Locally deforming continuation method based on a shooting method for a class of optimal control problems

Kiyoshi Hamada, Ichiro Maruta, Kenji Fujimoto and Kenniti Hamamoto

Department of Aeronautics and Astronautics, Kyoto University, Kyoto, Japan; Advanced Technology Incubation and Mechatronics Group, Kajima Technical Research Institute, Tokyo, Japan

ABSTRACT
This paper proposes a new continuation method for solving optimal control problems. The proposed method is based on a shooting method. In the proposed method, a cost function of an optimal control problem is locally deformed to find the solution in a stable way. This paper also analyses a relationship between the variation of the continuation parameter and the proximity of the solutions before and after a deformation in the proposed method. The obtained relation provides guidance on how to deform the continuation parameter. The effectiveness of this method is confirmed through numerical examples.

1. Introduction
Optimal control has been widely used in various fields, and various methods for solving optimal control problems exist. In optimal control problems for linear systems, the solution is often found analytically and is relatively easy to solve. On the other hand, in the case of nonlinear systems, the optimal solution is rarely found analytically, and therefore the solution is calculated numerically. There are two main methods for solving nonlinear optimal control problems numerically: the direct method and the indirect method [1]. The direct method first approximates the input of the solution of the optimal control problem with a function. Then it updates the function to minimize the cost function. On the other hand, in the indirect method, the optimal control problem is replaced by a two-point boundary value problem (TPBVP), and the TPBVP is solved numerically [2]. Generally speaking, the indirect method is more accurate than the direct method. Several indirect methods, such as the shooting methods [3,4] and the collocation methods [5], are known. The shooting methods are one of the most popular indirect methods. In the methods, the TPBVP is solved as an initial value problem, and they have an advantage that the computational complexity is very small because the search parameters are only the initial values. However, the shooting methods have a disadvantage that the convergence region is small, and therefore a good estimate of the solution has to be given beforehand. The continuation method is one of the ways to mitigate this difficulty.

The continuation method is a method to solve a difficult problem by iteratively deforming the problem continuously with a continuation parameter, and various methods have been proposed [6–9]. The idea of the method is to find a solution in a stable way by using the solution before deformation as an initial guess of the current solution after deformation. In the method, an easy optimal control problem is set up and solved first. Next, the cost function of the easy problem is deformed slightly towards the original problem, and the deformed optimal control problem is solved by using the previously obtained solution as an initial guess. By repeating this procedure and transforming the problem into the original problem in the end, the solution of the original problem is obtained. If each deformation is small, the solutions before and after a deformation are expected to be close enough, hence the method can solve the original problem in a stable way. The continuation method is widely used for solving various difficult problems such as obstacle avoidance problems [10], model predictive control problems [11], and sparse optimal control problems like an $L^1/L^2$-optimal control problem [12]. However, there is no continuation method that is effective for all problems, and it is unclear how the change of the cost function affects the closeness of the solutions before and after the deformation in those conventional methods.

In this paper, we propose a new continuation method, named a locally deforming continuation method, based on a shooting method. The idea of this method is to transform the cost function of the optimal
control problem locally during the iterations of the continuation method. In addition, we provide a relation between the variation of the continuation parameter and the closeness of the solutions before and after the deformation in the form of an upper bound of the errors in the shooting method. This relation gives clear guidance on how to change the continuation parameter during the algorithm for stably solving relatively difficult problems.

The rest of the paper is organized as follows. In Section 2, we introduce the problem formulation, and in Section 3, we briefly revisit some previous results. Next, Section 4 proposes the locally deforming continuation method and a modified shooting method and the results of our method are illustrated in Section 5. We summarize this article in Section 6.

**Notation:** The Euclidean norm of the vector $x$ is expressed as $\|x\|$. The symbol $x_i$ represents the $i$th element of the vector $x$. The function $\min(a)$ returns the minimum element of the vector $a$ and $\max$ returns the maximum element of the vector $a$.

2. Problem setting

Here we treat the optimal control problem to find the input which minimizes the cost function

$$ J = \int_0^{t_f} l(x(t), u(t), t) \, dt, $$

subject to the following dynamics and input constraints:

$$ \frac{dx(t)}{dt} = f(x(t)) + g(x(t))u(t), \quad x(0) = x^0, \quad x(t_f) = x^f, \quad |u_i(t)| \leq u_{bi}, \quad (i = 1, \ldots, m), $$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector, $u_{bi} \in \mathbb{R}^m$ is a positive constant vector, $t_f \in \mathbb{R}_+$ is a terminal time, $x^0 \in \mathbb{R}^n$ is an initial state vector, $x^f \in \mathbb{R}^n$ is a terminal state vector, $f : \mathbb{R}^n \to \mathbb{R}^n$, $g : \mathbb{R}^n \to \mathbb{R}^{n \times m}$, and $l : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}_+$. In this paper, we assume that $l$ is continuously differentiable for $x$. If the dynamics (2) is nonlinear, the input cannot be obtained analytically in general. There are several ways to solve the problem, and these methods are mainly classified into the direct method or the indirect method. In this study, we use a shooting method classified into the indirect method for solving this problem.

