On the Method of Generating a Circular-Clothoid Trajectory for Front-Steering Vehicles Based on $L^1/L^2$-Optimal Control

Kiyoshi HAMADA *, Ichiro MARUTA *, Kenji FUJIMOTO *, and Kenniti HAMAMOTO **

Abstract: The continuity and the constantness of both the curvature and the velocity in a trajectory are essential for the vehicle trajectory design from the viewpoint of the followability. This paper presents a new method of generating a vehicle trajectory whose curvature and velocity continuously change while both are constant in large part. The generated trajectory is called a circular-clothoid trajectory defined as the trajectory connecting circular curves with clothoid curves. In the method, the trajectory generation problem is formulated as an $L^1/L^2$-optimal control problem for front-steering vehicles. The desired trajectory is obtained numerically by solving a two-point boundary value problem that is relevant to the optimal control problem. The effectiveness of the proposed method is confirmed through some examples.

Key Words: nonlinear optimal control, trajectory generation.

1. Introduction

Autonomous driving systems become essential in various areas in recent years. For example, the aging population causes skilled workers shortage at the construction site in Japan. Hence the unmanned construction system is focused, where a construction vehicle machine moves autonomously [1]. Such systems are also useful for works under severe environments as the Moon [2].

In such a system, a vehicle moves autonomously by tracking a trajectory generated beforehand. A lot of trajectory generation (or path planning) methods have been proposed [3]–[7]. A circular-clothoid trajectory is one of the trajectories that are easy to track. In this trajectory, circular curves are connected by clothoid curves so that the curvature of the trajectory changes continuously. Since vehicles do not have to steer at the circular curves part, such trajectories are easy to track. Besides, these trajectories are useful to avoid the abrupt change of the curvature that causes trouble like a slip. Because of these reasons, such trajectories are often used in trajectory planning [8]–[10]. D.H. Shin et al. proposed a method of generating a path using clothoid segments [11]. In the article, a path composed of clothoid segments is shown to be easier to track than a path consisting of circular curves and straight lines. A method proposed by A. Kelly et al. focuses on generating a smooth trajectory, for example, clothoid curves [12], in which the proposed method generates a trajectory by using polynomial spirals with the assumption that the longitudinal velocity is constant. In some of those conventional methods, a designer first has to specify waypoints appropriately, and then the trajectory is generated so that it passes through the defined waypoints. Besides, the velocity of the generated trajectory changes discontinuously at the turning points where the vehicle changes the direction of travel, while the continuous change of the velocity is desirable for tracking. Some conventional methods are able to generate a trajectory whose velocity is continuous. Howard et al. presented a method of trajectory generation, in which they consider energy consumption and obstacle avoidance [13]. Tocker et al. also proposed a trajectory planning method [14], where a path is generated first, and then its velocity profile is optimized. In these methods, though the velocity is continuous, the generated trajectories have a property that the curvature of the trajectory changes discontinuously. The abrupt change of the curvature causes the steep change of the centrifugal force, and vehicles following such trajectories will slip. Hence such a trajectory is not always easy to track.

This paper presents a method to generate a trajectory whose curvature and velocity are continuous while they are constant in large part by using an $L^1/L^2$-optimal control technique. The generated trajectory becomes circular or clothoid at the part where the longitudinal velocity is constant. Since the trajectory becomes circular or clothoid almost everywhere, we call the generated trajectory as a circular-clothoid trajectory. In this trajectory, the velocity and the curvature change continuously even at the turning point where the vehicle switches the direction of travel. Hence the vehicles can track the generated trajectory with smooth steering and accelerator operation. Though we already presented a method of generating a hands-off trajectory that is easily trackable [15], the method presented in this paper is superior in terms of the continuity of the curvature.

The specific procedure for generating a circular-clothoid trajectory is as follows. First, a vehicle system is transformed into a system whose inputs are the longitudinal velocity and the time derivative of the curvature. The trajectory generation problem then reduces to another problem of generating a trajectory with continuous inputs. In the reduced problem, the longitudinal velocity input is desired to be constant in large part, and the input of the time derivative of the curvature is desired to be zero in large part. This problem is solved as an $L^1/L^2$-optimal control problem. The desired trajectory is obtained numerically by applying the shooting method with the continuation method for...
solving the two-point boundary value problem (TPBVP), the problem which is equal to the optimal control problem [15].

This paper is organized as follows. Section 2 gives the problem setting. Section 3 describes some existing mathematical results. In Section 4, we propose a circular-clothoid trajectory generation method based on $L_2$-optimal control. Section 5 shows the effectiveness of the proposed method by a numerical simulation. In Section 6, the paper is concluded.

