_3 represented by (13) is linear, we can apply SQP for model predictive control. We use CVXGEN [23] to generate a code which solves SQP for model predictive control at fast rate which can be implemented into embedded CPUs.

IV. Model Predictive Parking Control

A. Problem statement

In this study, a parking control problem involving back and forth motion is studied based on the application of model predictive control. The procedure of the parking control is comprised of the following three steps:

1) Approaching:
   The car moves forward to approach along the reference path.
2) Determining the switching point:
   The position and orientation are regulated to prepare that the car enters into the garage smoothly after the switching motion.
3) Driving into the garage:
   The car moves backward to go into the parking point with the specified altitude.

When the parking area is restricted, it is important to determine the switching point so that the car was driven into the narrow garage safely. Thus the model predictive control is a promising technique because it can include the future performance of the motion after switching and also the restriction of the range of movement. In this paper, we consider the parking problem of the course depicted in Fig.2. The lines l_1 through l_4 indicate the reference path only used as a guide of locomotion. The car approaches the garage along the reference path l_1. At the point A, the reference trajectory is switched into the curve line l_2 which is smoothly connected to l_1. Due to the limitation of the road width, l_2 curves at the other side of the road and smoothly connected to the line l_3 at the point B, which is parallel to the road. l_4 indicates the center line of the garage which is used at the backward motion to enter the garage. The point $G_p(x_p, y_p)$

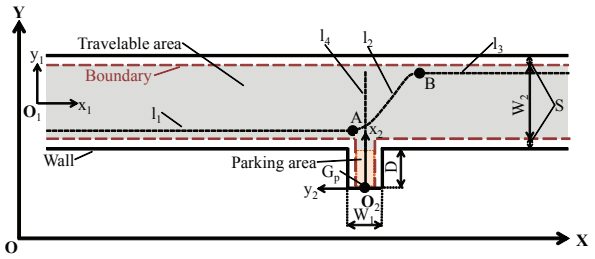

Fig. 2. Course of parking control

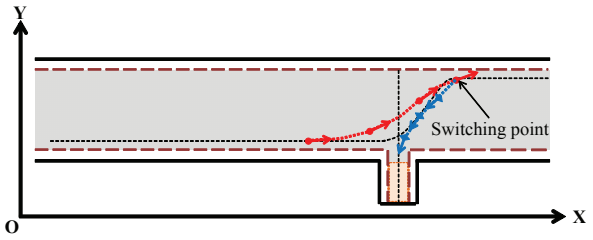

Fig. 3. The image of the predicted states in J_t, $n = 3$

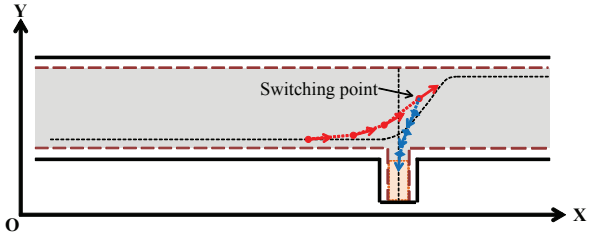

Fig. 4. The image of the predicted states in J_{t+}, $n = 3$

is the goal point of parking motion. It should be noted that the lines l_1 through l_4 are used simply as a guide, the actual locomotion path is modified based on the model predictive control.

B. Algorithm of parking control

In this section, we show the algorithm to determine the switching point automatically by MPC. The index function is described as a sum of the cost functions of forward and backward motion when the switching point is included in the horizon. Evaluating both forward and backward motion can optimize the switching point to realize the automatic switchback motion.

