Efficient Motion Planning for Fast Motion of Path Tracking Task with Constant Hand Speed using Redundant Manipulator

Kousuke Okabe and Yasumichi Aiyama

Abstract—In this paper, we propose an efficient motion planning method for fast motion of path tracking task with constant hand speed using redundant manipulator. The proposed method performs path finding on a state space of the manipulator instead of a motion planning. We previously proposed a motion planning method for fast motion of path tracking task with constant hand speed. However, this motion planning method has required long calculation time. Therefore, we propose an efficient motion planning method that based on partitioning the hand path into difficult sections and easy sections to do different path finding method on each sections. The difficult sections are the sections that difficult to operate fast motion within joint torque limits and joint velocity limits. The easy sections are the sections that easy to operate fast motion within the limits. In difficult sections, we look for paths to avoid limits on the state space using the previous method. In easy sections, we derive the connective paths between planned paths on the difficult sections. The 3-link planar manipulator has been used to show the efficacy of proposed method.

I. INTRODUCTION

In industrial field, manipulators are slower than other industrial equipments. Industrial manipulators are required to operate fast for cycle time. However, fast motion is hard in case of path tracking task with constant hand speed such as searing task. That because manipulators cannot operate fast on a curved path with small radius. Thus, the cycle time becomes long in case of path tracking task with constant hand speed.

In order to operate manipulator fast, J.E.Bobrow et al. [1] proposed the time-optimal path tracking control to project torque limits to phase-plane using the dynamics equation of a non-redundant manipulator, and finding motion without exceeding torque limits on the phase-plane. S.Ma et al. [2] applied the Bobrow’s method to redundant manipulator and proposed the time-optimal path tracking control of redundant manipulator. When Bobrow’s method applied to redundant manipulator, there were a problem that the torque limits could not project to phase-plane. S.Ma et al. solved the problem by using linear programming technique.

Previously, we proposed motion planning method [3] to speed up the path tracking task with constant hand speed using the redundancy of manipulator that has one redundant degrees of freedom (RDOF). The proposed method used path finding algorithm based on full search. As a result, the proposed method has required a long calculation time in case of long general paths. To reduce calculation time, we propose an efficient motion planning method of path tracking task with constant hand speed using hand path partitioning. The hand path is partitioned to difficult sections, that are difficult to move fast hand speed, and easy sections, that are easy to move fast hand speed. In difficult sections, we do motion planning using method of [3]. In easy sections, the motion has been planned using easy algorithm. The algorithm is working on connecting the motion between difficult sections.

II. FAST MOTION OF PATH TRACKING TASK WITH CONSTANT HAND SPEED

A path tracking task with constant hand speed is the task that manipulator’s hand tracks a reference path and operates constant hand speed on all hand path. We show an example of this task in Fig.1. This is the task to apply bonding agent to frame of displays. A manipulator applies the bonding agent to two displays alternately. The manipulator can send out the bonding agent from the end of the hand at a specific rate. The manipulator applies the bonding agent to frame of displays to do the path tracking task with constant hand speed in order to put the bonding agent of constant volume. The purpose of this paper is to reduce a cycle time per display to alter motion faster. A main problem on fast motion of path tracking task with constant hand speed is that the maximum hand speed becomes slow on a part of the curved path with small radius. For this reason, a cycle time becomes long because the task requires to move constant hand speed on all hand path. The parts of the curved path with small radius does not have majority in a general hand path. Important parts on the motion planning are parts of the curved path with small radius. In this paper, we propose an efficient motion planning method that reduces a calculation time to do different motion planning on curved sections and straight sections.
III. PROBLEM FORMULATION