When the cost is the $L^1$ norm of the input, the problem becomes a sparse optimal control problem, and the resulting optimal control input is sparse, i.e. the $L^0$ norm of the input is much smaller than $t_f$ [12]. However, in the case of nonlinear systems, problems such as $L^1$-optimal control problems are challenging to solve. The continuation method is one of the techniques to overcome this difficulty. The next section briefly reviews a conventional method.

3. Preliminaries

This section briefly reviews an existing way to solve an optimal control problem with a shooting method.

3.1. Transformation of the optimal control problem

Define a Hamiltonian function as follows:

$$ H(x(t), \lambda(t), u(t), t) = l(x(t), u(t), t) + \lambda(t)^\top (f(x(t)) + g(x(t))u(t)), $$

It follows from Pontryagin's minimum principle that the optimal control problem with (1)–(4) is reduced to a problem of solving the following simultaneous differential equation:

$$ \frac{dx(t)}{dt} = \frac{\partial H(x(t), \lambda(t), u^*(t), t)}{\partial \lambda(t)}, \quad \frac{d\lambda(t)}{dt} = -\frac{\partial H(x(t), \lambda(t), u^*(t), t)}{\partial x(t)} $$

subject to

$$ u^*(t) = \arg\min_{u(t)} H(x(t), \lambda(t), u(t), t), $$

with $\lambda(t) \in \mathbb{R}^n$ that is the costate of $x(t)$. This problem is called a TPBVP. Assume that (8) can be rewritten as the following by using a function of $x(t), \lambda(t), \text{and } t$.

$$ u^*(t) = f_u(x(t), \lambda(t), t), $$

where $f_u : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^m$. In this case, the TPBVP is rewritten as follows with $\xi(t) = (x(t)^\top, \lambda(t)^\top)^\top$.

Find $x(t)$ and $\lambda(t), \text{and } t \in [0, t_f]$ satisfying

$$ \frac{d\xi}{dt} = \begin{pmatrix} \frac{\partial H(x, \lambda, u, t)}{\partial x} \\ -\frac{\partial H(x, \lambda, u, t)}{\partial x} \end{pmatrix} \bigg|_{u = f_u(x, \lambda, t)} =: F(\xi, t), $$

satisfying (3) and (4).

Here the defined function $F$ is $F : \mathbb{R}^{2n} \times [0, t_f] \to \mathbb{R}^{2n}$. There are several ways to solve this TPBVP, such as the shooting methods and the collocation methods. Among them, the shooting methods have an advantage that the number of the search parameters is small. The next section briefly explains a standard shooting method.

3.2. Shooting method

If $x(0)$ and $\lambda(0)$ are given, $x(t), \lambda(t)$ satisfying (10) is uniquely determined. The shooting method takes advantage of this property and treats the TPBVP as an initial value problem. Since $x(0)$ is already given as the
boundary condition, the method searches for $\lambda(0)$ that satisfies
\[ \epsilon := x(t_f) - x^0 = 0, \quad \lambda(0) = x^0. \] (11)

Define the solution of the problem as $\lambda^*(0)$. In the algorithm, first an initial guess of the solution $\lambda^*(0)$ is given as $\lambda^\delta$, and then update $\lambda(0)$ so that the error $\|\epsilon\|$ decreases. The difficulty of this method is that the error cannot be reduced to 0 if the initial guess $\lambda^\delta$ is far from $\lambda^*(0)$. One of the way to mitigate this difficulty is to apply the continuation method. In the continuation method, a relatively easy optimal control problem is solved first. Then, the cost function is slightly changed towards the original problem, and solve the changed problem by using the previous solution as the initial guess. Since the previous solution is expected to be close to the solution of the current problem, the shooting method successfully finds the solution with the given initial guess. Iterating this procedure and finally we can obtain the solution of the original problem. There are several continuation methods for solving an optimal control problem [13]. However, there is no continuation method that is effective for all problems, and it is unclear how the change of the cost function affects the closeness of the solutions before and after the deformation in those conventional methods.

This paper proposes a new continuation method based on a shooting method and analyses the relation between the variation of the continuation parameter and the deviation of the initial estimate from the solution. This relation gives a clear guidance on how to change the continuation parameter during the algorithm for stably solving relatively difficult problems. The next section explains the proposed method.