First, the horizon is divided into two sections, forward motion and backward motion. We define n as a number of the sampling points at the backward motion to evaluate the performance index in the discretized dynamics. The performance index (14) is now represented as follows:

$$\begin{aligned} J_t(\zeta,\mu_2,z_1,H) &= \phi(\zeta[H]) + \sum_{k=0}^{H-n-1} L_+(\zeta[k],\mu_2[k],k) \\ &\quad + \sum_{k=H-n}^{H-1} L_-(\zeta[k],\mu_2[k],k), \end{aligned} \tag{24}$$

where L_+ and L_- correspond to the stage cost (16) for the dynamics (13) with positive Δ (forward movement) and with negative Δ (backward movement), respectively. The coordinates for l_1 through l_3 is O_1 and those for l_4 is O_2. When the vehicle is approaching the garage without the backward movement, then we set $n = 0$. We keep $n = 0$ to prevent the unnecessary computation of the performance index J_t when the vehicle moves along the road till it cuts across in front of the garage, i.e. $x > x_p$, where x_p is the x position of parking. During each control step, we evaluate the performance index J_t of the motion with backward movement for $n \to n+1$, which is defined as J_{t+} hereafter. The backward movement number n is increased by 1 when the performance index with the incremented n, J_{t+}, becomes better than the not incremented one: $J_{t+} < J_t$. The inputs use the computationally smaller result either J_t or J_{t+}, and compute alternately between J_t and J_{t+} every control cycle.

In this study, the backward movement number n is increased by 1 subject to above condition. As as result, proposed method can determine the switching point automatically.

In this study, if the control cycle is rapid and vehicle velocity is slow enough, variation of the state and the index function are small. Thus, it is enough to execute the optimized calculation at one-time only in every control step by computing J_t and J_{t+} alternatively. If $J_t > J_{t+}$ in the turn of calculation of J_t or if $J_t < J_{t+}$ in the turn of calculation of J_{t+}, the input is linearly-interpolated by the computation result of previous control step. The linearly-interpolated input is calculated by the state and input of previous sampling:

$$\mu_2 = \frac{\mu_2[1] - \mu_2[0]}{\Delta}(x - x[0]) + \mu_2[0] \tag{25}$$

where, $\Delta = z_1[1] - z_1[0]$, $z_1[1]$ is the z_1 position of $k = 1$, $z_1[0]$ is the z_1 position of $k = 0$, $\mu_2[0]$ is the input of $k = 0$, and $\mu_2[1]$ is the input of $k = 1$, respectively.

In addition to the change of dynamics at the switching point, we use the local coordinates O_1 in forward movement and use local coordinates O_2 in backward movement to prevent the singularity of the transformation of (6). In our past study [24], we used only local coordinates O_1 while the switching point is included in the horizon, however, it had a problem that the value of performance index varies drastically for the different backward step number n, because different dynamics and coordinates O_2 is evaluated during the backward movement. Thus it is preferred to avoid such discontinuous change of the performance index from the point of view of the numerical optimization. To resolve this issue, we use the same number of forward and backward steps to compare the performance index J_{t+} with J_t, by stretching the each distance of the forward steps and shrinking that of the backward steps to avoid discontinuous change of the value of performance index in Fig. 3 and Fig. 4. The red and blue arrows depict the positions during forward movement and backward movement respectively in which their directions mean the direction of movement.

C. Index of vehicle position to parking area[25]

Since the proposed method uses z_1-axis instead of time-axis, the stage cost L_+ or L_- reflects the deviation from the reference path, but it is not sensitive to the *time-state* position z_1 of the vehicles. Thus the above algorithm has a problem in the case that the vehicle moves forward even if vehicle passed through the garage and leaves far away from parking area. To reinforce the attraction to the garage, we introduce the criteria to evaluate the distance between the parking area and z_1 position of each predicted states. The index of the vehicle position to the parking area is given by

$$J_p = \sum_{k=0}^{H} (z_1[k] - x_p)^T Q_p (z_1[k] - x_p), \quad (26)$$

where Q_p is the positive scalar representing the weight for each position.