In our study, we consider about n-DOF manipulator that work on m = n1 dimensional work space. The manipulator has 1-RDOF. The joint angle vector is $q = [q_1, q_2, \cdots, q_n]^T \in \mathbb{R}^n$, and its time-differentials are $\dot{q} \in \mathbb{R}^n$, $\ddot{q} \in \mathbb{R}^n$. After this, $\dot{X}$ and $\dddot{X}$ represent first order time-differential and second order time-differential of $X$. The joint torque vector is $\tau = [\tau_1, \tau_2, \cdots, \tau_n]^T \in \mathbb{R}^n$. Dynamics equation of the manipulator derived from Lagrange’s equation is

$$\tau = M\ddot{q} + H + g + d$$  \hspace{0.5cm} (1)

Where, $M = M(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix and $H = H(q, \dot{q}) \in \mathbb{R}^n$ is the torque vector of Coriolis and centrifugal force, $g = g(q) \in \mathbb{R}^n$ is the torque vector of gravity, $d = d(q) \in \mathbb{R}^n$ is the torque vector of disturbance such as friction.

On the other hand, relationship of task space and joint space of manipulator are

$$r = P, \quad \dot{r} = J\dot{q}, \quad \dddot{r} = J\dddot{q} + J\dot{J}\ddot{q}$$  \hspace{0.5cm} (2)

Where, $r \in \mathbb{R}^m$ is the hand position vector, $P = P(q)$ is the forward kinematics function that consists m scalar functions, $J = J(q) = \frac{\partial P}{\partial q} \in \mathbb{R}^{m \times n}$ is the Jacobian matrix of the redundant manipulator, and $\dot{J} = J(q, \dot{q})$.

The hand path of manipulator is given by path function $f(s)$. The scalar parameter $s, (s_0 \leq s \leq s_1)$ is the position along the hand path. The $\dot{s}$ becomes zero because we are considering about the path tracking task with constant hand speed in this study. Thus, the hand path is given by (3).

$$r = f, \quad \dot{r} = f\dot{s}, \quad \dddot{r} = f'\dot{s}^2$$  \hspace{0.5cm} (3)

Where, $f = f(s)$, $f' = \frac{df}{ds}(s)$, and $f'' = \frac{d^2f}{ds^2}(s)$. $f(s_0)$ is a start position of the hand path, $f(s_1)$ is an end position of the hand path. Additionally, the hand position $r$ given by the path function does not impose restriction to motion by redundancy.

The manipulator has 1-DOF in internal mechanism that does not affect the hand position. Thus, we consider necessary parameters in order to indicate motion by redundancy. We define motion by redundancy to use redundant pose $R_x \in \mathbb{R}$, redundant velocity $R_v \in \mathbb{R}$, and redundant acceleration $R_a \in \mathbb{R}$ as

$$q = P^{-1}(r, R_x)$$  \hspace{0.5cm} (4)

$$\dot{q} = J^+\dot{r} + U R_v$$  \hspace{0.5cm} (5)

$$\ddot{q} = J^+(\dddot{r} - J\dot{q}) + UR_a$$  \hspace{0.5cm} (6)

$$R_x = \Sigma q$$  \hspace{0.5cm} (7)

Where, $\Sigma \in \mathbb{R}^{1 \times n}$ is the optional row vector, $P^{-1}(r, R_x)$ is the inverse kinematics function, $J^+ \in \mathbb{R}^{n \times m}$ is the Pseudo inverse matrix of $J$, and $U \in \mathbb{R}^n$ is the normalized basis vector of the orthogonal complement of the space spanned by $J$. The relationships of $R_x, R_v, \text{and} R_a$ are following. These equations have been derived on the [3].

$$\dot{R}_x = \Sigma^+ \dot{r} + \Sigma U R_v$$  \hspace{0.5cm} (8)

$$\dot{R}_v = R_a + U^T \dddot{q}$$  \hspace{0.5cm} (9)

Therefore, the state equation becomes (10). The states are the redundant pose $R_x$, the redundant velocity $R_v$, the position along the hand path $s$, and the constant hand speed $\dot{s}$. The input is the redundant acceleration $R_a$. In this study, we regard state space as $s - R_x - R_v$ space by determining the constant hand speed $\dot{s}$.