4. Proposed method

The proposed method solves the problem iteratively by successively transforming the problem from a relatively easy one to the original problem, in the same way as the existing methods. The characteristic feature of the proposed method is that the cost function is locally deformed during the iterations. Define the cost function of a relatively easy problem as
\[ J_E := \int_0^{t_f} \ell^E(x(t), u(t), t) \, dt, \] (12)
and define the cost function of the original problem as
\[ J_O := \int_0^{t_f} \ell^O(x(t), u(t), t) \, dt, \] (13)
with $\ell^E(x(t), u(t), t) \in \mathbb{R}_+$ and $\ell^O(x(t), u(t), t) \in \mathbb{R}_+$. Basically, the previous continuation method gradually changes the cost function by transforming the continuation parameter $c$ from 0 to 1, where the cost function

\[ \ell^c(x(c+h), u(c+h), c+h) = \ell^O(x(c+h), u(c+h), c+h), \]

and continuously differentiable at any $t \in [0, t_f]$. Figure 1 shows an example of the shape of $\ell^E(x(t), u(t), t)$, $\ell^O(x(t), u(t), t)$, and $\ell(x(t), u(t), t, c)$, where the horizontal axis is time and the vertical axis is the value of the function $\ell(x(t), u(t), t)$. The black solid line shows $\ell^E(x(t), u(t), t)$, the black dotted line shows $\ell^O(x(t), u(t), t)$, and the red dash-dotted line shows $\ell(x(t), u(t), t, c)$. Note that $J_c = J_E$ when $c \geq t_f$ and $J_c = J_O$ when $c \leq -h$. Our new continuation method transforms the cost function locally by changing the value of $c$. Note that we assume that each optimal control problem transformed iteratively has a solution. In the following, we define the solution of the optimal control problem with the cost function (15) as $x^E(t)$, $\lambda^E(t)$, and $u^E(t)$ for $t \in [0, t_f]$. The next section explains a shooting method used in the proposed continuation method. In the shooting method, the condition (11) is modified.

4.1. Modified shooting method

Define a Hamiltonian function as
\[ H_c(x(t), \lambda(t), u(t), t) \]
\[ = l(x(t), u(t), t, c) + \lambda^T (f(x(t)) + g(x(t))u(t)), \]

for the optimal control problem defined in the previous section. Then, from Pontryagin's minimum principle, \( x(t), \lambda(t), \) and \( u(t) \) minimizing \( J_c \) satisfy

\[
\begin{aligned}
\frac{d\xi}{dt} &= \left( \begin{array}{c}
\frac{\partial H_c(x, \lambda, u, t)}{\partial x} \\
\frac{\partial H_c(x, \lambda, u, t)}{\partial \lambda}
\end{array} \right) \bigg|_{u = f_c(x, \lambda, t)} =: F_c(\xi, t),
\end{aligned}
\]

(3) and (4),

with the function \( f_c^\xi : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^m \) derived from

\[ u^s(t) = \arg\min_{u(t)} H_c(x(t), \lambda(t), u(t), t). \] (19)

Here we modify the shooting method in Section 3.2 as in the following definition.

**Definition 4.1:** Modified shooting method: Find \( \lambda(0) \) and \( \lambda(t_f) \) that satisfy

\[ \|\epsilon_{\lambda}(t_{mid})\| = 0, \] (20)

for the error \( \epsilon_{\lambda}(t_{mid}) \) defined as

\[ \epsilon_{\lambda}(t_{mid}) := \int_0^{t_{mid}} F_c(\xi_{c+}(t), t) dt + \xi(0) \]

\[ - \left( \int_{t_f}^{t_{mid}} F_c(\xi_{c-}(t), t) dt + \xi(t_f) \right), \] (21)

with

\[ \xi(0) = \left( \begin{array}{c} x^0 \\ \lambda(0) \end{array} \right), \quad \xi(t_f) = \left( \begin{array}{c} x^f \\ \lambda(t_f) \end{array} \right). \] (22)

Here \( \xi_{c+}(t) \) and \( \xi_{c-}(t) \) are defined as

\[ \xi_{c+}(t) := \int_0^t F_c(\xi_{c+}(s), s) ds + \xi(0), \] (23)

\[ \xi_{c-}(t) := \int_{t_f}^t F_c(\xi_{c-}(s), s) ds + \xi(t_f). \] (24)

The difference between this method and the shooting method introduced in Section 3.2 is that there are two parameters to search for, not only \( \lambda(0) \) but also \( \lambda(t_f) \), and the boundary conditions are the continuity of \( x(t) \) and \( \lambda(t) \) at \( t = t_{mid} \). When the initial guess of the solution is different from the solution, \( \|\epsilon_{\lambda}(t_{mid})\| \) takes a value greater than zero, and if the norm of the error is sufficiently small for the initial guess, the guess is expected to be close enough to the solution. We define the error for the initial guess as “initial error.”

**Definition 4.2:** Define the initial guess of \( \lambda(0) \) and \( \lambda(t_f) \) as \( \lambda^0_0 \) and \( \lambda^0_{t_f} \), respectively. The corresponding initial error \( \epsilon_{\lambda}^I(t_{mid}) \) is defined as

\[ \epsilon_{\lambda}(t_{mid}) \] given by (21) with \( \xi(0) \)

In the next section, we explain the new continuation method and provides a clear relation between the variation of the continuation parameter \( c \) and the initial error \( \|\epsilon_{\lambda}^I(t_{mid})\| \).