D. Configurations of index function for parking control

In this paper, we use the index function of MPC for parking control comprised of J_t and J_p, where J_t works to evaluate the tracking performance and J_p to attract the vehicle position to the parking area: The index function J is given by (27):

$$J = J_t + J_p. \quad (27)$$

We solve the optimization problem with the performance index (27) under the constraints (13), (18), and (23).

V. Experimental result

A. Experimental system and condition

In the experiment of this paper, we use a 1/10 scale model car called RoboCar manufactured by ZMP Inc. which is equipped with laser range finder (LRF) available for localization and detection of obstacles. Figure 5 depicts the photograph of the experimental vehicle, and Table I shows its specification. RoboCar is the front wheel steering and rear-wheel drive vehicle; the velocity of motion is measured from the rotary encoders attached to the four wheels. The limitation of the front wheel steering is $(\delta, \underline{\delta}) = (30, -30)$ degrees; its wheelbase is $L_b = 0.256$ m. In this paper, the position and orientation of the vehicle are measured from the data of LRF, in which the sampling rate is 28 ms and resolution of the position is about 1 mm, and the angle is 0.36 degrees. The vehicle is equipped with the processor whose operation frequency is only 500 MHz. The processor is used to compute the proposed model predictive control and localization from the LRF data. The block diagram of the experimental system is depicted in Fig. 6. The position x, y and orientation θ are measured from the LRF data [26] and the velocity data from the four rotary encoders by fusing together using the extended Kalman filter (EKF) to estimate the $\hat{x}$, $\hat{y}$ and $\hat{\theta}$ [27]. MPC computes the velocity and wheel steering angle so that the optimal vehicle control is achieved. Table II shows the course parameters of the parking experiment, and Table III shows the experimental condition.

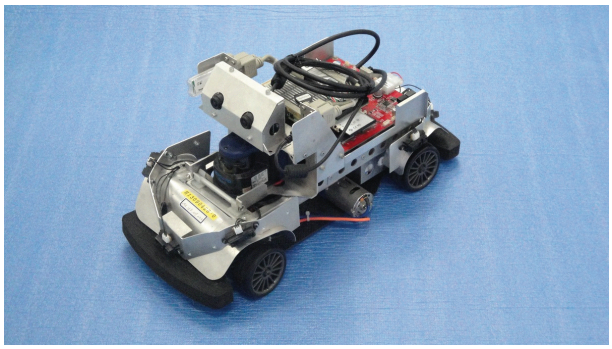

Fig. 5. Experimental vehicle (RoboCar®1/10 /ZMP RC-Z)

TABLE I
SPECIFICATION OF ROBOCAR®

Name/Number	RoboCar®1/10 /ZMP RC-Z
Size	429.0 × 195.0 × 212.2 mm
Weight	3 kg
Main controller	CPU:AMD Geode®LX800 Processor 500MHz OS:Linux(Fedora10)
Internal sensor	Rotary encoder (Wheel×4, DC motor×1)
External sensor	Laser range finder

TABLE II
SETTINGS OF PARKING CONTROL

	Parameters	Value
Local coordinate	$\mathbf{O_1}$	$(0.0, 3.0, 0.0)$
Local coordinate	$\mathbf{O_2}$	$(4.0, 2.0, \pi/2)$
Garage depth	D	0.5 m
Garage width	W_1	0.4 m
Passage width	W_2	1.0 m
Safety distance	S	0.1 m
1st reference trajectory	l_1	$Y = 2.6$
2nd reference trajectory	l_2	$Y = 1.2x^5 - 24.9x^4 + 211.9x^3 - 890.3x^2 + 1851.7x - 1526.1$
3rd reference trajectory	l_3	$Y = 3.4$
4th reference trajectory	l_4	$X = 4.0$
Intersection point	A	$(3.8, 2.6)$
Intersection point	B	$(5.0, 3.4)$
Goal point	$G_p(x_p, y_p)$	$(4.0, 2.0)$

