KSME International Journal, Vol. 14, No. 6, pp. 605~621, 2000

Optimal Configuration Control for a Mobile Manipulator

Jin-Gu Kang*, Tae-Seok Jin, Min-Gyu Kim, Jang-Myung Lee
Department of Electronics Engineering, Intelligent Robot Laboratory, Pusan National University

A mobile manipulator-a serial connection of a mobile platform and a task robot-is redundant by itself. Using its redundant freedom, a mobile manipulator can move in various modes, *i. e.*, can perform dexterous tasks. In this paper, to improve task execution efficiency utilizing redundancy, optimal configurations of the mobile manipulator are maintained while it is moving to a new task point. Assuming that a task robot can perform the new task by itself, a desired configuration for the task robot can be pre-determined. Therefore, a cost function for optimality can be defined as a combination of the square errors of the desired and actual configurations of the mobile platform and of the task robot. In the combination of the two square errors, a newly defined mobility of a mobile platform is utilized as a weighting index. With the aid of the gradient method, the cost function is minimized, so the tasle that the mobile manipulator performs is optimized. The proposed algorithm is experimentally verified and discussed with a mobile manipulator, PURL-II.

Key Words : Mobile Robot, Task Robot, Mobile Manipulator, Optimal Configuration, Mobility

Nomenclature

Suffix m	: Mobile robot, Suffix t : Task robot
Suffix i	: Initial, Suffix f : Finial, Suffix d : Desired
X, Y, Z	: Axis in Cartesian space
x, y, z	: Point in Cartesian space
p	: Position of robot, $\dot{p}$: Velocity of robot
q	: Robot joint variable
P	: Transformation matrix
$J(\cdot)$	: Jacobian matrix
$J(\cdot)^+$	: Jacobian pseudo inverse matrix
$v_{m,c}$	: Average velocity of gravity center of mobile robot
$v_{m,z}$	: Velocity of ball-screw joint along th z-axis
ω	: Angular velocity of the mobile robot
θ_m	: Angular position between x axis and mobile robot
v_R	: Velocity value of right wheel
v_L	: Velocity value of left wheel
l	: Distance between wheel and center
r	: Radius of wheel
L	: Cost function
$R(\cdot)$	: Rotation matrix
$f(\cdot)$	: Kinematics function

$c_1=\cos\theta_1$, $s_1=\sin\theta_1$

$c_{234}=\cos(\theta_2+\theta_3+\theta_4)$, $s_{234}=\sin(\theta_2+\theta_3+\theta_4)$

1. Introduction

While a mobile robot can expand the size of the workspace but does no work, a vertical multi-joint robot or manipulator cannot move but can do work. Therefore there has been an increase in interest in multiple robots and combination robots to accomplish the given task in cooperation (Francois G. Pin, 1994). At present, there has been a lot of research on redundant robots that have more degrees of freedom than non

* Corresponding Author,
E-mail : jgukang@hyowon.pusan.ac.kr
TEL : +82-51-510-1696 ; **FAX** : +82-51-515-5190
Department of Electronics Engineering, Intelligent Robot Laboratory, Pusan National University, 30, Jangjeon-dong, Kumjung-ku, Pusan 609-735, Korea.
(Manuscript **Received** October 5, 1999; **Revised** March 13, 2000)

combination robots in the given work space, so that it can have an optimal position and optimized job performance (Tsuneo Yoshikawa, 1985; Stephen L. Chiu, 1998). While there has been a lot of research on control for both mobile robot navigation and fixed manipulator motion, there are few reports on the cooperative control for a robot with movement and manipulation capability (Francois G. Pin, 1994). It is desirable to look at improvement in two areas: one is the case where multiple manipulators cooperate and perform the task in parallel, while the other case is where a mobile manipulator performs the task in serial. In this paper, we define a mobile manipulator as a mobile platform combined with a vertical multi-joint robot, and define a vertical multi-joint robot as a task robot. Different from a fixed redundant robot, a mobile manipulator has the characteristics that with respect to the given working environments, it has the merits of unconventional movement avoidance, collision avoidance, efficient application of the corresponding mechanical parts, and improvement of adjustment. Because of these characteristics, it is desirable that one uses the mobile manipulator with the transportation ability and dexterous handling in hazardous working environments. This paper utilizes the mobile manipulator PURL-II, consishing of a mobile robot with 3 degrees of freedom for efficient job accomplishment, and a task robot with 5 degrees of freedom. We analyze the kinematics and inverse kinematics to define the 'mobility' of the mobile robot as the most important feature of the mobile manipulator. We research the optimal position and movement of the robot so that the combined robot can perform the task with minimal joint displacement, adjusting the weighting value based on this 'mobility'. When the mobile robot performs the job with the cooperation of the task robot, we search for the optimal configuration for the task using the ' gradient method' to minimize the movement of the whole robots as well as to drive to the desired configuration. The results acquired by implementing the proposed algorithm through computer simulation and experiments using PURL-II are demonstrated.

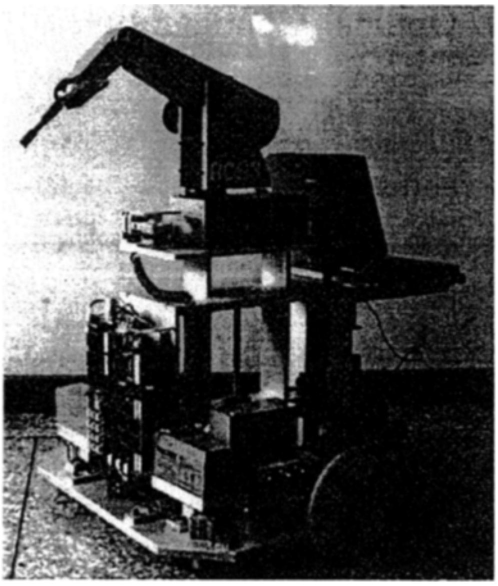


Fig. 1 Complete PURL-II

2. Mobile Manipulator

2.1 Configuration of the mobile manipulator

The robot that was used in our research is shown in Fig. 1. The robot PURL-II consists of a task robot with 5 degrees of freedom and a mobile robot with 3 degrees of freedom. The mobile robot has three DC servo-motors for three different directions of movement. We use 80C196KC microprocessors as the motor controllers, so that we control the three motors concurrently. The motor-drive port is configured with an IGBT in H-bridge form. We mount the ROB3 with 5 joints as the task robot, and install a gripper at the end-effector. In addition, we mount a portable PC to be used as the host computer to monitor the controller of the mobile manipulator and to monitor the states of the robot. The block diagram of the mobile robot configuration is shown in Fig. 2.

2.2 Kinematics analysis of the mobile robot

2.2.1 Forward kinematics analysis

We analyze the forward kinematics to calculate the position in the Cartesian coordinate system using the variables of the mobile robot (Mark W. Spong, 1989). The coordinate system and model-

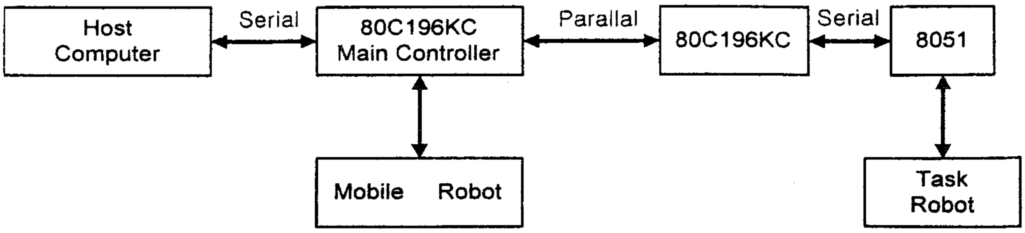


Fig. 2 Block diagram of PURL-II

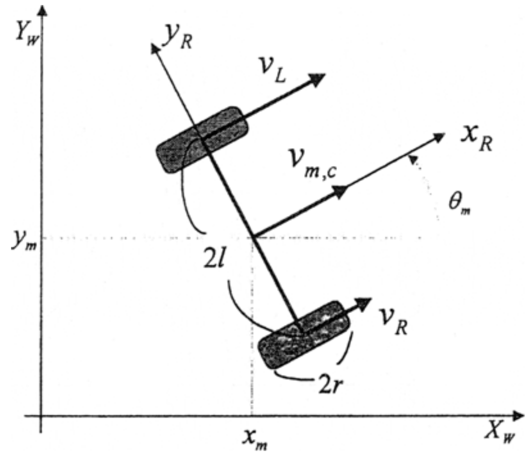


Fig. 3 Mobile robot modeling and coordinate system

ing for the forward kinematics is shown in Fig. 3.