### 4.2. Locally deforming continuation method

The idea of the locally deforming continuation method is to transform the cost function locally. In the existing methods, the cost function is deformed by changing the continuation parameter \( c \) from 0 to 1 as in Figure 2. On the other hand, the locally deforming continuation method locally changes the cost function by changing \( c \) from \( t_f \) to \( -h \), where the function \( l(x(t), u(t), t, c) \) changes as shown in Figure 3. Both Figures 2 and 3 are depicted in the same way as in Figure 1. The algorithm of the locally deforming continuation method with the modified shooting method is shown in Algorithm 1. Here \( t_{mid} \) is set as

\[ t_{mid} = f_{mid}(c), \] (25)

\[ f_{mid}(c) = \begin{cases} t_f & c \geq t_f - h, \\ c + h & t_f - h > c > -h, \\ 0 & -h \geq c. \end{cases} \] (26)

The parameters \( t_f, x^0, x^f, h, \delta \) are given before executing the algorithm and assume that \( x^h(t), u^h(t), \) and

![Figure 2. Transition of the function \((1-c)^2 + \delta^2\) in the algorithm of the existing continuation method for a given \( x(t) \) and \( u(t) \).](image-url)

![Figure 3. Transition of the function \(l(t)\) in the algorithm of the locally deforming continuation method for a given \( x(t) \) and \( u(t) \).](image-url)
$\lambda^I(t)$ are obtained. Here $h$ and $\delta$ are scalars satisfying $0 < \delta \leq h$, and $\delta$ represents the variation of the continuation parameter $c$.

**Algorithm 1** locally deforming continuation method

**Input:** $t_f, x^0, x^1, h, \delta, x^H(t), u^H(t), \lambda^H(t)$

**Output:** $x^0(t), u^0(t)$

1: $c \leftarrow t_f - \delta$
2: loop

3: $t_{mid} \leftarrow f_{mid}(c)$
4: $\lambda^f_0 \leftarrow \lambda^f_{c+}(0), \lambda^f_c \leftarrow \lambda^f_{c+}(t_f)$
5: Find $\lambda^f_0(0), \lambda^f_c(t_f)$ by solving TPBVP (18) from the initial guess
6: if $c \leq -h$ then
7: return $x^c(t), u^c(t)$
8: else
9: $c \leftarrow c - \delta$
10: end if
11: end loop

In the next section, we analyse the relation between $\delta$ and the initial error $\epsilon^I_h(t_{mid})$ with respect to the proposed algorithm.

### 4.3. Analysis of the relation between $\delta$ and $\epsilon^I_h(t_{mid})$

First, we prove the following lemmas for $\epsilon^I_h(t_{mid})$ and then we introduce the relation between $\delta$ and $\epsilon^I_h(t_{mid})$.

**Lemma 4.1:** Select $c$ so that $t_f - h \leq c \leq t_f$ and $t_{mid} = f_{mid}(c)$. Assume that $u^H(t), \lambda^I(t)$, and $\lambda^I_{c+}(t_f)$ are obtained. If we give $\lambda^f_0(0)$ and $\lambda^f_c(t_f)$ respectively, the initial error $\epsilon^I_{t_{mid}}(t_{mid})$ for $I_{c-\delta}$ satisfies

$$
\epsilon^I_{c-\delta}(t_{mid}) = \int_{c-\delta}^{t_f} F_{c-\delta}(\xi_{c-\delta,+}(t), t) - F_c(\xi_{c,+}(t), t) \, dt.
$$

**Proof:** It follows from (16) that

$$l(x(t), u(t), t, c) = l(x(t), u(t), t, c - \delta),$$

holds for $t \in [0, c - \delta]$, therefore

$$F_c(\xi_{c,+}(t), t) = F_{c-\delta}(\xi_{c-\delta,+}(t), t),$$

also holds for $t \in [0, c - \delta]$. Note that

$$\xi^f = \int_{0}^{t_f} F_c(\xi_{c,+}(t), t) \, dt + \xi^0,$$

with $\xi^f := (x^T, \lambda^I(t_f))^T$ and $\xi^0 := (x^0_T, \lambda^I(0))^T$ holds for $\lambda^I(0)$ and $\lambda^I_{c+}(t_f)$. Hence, for $\epsilon^I_{c-\delta}(t_{mid})$, (29) and (30) lead

$$\epsilon^I_{t_{mid}}(t_{mid}) = \int_{0}^{t_f} F_{c-\delta}(\xi_{c-\delta,+}(t), t) \, dt + \xi^0$$

and this completes the proof.