TABLE III
EXPERIMENTAL CONDITION

	Parameters	Value
Initial state	(x, y, θ)	$(1.0, 3.0, 0.0)$
Vehicle Velocity	V	200 mm/s
Step width	Δ	0.2 m
Horizon	H	6
Weighting matrix	Q	diag(5.5, 1.0)
Weighting matrix	R	1.0×10^{-3}
Weighting matrix	Q_{fin}	diag(5.5, 1.0)
Weighting matrix	Q_p	5.0
Feedback gain (Forward)	(κ_1, κ_2)	(4.0, 4.0)
Feedback gain (Backward)	(κ_1, κ_2)	$(4.0, -4.0)$

B. Experimental result

In this section, we show the results of experiment comparing the proposed model predictive control method with TSCF

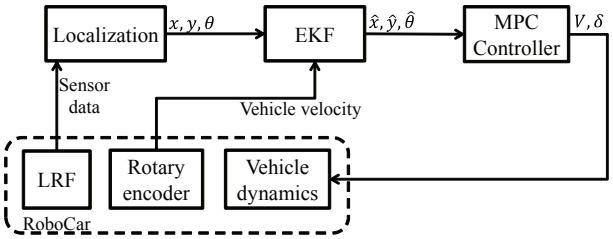

Fig. 6. System block diagram

and a simple tracking feedback PI control with TSCF which uses the linear controller (12) with $\kappa_1 > 0$ and $\kappa_2 > 0$ at the forward movement and $\kappa_1 > 0$ and $\kappa_2 < 0$ at the backward movement, respectively.

Figure 7 depicts the calculation time used for optimization of MPC, where the computation is done within the control cycle 10 ms, which indicates that the proposed real time model predictive control is feasible even on the low speed CPU. Figure 8 depicts the value of the index function J and J_+ and backward step number n. It indicates that the nunber of n is increased by 1 from 0 to H, which shows that switching point is determined automatically. Figure 9 and Fig. 12 depict the path of proposed MPC and TSCF, and Fig. 11 and Fig.14 depict their front wheel steering angle. The proposed method satisfies both the steering angle and the travel range constraints as depicted in Fig. 9 and Fig. 11, respectively. On the other hand, TSCF cannot satisfy the travel range constraint during the backward movement as depicted in Fig. 12. Additionally, the steering angle of TSCF hits the limitation of front steering angle when the reference trajectory changes l_1 to l_2. Since it is not straightforward to include the constraints like steering angle or travel range in TSCF, the saturation of the input due to the limitation of front wheel steering appears and the path tracking performance of TSCF was deteriorated. Figure 10 and Fig. 13 depict the time response of position and heading angle in MPC and TSCF. At $t = 30$ s, the states of the proposed method converged to the reference states, but the states of TSCF have the error between the states and reference states. Furthermore, the switching point of TSCF is the pre-designed parameter: if the switching point is changed, then the trajectory is varied. Figure 15 depicts the path of the proposed MPC and paths of TSCF in which switching point is changed to $x = 4.3$ m, $x = 4.5$ m, and $x = 4.8$ m. It indicates that the switching point is important parameter for TSCF to realize safety locomotion into the garage; but it is not trivial to optimize the switching point under the constraints. Thus the experiments show the advantage of the proposed method. Figure 16 and Fig. 17 depict the successful area for parking control in this study and our past study [24]. The successful area of parking control in this study is larger than the our past study, because the horizon length is adaptively adjusted to realize the smooth comparison of switching point.

