Sub-Optimal Trajectory Planning of Mobile Manipulator

Akira Mohri, Seiji Furuno, Makoto Iwamura and Motoji Yamamoto

Department of Intelligent Machinery and Systems
Faculty of Engineering, Kyushu University
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan
E-mail: mohri@mech.kyushu-u.ac.jp

Abstract

In this paper, a trajectory planning method of a mobile manipulator is presented. We derive the dynamics of the mobile manipulator considering it as the combined system of the manipulator and the mobile platform. The planning problem is formulated as an optimal control problem. To solve the problem, we use the concept of the order of priority. A gradient-based iterative algorithm which synthesizes the gradient function in a hierarchical manner based on the order of priority is used. The simulation results of the 2-link planar nonholonomic mobile manipulator are given to show the effectiveness of the proposed algorithm.

1 Introduction

A mobile manipulator composed of a manipulator and a mobile platform has a much larger workspace than a fixed-base manipulator due to the mobility provided by the platform.

Until now many papers treat the mobile manipulator’s problems. Some papers consider multiple task execution problems [1,2,3]. Path planning problems of the mobile manipulator are treated based on various ideas also [4,5,6,7]. All these papers do not consider the dynamics of the mobile manipulator. Yamamoto and Yun [8,9,10,11] consider the coordination problems based on the concept of preferred operation region. Main themes of their papers are considered to be control problems. Desai and Kumar [12] treat nonholonomic motion planning for the mobile manipulator considering its dynamics. They reformulate the problem as a variational problem.

In this paper we treat the trajectory planning problem of the nonholonomic mobile manipulator taking the dynamics into considerations also. Our method is based on the gradient function. First we derive the dynamics of the mobile manipulator considering it as the combined system of the manipulator and the nonholonomic mobile platform. Then the trajectory planning problem is formulated as the optimal control problem with some constraints. To solve the problem numerically, we use the concept of the order of priority and the gradient function which are synthesized in the hierarchical manner [13]. Simulation results are given to show the effectiveness of the proposed algorithm.

2 Modeling of Dynamic Equation

We consider a mobile manipulator shown by Fig.1. Dynamic equation of the mobile manipulator is given...
by the following equation.

\[ M_m(q)\ddot{q} + C(q, \dot{q}) = F \]  

(1)

where \( M_m(q) \) is the inertia matrix and \( C(q, \dot{q}) \) represents the Coriolis and centrifugal forces. Their details are omitted. \( q \) and \( F \) are the following vectors.

\[ q = (x, y, \phi, w_1, w_2, \theta_1, \theta_2)^T \]

(2)

\[ F = (0, 0, \tau_{w_1}, \tau_{w_2}, \tau_{\theta_1}, \tau_{\theta_2})^T \]

(3)

\( R_c(x, y) \) is the coordinates of the center of gravity of the mobile platform, \( \phi \) is the heading angle of the mobile platform, \( w_1 \) and \( w_2 \) are angular positions of two driving wheels, \( \theta_1 \) and \( \theta_2 \) are joint angles of the manipulator and \( \tau_{w_1} \) and \( \tau_{w_2} \) are input torques of the platform and \( \tau_{\theta_1} \) and \( \tau_{\theta_2} \) are joint torques of the manipulator. Eq.(1) is derived using Lagrange equation. Details of derivation are omitted.

Next, the constraint equation to which the platform is subjected is given

\[ A(q)\ddot{q} = 0 \]

(4)

where

\[
A(q) = \begin{bmatrix}
1 & 0 & 0 & -\frac{r}{2} \cos \phi & -\frac{r}{2} \cos \phi & 0 & 0 \\
0 & 1 & 0 & -\frac{r}{2} \sin \phi & -\frac{r}{2} \sin \phi & 0 & 0 \\
0 & 0 & 1 & \frac{r}{2w} & \frac{r}{2w} & 0 & 0
\end{bmatrix}
\]

(5)

where \( r \) is the radius of the driving wheel and \( 2w \) is the distance of two wheels.