$$\begin{bmatrix} d \tau_x \cr \tau_x \end{bmatrix} = \begin{bmatrix} 0 \cr 0 \end{bmatrix} \begin{bmatrix} \Sigma U \cr \dot{R}_x \end{bmatrix} + \begin{bmatrix} 0 \cr 1 \end{bmatrix} R_a + H_R$$  \hspace{0.5cm} (10)

Where,

$$H_R = \begin{bmatrix} \Sigma J^+ f'\dot{s} \\
U^T J^+ f'\dot{s} \end{bmatrix}$$

If matrix $[J_T, \Sigma T]^T$ is not regular matrix, $\Sigma U$ becomes zero and the redundant pose $R_x$ becomes unrelated to $R_v$ and $R_a$. Hence, the $\Sigma$ is required that the matrix $[J_T, \Sigma T]^T$ is regular matrix.

The manipulator has joint torque limits $\Omega_t$ and joint velocity limits $\Omega_v$, as following.

$$\Omega_t = \begin{bmatrix} \tau - \tau_{lim} \leq \tau \leq \tau_{lim} \end{bmatrix}$$  \hspace{0.5cm} (11)

$$\Omega_v = \begin{bmatrix} \dot{q} - \dot{q}_{lim} \leq \dot{q} \leq \dot{q}_{lim} \end{bmatrix}$$  \hspace{0.5cm} (12)

Where, $\tau_{lim} = [\tau_{lim1}, \tau_{lim2}, \cdots, \tau_{limn}]^T \in \mathbb{R}^n \geq 0$, $\dot{q}_{lim} = [\dot{q}_{lim1}, \dot{q}_{lim2}, \cdots, \dot{q}_{limn}]^T \in \mathbb{R}^n \geq 0$. These limits can project to state space. The joint torque limits become a redundant acceleration limit. The joint velocity limits become a redundant velocity limit. In this study, an area of filling these limits are called operational area, and an area of besides the operational area are called prohibited area. The redundant acceleration limit and the redundant velocity limit are

$$F_x(s, \dot{s}, R_x, R_v, \tau_{lim}) \leq R_a \leq G_x(s, \dot{s}, R_x, R_v, \tau_{lim})$$  \hspace{0.5cm} (13)

$$F_v(s, \dot{s}, R_x, \dot{q}_{lim}) \leq R_v \leq G_v(s, \dot{s}, R_x, \dot{q}_{lim})$$  \hspace{0.5cm} (14)

The purpose of our study is off-line motion planning in order to get fast motion of path tracking constant hand speed task. The motion planning derives fast motion to use redundancy within joint torque limits $\Omega_t$ and joint velocity limits $\Omega_v$. This paper aims to reduce the calculation time without slowing maximum constant hand speed because the method of [3] requires large calculation time.

IV. MOTION PLANNING

A. Motion Planning Algorithm

The previous method derived a motion to be path finding on the state space in stead of motion planning. However, The previous method required long calculation time in case of planning to general path. In this paper, we partition the hand path to easy sections and difficult sections based on curvature of the hand path. The easy sections are sections that are easy to operate fast hand speed within the limits. The difficult sections are the sections that are difficult to operate fast hand speed. The easy sections and the difficult sections don’t continue and don’t overlap. We plan the path on the state space efficiently to use different path finding methods.
The first path finding method on the difficult sections looks for the path which avoids each limits on the state space. The second path finding method on the easy sections derives the connective paths between planned path on difficult sections. These path finding methods derive the path without entering the prohibited area on one of the hand constant speed. The states $(R_s, R_v)$ on the section switching points are called edge states. The proposed method starts path finding methods from proper slow $\dot{s}$. By speed-up the constant hand speed gradually, the area of edge states that paths have been found is reduced. Where we do path finding from the area of edge states found paths in order to reduce calculation time. The fastest motion of path tracking task with constant hand speed is decided with the method of path which remains to the end.