**Lemma 4.2:** Select $c$ so that $\delta \leq c \leq t_f - h$ and $t_{mid} = f_{mid}(c)$. Assume that $u^H(t), \lambda^I(t)$, and $\lambda^I_{c+}(t_f)$ are obtained. If we give $\lambda^f_0(0)$, $\lambda^f_0(0)$, and $\lambda^f_c(t_f)$ respectively, the initial error $\epsilon^I_{t_{mid}}(t_{mid})$ for $I_{c-\delta}$ satisfies

$$\epsilon^I_{t_{mid}}(t_{mid}) = \int_{c-\delta}^{t_f} F_{c-\delta}(\xi_{c-\delta,+}(t), t) - F_c(\xi_{c,+}(t), t) \, dt.$$  

**Proof:** The proof of the Lemma 4.2 is similar to the one of Lemma 4.1. It follows from (16) that

$$l(x(t), u(t), t, c) = l(x(t), u(t), t, c - \delta),$$

holds for $t \in [0, c - \delta]$ and $t \in [c + h, t_f]$, therefore

$$F_c(\xi_{c,+}(t), t) = F_{c-\delta}(\xi_{c-\delta,+}(t), t),$$

also holds for $t \in [0, c - \delta]$ and

$$F_c(\xi_{c,+}(t), t) = F_{c-\delta}(\xi_{c-\delta,+}(t), t),$$

holds for $t \in [c + h, t_f]$. Note that

$$\xi(t) = \int_{0}^{t} F_c(\xi_{c,+}(t), t) \, dt + \xi^0$$

and

$$\xi(t) = \int_{0}^{t} F_c(\xi_{c,+}(t), t) \, dt + \xi^f,$$
with \( \xi^f = (x^f, \lambda^f(t)^T)^T \) and \( \xi^0 = (x^0^T, \lambda^0(0)^T)^T \) holds for \( \lambda^J(0) \) and \( \lambda^J(t) \) from the assumption. Hence, it follows from (33)–(35) that

\[
\epsilon^1_{J_c-\delta}(t_{\text{mid}}) = \int_0^{c+h} F_c - \delta(\xi_{c-\delta}, (t), t) \, dt + \xi^0 \\
- \left( \int_0^{c+h} F_c - \delta(\xi_{c-\delta}, (t), t) \, dt + \xi^f \right) \tag{36}
\]

\[
= \int_0^{c+h} F_c - \delta(\xi_{c-\delta}, (t), t) \, dt + \xi^0 \\
- \left( \int_0^{c+h} F_c(\xi_{c-\delta}, (t), t) \, dt + \xi^f \right) \\
= \int_0^{c+h} F_c - \delta(\xi_{c-\delta}, (t), t) \, dt \\
+ \int_0^{c-h} F_c(\xi_{c-\delta}, (t), t) \, dt \\
- \left( \int_0^{c+h} F_c(\xi_{c-\delta}, (t), t) \, dt + \xi^f \right) \\
= \int_0^{c+h} F_c - \delta(\xi_{c-\delta}, (t), t) \, dt \\
- \int_0^{c+h} F_c(\xi_{c-\delta}, (t), t) \, dt,
\]

and this completes the proof.

**Lemma 4.3:** Select \( c \) so that \(-h \leq c \leq \delta \) and \( t_{\text{mid}} = f_{\text{mid}}(c) \). Assume that \( l^J(t), x^J(t), \) and \( \lambda^J(t) \) are obtained. If we give \( \lambda^0_\delta \) and \( \lambda^\delta(t) \) as \( \lambda^J(0) \) and \( \lambda^J(t) \) respectively, the initial error \( \epsilon^1_{J_c-\delta}(t_{\text{mid}}) \) for \( I_{c-\delta} \) satisfies

\[
\epsilon^1_{J_c-\delta}(t_{\text{mid}}) = \int_0^{c+h} F_c - \delta(\xi_{c-\delta}, (t), t) \, dt - F_c(\xi_{c-\delta}, (t), t) \, dt. \tag{36}
\]

We omit the proof since it is proved in the same way as Lemma 4.1. In addition to Lemmas 4.1–4.3, under certain conditions, the following theorem holds.

**Theorem 4.1:** Assume that \( l(x), u(t), t, c) = l(u(t), t, c) \). Then the costate system of (18) can be written as

\[
\frac{d\lambda(t)}{dt} = -\frac{\partial H_c(\xi(t), u(t), t)}{\partial x(t)} \bigg|_{u(t) = f_{\text{mid}}(\xi(t), t)} \tag{37}
\]

\[
= \tilde{f}(\xi(t)) + \tilde{g}(\xi(t))f^J_{\text{mid}}(\xi(t), t),
\]

with \( \tilde{f} : \mathbb{R}^{2n} \to \mathbb{R}^n, \tilde{g} : \mathbb{R}^{2n} \to \mathbb{R}^{n \times m} \), and

\[
\frac{dx(t)}{dt} = f(x(t)) + g(x(t))f^J_{\text{mid}}(\xi(t), t). \tag{38}
\]