Using the \( B(q) \) matrix which satisfies

\[ A(q)B(q) = 0 \]

(6)

\( \ddot{q} \) can be expressed as follows

\[ \ddot{q} = B(q)\dot{r} \]

(7)

where

\[ \dot{r} = (w_1, w_2, \theta_1, \theta_2)^T \]

(8)

One choice of \( B(q) \) is as follows.

\[
B(q) = \begin{bmatrix}
\frac{r}{2} \cos \phi & \frac{r}{2} \cos \phi & 0 & 0 \\
\frac{r}{2} \sin \phi & \frac{r}{2} \sin \phi & 0 & 0 \\
\frac{r}{2w} & \frac{r}{2w} & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

(9)

Differenting (7), we obtain

\[ \ddot{r} = \dot{B}(q)\dot{r} + B(q)\ddot{r} \]

(10)

Adding (4) to (1) by use of Lagrange multiplier \( \lambda \), we obtain

\[ M_m(q)\ddot{q} + C(q, \dot{q}) = F + A^T(q)\lambda \]

(11)

Premultiplying (11) by \( B^T(q) \), we have

\[ B^T(q)M_m(q)\ddot{q} + B^T(q)C(q, \dot{q}) = B^T(q)F \]

(12)

Where we use \( A(q)B(q) = 0 \). Substituting (10) into (12) and rearranging, we obtain

\[
\ddot{r} = (B^T(q)M_m(q)B(q))^{-1}(B^T(q)F - B^T(q)C(q, \dot{q}) \times (C(q, \dot{q}) + M_m(q)B(q)\ddot{r} - F) \]

(13)

Combining (7) and (13), we obtain

\[
\frac{d}{dt}q_r = B_r(q_r)\dot{r}
\]

(15)

where

\[
q_r = (x, y, \phi, \theta_1, \theta_2)^T
\]

(16)

\[
B_r(q_r) = \begin{bmatrix}
\frac{r}{2} \cos \phi & \frac{r}{2} \cos \phi & 0 & 0 \\
\frac{r}{2} \sin \phi & \frac{r}{2} \sin \phi & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

(17)

Finally we obtain the state space equation of the mobile manipulator considered in this paper. Here in the second equation we use \( q_r \) instead of \( q \).

\[
\frac{d}{dt}q_r = B_r(q_r)\dot{r}
\]

\[
\frac{d}{dt}\dot{r} = -(B^T(q_r)M_m(q_r)B(q_r))^{-1}B^T(q_r) \times (C(q_r, \dot{q}_r) + M_m(q_r)B(q_r)\ddot{r} - F)
\]

(18)

Using these equations, the optimal trajectory planning problem can be formulated.

3 Optimal Control Problem

We define the state and control vectors as follows.
\[ x = (x, y, \phi, \theta_1, \theta_2, \dot{w}_1, \dot{w}_2, \dot{\theta}_1, \dot{\theta}_2)^T \]  
(19)

\[ u = (r_{w_1}, r_{w_2}, r_{\theta_1}, r_{\theta_2})^T \]  
(20)

Then (18) is rewritten as

\[ \dot{x}(t) = f(x(t), u(t)) \]  
(21)

The initial state of \( x \) is defined as follows.

\[ x(0) = x_0 \]  
(22)

Let \( x_d \) be the desired final state. Then the terminal constraint is expressed as

\[ r_1(t_f) = x(t_f) - x_d = 0 \]  
(23)

where \( t_f \) is the final time.

If some obstacles are present in the workspace, the constraint conditions of obstacles avoidance are defined as follows.

\[ g^i_j(x(t)) \geq 0 \quad (0 \leq t \leq t_f) \]  
(24)

\[ (i = 1, \ldots, l \ ; \ j = 1, \ldots, m_i) \]

where \( l \) is the number of obstacles and \( m_i \) is the number of inequalities that are necessary to describe the conditions of \( i \)-th obstacle. The inequality constraint (24) can be rewritten as the following equality condition.

\[ r_2(t_f) = \int_0^{t_f} c_2(x(t)) dt = 0 \]  
(25)

where

\[ c_2 = (c_{21}, \ldots, c_{1m_1}, c_{21}, \ldots, c_{2m_2}, c_{21}, \ldots, c_{2m_1})^T \]
and \( c_{2j}(x) \) is a following continuous function

\[ c_{2j}(x) = \begin{cases} 
\delta_{2j}(x) (>0) & \text{if } g_{1j}^i(x) < 0 \\
0 & \text{if } g_{1j}^i(x) \geq 0
\end{cases} \]  
(26)

where \( \delta_{2j}(x) \) is a positive function.