The motion planning algorithm is shown in Fig.2. The explanation of the function in Fig.2 are as following:

**Arranging initial edge states and setting an initial constant hand speed**

The constant hand speed $\dot{s}$ is set to a proper slow hand speed, and initial edge states are arranged on all initial hand position of difficult sections. Distances among the initial edge states are $w = [w_{R_s}, w_{R_v}]^T$.

**Path finding on all difficult sections**

The path finding on the difficult sections looks for paths from each edge states on all difficult sections. At last, the edge states on the initial and terminal hand positions of difficult sections are renewed to the edge states derived by the path finding on difficult sections.

**Path finding on all easy sections**

The path finding on easy sections is performed from near initial hand position $s = s_0$ to near terminal hand position $s = s_1$. The path finding on the easy sections is performed to all combinations of initial and terminal edge states derived by the path finding method on the difficult section. Where, this path finding is not performed to combinations with initial edge states that cannot connect a path from the edge states on the initial hand position $s = s_0$.

**Stepping up the hand constant speed**

The constant hand speed $\dot{s}$ is renewed to fast constant hand speed with a small rate $\delta \dot{s}$. As a result, the hand constant speed becomes $\dot{s} = \dot{s} + \delta \dot{s}$.

**Addition to edge state**

The edge states are added to 8-neighbor of edge states existing a path on state space of $\dot{s} - \delta \dot{s}$. Then, the distances of edge states $w$ are reduced to $w = \frac{1}{\tau} w$.

**Generating a motion**

We decide a maximum hand constant speed with $\dot{s} - \delta \dot{s}$. The fast motion of path tracking constant hand speed task is generated by tracing planned path from the terminal hand position $s = s_1$ to the initial hand position $s = s_0$.

**B. Path Finding on State Space**

1) In case of difficult sections: We use the path finding method of [3] for path finding on difficult sections. Where, an evaluation function is as follows.

$$E = E_t + E_v$$

$$E_t = 1 - e^{-\min(G_{t(ke)} - R_i(ke), R_i(ke) - F_{t(ke)})}$$

$$E_v = 1 - e^{-\min(G_{v(ke)} - F_{t(ke)})}$$

The path finding on the difficult section derives the path that does not enter the prohibited area, and the path finding is started from initial edge states $(R_{i0}, R_{t0})$ on the initial hand position of the difficult section. A path on the state space is decided by redundant acceleration $R_a$ of (10). Thus, we decide a redundant acceleration with evaluation function $E$ on each sampling time. The paths on the difficult sections don’t branch or join, because the paths derive from each initial edge state by the path finding.

2) In case of easy sections: The path finding method on the easy sections derives the path which connects the initial edge state $(R_{i0}, R_{t0})$ and the terminal edge state $(R_{i1}, R_{t1})$ on the easy section. The initial edge state and the terminal edge state decide by path on the front and the rear difficult sections. The path which connects the initial edge state and the terminal edge state cannot connect optionally, because a path follow the state equation of (10). Accordingly, We do the path finding on easy sections to suppose switching surface that paths finally arrive at the terminal edge state, such as Fig.3-c.

The switching surface can be obtained in two steps. First, we derive an accelerative path and a decelerative path such
as the paths a and b in Fig.3. The accelerative path is a path operated by maximum redundant acceleration $R_a = G_I$ from the terminal edge state $(R_{t1}, R_{t1})$ toward initial hand position on the easy section. The decelerative path is a path operated by minimum redundant acceleration $R_d = F_I$ from the terminal edge state $(R_{t1}, R_{t1})$ toward initial hand position on the easy section. Second, we make a surface to derive many paths that are operated by redundant acceleration to be minimum joint torque vector norm from the accelerative path and the decelerative path toward initial hand position, such as surface c in Fig.3.