2. Problem Setting

The objective of the proposed method is to generate a trajectory whose curvature and longitudinal velocity are continuous while they are constant in large part. The generated trajectory is classified into a circular-clothoid trajectory. Here we define the circular-clothoid trajectory as follows.

**Definition 1** If the time derivative of the curvature and the longitudinal velocity are piecewise constant in a trajectory, the trajectory is a circular-clothoid trajectory.

The target of the method is a front steering vehicle. The physical parameters of the vehicle are defined as in Fig. 1. The symbol $\theta_c$ denotes the attitude angle, $v$ denotes the longitudinal velocity, and $\phi_c$ denotes the steering angle. The coordinate of the center of the axle of the front wheel and the rear wheel are $(x_a, y_a)$ and $(x_c, y_c)$, respectively. The distance between $(x_a, y_a)$ and $(x_c, y_c)$ is defined as $L$. Defining $\theta_c$ and $v$ as inputs $u_1, u_2$, respectively, a state space model of the vehicle is denoted by

$$
\frac{d}{dt} \begin{bmatrix} x_c \\ y_c \\ \theta_c \\ \phi_c \end{bmatrix} = \begin{bmatrix} 0 & \tan \theta_c/L & 0 & \cos \theta_c \\ 0 & \cos \theta_c & 0 & \sin \theta_c \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. 
$$ (1)

For simplicity, we use the following notation

$$
X = G(X)U
$$ (2)

instead of (1), where

$$
G(X) := \begin{bmatrix} 0 & \tan \theta_c/L & 0 & \cos \theta_c \\ 0 & \cos \theta_c & 0 & \sin \theta_c \\ 1 & 0 \\ 0 & 1 \end{bmatrix}. 
$$ (3)

In (2), $X := [\theta_c, x_c, y_c, \phi_c]^T$ and $U := [u_1, u_2]^T$ are the state and the input of the system, respectively. The input $U(t)$ is constrained in magnitude by the relation

$$
|u_i(t)| \leq m_i,
$$

where $m_i \in \mathbb{R}$, $m_i > 0$ ($i = 1, 2$). The value of $m_i$ is determined by the performance of the vehicle.

The problem setting is shown in Fig. 2. The axis $x$ and $y$ denote the Cartesian coordinates of the horizontal plane. The time interval of the trajectory is set to $[0, t_f]$, where $t_f > 0$ is the terminal time. The objective in the proposed method is to generate a circular-clothoid trajectory from the initial state $X^0$ to the terminal state $X^f$ for the front steering vehicle, illustrated as the dotted curve in Fig. 2.

3. Preliminaries

The mathematical results used in the proposed method are reviewed in this section. The following discussion is based on the system (2) and the problem explained in Section 2.

3.1 $L_1^1/L_2^2$-Optimal Control Theory

The objective of an $L_1^1/L_2^2$-optimal control problem is to find an optimal input vector $U^{\ast}(t), t \in [0, t_1]$ which is the solution of the following problem.

$$
\begin{align*}
\text{minimize} & \quad J_{12} := \int_0^{t_1} \sum_{i=1}^{2} \left( w_{i} |u_i(t)| + r_{i} |u_{i}(t)|^2 \right) \, dt \\
\text{subject to} & \quad X(0) = X^0, X(t_f) = X^f,
\end{align*}
$$ (5)

where $w_i > 0, r_i > 0$ ($i = 1, 2$) are weights of the norm. The time $t_i$ has to be large enough than the minimum time $t_i^{\ast}$ that is obtained by solving a minimum time optimal control problem [16].

The optimal input is obtained analytically as a function of $X(t)$ and $P(t)$ by analyzing a Hamiltonian function with respect to the $L_1^1/L_2^2$-optimal control problem, where $P(t) \in \mathbb{R}^4$ is a costate vector of $X(t), P(t) \in \mathbb{R}^4$. The Hamiltonian function is defined as

$$
H(X(t), P(t), U(t)) := \sum_{i=1}^{2} \left( w_{i} |u_i(t)| + \frac{r_{i}}{2} u_{i}(t)^2 \right) + P(t)^T G(X(t)) U(t).
$$ (6)

According to the minimum principle, the optimal state $X^{\ast}(t)$, the optimal costate $P^{\ast}(t)$, and the optimal input $U^{\ast}(t)$ minimizing the Hamiltonian function (6) also minimize the cost function (5).