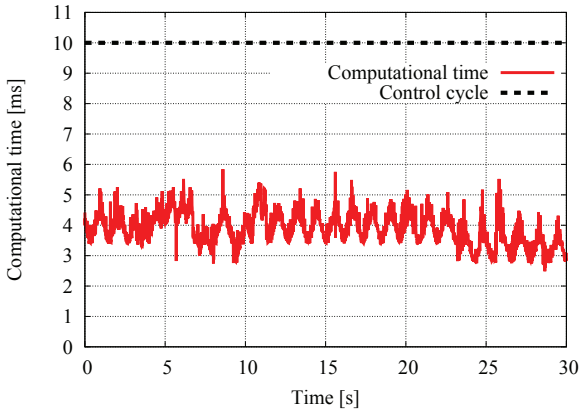

Fig. 7. Computational time (Proposed method)

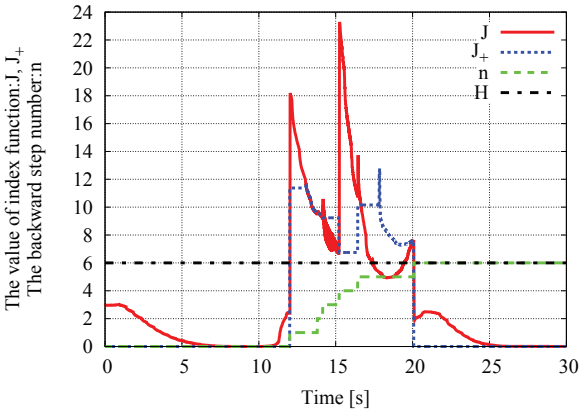

Fig. 8. The value of index function (Proposed method)

VI. Conclusion

In this paper, a model predictive parking control based on the time-state control form is proposed. MPC automatically computes the steering angle and determines the switching point so that the optimal path without collision with the wall is achieved. The experimental results show that the safety driving is achieved while keeping the range of movement and limitation of the steering angle. In addition, it shows that the real-time control is achieved at the computational time less than 10 ms.

In this paper, only single switching is considered in the proposed method by assuming that the backward step number n is not decreasing. For the future work, it is interesting to consider the movement involving the multiple switching motions. In addition, we are planning to compare the computational effort of proposed method with the nonlinear model predictive control.

Acknowledgments

The authors gratefully acknowledge the support of Grantin-Aid for Scientific Research (C) No. 24560555 of Japan. In this research, we gratefully acknowledge and thank Prof. Katsumasa Suzuki and Assistant Prof. Kazuma Sekiguchi of Tokyo City University for their advise.

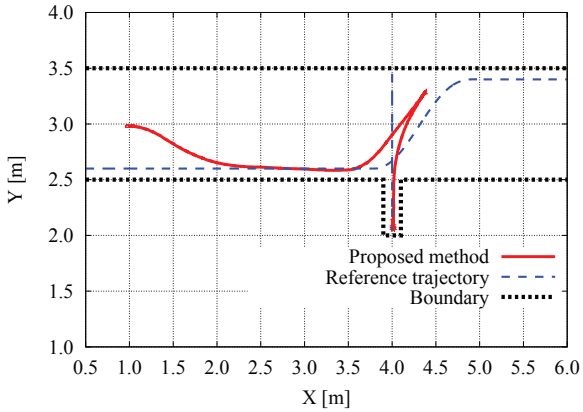

Fig. 9. Trajectory (Proposed method)

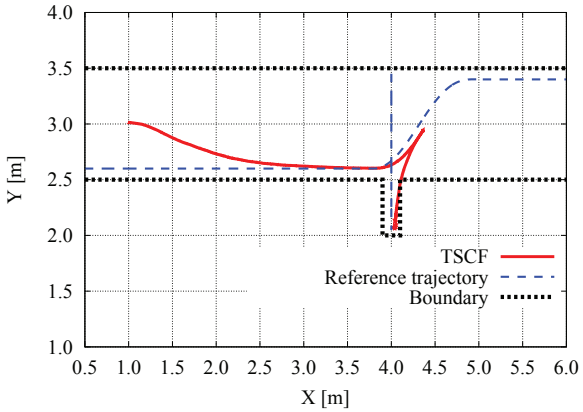

Fig. 12. Trajectory (TSCF)