Assume that there are constants \( \alpha_1, \alpha_2, \beta_{1,i}, \) and \( \beta_{2,i}, \) \( i = 1, \ldots, m \) such that

\[
\|f(x_1) - f(x_2)\| \leq \alpha_1 \|x_1 - x_2\|, \tag{39}
\]

\[
\|\tilde{f}(\xi_1) - \tilde{f}(\xi_2)\| \leq \alpha_2 \|\xi_1 - \xi_2\|, \tag{40}
\]

\[
\|g_i(x^J(\tau))\| \leq \beta_{1,i}, \tag{41}
\]

\[
\|\tilde{g}_i(\xi^J(\tau))\| \leq \beta_{2,i}, \tag{42}
\]

where \( x_1, x_2 \in \mathbb{R}^n, \xi_1, \xi_2 \in \mathbb{R}^n, \tau \in [\max(c - \delta, 0), \min(c + h, t_1)], g_i \) and \( \tilde{g}_i \) are the column vectors of \( g = (g_1, \ldots, g_m) \) and \( \tilde{g} = (\tilde{g}_1, \ldots, \tilde{g}_m) \), and \( x^J(\tau) \) and \( \xi^J(\tau) \) are arbitrary vectors that satisfy (3), (37) and (38) for \( \tau \). Then, if we give \( \lambda^\delta_0 \) and \( \lambda^\delta(t) \) as \( \lambda^J(0) \) and \( \lambda^J(t) \) respectively, for \( t_{\text{mid}} = f_{\text{mid}}(c), c \in [-h, t_1] \), the following relation

\[
\|\epsilon^1_{J_c-\delta}(t_{\text{mid}})\| \leq \frac{2 \sum_{i=1}^{m} (\beta_{1,i} + \beta_{2,i}) u_{b,i} (\alpha(h + \delta) - 1)}{\alpha} \tag{43}
\]

with \( \alpha = \alpha_1 + \alpha_2 \) holds.

**Proof:** This proof refers to the proof of Theorem 2.1 in [14]. Consider the case \( \delta \leq c \leq t_1 - h \) and define \((x_{c+}, \lambda_{c+}^T)^T := \xi_{c+} \). Here \( \xi_{c+} \). For \( t \in [c - \delta, c + h] \),

\[
\frac{d}{dt} \|x_{c-\delta,+} - x_{c,+}\| + \frac{d}{dt} \|\lambda_{c-\delta,+} - \lambda_{c,+}\|
\leq \|\hat{x}_{c-\delta,+} - \hat{x}_{c,+}\| + \|\lambda_{c-\delta,+} - \lambda_{c,+}\|
\leq \|f(x_{c-\delta,+}) - f(x_{c,+})\| + \|f(\xi_{c-\delta,+}) - f(\xi_{c,+})\|
+ \|g(x_{c-\delta,+})f^J_{\text{mid}}(\xi_{c-\delta,+}, t) - g(x_{c,+})f^J_{\text{mid}}(\xi_{c,+}, t)\|
+ \|\tilde{g}(\xi_{c-\delta,+})f^J_{\text{mid}}(\xi_{c-\delta,+}, t) - \tilde{g}(\xi_{c,+})f^J_{\text{mid}}(\xi_{c,+}, t)\|
\leq \alpha_1 \|x_{c-\delta,+} - x_{c,+}\| + \alpha_2 \|\xi_{c-\delta,+} - \xi_{c,+}\|
+ \sum_{i=1}^{m} (\|g_i(x_{c-\delta,+})\| + \|g_i(x_{c,+})\|)
+ \|\tilde{g}_i(\xi_{c-\delta,+})\| + \|\tilde{g}_i(\xi_{c,+})\|) u_{b,i}
\leq \alpha_1 \|x_{c-\delta,+} - x_{c,+}\| + \alpha_2 \|\xi_{c-\delta,+} - \xi_{c,+}\|
+ 2 \sum_{i=1}^{m} (\beta_{1,i} + \beta_{2,i}) u_{b,i}
\leq (\alpha_1 + \alpha_2) \|x_{c-\delta,+} - x_{c,+}\| + \alpha_2 \|\lambda_{c-\delta,+} - \lambda_{c,+}\|
+ 2 \sum_{i=1}^{m} (\beta_{1,i} + \beta_{2,i}) u_{b,i}
\leq (\alpha_1 + \alpha_2) \|x_{c-\delta,+} - x_{c,+}\|
+ (\alpha_1 + \alpha_2) \|\lambda_{c-\delta,+} - \lambda_{c,+}\|
+ 2 \sum_{i=1}^{m} (\beta_{1,i} + \beta_{2,i}) u_{b,i}.
Hence the following relation
\[
\frac{d}{dt} \left[ x_{cδ} - x_{c+} \right] = (α_1 + α_2)x_{cδ} - x_{c+} \\
+ \frac{d}{dt} \left[ λ_{cδ} - λ_{c+} \right] \leq (α_1 + α_2)∥x_{cδ} - x_{c+}∥ \\
\leq 2 \sum (β_{1,i} + β_{2,i})u_{b,i} \tag{44}
\]
holds. By multiplying the integrating factor \(e^{-(α_1+α_2)t}\) to (44), we have
\[
\frac{d}{dt} \left( e^{-(α_1+α_2)t} \left[ x_{cδ} - x_{c+} + λ_{cδ} - λ_{c+} \right] \right) \leq 2 e^{-(α_1+α_2)t} \sum (β_{1,i} + β_{2,i})u_{b,i} \tag{45}
\]
From Lemma 4.2, the initial error is gained by integrating \(\xi_{cδ} - \xi_{c+}\) with (37) and (38) from \(c - δ\) to \(c + h\). The integration of (45) from \(c - δ\) to \(c + h\) leads to
\[
e^{-α(c+h)} \left[ x_{cδ} - x_{c+} + λ_{cδ} - λ_{c+} \right] \leq 2 \sum (β_{1,i} + β_{2,i})u_{b,i} \frac{e^{α(c+h)}}{α}. \tag{46}
\]
Since \(x_{cδ} - x_{c+} + λ_{cδ} - λ_{c+}\) hold from (33), it follows from (46) that
\[
\frac{d}{dt} \left[ x_{cδ} - x_{c+} + λ_{cδ} - λ_{c+} \right] \leq 2 \sum (β_{1,i} + β_{2,i})u_{b,i} \frac{e^{α(c+h)}}{α}.
\]
and the relation
\[
\frac{d}{dt} \left[ x_{cδ} - x_{c+} + λ_{cδ} - λ_{c+} \right] \leq 2 \sum (β_{1,i} + β_{2,i})u_{b,i} \frac{e^{α(c+h)}}{α} - 1,
\]
holds since \(t_f - (c - δ) \leq h + δ\), and for the case \(-h \leq (c - δ)\)
\[
\|\epsilon_{cδ}^i(c + h)\| \leq 2 \sum (β_{1,i} + β_{2,i})u_{b,i} \frac{e^{α(c+h)}}{α} - 1,
\]
holds since \(c + h \leq h + δ\). This completes the proof.