Let us denote the present position of the mobile robot as p_m, the velocity as $\dot{I}_m$, the average velocity of the gravity center as $v_{m,c}$, the angular velocity of the mobile robot as ω, and the angle between X coordinate and the mobile robot as θ_m. The Cartesian velocity $\dot{I}_m$, is represented in terms of joint variables as follows:

$$\dot{I}_m = J(p_m)\,\dot{q}_m \tag{1}$$

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{z}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos\theta_m & 0 & 0 \\ \sin\theta_m & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_{m,c} \\ v_{m,z} \\ \omega \end{bmatrix} \tag{2}$$

where $v_{m,z}$ is provided by a ball-screw joint along the Z axis.

2.2.2 Inverse kinematics analysis

Because $J(p_m)$ in (1) is not a square matrix, using the pseudo inverse matrix, $J(p_m)^+$, the joint velocity, $\dot{I}_m$, is obtained as

$$\dot{I}_m = J(p_m)^+ \dot{I}_m = (J^T\ J)^{-1} J^T \dot{I}_m \tag{3}$$

$$\begin{bmatrix} v_{m,c} \\ v_{m,z} \\ \omega \end{bmatrix} = \begin{bmatrix} \cos\theta_m & \sin\theta_m & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{z}_m \\ \dot{\theta}_m \end{bmatrix} \tag{4}$$

The mobile robot can deliver velocity to the two wheels, say v_L and v_R for the left and right wheel velocities, respectively. It is supposed that the *pure rolling condition and non-slipping condition* are satisfied. The pure rolling condition, in while the relative velocity between the wheel of the robot and the contact surface is zero, is adapted for the right wheel by (5):

$$-\cos(\theta_m)\,\dot{x}_m - \sin(\theta_m)\,\dot{y}_m - l\dot{\theta}_m + r\dot{q}_{m,r} = 0 \tag{5}$$

For the left wheel, the condition is represented as in (6):

$$-\cos(\theta_m)\,\dot{x}_m - \sin(\theta_m)\,\dot{y}_m - l\dot{\theta}_m + r\dot{q}_{m,l} = 0 \tag{6}$$

The non-slipping condition is shown in (7):

$$-\sin(\theta_m)\,\dot{x}_m + \cos(\theta_m)\,\dot{y}_m = 0 \tag{7}$$

Let us denote the wheel radius as r, and the distance between wheel and the center as l. Then $\dot{x}_m$, $\dot{y}_m$, and $\dot{\theta}_m$ are calculated by (8) as follows, and we can express the Jacobian matrix as (9):

$$\dot{x}_m = \frac{r}{2}(\dot{q}_{m,l} + \dot{q}_{m,r})\cos\theta_m \tag{8a}$$

$$\dot{y}_m = \frac{r}{2}(\dot{q}_{m,l} + q_{m,r})\sin\theta_m \tag{8b}$$

$$\dot{\theta}_m = \frac{r}{2l}(\dot{q}_{m,l} - \dot{q}_{m,r}) \tag{8c}$$

$$p_m = J\dot{q}_m \tag{9}$$

where $p_m = [\dot{x}_m\ \ \dot{y}_m\ \ \dot{z}_m\ \ \dot{\theta}_m]^T$, $\dot{q}_m = [\dot{q}_{m,r}\ \ \dot{q}_{m,l}\ \ \dot{q}_{m,z}]$, and the Jacobian matrix $J_{4\times3}$ can be defined by (10) (Jin-Hee Jang, and Chang-Soo

Han, 1997):

$$J_{4\times3}=\begin{bmatrix} \frac{r}{2}\cos\theta_m & \frac{r}{2}\cos\theta_m & 0 \\ \frac{r}{2}\sin\theta_m & \frac{r}{2}\sin\theta_m & 0 \\ 0 & 0 & 1 \\ -\frac{r}{2l} & \frac{r}{2l} & 0 \end{bmatrix} \tag{10}$$

2.3 Kinematics analysis of the task robot

2.3.1 Forward kinematics

The forward kinematics of the task robot is represent by (11); the coordinate system is shown in Fig. 4.

$$p_t={}^tP_j \cdot q_t \tag{11}$$

where ${}^tP_j$ defines the transformation from the joint space to the Cartesian space (Keum-Shik Hong, Young-Min Kim, and Chiutai Choi, 1997).

With respect to the coordinate system in Fig. 4, the D-H parameters can be obtained as shown in Table 1.

From Table 1, the homogeneous matrix is represented by (12) as follows

Table 1 D-H parameter of the task robot

Joint	α	a	d	θ
1	0	0	d_1	θ_1
2	90°	a_1	0	θ_2
3	0	a_2	0	θ_3
4	0	0	d_4	θ_4
5	90°	0	0	θ_5

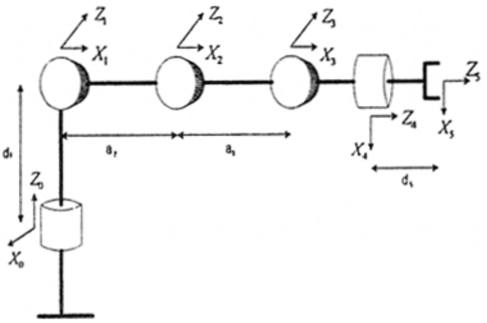


Fig. 4 Task robot modeling and coordinate system assignment

$${}^tP_j={}^0A_1{}^1A_2{}^2A_3{}^3A_4{}^4A_5$$

$$=\begin{bmatrix} -c_1s_{234}s_5-s_1c_1 & -c_1s_{234}c_5+s_1s_5 & c_1c_{234} & a_3c_1c_{23}+a_2c_1c_2-d_4c_1c_{23} \\ -s_1s_{234}s_5+c_1c_5 & -s_1s_{234}s_5+c_1c_5 & s_1c_{234} & a_3s_1c_{23}+a_2s_1c_2-d_4 \\ s_{234}s_5 & c_{23}c_5 & s_{234} & a_3s_{23}+a_2s_2+d_4c_{23}+d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

where

$${}^0A_1=\begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\quad {}^1A_2=\begin{bmatrix} c_2 & 0 & s_2 & a_2c_2 \\ s_2 & 0 & -c_2 & a_2s_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$${}^2A_3=\begin{bmatrix} c_3 & -s_3 & 0 & a_3c_3 \\ s_3 & c_3 & 0 & a_3s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\quad {}^3A_4=\begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_2 & c_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and $${}^4A_5=\begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