Assume that $X^{\ast}(t)$ and $P^{\ast}(t)$ are given. By completing the square, the Hamiltonian function is written by

$$
H(X^{\ast}(t), P^{\ast}(t), U(t))
$$

$$
= \sum_{i=1}^{2} \left( w_{i} \tilde{u}_i(t) + P^{\ast}(t)^T g_i(X^{\ast}(t)) \tilde{u}_i(t) + \frac{r_{i}}{2} \tilde{u}_{i}(t)^2 \right)
$$

$$
= \sum_{i=1}^{2} \left( \frac{a}{2} \left( \tilde{u}_i + \frac{w_i P^{\ast} g_i(X^{\ast}(t))}{r_i} \right)^2 + \tilde{C}_i \right) \quad \text{if } u_i \geq 0,
$$

$$
= \sum_{i=1}^{2} \left( \frac{a}{2} \left( \tilde{u}_i - \frac{w_i - P^{\ast} g_i(X^{\ast}(t))}{r_i} \right)^2 + \tilde{C}_i \right) \quad \text{if } u_i < 0
$$

with $X^{\ast}(t)$ and $P^{\ast}(t)$. In the last line of (7), we omit $t$ for simplicity. The symbols $g_i$ ($i = 1, 2$) denote the column vectors of $G$, and they are in the relation $G(X(t)) = [g_1(X(t)), g_2(X(t))]$. In (7), $C_i$ and $\tilde{C}_i$ ($i = 1, 2$) are functions of $X^{\ast}(t)$ and $P^{\ast}(t)$, and they are independent of $U(t)$. Hence, from (7), the optimal input $U^{\ast}(t) = [u_1^{\ast}(t), u_2^{\ast}(t)]^T$ minimizing the Hamiltonian function (6) is denoted by
The symbol $S_{α}(·)$ denotes the shrinkage function represented by

$$S_{α}(v) := \begin{cases} v + α & \text{if } v < -α, \\ 0 & \text{if } -α \leq v \leq α, \\ v - α & \text{if } v > α, \end{cases}$$

and $\text{sat}_m(·)$ denotes the saturation function

$$\text{sat}_m(v) := \begin{cases} -m & \text{if } v < -m, \\ v & \text{if } -m \leq v \leq m, \\ m & \text{if } v > m. \end{cases}$$

Figure 3 shows the optimal input as a function of $X^*(t)$ and $P^*(t)$. As the figure shows, the optimal input is more likely to become zero. The $L^1/L^2$-optimal input becomes like a bang-off-bang form when the weight of the $L^1$ norm becomes greater enough than the weight of the $L^2$ norm. This property is explained in the next subsection.

### 3.2 Bang-off-Bang Property of $L^1/L^2$-Optimal Control

To show that the $L^1/L^2$-optimal control input has a form like bang-off-bang, we explain the limiting property of $L^1/L^2$-optimal control. The $L^1/L^2$-optimal control is a mixture of $L^1$-optimal control and $L^2$-optimal control. A cost function of an $L^1$-optimal control problem is defined as

$$J_1 := \int_0^t \sum_{i=1}^2 (w_i|h_i(t)|) \, dt$$

s.t. $X(0) = X^0$, $X(t_i) = X^i$.

The $L^1$-optimal input $U^*$ minimizing (11) is given by

$$u^*_i(t) = -D_{m,w}(P^*(t)^Tg_i(X^*(t))),$$

where $D_{m,w}$ is a dead-zone function defined as

$$D_{m,w}(v) := \begin{cases} -m & \text{if } v < -w, \\ 0 & \text{if } -w \leq v < w, \\ v & \text{if } v \leq w, \end{cases}$$

$$D_{m,w}(v) \in [-m,0] \text{ if } v = -w,$$

$$D_{m,w}(v) \in [0,m] \text{ if } v = w.$$  

Note that the value of the dead-zone function $D_{m,w}$ is not uniquely determined when the argument is equal to ±$w$. The problem is called normal if the $L^1$-optimal control input is uniquely determined at almost all $t \in [0,t_i]$ [17].

On the other hand, a cost function of an $L^2$-optimal control problem is defined as

$$J_2 := \int_0^t \sum_{i=1}^2 \left( \frac{1}{r_i} \cdot |u_i(t)|^2 \right) \, dt$$

s.t. $X(0) = X^0$, $X(t_i) = X^i$.