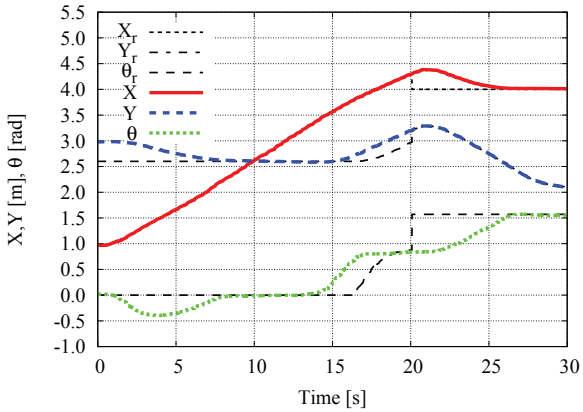

Fig. 10. Position and heading angle (Proposed method)

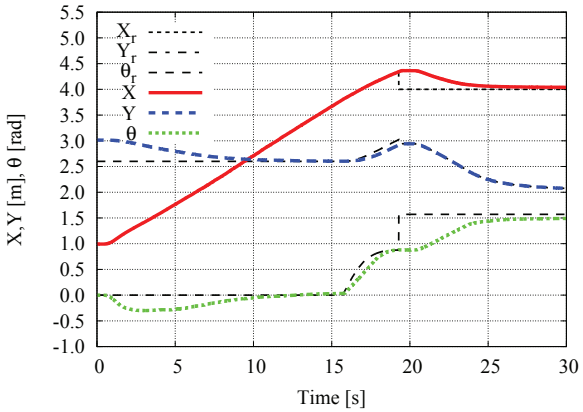

Fig. 13. Position and heading angle (TSCF)

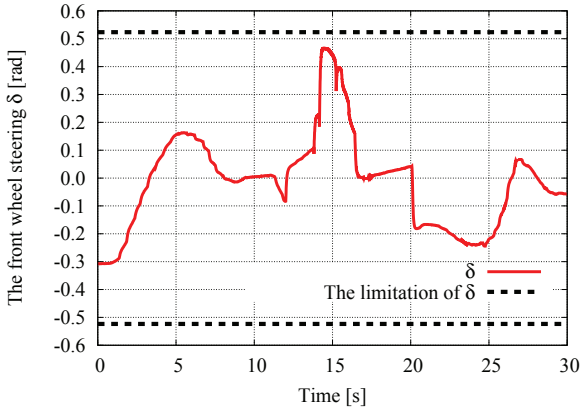

Fig. 11. The front wheel steering (Proposed method)

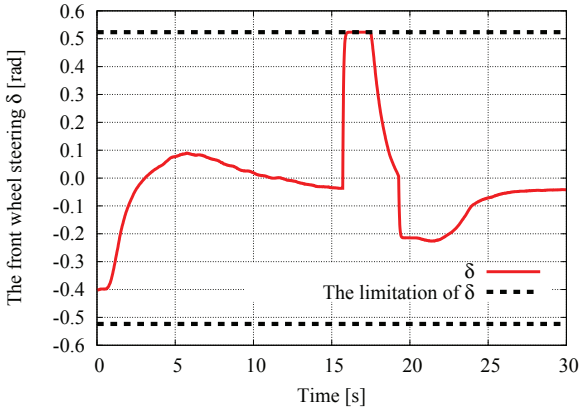

Fig. 14. The front wheel steering (TSCF)