2.3.2 Inverse kinematics

If the object position is known, the inverse kinematics is need for calculating the angle of each joint. The Inverse kinematics is defined as in (13). Also, the velocity kinematics can be derived in as (14):

$$\dot{q}_t=J^{-1}\dot{p}_t \tag{13}$$

$$J^{-1}=\begin{bmatrix} J_{vx} \\ J_{vy} \\ J_{vz} \\ J_{\omega x} \\ J_{\omega y} \\ J_{\omega z} \end{bmatrix}=\begin{bmatrix} J_{11} & J_{12} & J_{13} & 0 & 0 \\ J_{21} & J_{22} & I_{23} & 0 & 0 \\ 0 & J_{32} & J_{33} & 0 & 0 \\ 0 & 0 & J_{43} & J_{44} & J_{45} \\ 0 & 0 & J_{53} & J_{54} & J_{55} \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{14}$$

where ·

$$J_{11}=-2a_3c_1c_2s_1+s_1c_1s_2-a_2s_1c_2+d_4s_1c_2s_3$$
$$+d_4s_1c_2s_3,$$
$$J_{12}=-a_3c_1^2s_2-c_1s_1c_2-a_2c_1s_2-d_4c_1c_2c_3$$
$$+d_4c_1s_2s_3,$$
$$J_{13}=d_4c_1s_2s_3-d_4c_1c_2c_3,$$
$$J_{21}=a_3c_1^2c_2-a_3s_1^2c_2-a_2c_1c_2-d_4c_1s_2c_3$$

$$-d_4c_1c_2c_3,$$

$$J_{22}=-a_3s_1s_2-a_2s_1s_2-d_4s_1c_2c_3+d_4s_1s_2s_3$$

$$J_{23}=d_4s_1s_2s_3-d_4s_1c_2{}^2,$$

$$J_{32}=a_3c_2s_3-a_3s_2c_3+a_2c_2-d_4s_2c_3-d_4c_2s_3,$$

$$J_{43}=c_1s_2+s_1c_2,\ J_{53}=s_1s_2-c_1c_2,$$

$$J_{44}=c_1s_2+s_1c_2,\ J_{54}=s_1s_2-c_1c_2,$$

$J_{45}=c_1s_2+s_1c_2$, and $J_{55}=s_1s_2-c_1c_2$.

2.4 Kinematics analysis of the mobile manipulator

Each robot that is designed to accomplish an independent objective concurrently should perform its corresponding movement to complete the task. The trajectory is needed for a kinematic analysis of the whole system, so that we can make the combination robot perform the task efficiently using the redundant degrees of freedom generated by the combination of the two robots (N. Hare and Y. Fing, 1997). From Fig. 5, we can see the Cartesian coordinate of the implemented mobile/task robot system and the link coordinate system of each joint in space (Mark W. Spong, 1989).

This system is an independent mobile manipulator without wires. The vector $\dot{q}$ of the whole system joint variables can be defined by $\dot{q}_t=[\dot{q}_{t1}\ \dot{q}_{t2}\ \dot{q}_{t3}\ \dot{q}_{t4}\ \dot{q}_{t5}]$ and $\dot{q}_m=[\dot{q}_{m6}\ \dot{q}_{m7}\ \dot{q}_{m8}\ \dot{q}_{m9}]$ such that it represents the joint variable vector of the task robot, i.e.,

$$\dot{q}=\begin{bmatrix}\dot{q}_t\\ \dot{q}_m\end{bmatrix}=[\dot{q}_{t1}\ \dot{q}_{t2}\ \dot{q}_{t3}\ \dot{q}_{t4}\ \dot{q}_{t5}\ \dot{q}_{m6}\ \dot{q}_{m7}\ \dot{q}_{m8}\ \dot{q}_{m9}]^T \tag{15}$$

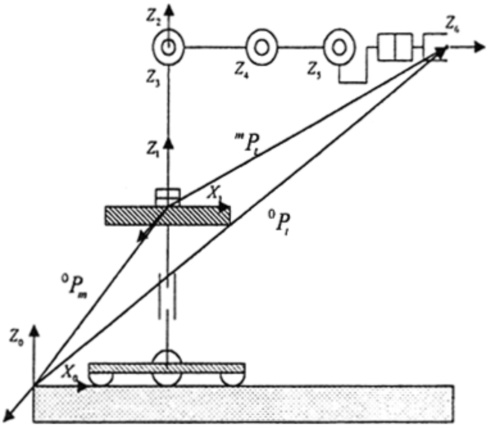


Fig. 5 Coordinate system of the mobile manipulator

From Fig. 5, the mobile robot is a three degree-of-freedom robot with three individual axes of movement, which can move in directions X, Y, and Z. The end-plate position (vertical multi-joint robot base) is determined from the joint variables with respect to each axis of movement of the mobile robot. Let us denote this as $^0P_m=[p_x\ p_y\ p_z\ \theta_m]^T$ and define frame {1} as shown. ^0P_m is the position vector from the world frame {0} to the frame {1} that is assigned at the end plate of the mobile robot (Jae-Kyung Lee and Hyung Suck Cho, 1997). The linear velocity and angular velocity of the mobile robot in Cartesian space with respect to the fixed frame of the world frame can be expressed as in (16).

$$^0\dot{P}_m=\begin{bmatrix}{}^0V_m\\ {}^0\omega_m\end{bmatrix}=\begin{bmatrix}{}^0J_{m,v}\\ {}^0J_{m,\omega}\end{bmatrix}\dot{q}_m={}^0J_m\dot{q}_m \tag{16}$$

In view of Fig. 5, let us represent the Jacobian of vector $\dot{q}_t$ (task robot joint variable) with respect to frame {1}. These results are shown in (17) as follows (Sam-Sang You, 1996):

$$^m\dot{P}_t=\begin{bmatrix}{}^mV_t\\ {}^m\omega_t\end{bmatrix}=\begin{bmatrix}{}^mJ_{t,v}\\ {}^mJ_{t,\omega}\end{bmatrix}\dot{q}_t={}^mJ_t\dot{q}_t \tag{17}$$

Given the Jacobians ^0J_m and ^mJ_t for each robot, if we express the Jacobian of the mobile manipulator as ^0J_t, then the linear and angular velocity of the end-effector, $^0\dot{P}_t=[^0V_t,\ ^0\omega_t]^T$, with respect to the world frame is represented as (18):

$$^0\dot{P}_t=\begin{bmatrix}{}^0V_t\\ {}^0\omega_t\end{bmatrix}=\begin{bmatrix}{}^0V_m\\ {}^0\omega_m\end{bmatrix}+\begin{bmatrix}{}^0\omega_m+{}^0R_1{}^1V_t\\ {}^0R_1{}^1\omega_t\end{bmatrix}$$

$$={}^0J_m\dot{q}_m+{}^0J_t\dot{q}_t=[{}^0J_m\ {}^0J_t] \tag{18}$$

Here $^0R_1$ is a rotational transformation from the world frame to the base frame of the task robot. Namely, in view of (16) ~ (18), the movements of the mobile robot and task robot are involved with the movement of end-effector.

3. Algorithm for System Application

3.1 Task planning for minimal movement

In the task of moving an object from some point to the desired point for the mobile manipulator, the position of the base frame of the task robot varies according to the movement of the mobile robot. Therefore through the inverse kinematics, the task planning has many solutions with respect to the robot movement. For the robot to perform the task efficiently, after defining the constraint condition, we must find an accurate solution satisfying both the optimal accomplishment of the task and the efficient completion of the task. In this paper, we have the objective of minimization of movement of the whole robot in performing the task, so we express the vector for the mobile manipulator states as (19):

$$q=\begin{bmatrix} q_m \\ q_t \end{bmatrix} \tag{19}$$

where $q_m=[x_m\ y_m\ z_m\ \theta_m]^T$ and $q_t=[\theta_1\ \theta_2\ \theta_3\ \theta_4\ \theta_5]^T$.

Here, q is the vector for the mobile manipulator, and consists of q_m, representing the position and direction of mobile robot in Cartesian space, and q_t, the joint variable to each link i of the task robot. Now to plan the task to minimize the whole movement of mobile manipulators, a cost function, L, is defined as

$$\begin{aligned} L&=\Delta q^T\Delta q=(q_f-q_i)^T(q_f-q_i) \\ &=(q_{m,f}-{}_{m,i})^T(q_{m,f}-q_{m,i}) \\ &\quad+(q_{t,f}-q_{t,i})^T(q_{t,f}-q_{t,i}) \end{aligned} \tag{20}$$

Here, $q_i=[q_{m,i}\ \ q_{t,i}]^T$ represents the initial states of the mobile manipulator, and $q_f=[q_{m,f}\ q_{t,f}]^T$ represents the final states after having accomplished the task. In the final states, the end-effector of the task robot must be placed at the desired position $X_{t,d}$. For that, Eq. (21) must be satisfied. In (21), we denote as $R(\theta_{m,f})$ and $f(q_{t,f})$, respectively, the rotational transformation to the $X-Y$ plane and kinematics equation of the task robot (Sam-Sang You, and Seok-Kwon Jeong, 1998).

$$X_{t,d}=R(\theta_{m,f})f(q_{t,f})+X_{m,f} \tag{21}$$

where $X_{t,d}$ represents the desired position the of task robot, and $X_{m,f}$ is the final position of the mobile robot.