The cost function will be assumed to take the form

\[ P = \int_0^{t_f} f_0(x, u) dt \]  
(27)

4 Gradient Function in a Hierarchical Manner

In this section, the method to synthesize the gradient function in a hierarchical manner considering the order of priority is presented [13]. If a input vector \( u(t) \) is given, a trajectory \( x(t) \) is obtained by solving

\[ (21) \]. Then, the constraint vector \( r_i \) \((i = 1, 2)\) can be calculated from (23), (25). Therefore, the constraint vector \( r_i \) can be written as a function of \( u \) as follows.

\[ r_i = h_i(u) \quad (i = 1, 2) \]  
(28)

From (28), the variation in \( r_i \) due to variation in the input vector \( u \) is expressed as follows.

\[ \delta r_i = \int_0^{t_f} J_i \delta u dt \quad (i = 1, 2) \]  
(29)

(Refer to Appendix). Based on (29), we will synthesize the gradient function.

First, we compute \( \delta u \) considering the terminal constraint condition \( r_1 = 0 \) which is given first-priority. If we choose \( \delta u \) so that it yields \( \delta r_1 = \alpha r_1 \) \((\alpha \text{ : negative constant})\), then it is guaranteed that \( \| r_1 \| \to 0 \). In general, such \( \delta u \) is not unique and its general solution can be expressed as follows.

\[ \delta u = \alpha J_1^T I_{11}^{-1} r_1 + \delta y - J_1^T I_{11}^{-1} \int_0^{t_f} J_1 \delta y dt \]  
(30)

\[ I_{11} = \int_0^{t_f} J_1 J_1^T dt, \quad \delta y : \text{arbitrary vector} \]

Equation (30) covers all the least-squares solutions that minimize the error norm

\[ \| \delta r_1 - \alpha r_1 \| = \| \int_0^{t_f} J_1 \delta u dt - \alpha r_1 \| \]

Now, substituting (30) into (29) for \( i = 2 \), we have the following equation.

\[ \delta r_2 = \alpha J_{21}^T I_{11}^{-1} r_1 + \int_0^{t_f} J_2 \delta y dt \]  
(31)

\[ I_{21} = \int_0^{t_f} J_2 J_2^T dt, \quad J_2 = J_2 - I_{21} I_{11}^{-1} J_1 \]

Based on (31), we next compute the arbitrary vector \( \delta y \) considering the constraint condition of obstacle avoidance \( r_2 = 0 \) which is given second-priority. We can obtain \( \delta y \) that minimize the error norm

\[ \| \delta r_2 - \alpha r_2 \| = \| \int_0^{t_f} J_2 \delta y dt - \alpha (r_2 - I_{21} I_{11}^{-1} r_1) \| \]

in the same fashion as follows.

\[ \delta y = \alpha J_2^T I_{22}^{-1} (r_2 - I_{21} I_{11}^{-1} r_1) \]

\[ + \delta z - J_2^T I_{22}^{-1} \int_0^{t_f} J_2 \delta z dt \]  
(32)

\[ I_{22} = \int_0^{t_f} J_2 J_2^T dt, \quad \delta z : \text{arbitrary vector} \]
Therefore, the gradient function $\delta u$ is obtained by substituting (32) into (30) as follows.

$$
\delta u = \alpha J_1^T I_{11}^{-1} r_1 + \alpha K (r_2 - I_2 I_{11}^{-1} r_1) + \delta z - J_1^T I_{11}^{-1} \int_0^t J_1 \delta z dt - K \int_0^t J_2 \delta z dt \tag{33}
$$

$$
I_{12} \equiv \int_0^t J_1 J_2^T dt, \quad K \equiv J_1^T I_{22}^{-1} - J_1^T I_{11}^{-1} I_{12} I_{22}^{-1}
$$