Remark 4.1: Theorem 4.1 provides a clear guideline for adjusting the parameters \(δ\) and \(h\) in solving problems with the locally deforming continuation method. For example, the theorem shows that both \(h\) and \(δ\) should be decreased for reducing the initial error, that is, reducing \(h + δ\) means the guess of the solution in each iteration becomes close to the solution and thus makes the algorithm more likely to succeed. Hence, \(h\) should be chosen so that it is as large as \(δ\), and vice versa. In addition, if \(h + δ\) is sufficiently small, the upper bound of the initial error \(\epsilon_{cδ}^i(f_{\text{final}}(c))\) also expected to change linearly.

The next section shows the effectiveness of the proposed method through numerical examples.

5. Numerical example

In this example, we apply an \(L^1/L^2\)-optimal control \([12]\) to a two-wheeled rover depicted in Figure 4, where the horizontal axis is \(X\), the vertical axis is \(Y\), \(x_1\) is the angle of the rover, and \(x_2\) and \(x_3\) are \(x\)-position and \(y\)-position of the rover, respectively. The system of the rover is denoted by
\[
\frac{dx(t)}{dt} = g(x)u = \begin{pmatrix} 1 & 0 \\ 0 & \cos(x_1(t)) \\ 0 & \sin(x_1(t)) \end{pmatrix} \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}, \tag{47}
\]
and the input is restricted by \(u_{b,1} = u_{b,2} = 5\). Define an optimal control problem with the cost function
\[
J_0 = \int_0^1 \frac{1}{2} \sum_{i=1}^2 (u_i(t)^2) + 2|u_1(t)| dt \tag{48}
\]

Figure 4. Two-wheeled rover.
and a boundary condition
\[
x(0) = (0, 0, 0)^T, \quad x(2) = \left( \frac{\pi}{5}, 1, 1 \right)^T \in \mathbb{R}^3.
\] (49)

The resultant trajectory of this optimal control problem is known as a hands-off trajectory [15], where the input \( u_1(t) \) takes zero in a large part of the control sequence.

5.1. Solve the problem without continuation method

In this example, we solve the problem with the shooting method without using the continuation method. Since the shooting method requires an initial guess of the solution, we first solve an \( L^2 \)-optimal control problem with the cost function
\[
J_E = \int_0^2 \frac{1}{2} \sum_{i=1}^{2} (u_i(t))^2 \, dt,
\] (50)
and set its solution as an initial guess for solving the \( L^1/L^2 \)-optimal control problem. Figures 5 and 6 show the generated trajectories and inputs respectively. In Figure 5, the first row figure shows \( x_1(t) \), the second figure shows \( x_2(t) \), and the third figure shows \( x_3(t) \). The horizontal axis is time and the vertical axis is the state. The plus sign markers show the boundary conditions of each state. In Figure 6, the blue dash-dotted line is \( u_1(t) \) and the blue dashed line is \( u_2(t) \), where the horizontal axis is time and the vertical axis is the input. As this result shows, the generated trajectories do not satisfy the boundary conditions, that is, the shooting method fails to solve the \( L^1/L^2 \)-optimal control problem.

Figure 5. Generated trajectory without continuation method.

Figure 6. Input of the generated trajectory.

In the next problem, we use the locally deforming continuation method for solving the \( L^1/L^2 \)-optimal control problem.