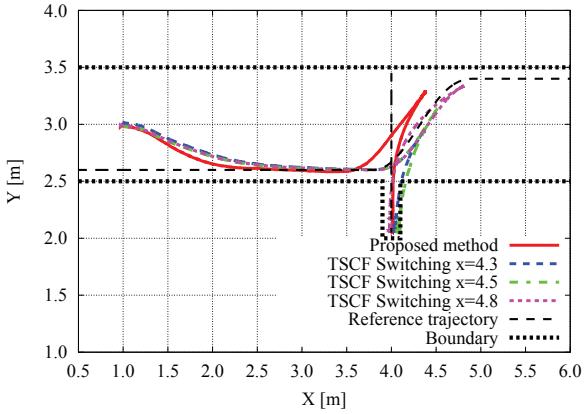

Fig. 15. Comparison of MPC and TSCF

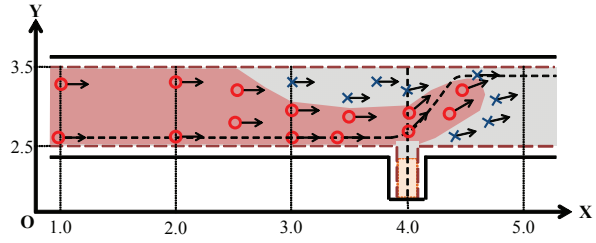

Fig. 16. The successful area of parking control (Proposed method)

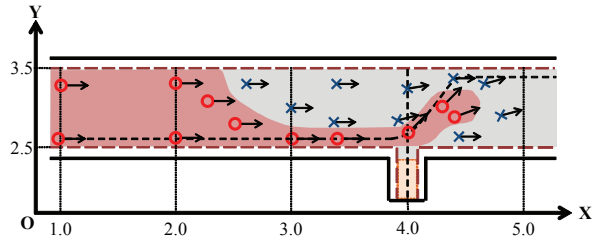

Fig. 17. The successful area of parking control [24]

REFERENCES

[1] M. Sampei, H. Kiyota and M. Ishikawa, "Control strategies for mechanical systems with various constraints-control of non-holonomic systems, "*Proceedings of the 1999 IEEE International Conference on Systems,Man, and Cybernetics*, Vol.3, No.3, 1999, pp. 158-165.

[2] T. Ohtsuka, "A continuation/GMRES method for fast computation of nonlinear receding horizon control, "*Automatica*, Vol.40, No.4, 2004, pp. 563-574.

[3] Mattingley J., Wang Y. and Boyd S., "Receding Horizon Control, "*Control Systems, IEEE*, Vol.31, No.3, 2011, pp. 52-65.

[4] I. Okawa and K. Nonaka, "Optimal Online Generation of Obstacle Avoidance Trajectory running on a Low Speed Embedded CPU for Vehicles, "*International Conference on IEEE Control Applications*, 2010, pp. 1257-1262.

[5] T. Kobayashi and T. Ohtsuka, "Tracking Control of Vehicles Using Nonlinear Model, "*Proceedings of the 1999 IEEE International Conference on Control Applications*, 1999, pp. 1667-1672.

[6] D. Gu and H. Hu, "Receding Horizon Tracking Control of Wheeled Mobile Robots, "*IEEE Transactions on Control Systems Technology*, Vol.14, No.4, 1996, pp. 743-749.

[7] L. Li, Z. Jia, T. Cheng and X. Jia, "Optimal Model Predictive Control for Path Tracking of Autonomous Vehicle, "*Third International Conference on Measuring Technology and Mechatronics Automation*, 2004, pp 791-794.

[8] N. Takahashi and K. Nonaka, "Model Predictive Obstacle Avoidance and Wheel Allocation Control of Mobile Robots Using Embedded CPU, "*Journal of System Design and Dynamics*, Vol.6, No.4, 2012, pp. 447-465.

[9] N.Masashi and T.Ohtsuka, "Vehicle Dynamics Control for Collision Avoidance Considering Physical Limitations, "*SICE Annual Conference*, 2011, pp. 688-693.

[10] H.Okajima, T.Asai and S.Kawaji, "Optimal Velocity Control Method in Path Following Control Problem, "*Proceedings of the 17th IFAC World Congress*, 2008, pp. 90-95.