Finally, we determine the arbitrary vector $\delta z$ in (33) so that the cost function decreases as much as possible. Now we define the Hamiltonian $H$ using the multiplier function $\psi$.

$$
H(x, u, \psi) \equiv f_0(x, u) + \psi^T f(x, u) \tag{34}
$$

And $\psi$ satisfies the following equations.

$$
\dot{\psi} = -H_x; \quad \psi(t_f) = 0 \tag{35}
$$

where $H_x = \partial H / \partial x$. Then $\delta P$ is given by

$$
\delta P = \int_0^t H_u \delta u dt \tag{36}
$$

where $H_u = \partial H / \partial u$. We choose here the arbitrary vector $\delta z$ in (33) as $\delta z = \alpha H_u^T$. Although this choice does not guarantee monotonous decrease of $P$ strictly, the inequality

$$
\delta P = \alpha \int_0^t H_u H_u^T dt \leq 0 \tag{37}
$$

can be obtained if both $r_1$ and $r_2$ converge to zeros.

Finally, we obtain the gradient function that considers the order of priority as follows.

$$
\delta u = \alpha \left[ J_1^T I_{11}^{-1} r_1 + K (r_2 - I_2 I_{11}^{-1} r_1) + H_u^T \right] - J_1^T I_{11}^{-1} \int_0^t J_1 H_u^T dt - K \int_0^t J_2 H_u^T dt \tag{38}
$$

5 Numerical Algorithm

The numerical algorithm to find the solution that is based on the gradient function synthesized in a hierarchical manner considering the order of priority is given as follows.

Step 1. Estimate a set of control variable histories $u^0(t)$.

Determine a negative constant $\alpha$, small positive constants $\epsilon_1$, $\epsilon_2$, $\epsilon_3$. Set $k = 0$.

Step 2. Solve the state equation (21) and save $x^0(t)$. Calculate $r_1^0$, $r_2^0$, $P^0$ by the equation (23),(25),(27).

Step 3. Calculate $J_1^k$, $J_2^k$, $H_u^k$ and compute the gradient function $\delta u^k(t)$ by the equation (38).

Step 4. Modify $u^k(t)$ as follows.

$$
u^{k+1}(t) = u^k(t) + \delta u^k(t)
$$

Then, calculate $x^{k+1}(t)$, $r_1^{k+1}$, $r_2^{k+1}$, $P^{k+1}$.

Step 5. If the following conditions

$$
\|r_1^{k+1}\| < \epsilon_1, \quad \|r_2^{k+1}\| < \epsilon_2, \quad \|P^{k+1} - P^k\| / P^k < \epsilon_3
$$

are satisfied simultaneously, stop calculation. If not, then set $k = k + 1$ and return to Step 3.

6 Example

In this section, the proposed algorithm is applied to plan the sub-optimal trajectory of the 2-link planar nonholonomic mobile manipulator. Parameter values are as follows; mass of the platform : $m_v = 50.0$ [kg], mass of the wheel : $m_w = 15.0$ [kg], mass of the link : $m_l = m_s = 10.0$ [kg], length of the link : $l_1 = l_2 = 2.0$ [m], $t_v$ and $t_w$ in Fig. 1 are 1.7 [m] and 1.5[m] respectively. $x_0$ and $x_d$ are assumed as follows.

$$
x_0 = (0.0, 0.45, 0.0, -90.0, 0, 0, 0, 0)^T
$$

$$
x_d = (9.2, 5.5, 80.0, -20.0, 30.0, 0, 0, 0, 0)^T
$$

$t_f$ is set to 5.0. Performance index is given by

$$
P = \int_0^{t_f} u^T W u dt
$$

Where $W$ is the weighting matrix and set to $W = \text{diag}([1.0, 1.0, 1.0, 1.0])$. A obstacle which is the circle of radius $r_o = 2.0$ [m] centered at $P_o(x, y) = (5.0, 5.0)$ is present.

Here we consider the constraint condition of obstacle avoidance [14]. We assume that the mobile platform is the circle whose center is at $P_c(x, y)$ and radius is $r_m$ and links of the manipulator are straight lines.