The $L^2$-optimal input $U^*(t)$ minimizing (14) is given by

$$u^*_i(t) = -\text{sat}(P^*(t)^T g_i(X^*(t)))/r_i.$$  

The $L^1/L^2$-optimal input has the following limiting property with respect to the $L^1$-optimal input and the $L^2$-optimal input [17].

**Proposition 1** Assume the $L^1$-optimal control problem is normal. Let $u^1(w)$, $u^2(r)$, and $u^{12}(w;r)$ be the $L^1$-optimal input, the $L^2$-optimal input, and the $L^1/L^2$-optimal input with weight parameters $w := (w_1,w_2)$, $r := (r_1,r_2)$, respectively. For any fixed $w > 0$,

$$\lim_{r \to 0} u_{12}(w,r) = u_1(w)$$

holds. Also, for any fixed $r > 0$,

$$\lim_{w \to 0} u_{12}(w,r) = u_2(r)$$

holds.

Hence, if the weight of the $L^1$ norm becomes greater enough than the weight of the $L^2$ norm, the $L^1/L^2$-optimal input becomes like a bang-off-bang form. Note that the $L^1/L^2$-optimal input is continuous though the $L^1$-optimal input is discontinuous. The proposed method generates a circular-clothoid trajectory by using this property.

### 4. Proposed Method

The objective of the proposed method is to generate a trajectory whose curvature and longitudinal velocity are continuous while they are constant in large part. The proposed method is based on two new ideas. One is to transform the original plant system into another system that has two inputs, the longitudinal velocity and the time derivative of the curvature. The other is to reduce the trajectory generation problem with the transformed system to an $L^1/L^2$-optimal control problem. The optimal control problem is transformed into a TPBVP, and the solution is obtained numerically by the shooting method with the continuation method [15].

#### 4.1 Transformation of the Vehicle System

From the definition, the curvature $κ_c$ of the trajectory of the vehicle is denoted as

$$κ_c = \frac{dφ}{ds} = \frac{dθ_c}{dt} \frac{dr}{ds},$$

where $s$ is the arc length of the trajectory. Since the time derivative of $s$ is equal to $u^2$ in the problem,

$$\frac{dθ_c}{dt} = κ_c u_2 = \frac{\tan φ_c}{L} u_2$$

(19)
holds from (18). Therefore, define a new input \( \tilde{U} = [\tilde{a}_1, \tilde{a}_2] \)' as
\[
\tilde{a}_1 = \dot{k}_c, \\
\tilde{a}_2 = u_2,
\]
and using \( k_c \) instead of \( \phi_c \), (1) is transformed to
\[
\begin{bmatrix}
\frac{d}{dt} \theta_c \\
\frac{d}{dt} x_c \\
\frac{d}{dt} y_c \\
\frac{d}{dt} \kappa_c
\end{bmatrix} = \begin{bmatrix}
0 & k_c \\
\cos \theta_c & 0 \\
\sin \theta_c & 0 \\
1 & 0
\end{bmatrix} \begin{bmatrix}
\tilde{a}_1 \\
\tilde{a}_2
\end{bmatrix}. 
\]
(22)

For simplicity, we use the following notation
\[
\dot{X} = G(\tilde{X}) \tilde{U}
\]
instead of (22), where
\[
G(\tilde{X}) := \begin{bmatrix}
0 & k_c \\
0 & \cos \theta_c \\
0 & \sin \theta_c \\
1 & 0
\end{bmatrix}. 
\]
(24)

In (23), \( \dot{X} := [\theta_c, x_c, y_c, \kappa_c] \)' is the state of the system. In the proposed method, \( \tilde{U}(t) \)' is constrained in magnitude by the new relation
\[
|\tilde{a}_i(t)| \leq \tilde{m}_i,
\]
(25)
where \( \tilde{m}_i \in \mathbb{R}, \tilde{m}_i > 0 \) (\( i = 1, 2 \)). The value of \( \tilde{m}_i \) is determined by the performance of the vehicle.