We can express the final position of the mobile robot $X_{m,f}$ as the function of the desired coordinate $X_{t,d}$, joint variables $\theta_{m,f}$ and $q_{t,f}$. Then the cost function that represents the robot movement is expressed as the $n\times 1$ space function of $\theta_{m,f}$ and $q_{t,f}$ as (22):

$$\begin{aligned} L=&\{X_{t,d}-R(\theta_{m,f})f(q_{t,f})-X_{m,i}\}^T \\ &\{X_{t,d}-R(\theta_{m,f})f(q_{t,f})-X_{m,i}\} \\ &+\{q_{t,f}-q_{t,i}\}^T\{q_{t,f}-q_{t,i}\} \end{aligned} \tag{22}$$

In (22), $\theta_{m,f}$ and $q_{t,f}$ which minimize the cost function L must satisfy the condition in (23).

$$\nabla L=\begin{bmatrix} \dfrac{\partial L}{\partial \theta_{m,f}} \\ \cdots\cdots\cdots\cdots \\ \dfrac{\partial L}{\partial q_{t,f}} \end{bmatrix}=0 \tag{23}$$

Because the cost function is nonlinear, it is difficult to find analytically the optimum solution that satisfies (23). So in this paper, we find the solution through the numeric analysis using the gradient method described by (24):

$$\begin{bmatrix} \theta_{m,f(k+1)} \\ q_{t,f(k+1)} \end{bmatrix}=\begin{bmatrix} \theta_{m,f(k)} \\ q_{t,f(k)} \end{bmatrix}-\eta\nabla L|_{\theta_{m,f(k)},q_{t,f(k)}} \tag{24}$$

This recursive process will stop when $\|\nabla L\|<\varepsilon\approx 0$. That is, $\theta_{m,f(k)}$ and $q_{t,f(k)}$ are optimum solutions. Through the optimum solutions of $\theta_{m,f}$ and $q_{t,f}$ the final robot state q_f can be calculated as in (25):

$$q_f=\begin{bmatrix} q_{m,f} \\ q_{t,f} \end{bmatrix}=\begin{bmatrix} X_{t,d}-R(\theta_{m,f})f(q_{t,f}) \\ q_{t,f} \end{bmatrix} \tag{25}$$

There are several efficient searching algorithms. However, the simple gradient method is applied for this case.

3.2 Mobility of mobile robot

As shown in Fig 6, the mobile robot transforms the input into the output in the form of rotation of the wheels, and reaches a different position in the Cartesian coordinate system depending on the initial orientation. By considering the relationship between the traveling and the change of direction, we can have the degree of the robot mobility with respect to the direction that represents the adaptability of the robot-configuration in a given path.

In this research, we define "mobility of the mobile robot" as the amount of the mobile robot movement when the input magnitude of the wheel velocity is unity. That is, the mobility is defined

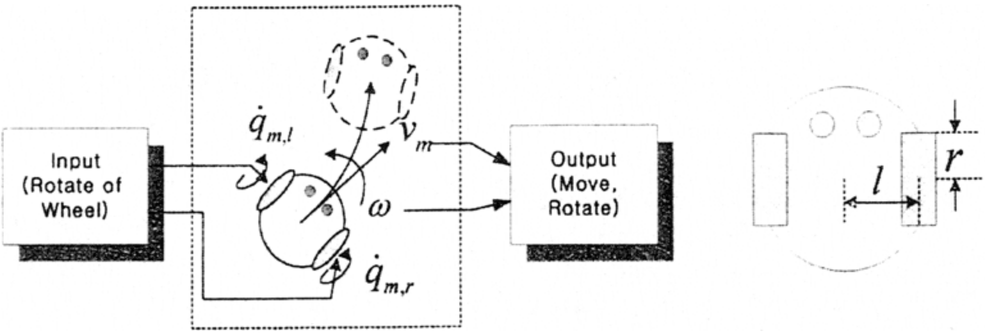


Fig. 6 Task of mobile robot

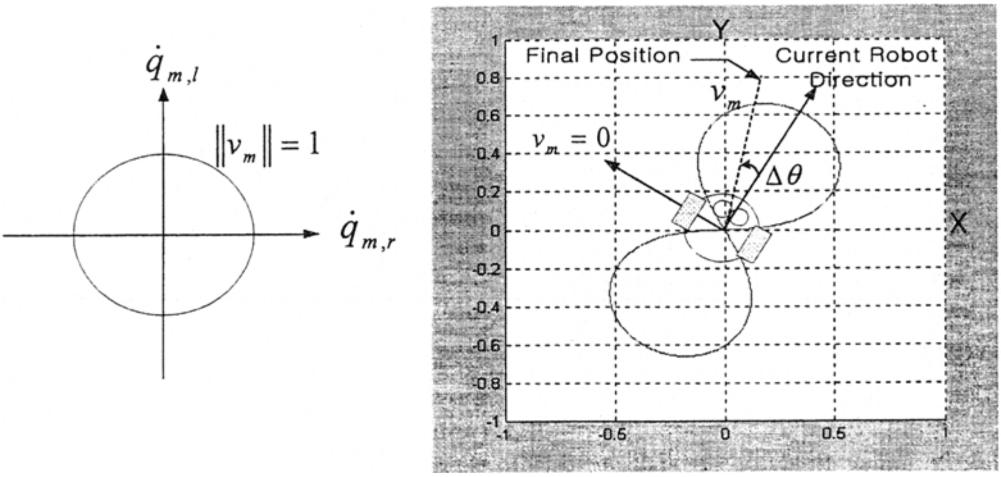


Fig. 7 Motion generation efficiency

as the corresponding quality of movement in any direction. The mobile robot used in this research does move and rotate because each wheel is rotated independently under control. The robot satisfies (26) with the noted kinematics by denoting left and right wheel velocities by $(\dot{q}_{m,l}, \dot{q}_{m,r})$ and the linear velocity and angular velocity by (v_m, ω).

$$v_m = r\frac{\dot{q}_{m,l} + \dot{q}_{m,r}}{2} \tag{26a}$$

$$\omega = \frac{r}{l}\frac{\dot{q}_{m,l} - \dot{q}_{m,r}}{2} \tag{26b}$$

Rewriting equations (26a), (26b), we get equations (27c) and (27d):

$$\dot{q}_{m,r} = \frac{v_m + \omega l}{r} \tag{27c}$$

$$\dot{q}_{m,l} = \frac{v_m - \omega l}{r} \tag{27d}$$

Mobility is the output to input ratio with a unit vector, $\|v_m\| = 1$, or $\dot{q}_{m,l} + \dot{q}_{m,r} = 1$, and the mobility v_m in any angular velocity ω is calculated by Eq. (28):

$$v_m = r\sqrt{\frac{1}{2} - \omega^2\frac{l^2}{r^2}} \tag{28}$$

When the mobile robot has a velocity of unit norm, the mobility of the mobile robot is represented as in Fig. 7. It shows that the output, v and ω in the workspace for all direction inputs that are variations of robot direction and movement. For any input, the direction of maximum movement is the current robot direction when the velocities of the two wheels are the same. In this situation, there does not occur any angular movement of the robot.

3.3 Desired configuration of a task robot for a given task

A task can be described by a set of artificial constraints (M. T. Mason, 1981) which can be controlled. For a given set of motion components, we can define a desired manipulability ellipsoid based upon the task requirements, *i. e.*, either force control or motion control is required along each direction. TOMM (Sukhan Lee and Jang M. Lee, 1988) represents the discrepancy between the desired and actual manipulability ellipsoid. Using this TOMM, an optimal configuration of a task robot for a given task can be searched in its joint space. In this paper, it is assumed that an optimal configuration of a task robot for a given task is pre-obtained and specified.