[11] K.Oyama and K.Nonaka, "Model Predictive Control for Non-holonomic Vehicles with Travel Range Constraint, "*Proceedings of the 29th Symposium on Guidance and Control*, 2012, pp. 65-72 (in Japanese).

[12] K. Kozlowski and D. Pazderski, "Stabilization of Two-Wheeled Mobile Robot Using Smooth Control Laws -Experimental Study, "*Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pp. 3387-3392.

[13] M. Deng, A. Inoue and K. Sekiguchi, "Parking Control of a Two Wheeled Mobile Robot, "*Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, 2007, pp. 539-544.

[14] D. Lyon, "Parallel parking with curvature and nonholonomic constraints, "Proceedings of the Intelligent Vehicles 1992 Symposium, 1992, pp. 341-346.

[15] K. Lee, D. Kim, W. Chung, Hyo W. Chang, and P. Yoon, "Car parking control using a trajectory tracking controller, "SICED-ICASE International Joint Conference 2006, pp. 2058-2063.

[16] H. Kwon and W. Chung, "Comparative Analysis of Path Planners for a Car-like Mobile Robot in a Cluttered Environment, "Proceedings of the 2011 IEEE International Conference of Robotics and Biomimetics, 2011, pp. 602-607.

[17] Igor E. Paromtchik and C. Laugier, "Autonomous parallel parking of a nonholonomic vehicle, Proceedings of the 1996 IEEE Intelligent Vehicles Symposium, pp.13-18, 1996.

[18] K. Jiang, "A sensor guided parallel parking system for nonholonomic vehicles", 2000 IEEE Intelligent Transportation Systems, 2000, pp.270-275.

[19] Muhammad U. Rafique and K. Faraz, "Modified Trajectory Shaping Guidance for autonomous parallel parking, "2010 IEEE Conference on Robotics Automation and Mechatronics, 2010, pp. 458-463.

[20] Moritz B. Oetiker, Gion P. Baker, and Lino Guzzella, "A Navigation-Field-Based Semi-Autonomous Nonholonomic Vehicle-Parking Assistant, "IEEE Transactions on Vehicular Technology, Vol. 58, No. 3, 2009, pp. .

[21] M.Yamamoto, Y.Hayashi and A.Mohri, "Garage parking planning and control of car-like robot using a real time optimization method, "*The 6th IEEE International Symposium on Assembly and Task Planning: From Nano to Macro Assembly and Manufacturing*, 2005, pp. 248-253.

[22] M.Sampei and T.Itoh, "Path Planning and Path Tracking Control of Wheeled Vehicles Using Nonlinear System Theory -Parking Control Using Forward and Backward Movement", *ISCIE Journal 'Systems, Control and Information'"*, Vol.6, No.1, 1993, pp. 37-47 (in Japanese).

[23] Mattingley J. and Boyd S., "CVXGEN : A Code Generator for Embedded Convex Optimization, "*Optimization and Engineering*, Vol.13, No.1, 2012, pp. 1-27.

[24] K.Oyama, K.Nonaka, "Model Predictive Control for Non-holonomic Vehicles with Switching Movement using Time-State Control Form, "*41st SICE Symposium on Control Theory*, 2012, 79-84 (in Japanese).

[25] K.Oyama, K.Nonaka, "Model Predictive Parking Control in Environment with Obstacles, "*13th SICE System Integration Division Annual Conference*, 2012, pp. 2111-2114 (in Japanese).

[26] I.Okawa and K.Nonaka, "Obstacle Avoidance by Model Predictive Control using Self-localization and Obstacle Detection, " *11th SICE System Integration Division Annual Conference*, 2010, pp. 1670-1673 (in Japanese).

[27] Y.Hiromachi, K.Nonaka, "Localization for the fusion of dead-reckoning and LRF of mobile robots, "*Proceedings of the 29th Symposium on Guidance and Control*, 2012, pp. 31-38 (in Japanese)