4.2 Design of the Optimal Control Problem

Next, the trajectory generation problem is reduced to an \( L^1/L^2 \)-optimal control problem in the proposed method. In the proposed method, the cost function of the optimal control problem is defined as follows.
\[
J := \int_0^{t_f} \left[ w_1 |\tilde{a}_1(t)| + w_2 |\tilde{a}_2(t)| \right]
\]
\[
+ \frac{r_1}{2} |\tilde{a}_1(t)|^2 + \frac{r_2}{2} |\tilde{a}_2(t)|^2 dt
\]
(26)

s.t. \( \dot{X}(0) = \bar{X}^0, \dot{X}(t_f) = \bar{X}^f \). The weights satisfy \( w_i > 0 \) and \( r_i > 0 \) (\( i = 1, 2 \)). The boundary conditions are defined as
\[
\bar{X}^0 = \phi(X^0), \quad \bar{X}^f = \phi(X^f),
\]
(27)
where
\[
\Phi : \left[ \theta_c, x_c, y_c, \phi_c \right] \rightarrow \left[ \theta_c, x_c, y_c, \tan(\phi_c)/L \right]. 
\]
(28)

It follows from (8) that the optimal input \( \bar{U}^*(t) \) minimizing the cost function (26) is denoted by the following equations :
\[
\bar{a}_i(t) = -\text{sat}_{\bar{m}_i} \left( S \frac{P_i(t) \bar{G}_i(\bar{X}(t))}{r_i} \right) \quad (i = 1, 2),
\]
(29)
where \( G(\bar{X}(t)) = \left[ \bar{g}_1(\bar{X}(t)), \bar{g}_2(\bar{X}(t)) \right] \), and \( P_i(t) \) is the optimal costate of the optimal state \( \bar{X}(t) \). Note that a circular trajectory and a clothoid trajectory are characterized in terms of the time derivative of the curvature with the constant longitudinal velocity, as \( \kappa_c = 0 \) and \( \kappa_c = \text{const.} \), respectively. Since the \( L^1/L^2 \)-optimal input is more likely to become zero, the curvature becomes constant in the large part of the trajectory. Besides, because of the bang-off-bang property of the \( L^1/L^2 \)-optimal control, the optimal inputs \( \bar{a}_i(t) = k_c(t) \) and \( \bar{a}_2(t) \) become like bang-off-bang if the weight \( w_i \) is set to be larger than \( r_i \). Hence the solution of the \( L^1/L^2 \)-optimal control problem is classified into the circular-clothoid trajectory.

In the calculation of the trajectory and the optimal input, the above optimal control problem is reduced to a TPBVP [18], a problem to solve the following equations for \( \dot{X}^*(t), P^*(t) \) :
\[
\frac{d\dot{X}^*(t)}{dt} = \frac{\partial H(\dot{X}^*(t), P^*(t), \dot{U}^*(t))}{\partial P^*(t)} \bigg|_{U^*=U^*(\dot{X}^*(t), P^*(t))}, \\
\frac{dP^*(t)}{dt} = -\frac{\partial H(\dot{X}^*(t), P^*(t), \dot{U}^*(t))}{\partial \dot{X}^*(t)} \bigg|_{U^*=U^*(\dot{X}^*(t), P^*(t))}
\]
(30)
s.t. \( \dot{X}^*(0) = \bar{X}^0, \quad \dot{X}^*(t_f) = \bar{X}^f \).

where \( \bar{U}^*(\dot{X}^*(t), P^*(t)) \) is obtained as (29) and the symbol \( H \) denotes the Hamiltonian function defined as
\[
H(\dot{X}^*(t), P^*(t), \dot{U}^*(t)) = w_1 |\bar{a}_1(t)| + w_2 |\bar{a}_2(t)| + \frac{r_1}{2} |\bar{a}_1(t)|^2 + \frac{r_2}{2} |\bar{a}_2(t)|^2 + P_i(t)^2 \dot{X}^*(t).
\]
(31)

Hence, by solving the TPBVP (30) for \( \dot{X}^*(t), \dot{P}^*(t) \), we can obtain the optimal input \( \bar{U}^*(t) \) and the desired trajectory \( \dot{X}^*(t) \). There are a lot of numerical methods for solving TPBVPs, for example, the iterative method and the shooting method [19]. In this paper, we use the shooting method with the continuation method for solving the problem. The shooting method searches \( \bar{P}^*(0) \) satisfying (30). This method requires an initial guess \( \bar{P}_0 \) of \( \bar{P}^*(0) \) for searching. The algorithm of searching is numerically stable if \( \bar{P}_0 \) is close enough to \( \bar{P}^*(0) \). The continuation method ensures that the algorithm is always numerically stable. The details of the algorithm are explained in the next section.