3.4 Assigning of weighting value using mobility

From the mobility, we can obtain the mobility of the robot in any direction, and the adaptability to a given task in the present posture of the mobile robot. If the present posture of the mobile robot is adaptable to the task, that is, the mobility is large in a certain direction, we impose the lower weighting value on the term in the cost function of Eq. (29) to assign a large amount of movement to the mobile robot along this direction. If not, by imposing a higher weighting value on the term we can make the movement of the mobile robot small. Equation (29) represents the cost function with weighting value:

$$L=\{X_{t,d}-R(\theta_{m,f})f(q_{t,f})-X_m\}^T W_m\{X_{t,d}-R(\theta_{m,f})f(q_{t,f})-X_{m,i}\}+\{q_{t,f}-q_{t,i}\}^T W_t\{q_{t,f}-q_{t,i}\} \quad (29)$$

Here, W_m and W_t are weighting matrices imposed on the movement of the mobile robot and task robot, respectively. In the cost function, the mobility of the mobile robot is expressed in the Cartesian coordinate space, so the weighting matrix W_m of the mobile robot must be applied after decomposing each component to each axis in the Cartesian coordinate system.

As in Fig. 8, we can decompose mobility into two components for the X and Y axes in the Cartesian coordinate system, where α, the relationship between mobility and weighting value, must be satisfied. So, we impose a weighting value on the mobile robot as in Eq. (30):

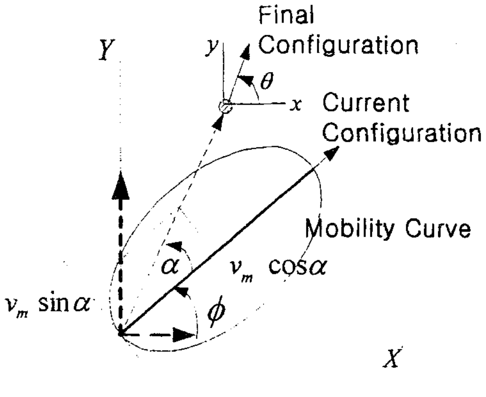


Fig. 8 Decomposing mobility

$$W_m=\begin{bmatrix} w_x & 0 & 0 & 0 \\ 0 & w_y & 0 & 0 \\ 0 & 0 & w_z & 0 \\ 0 & 0 & 0 & w_\theta \end{bmatrix} \quad (30)$$

where

$$w_x=\frac{1}{v\cdot\cos(\phi)\cos(\alpha)+e},$$
$$w_y=\frac{1}{v\cdot\sin(\phi)\sin(\alpha)+e},$$
$$w_z=\frac{k_1}{(z_d-f_z(q_t))^2}, \text{ and } w_\theta=1.$$

The z directional portion of the mobile robot movement is controlled by the weighting value of w_z, which implies that if the current height is far from the desired, the mobile robot takes care of controlling more w_z. For that, we define the weighting index in the form of a second order function as the difference between the current and desired heights of the mobile robot. We also assign the weighting value W_t for the task robot to be fixed in an identity matrix, so that the importance varies relative to the weighting value of the mobile robot. Therefore, by considering the mobility of the mobile robot along the task direction, the robot can accomplish a given task more efficiently through the application of a weighting value in the cost function. Note that w_θ is kept constant since the z directional angular velocity, w_z, is a dependent variable to v_x and v_y.

3.5 Mobile robot control

When the mobile manipulator is far from the desired position, the cooperative control of the mobile and task robots is not considered. Instead the mobile robot carries the task robot to the reachable boundary to the goal position, *i. e.*, within the reachable workspace. We establish the coordinate system as shown in Fig. 9 so that the robot can take the desired posture and position movement from the initial position according to the assignment of the weighting value of the mobile robot to the desired position. After starting at the present position, (x, y_i), the robot reaches the desired position, (x_d, y_d). Here the current robot direction ϕ, the position error α from present position to the desired position, the distance error e to the desired position, and the direction of mobile robot at the desired position θ are noted (N. Hare and Y. Fing, 1997; M. Aicardi, 1995). These are explained in Fig. 9.

Figs 10～12 represent the position movement of a mobile robot by an imposed weighting value. When the mobile robot moves from $P_{m,i}$ to $P_{m,d}$, we want to minimize α and e, and to make θ be in the desired direction at the desired position (James C. Alexander and John H. Maddocks, 1998). For this, the relationship of α and e, θ are described in Eq. (31).

$$\dot{e} = -\nu\cos\alpha \tag{31a}$$

$$\dot{c} = -\omega + \frac{\nu\sin\alpha}{e} \tag{31b}$$

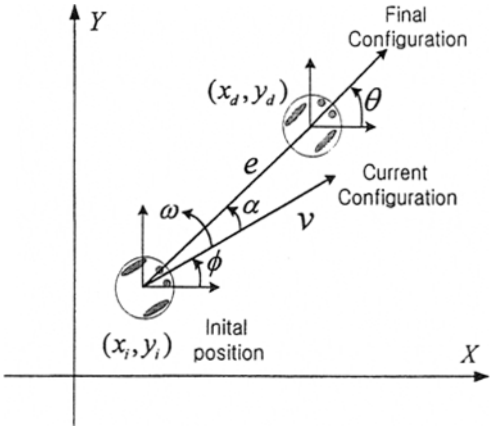


Fig. 9 Position movement of mobile robot by imposed weighting value

$$\dot{\theta} = \frac{\nu\sin\alpha}{e} \tag{31c}$$

A Lyapunov candidate function is defined as in Eq. (32):

$$V = V_1 + V_2 = \frac{1}{2}\lambda e^2 + \frac{1}{2}(\alpha^2 + h\theta^2) \tag{32}$$

where V_1 means the error energy to the distance and V_2 means the error energy in the direction. After differentiating both sides in Eq. (32) with respect to time, we can acquire the result as in Eq. (33).

$$\dot{V} = \dot{V}_1 + \dot{V}_2 = \lambda e\dot{e} + (\alpha\dot{\alpha} + h\theta\dot{\theta}) \tag{33}$$

Let us substitute Eq. (31) into the corresponding part in Eq. (33), so that it results in Eq. (34):

$$V = -\lambda e\nu\cos\alpha + \alpha\left[-\omega + \frac{\nu\sin\alpha}{\alpha}\cdot\frac{(\alpha + h\theta)}{e}\right] \tag{34}$$

Note that $\dot{V} < 0$ is required for a given V to be a stable system. On this basis, we can design the nonlinear controller of the mobile robot as in Eq. (35).

$$\nu = \gamma(e\cos\alpha),\quad (\gamma > 0) \tag{35a}$$

$$\omega = k\alpha + \gamma\frac{\cos\alpha\ \sin\alpha}{\alpha}(\alpha + h\theta),\quad (k,\ h > 0) \tag{35b}$$

Therefore, using this controller for the mobile robot, $\dot{V}$ approaches zero as $t \to \infty$; e α also approaches almost to zero as shown in (36):

$$\dot{V} = -\lambda(\gamma\cos^2\alpha)e^2 - k\alpha^2 \leq 0 \tag{36}$$

3.6 PID Algorithm

In this paper, we use a PID algorithm for controlling the position and velocity of the mobile manipulator. Using the PID algorithm, we can control the important system characteristics, e.g., rise time, steady state error, system stability, etc. Each term in the control algorithm has a different effect on the system characteristics (F. L. Lewis, 1993). In the PID control, the input for control of a standard PID controller in continuous time is stated as in Eq. (37):

$$\begin{aligned} m(t) &= K\left[e(t) + \frac{1}{T_i}\int_0^t e(\tau)\,d\tau + T_d\frac{de(t)}{dt}\right] \\ &= K_p e(t) + K_i\int_0^t e(\tau)\,d\tau + K\frac{de(t)}{dt} \end{aligned} \tag{37}$$