To minimize the joint torque vector norm to use redundant acceleration, the joint torque vector norm is derived as follow: By substituting (6) with (1), we get

$$\tau = MU_R a + \tau_J \quad (16)$$

Where,

$$\tau_J = MJ^+ (\dot{\mathbf{r}} - J\dot{\mathbf{q}}) + H$$

The joint torque vector needs to be normalization because the manipulator that considered in our study has the joint torque limits $\Omega_T$. The normalized joint torque vector is as follow Where $W = \text{diag}(\tau_{lim})$.

$$W\tau = WMU_R a + W\tau_J \quad (17)$$

We solve partial derivative of (17) with respect to $R_a$. Its minimal becomes a redundant acceleration to be minimum joint torque vector norm.

$$R_a = -(U^TMW^2MU)^{-1}U^TMW^2\tau_J \quad (18)$$

Next, we explain about the path finding algorithm connecting the initial edge state $(R_{a0}, R_{d0})$ and the terminal edge state $(R_{t1}, R_{t1})$ on the easy section.

We make Phase = 1.

for(k=1,2,...,Number of sampling on easy section) {
 Path finding up to the switching surface
 if(Phase == 1) {
     We compare a tip of the path and the switching surface in order to derive a path from the initial edge state $(R_{a0}, R_{d0})$ to the switching surface. If a $R_{a(k)}$ of tip of the path is smaller than the switching surface, the redundant acceleration on k-th sampling $R_{a(k)}$ becomes $R_{a(k)} = G_I(k)$. If a $R_{a(k)}$ of tip of the path is bigger than the switching surface, the redundant acceleration on k-th sampling $R_{a(k)}$ becomes $R_{a(k)} = F_I(k)$. Additionally, if the path strides over the switching surface between k-1-th sampling and k-th sampling, we make Phase = 2.
     }
     Path finding to trace the switching surface
     if(Phase == 2) {
         We approximate the switching surface by two points which are close to an intersection with the path such as Fig.4. We derive the redundant acceleration on k-th sampling $R_{a(k)}$ to locate the tip of path on the approximated switching surface by (8) and (9). Where, the redundant acceleration becomes a value in the range of $F_I(k) \leq R_{a(k)} \leq G_I(k)$. Additionally, if the closest switching surface is accelerative path or decelerative path, we make Phase = 3.
     }
     if(Phase == 3) {
         If the tip of path is located on the accelerative path, the redundant acceleration on k-th sampling $R_{a(k)}$ becomes $R_{a(k)} = G_I(k)$. If the tip of path locate on the decelerative path, the redundant accelerative on k-th sampling $R_{a(k)}$ becomes $R_{a(k)} = F_I(k)$. In case of other, the redundant acceleration on k-th sampling $R_{a(k)}$ becomes the value of (18).
     }
     We derive the path on k-th sampling using by $R_{a(k)}$.
     }
}
V. MOTION EXPERIMENT

A. Verifying Relationship of Redundant States

We verify the relationship of \( R_v, R_a, \) and \( R_{a_{est}} \) in (10). We use an actual redundant manipulator shown in Fig.5 in order to be verified. The manipulator is made by authors using TBL-V series of AC servo motor made of TAMAGAWA SEIKI Co. Controllers are perfect following controller using speed input mode on each joint. The controller execution cycle is \( 1[ms] \). Parameters of the manipulator are shown in Table.I.

In this verification, we derive a redundant pose reference \( R_{v_{ref}} \) and redundant velocity reference \( R_{v_{ref}} \) from redundant acceleration reference \( R_{a_{ref}} \) are shown in Fig.6 using (10). We operate the manipulator by reference of joint angle, joint velocity, and joint acceleration that are generated from the references of redundancy by (4) (5), and (6). After operating, we estimate responses of joint velocity and joint acceleration from joint angle responses by observer. Next, we derive estimated values of redundant pose \( R_{v_{est}} \), redundant velocity \( R_{v_{est}} \), and redundant acceleration \( R_{a_{est}} \) using (7),(19), and (20). We compare the references of redundancy and the responses of redundancy in order to verify the relationship of \( R_v, R_a, \) and \( R_{a_{est}} \).

\[
R_v = U^T \dot{q} \quad \text{(19)}
\]

\[
R_a = U^T \ddot{q} \quad \text{(20)}
\]

We show the references of redundancy and the responses of redundancy in Fig.6. For these results, the redundant acceleration response has vibration. The responses have been arisen transient response. As a result, we can find that the relationship of (10) is collect.