4.3 Algorithm of the Proposed Method

The proposed method calculates the desired trajectory numerically by solving the TPBVP (30). Since it is difficult to solve the \( L^1/L^2 \)-optimal control problem directly, the continuation method is used for solving the problem [15],[20]. In the algorithm, we first set the weight \( w_i \) (\( i = 1, 2 \)) of the cost function to zero, and the optimal control becomes the \( L^2 \)-optimal control problem that is relatively easy to solve. The algorithm solves (30) with \( w_i = 0 \) and obtains the solution \( \bar{P}^*(0) \). Next, the weight \( w_i \) is slightly increased by the parameter \( \Delta w_i \) and the algorithm again solves (30) by using obtained \( \bar{P}^*(0) \) as an initial guess \( \bar{P}_0 \). Since the guess is close enough to a solution, the shooting method can solve the TPBVP in a numerically stable manner. The previous procedure is iterated until more than \( \alpha \% \) of the inputs become piecewise constant. Finally, the desired trajectory, which is classified into the circular-clothoid trajectory, is generated. Note that the differential equation (30) depends on \( w = [w_1, w_2] \).

The specific algorithm of the proposed method is shown in Algorithm 1. Before executing the algorithm, \( \bar{X}^0, \bar{X}^f, r_i, r_2, \tilde{m}_1, \tilde{m}_2 \) have to be defined as the problem setting. Note that the value \( r_i \) has to be larger enough than \( \tilde{r}_i \) as mentioned in Section 3.1. The values of \( r_1, r_2 \) should be set to small so that
the algorithm ends in the fewer iterations. Besides, the design parameters $\Delta w_1, \Delta w_2, \alpha$ have to be defined. The values of $\Delta w_1, \Delta w_2$ are set to small so that the algorithm is stable, and $\alpha$ is set to high so that the trajectory becomes a circular-clothoid trajectory.

**Algorithm 1** Trajectory generation based on $L^1/L^2$-optimization

*Input:* $t_i, \dot{x}_i, \dot{y}_i, \rho, r_1, r_2, \Delta w_1, \Delta w_2, \alpha$

*Output:* $\tilde{X}(t)$

1: $w_1, w_2 \leftarrow 0$
2: loop
3: Find $\tilde{X}^2(t), \tilde{P}^2(t), \tilde{U}^2(t)$ by solving TPBVP(30)
4: if $\int_0^t ||\tilde{u}_i(t)|| dt > \frac{1}{\max} t_i (i = 1 \text{ and } 2)$ then
5: return $\tilde{X}^2(t)$
6: end if
7: $w_j \leftarrow w_j + \Delta w_j (i = 1, 2)$
8: end loop

In the next section, the effectiveness of the proposed method is demonstrated by numerical simulations of a front steering vehicle.

### 5. Numerical Example

#### 5.1 Example 1: Reduction of the Steering Operation

The parameters are defined as $X^0 = [0, 0, 0, 0]^T, X^f = [0.45, 0, 0, 0]^T, t_i = 6, r_1 = 1, r_2 = 1, \dot{m}_1 = 0.3, \dot{m}_2 = 0.8, \alpha = 90$, in this example. The incremental values of the weight are set as $\Delta w_1 = 0.05, \Delta w_2 = 0.1$.

Figure 4 shows the generated trajectory. The horizontal and vertical axes are $x$ and $y$, respectively. The solid line illustrates the generated trajectory. Figure 5 shows the response of the inputs, where the horizontal axis is time, and the vertical axis is the input. In the figure, the thick solid line denotes the input of the time derivative of the curvature $\tilde{\kappa}_c(t)$ while the thin solid line is the longitudinal velocity input of the vehicle. The dashed lines illustrate the upper and lower bounds of $\tilde{\kappa}_c(t)$ and $\tilde{u}_2(t)$. As Fig. 5 shows, the time derivative of the curvature is zero in large part, that is, the curvature is constant in large part. Since the time derivative of the curvature and the longitudinal velocity is piecewise constant in large part, the generated trajectory is classified into the circular-clothoid trajectory. In the example, about 90.2% of the time derivative of the curvature and 92.9% of the longitudinal velocity are piecewise constant.

To make clear the advantage of the proposed method, Fig. 6 shows the generated circular-clothoid trajectory, in which the vehicle goes forward first and then goes backward. Note that any waypoint is not specified. The horizontal and vertical axes are $x$ and $y$, respectively. The solid line illustrates the generated trajectory. In Fig. 7, the horizontal axis is time, and the vertical axis is the input. In the figure, the thick solid line denotes the input of the time derivative of the curvature $\tilde{\kappa}_c(t)$ while the thin solid line is the longitudinal velocity input of the vehicle. The dashed lines illustrate the upper and lower bounds of $\tilde{\kappa}_c(t)$ and $\tilde{u}_2(t)$. As Fig. 7 shows, both the time derivative of the curvature and the longitudinal velocity are not piecewise constant. Hence the trajectory generated by the conventional method requires much steering operation.