Here $e(t)$ represents the error signal that is the difference between the desired input and output signal. The digital control PID equation in discrete time is expressed as in Eq. (38):

$$\Delta m(k) = K_p \Delta e(k) + K_i e(k) + K_d \Delta^2 e(k) \quad (38)$$

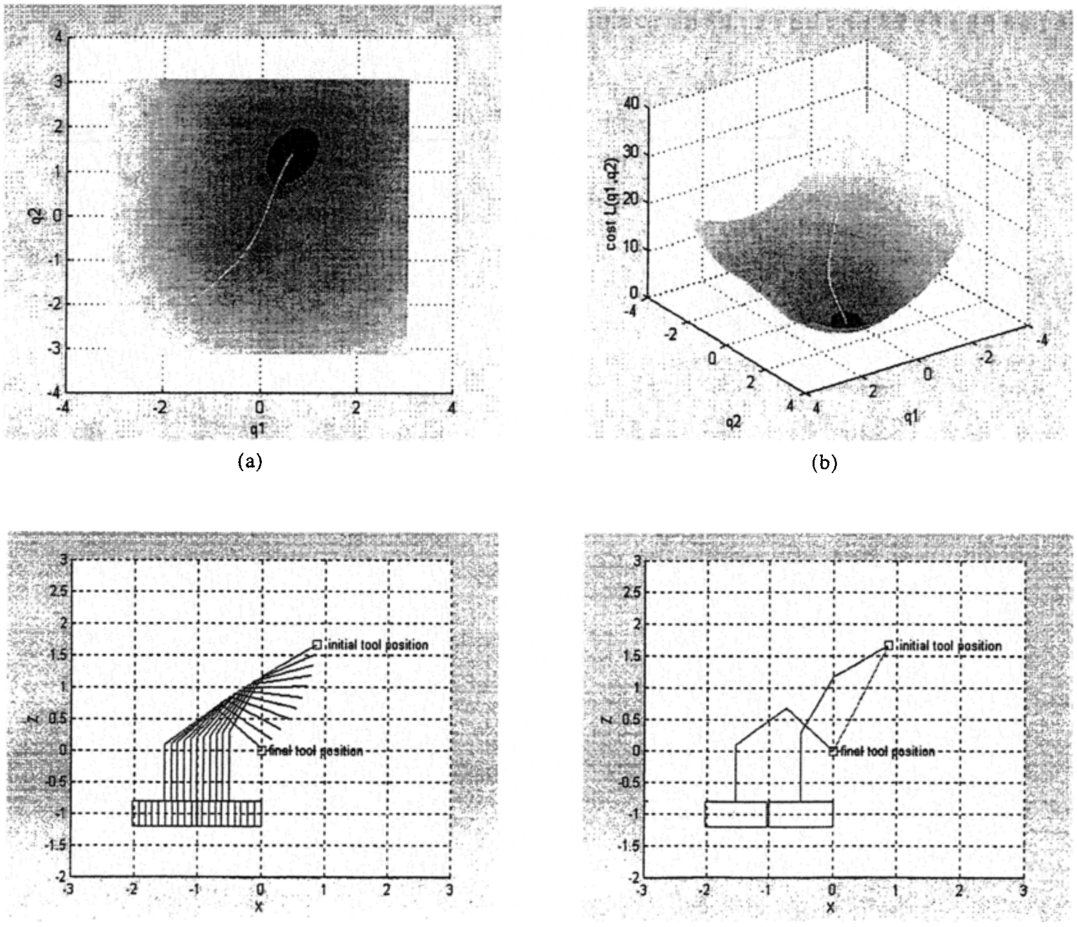
Also, the above Eq. (38) can be rewritten as Eq. (39):

$$\Delta m(k) = \nu_0 e(k) + \nu_1 e(k-1) + \nu_2 e(k-2) \quad (39)$$

where $\nu_0 = K_p + K_i + K_d$, $\nu_1 = -K_p - 2K_d$, and $\nu_2 = 2K_d$.

4. Simulation

For verifying the proposed algorithm, simulations were performed with PURL-Ⅱ. Figure 9 shows an optimal posture for a job for the mobile manipulator. In the simulation, assuming the task robot has 2 links and the mobile robot doesn't change direction, a cost function is represented in the 2-dimensional space of q_1 and q_2. Figure 10 (a) represents the cost function in the $q_1 - q_2$ space; the dark area has a low value. Figure 10 (b) shows the tracking trace to the optimal value of $q_1 - q_2$, (36.2°, 79.3°); dynamic configurations of the task robot during the tracking are illustrated in Fig. 10 (c) and Fig. 10 (d). There may be a lot of methods that enable the task robot to move the end-effect to a desired point through the cooperative control of the task robot and a mobile robot. However, the proposed algorithm is



(a)

(b)

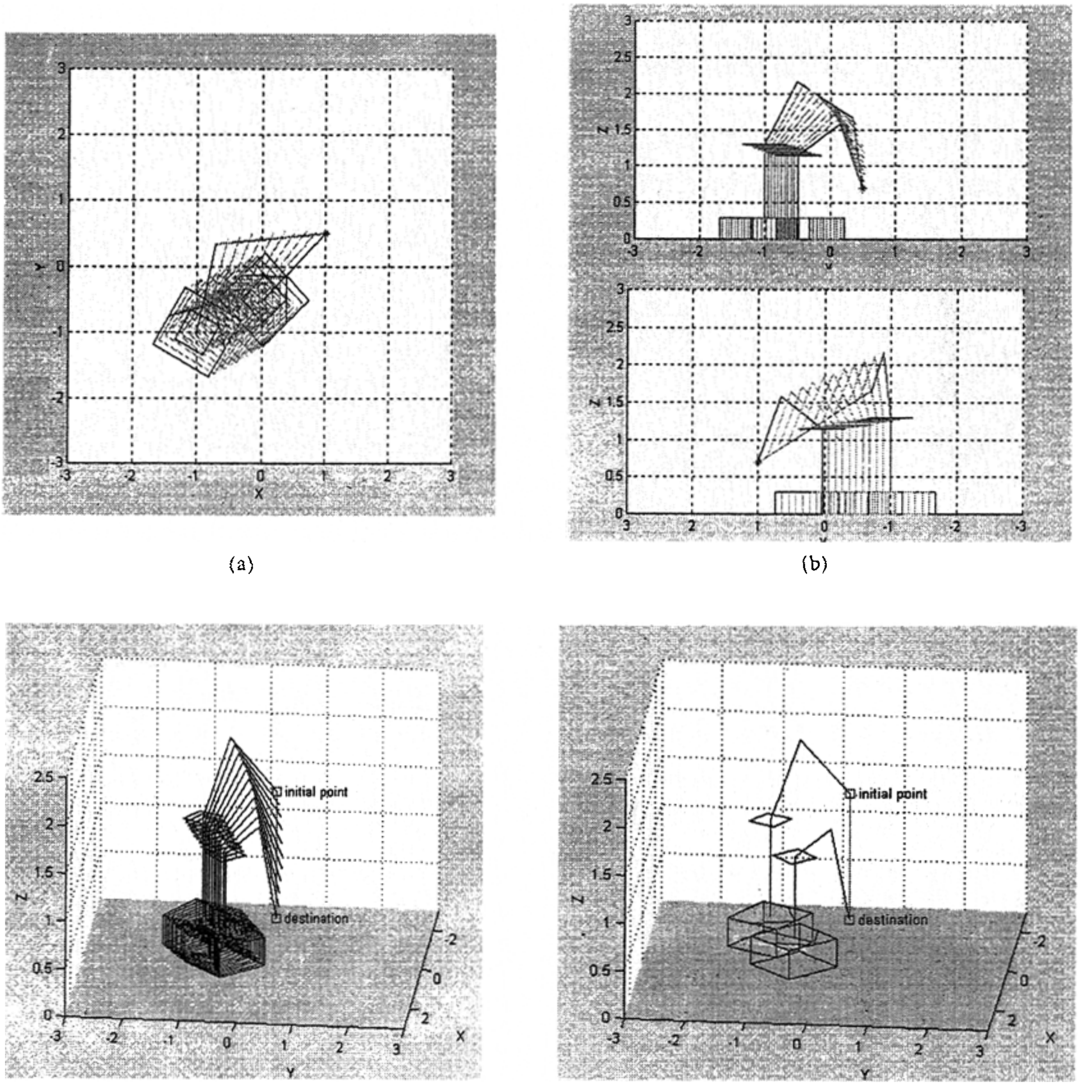
(c)

(d)

Fig. 10 Tracking trace to the optimal position

unique in that it guarantees the minimum movement to a desired configuration. Figure 11 shows the simulation results with a 3 DOF task robot and a 3 DOF mobile robot. The goal is positioning the end-effect to (1, 0.5, 0.7), while the initial configuration of the mobile robot is (−1.0, −1.0, 1.3, 60°) and that of the task robot is (18°, 60°, 90°). The optimally determined configuration of the mobile robot is (0.04, −0.50, 1.14, 44.1°) and that of the task robot is (1.99°, 25.57°, 86.63°). Figure 11 shows movements of the task robot from different points of view. Figure 12 shows simulation results for the initial condition of mobile robot, (−2.0, −0.5, −0.5, −163.6°), the initial condition of task robot (−45.0°, 60.0°, 105.9°) and the desired position of the end-effector, (−1, −1, 1.3). The final configuration of the mobile robot is (0.27, −0.56, 0.70, −139.79 °) and that of the task robot is (1.98°, 25.57°, 86.63°).