5.2. Solve the problem with the locally deforming continuation method

In this example, we use the locally deforming continuation method for solving the \( L^1/L^2 \)-optimal control problem. In the method, \( J_E \) and \( J_O \) are defined as (48) and (50), and \( J_C \) is defined by (15) and (16) with
\[
J_C(x(t), u(t), t) = \frac{1}{2} \sum_{i=1}^{2} (u_i(t))^2 + f_w(t)|u_1(t)|,
\] (51)
\[
f_w(t) = \cos \left( \frac{\pi (t - c)}{h} \right) + 1.
\] (52)
To confirm the effectiveness of our method, we performed Algorithm 1 under several combinations of \( h \) and \( \delta \) listed in Table 1. Here we define \( h_0 = 0.500 \) and \( \delta_0 = 0.125 \). Figure 7 shows the log–log plot of the relation between the mean of the initial errors of each case and \( h + \delta \). The horizontal axis is \( h + \delta \), and the vertical axis is the mean of the Euclidean norm of the initial errors of the shooting method during Algorithm 1. The green line shows the line of \( y = 0.2x \), where \( y \) is the mean of the initial errors and \( x \) is \( h + \delta \). From Figure 7, we can see that the error decreases linearly with respect to \( h + \delta \) if \( h + \delta \) is small enough. Figures 8 and 9 show the generated trajectory and input in Case 4 in the same way as Figures 5 and 6. As can be seen from Figures 8–9, the proposed method successfully generates trajectories satisfying the boundary conditions and the generated input \( u_1(t) \) takes zero in large part. These results show the locally deforming continuation method is effective.

Table 1. Combinations of \( h \) and \( \delta \).

| Case | 1 | 2 | 3 | 4 |
|------|---|---|---|---|
| \( h \) | \( h_0 \) | \( h_0 \) | \( h_0 \) | \( h_0 \) |
| \( \delta \) | \( \delta_0 \) | \( \delta_0 \) | \( \delta_0 \) | \( \delta_0 \) |

Figure 7. The log–log plot of the mean of the errors in each case listed in Table 1.
for solving a relatively difficult optimal control problem like an $L^1/L^2$-optimal control problem.

### 5.3. Comparison between the proposed method and the conventional method

In the example, we compare the initial errors of the conventional method and the proposed method in five cases, in which the number of iterations of each continuation method is the same in each case. Here the boundary condition at $t_f = 2$ is set to $x(2) = (1,1,1)^T$. The conventional method with the cost function (14) is solved by the shooting method introduced in Section 3.2, and the initial error in this method is defined as follows.

**Definition 5.1:** Define the initial guess of $\lambda^*(0)$ as $\lambda_0^0$. The corresponding initial error $\epsilon_{x}^1$ is defined as

$$\epsilon_{x}^1 := x(t_f) - x^f \text{ with } \xi(0) = \begin{pmatrix} x_0^0 \\ \lambda_0^0 \end{pmatrix} \text{ s.t. } (10) \quad (53)$$

In addition, the continuation parameter $c$ in (14) is changed in each iteration as $c \leftarrow c - \delta$ from 0 towards 1.

In Figure 10, the upper figure shows the means of the initial errors of the conventional method, and the lower figure shows those of the proposed one, where the cases that succeeded to find the solutions are marked in blue and the failed cases are coloured in red. At each case, $\delta$ and $h + \delta$ take the values as in horizontal axes, where $\delta_0 = 1/2$, $h_0 = 2$, $b_0 = 2$ and $h = \delta$. The vertical axes are the mean of the Euclidean norm of the initial errors. The iteration number of each case is listed in Table 2. In the conventional method, if the iteration number is increased, the estimate of the solution is expected to become close to the solution. However, as shown in the upper figure of Figure 10, the means of the initial errors in Cases 1 and 2 are almost the same in the conventional methods. On the other hand, the means of the initial errors of the proposed method decrease as in the lower figure of Figure 10, as mentioned in Remark 4.1. Though there is no difference in the success or failure of searching the solution, this example shows that the proposed method is superior to the conventional method in terms that it provides a guideline on how to change the continuation parameter for successfully obtaining the solution.

### 6. Conclusion

In this study, we proposed a new continuation method based on the modified shooting method for optimal control problems. We clarify the relationship between the continuation parameter and the proximity of the solutions before and after a deformation in terms of the error in the shooting method, which has been unclear in the conventional continuation methods. This relation is useful in solving the optimal control problem in the sense that it provides a guideline on how to change the continuation parameter when applying the locally deforming continuation method. Our future work is to work on more difficult problems like $L^0$-optimal control problems [16] or CLOT optimal control problems [17] by using the locally deforming continuation method.

**Disclosure statement**

No potential conflict of interest was reported by the author(s).
Notes on contributors

Kiyoshi Hamada received his Bachelor of Engineering and Master of Engineering degrees from Kyoto University, Kyoto, Japan, in 1994, 1996, and 2001, 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.

Ichiro Maruta 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, where he is currently a Ph.D. student.

Kenji Fujimoto 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, Kyoto 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 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.