First, we consider the constraint condition as to the platform. $d_1^{21}$: Define $d_1 = ||P_c - P_o||$. Then

$$
d_1^{21} = \begin{cases} (r_o + r_m + s_p - d_1)^2 & \text{if } d_1 < r_o + r_m + s_p \\ 0 & \text{if } d_1 \geq r_o + r_m + s_p \end{cases}
$$
where $s_p$ is the safety factor.

Second, we consider the constraint condition as to the manipulator, $c_2^1$ and $c_2^3$. Where $a_i$ is the position vector of $i$-th joint ($i = 1, 2$) and $b_i$ is the position vector of the point where the $i$-th link and the perpendicular on the $i$-th link from the center $P_o$ intersects. Define $d_{1i}^2 = \|b_i - P_o\|$, $d_{2i}^2 = \|a_{i+1} - P_o\|$. $\gamma$ as $(b_i - a_i) = \gamma(a_{i+1} - a_i)$. Where $a_3$ is the position of the end-effector. Then

$$c_{2i+1}^1 = \begin{cases} \left(r_o + s_m - d_{1i}^2\right)^2 & \text{if } 0 \leq \gamma \leq 1 \\ \text{and } d_{2i}^2 < r_o + s_m \end{cases}$$
$$c_{2i+1}^2 = \begin{cases} \left(r_o + s_m - d_{2i}^2\right)^2 & \text{if } \gamma > 1 \\ \text{and } d_{1i}^2 < r_o + s_m \end{cases}$$
$$c_{2i+1}^3 = \begin{cases} 0 & \text{the other case} \end{cases}$$

(i = 1, 2)

where $s_m$ is the safety factor also.

Fig.2 shows the simulation result without consideration of obstacle [case 1] and Fig.3 shows the result with obstacle [case 2]. The performance values are 78.98 for case 1 and 99.24 for case 2.

Next, we consider two obstacles problem [case 3]. One is the same position as case 2 and one more obstacle, which has the same radius $r_o = 2.0$ [m], is added to the position $P_o(x, y) = (15.0, 10.0)$. The constraint conditions of obstacle avoidance, $c_{21}^1$, $c_{22}^1$ and $c_{23}^1$ are given in the same way as $c_{11}^1$ and so on. $x_0$ is the same as case 1 and case 2. In this case $x_d$ is given by

$$x_d = (15.5, 10.0, 45.0, 0, 0, 0, 0, 0, 0, 0)^T$$

Fig.4 shows the simulation result. The performance value of this case is 395.5.

7 Conclusions

This paper discussed the sub-optimal trajectory planning problem of the nonholonomic mobile manipulator. We derive the dynamics of the mobile manipulator considering it as the combined system of the manipulator and the mobile platform. The trajectory planning problem is formulated as the optimal control problem. We used iterative algorithm that is based on the gradient function synthesized in the hierarchical manner. Simulation results were given to show the effectiveness of the proposed method.

Appendix

Calculation of $J_i$

Define the error vector $e_i(t)$ ($i = 1, 2$) as follows

$$e_1(t) \equiv r_1(t) = x(t) - x_d$$
$$e_2(t) \equiv r_2(t) = \int_0^t c_2(x(\tau))d\tau$$

Then, $\delta r_i$ is given as follows.

$$\delta r_i = \int_0^{T_f} J_i \delta u dt, \quad (i = 1, 2)$$

$$J_i \equiv e_i^T f_u$$

where $f_u = \partial f/\partial u$ and $e_i$ are given by the following.
Fig. 4 Simulation result: Case 3

Equations

\[ \begin{align*}
\dot{e}_{1x} &= -e_{1x} f_x; & e_{1x}(t_f) &= I \\
\dot{e}_{2x} &= -e_{2x} f_x - c_{2x} e_{2x}(t_f) = 0
\end{align*} \]

where \( f_x = \frac{\partial f}{\partial x}, c_{2x} = \frac{\partial c_2}{\partial x} \) and \( I \) is the \( n \times n \) unit matrix (\( n \) is the dimension of \( x(t) \) and equal to 9 in this paper). Details are omitted. Please refer to [13].

References