Figure 10(a) is the cost function $q_1 - q_2$ in the plane. (b) is the optimum tracking by the gradient method where initial position of mobile robot is (−0.5, 0.3), initial position of the task robot is (60°, 30°), the final position of the mobile robot



(a)

(b)

(c)

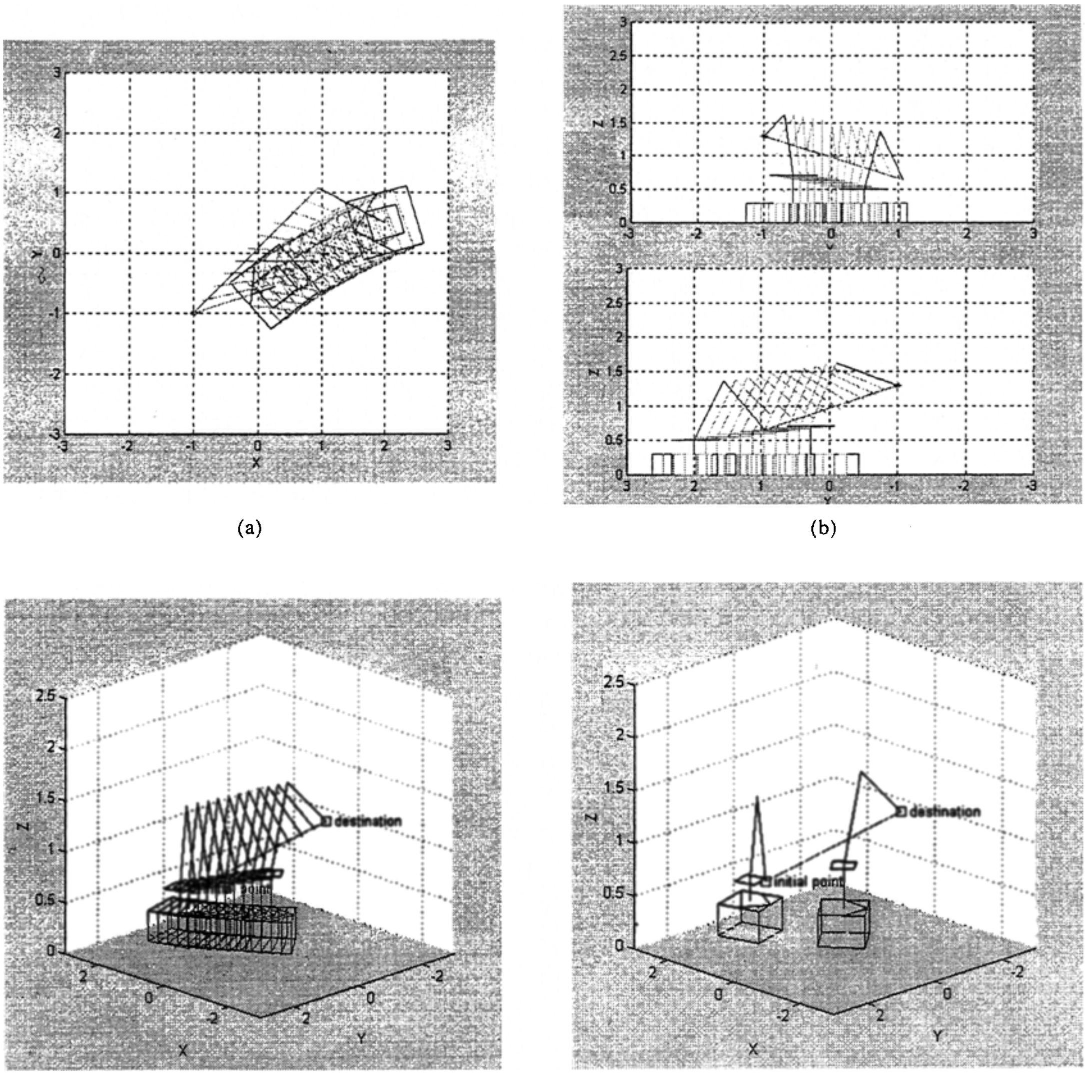
(d)

Fig. 11 The optimal position planning to move a point of action of a robot to (1, 0.5, 0.7)

is (−1.54, 0.09), and the final position of the task robot is (36.22°, 79.28°). (c) is the task of moving a point of action by optimal position planning where (866,1666) → (0, 0). (d) is the initial and final position of the robot.

Figure 11(a) is the movement of the robot in the Z-axis. (b) is the movement of the robot in the Y and Z-axis. (c) is the task of moving a point of action of the robot. (d) is the initial and final position of a robot where the initial position of mobile robot is (−1, −1, 1.3, 60°) the, initial position of task-robot is (18°, 60°, 90°), a desired point of action of a robot is (−0.72, 0.34, 1.66) the, final position of the mobile robot is (0.04, −0.49, 1.14, 44.05°) the, final position of task-robot is (1.98°, 25.57°, 86.63°), and a final point of action of a robot is (1, 0.5, 0.7).

Figure 12(a) is the movement of the robot in the Z-axis. (b) is the movement of the robot at Y and X-axis. (c) is the task moving a point of action of the robot. (d) is the initial and final position of the robot where initial position of mobile robot is (−2, −0.5, −0.5, −163.64°), the initial position of the task robot is (−45°, 60°,



(a) (b)

(c) (d)

Fig. 12 The optimal position planning to move a point of action of a rotot to (−1, −1, −1.3)

105.88°), and a desired point of action of the robot is (−0.95, 1.07, 0.64), the final position of the mobile robot is (0.27, −0.56, 0.70, −139.79°), the final position of the task robot is (−21.15°, 66.64°, 85.48°), and a finial point of action of the robot (−1, −1, 1.3).

5. Experiment

The displacement on each joint of the task robot and the velocity, and position of the mobile robot are calculated on a PC. We can calculate the end-effector position using the inverse kinematics because the z-axis of mobile robot is equal to that of the base frame of the task robot. The corresponding velocity profile of each joint is generated and transferred to the controller. The controller of the mobile robot reads the value from encoders mounted on each motor every 1 -msec and performs PID control, which takes 0.2 msec; it generates a control input in the form of PWM from the desired velocity. The PWM signal is applied to the motor drive of H-bridge type, and a bi-directional motor control is used. At the same time the task robot applies the displacement value of each axis to the task robot controller in serial communication according to the defined protocol, and PID control is then performed with respect to each axis.

The calculated inputs for velocity and position of the mobile robot and the task robot are transferred to the mobile robot controller (CPU1) in serial communication, and the control input to the task robot is transferred to the sensor-interfaced controller in parallel communication. CPU2 applies the control input to the task robot protocol. From Fig. 13 that expresses the control system block diagram, the baud rate for serial communication is 9600 bps while that for parallel communication is 10Mbps.