B. Results of Motion Planning

We show efficacy of proposed method to compare results of motion planning with proposed method, previous method [3] and general method.

We plan motions from the constant hand speed \( \dot{s} = 0.15[m/s] \) with the manipulator of Fig.5. The step of hand speed is \( \Delta \dot{s} = 0.01 \). The hand path is a path connecting straight paths and arc paths with the radius of \( 5[mm] \), shown in Fig.7. The previous method is same as a case of hand path is all difficult section. The general method is a simplest method. This method plans a motion that a joint acceleration vector norm is minimum. This motion is same as a case of a redundant acceleration \( R_a = 0 \). We calculate these motion

<table>
<thead>
<tr>
<th>Table II</th>
<th>Results of motion planning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed method</td>
<td>0.16[m/s]</td>
</tr>
<tr>
<td>Previous method</td>
<td>0.16[m/s]</td>
</tr>
<tr>
<td>Motion of minimum joint acceleration norm</td>
<td>0.01[m/s]</td>
</tr>
</tbody>
</table>

![Fig. 7. Hand path used for motion planning and motion planning section](image_url)
planning using MATLAB on computer that has Intel Core i7, 3.4Ghz CPU, and 8GB memory.

We shown maximum constant hand speeds and calculation time of results of motion planning in Table.II. The circle marks on the hand path in Fig.7 show switching points of easy sections and difficult sections. The first section is easy section, and the last section is difficult section. Other sections are arranged alternately.

For results of Table.II, When we use proposed method to plan the fast motion of path tracking task with constant hand speed, the maximum constant hand speed becomes faster greatly and the calculation time is increased comparing with the general method. However, when we compare the calculation time of the proposed method with the previous method, the calculation time is decreasing in spite of deriving same maximum constant hand speed.

C. Experimentation by Actual Manipulator

We operate the actual manipulator to use a motion planned on previous subsection.

We located an acceleration section and a deceleration section on front and rear of the hand path reference such as Fig.9, because a subject in our study is constant hand speed motion. The waveform of hand speed is a trapezoidal velocity waveform. We show responses of joint angles \( \dot{q} \), joint velocities \( q \), and joint torques \( \tau \) in Fig.8. Where, the joint velocities are estimation values using by observer. Chain lines on 0.5\([s]\) and 0.9\([s]\) indicate motion start and end points. The hand trajectory shown in Fig.9.

For the experiment, the second joint and the third joint show a good response. However, the first joint torque response runs off the planned value. We assume the joint torque can be output discontinuous value. However, if we consider a motor controller inside a servo pack, the joint torque can not be output discontinuous values because the joint torque is proportional to electrical current. Thus, we can find that the joint velocity estimation values and the joint torque estimation values are vibrated. That because the joint torques can not be output discontinuous values. Therefore, when we use proposed method to plan the fast motion of path tracking task with constant hand speed, we need to take a safety margin in order not to exceed the limits under the influence of control delay.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed a method that partitions hand path into easy sections and difficult sections and performs different path finding methods, for reduce calculation time. The method could reduce the calculation time greatly without slowing the maximum constant hand speed. We verified a relationship of redundant states by actual experiment, and
we showed efficacy of proposed method. Moreover, we addressed the problem of applying the proposed method on actual manipulator.

Now, we are improving a method partitioning the hand path to a method for partitioning automatically.

REFERENCES