#### 5.2 Example 2: Automatic Generation of the Turning Point

The parameters are defined as $X^0 = [\pi/6, 0, 0, 0]^T, X^f = [0, 2, 1.5, 0]^T, t_i = 6, r_1 = 1, r_2 = 1, \dot{m}_1 = 0.3, \dot{m}_2 = 0.8, \alpha = 85$, in this example. The incremental values of the weight are set as $\Delta w_1 = 0.5, \Delta w_2 = 1$.

Figure 8 shows the generated circular-clothoid trajectory, in which the vehicle goes forward first and then goes backward. Note that any waypoint is not specified. The horizontal and vertical axes are $x$ and $y$, respectively. The solid line illustrates the generated trajectory. In Fig. 9, the horizontal axis is time, and the vertical axis is the input. In the figure, the thick solid line denotes the input of the time derivative of the curvature $\tilde{\kappa}_c(t)$ while the thin solid line is the longitudinal velocity input of the vehicle. The dashed lines illustrate the upper and lower bounds of $\tilde{\kappa}_c(t)$ and $\tilde{u}_2(t)$. As Fig. 9 shows, though the curvature is not constant in large part...
because of the boundary conditions, the generated trajectory is the circular-clothoid trajectory. Besides, the longitudinal velocity changes continuously, even at the turning point. In the example, about 95.2% of the time derivative of the curvature and 88.1% of the longitudinal velocity are piecewise constant.

### 6. Conclusion

In this paper, we proposed the method for generating a trajectory whose curvature and longitudinal velocity are continuous while they are constant in large part. The method is based on $L^1/L^2$-optimal control. In the proposed method, the vehicle system is reduced to the system whose inputs are the time derivative of the curvature and the longitudinal velocity. The trajectory generation problem is then reduced to the specific $L^1/L^2$-optimal control problem. The systematic algorithm based on the continuation method is proposed for solving the optimal control problem. The trajectory generated by the proposed method is confirmed to be classified into the circular-clothoid trajectory through the numerical simulations. The computational cost depends on the conditions of the trajectory generation problem. Though it takes 11 minutes and 25 seconds for generating the desired trajectory in example 1, it takes only 31 seconds in example 2. Our future work is to consider the dynamics of the vehicle to make this method more practical.

### Acknowledgments

This work was supported by the Space Exploration Innovation Hub Center (TansaX), Japan Aerospace Exploration Agency (JAXA) under Support Program for starting up Innovation Hub on the National Research and Development Agency promoted by the Japan Science and Technology Agency (JST).

### References

[1] S. Miura, I. Kuronuma, and K. Hamamoto: Next generation construction production system: On automated construction machinery, *Proceedings of the Seventh Civil Engineering Conference in the Asian Region*, pp. 1–6, 2016.

[2] T. Narumi, S. Aoki, T. Yokosima, N. Uyama, S. Wakabayashi, G. Tabuchi, and H. Kanamori: Preliminary system design for unmanned building construction in extreme environments, *IEEE International Conference on Research and Education in Mechatronics (REM)* 2017, pp. 1–6, 2017.

[3] H. Delingette, M. Hebert, and K. Ikeuchi: Trajectory generation with curvature constraint based on energy minimization, *Proc. IEEE/RSJ International Workshop on Intelligent Robots and Systems Intelligence for Mechanical Systems (IROS’91)*, pp. 206–211, 1991.

[4] R.M. Murray and S.S. Sastry: Nonholonomic motion planning: Steering using sinusoids, *IEEE Transactions on Automatic Control*, Vol. 38, No. 5, pp. 700–716, 1993.

[5] Y.J. Kanayama and B.I. Hartman: Smooth local-path planning for autonomous vehicles, *The International Journal of Robotics Research*, Vol. 16, No. 3, pp. 263–284, 1997.

[6] J. Reuter: Mobile robots trajectories with continuously differentiable curvature: An optimal control approach, *Proc. IEEE/RSJ 1998 The International Conference on Intelligent Robots and Systems*, Vol. 1, pp. 38–43, 1998.

[7] B. Nagy and A. Kelly: Trajectory generation for car-like robots using cubic curvature polynomials, *Field and Service Robots*, Vol. 11, 2001.

[8] D.K. Wilde: Computing clothoid segments for trajectory generation, *Proc. IEEE/RSJ The International Conference on Intelligent Robots and Systems 2009*, pp. 2440–2445, 2009.

[9] C. Alia, T. Gilles, T. Reine, and C. Ali: Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method, *IEEE Intelligent Vehicles Symposium (IV)* 2015, pp. 674–679, 2015.

[10] V. Schneider, P. Piprek, S.P. Schatz, T. Baier, C. Dörhoff, M. Hochstrasser, A. Gabrys, E. Karlsson, C. Krause, P.J. Lauffs et al.: Online trajectory generation using clothoid segments, *IEEE 14th International Conference on Control, Automation, Robotics and Vision (ICARCV) 2016*, pp. 1–6, 2016.

[11] D.H. Shin, S. Singh, and W. Whittaker: Path generation for a robot vehicle using composite clothoid segments, *IFAC Proceedings Volumes*, Vol. 25, No. 6, pp. 443–448, 1992.

[12] A. Kelly and B. Nagy: Reactive nonholonomic trajectory generation via parametric optimal control, *The International Journal of Robotics Research*, Vol. 22, No. 7–8, pp. 583–601, 2003.

[13] T.M. Howard and A. Kelly: Optimal rough terrain trajectory generation for wheeled mobile robots, *The International Journal of Robotics Research*, Vol. 26, No. 2, pp. 141–166, 2007.

[14] P. Tokekar, N. Karnad, and V. Isler: Energy-optimal trajectory planning for car-like robots, *Autonomous Robots*, Vol. 37, No. 3, pp. 279–300, 2014.

[15] K. Hamada, I. Maruta, K. Fujimoto, and K. Hamamoto: On hands-off trajectory generation for a two-wheeled rover based on $L^1/L^2$-optimal control, *Proceedings of IECON 2018: 44th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2601–2606, 2018.

[16] M. Athans and P.L. Falb: *Optimal Control: An Introduction to the Theory and Its Applications*, Chap. 3, 6, *Corporar Courier Corporation*, 2013.

[17] M. Nagahara, D.E. Quevedo, and D. Nesic: Maximum hands-off control and $L^1$ optimality, *IEEE 52nd Annual Conference on Decision and Control (CDC)* 2013, pp. 3825–3830, 2013.

[18] H.P. Geering: *Optimal Control with Engineering Applications*, Springer, 2007.

[19] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery: *Numerical Recipes in C: Japanese Edition*, Gijutsu-Hyohon, 1993 (in Japanese).

[20] S.L. Richter and R.A. Decarlo: Continuation methods: Theory and applications, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 4, pp. 459–464, 1983.
Kiyoshi Hamada (Student Member)

He received his Bachelor of Engineering and Master of Engineering degrees from Kyoto University, Kyoto, Japan, in 2017 and 2019, respectively. In 2017, he joined the Graduate School of Engineering, Kyoto University, where he is currently a Ph.D. student.

Ichiro Maruta (Member)

He received the Bachelor of Engineering, Master of Informatics, and Doctor of Informatics degrees from Kyoto University, Kyoto, Japan, in 2006, 2008, and 2011, respectively. He was a research fellow of the Japan Society for the Promotion of Science from 2008 to 2011. From 2012 to 2017, he was an Assistant Professor at the Graduate School of Informatics, Kyoto University. In 2017, he joined the Graduate School of Engineering, Kyoto University, as a Lecturer of the Department of Aeronautics and Astronautics, and since 2019, he is an Associate Professor.

Kenji Fujimoto (Member)

He received his B.Sc. and M.Sc. degrees in Engineering and Ph.D. degree in Informatics from Kyoto University, Japan, in 1994, 1996, and 2001, respectively. He is currently a professor of Graduate School of Engineering, Nagoya University, Japan. From 1997 to 2004, he was a research associate of Graduate School of Engineering and Graduate School of Informatics, Kyoto University, Japan. From 2004 to 2012, he was an associate professor of Graduate School of Engineering, Nagoya University, Japan. From 1999 to 2000, he was a research fellow of Department of Electrical Engineering, Delft University of Technology, The Netherlands. He has held visiting research positions at the Australian National University, Australia and Delft University of Technology, The Netherlands in 1999 and 2002, respectively. His research interests include nonlinear control and stochastic systems theory.

Kenniti Hamamoto (Member)

He received the Bachelor of Engineering, Master of Engineering, and Doctor of Informatics degrees from Kyoto University, Kyoto, Japan, in 1995, 1997, and 2000, respectively. He was a JSPS research fellowship for young scientists (PD) from 2000 to 2001. Since 2001, he is a research engineer at Kajima Technical Research Institute of Kajima Corporation.