Before the real experiments, assumptions for the manipulator's operational conditions are set as follows: 1. In the initial stage, the object is in the end-effect or of the task robot. 2. The mobile robot satisfies *pure rolling and non-slippage* conditions. 3. There is no obstacle in the mobile robot path. 4. There is no disturbance of the total system. The scenario for this experiment is illustrated in Fig. 14. The mobile robot that is at the origin of the world frame, initially moves towards the x-axis (the left-most one in Fig. 14). The task robot is configured at the joint angles (18°, 60°, 90°); then the coordinate of the end-effecton is set up for (0.02, 0.04, 1.30). From this location, the mobile manipulator must bring the object to (1, 1.5, 0.5). In the first stage, the PURL-Ⅱ robot as a mobile manipulator carries the object from the initial space to the final operating space, and the object is placed at the designed position by PURL-Ⅱ as a simple task robot. The final operation

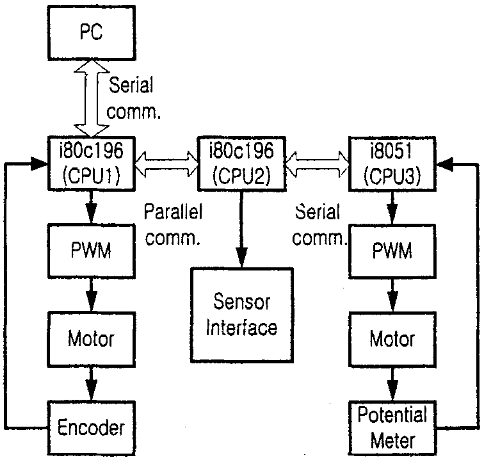


Fig. 13 Control system block diagram

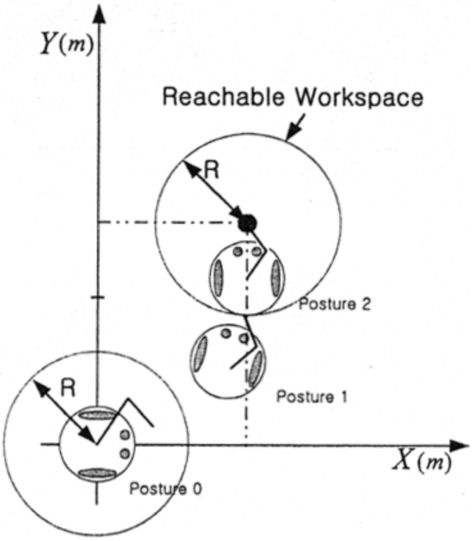


Fig. 14 Scenario of an ideal experiment

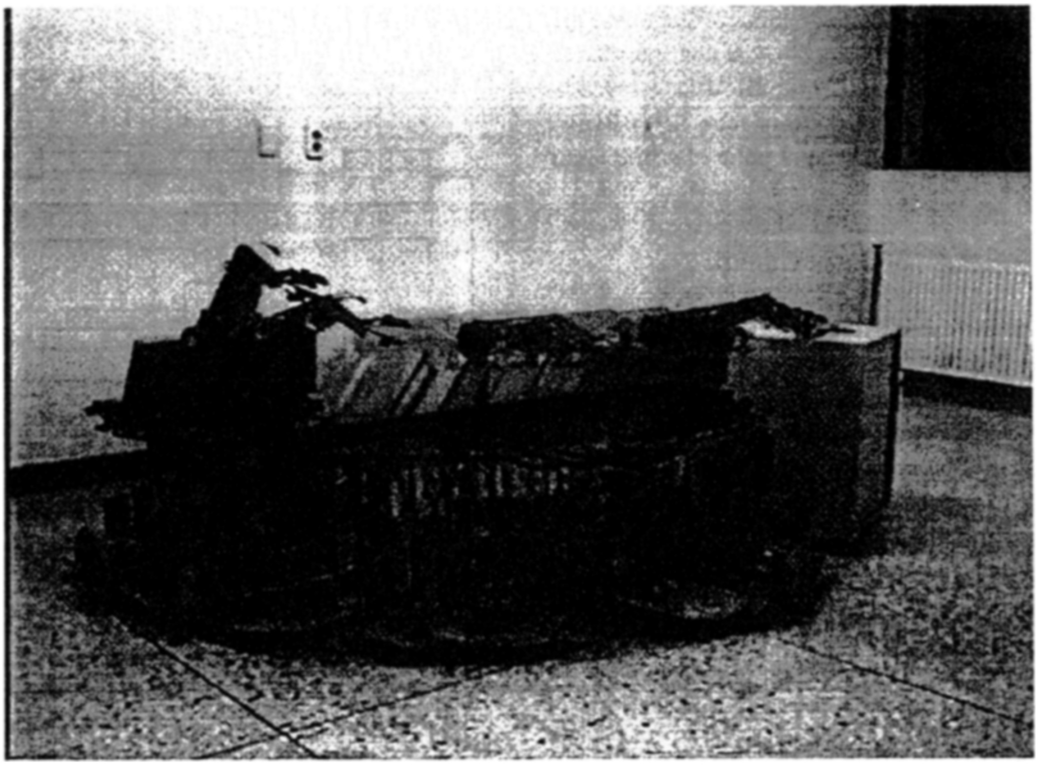


Fig. 15 Response of robot posture

space is pre-determined with the given final location of the object and the desired configuration for the task robot.

First, the mobile robot goes ahead in running down the α and e from the initial place of the task-robot's end-effector to the operating space. With the mobility and gradient algorithm described previously, the mobile robot and the task-robot are moving into their optimal configuration under a cooperative control mode.

An optimal path which is calculated using the algorithm as stated in the previous section has $W_x=10.0$, $W_y=10.0$, and $W_z=2.66$. Forelly the mobile robots angle is 76.52° from the X axis; the difference is coming from the moving of the right wheels by 0.8m and the moving of the left wheels by 1.4m. Next, the mobile robot is different from the X axis by 11.64° with the right wheels moving 0.4m and the left wheels moving. 0.5m Hence, the total moving distance of the mobile robot is (1.2m, 1.9m), the total angle is 88.16°, and each joint angle of the task robot is (−6.45°, 9.87°, 34.92°).

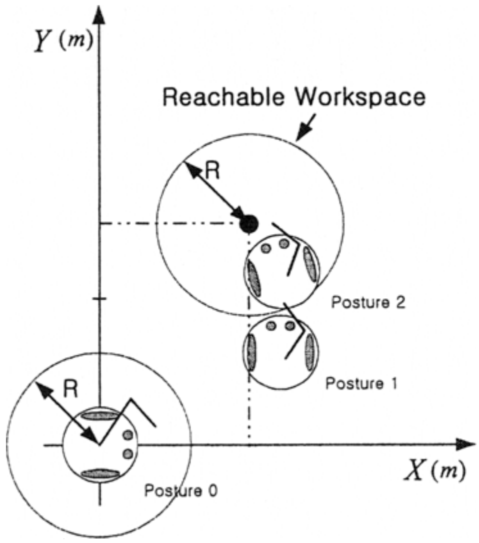


Fig. 16 Real experimental coordinates model

The experimental results are shown by the photograph in Fig. 15. 24 frames per second are being stored in the results, including an initial posture and a final posture. The displayed frames are selected frames among 10 consecutive frames, and they are overlapped one by one. With the photograph, the coordinate and the distance is difficult to see so the real experimental coordinate model is represented in Fig. 16.

In the real experiment, the wheel *pure rolling* condition is not satisfied; also by the control of the velocity through the robot kinematics, the distance error occurs from the cumulative velocity error. Using a timer in the CPU for estimating velocity, timer error also causes velocity error. Hence, real experimental result is shown in Fig. 16. First, the right wheel is moving by 1.1m, the left wheel is moving by 1.8m, and the error is (0. 3m, 0.4m). For this period, the mobile robot reduces α and e gradually until it carries the task robot to the reachable workspace, posture 1. Next, the moving distance of 0.43m, 0.52m causes an error of (0.03m, 0.02m). During this period, the mobile and task robots are cooperating to arrive at the desired position with a desired configuration of the task robot. Because controlling the position of the task robot is implemented through the link angle measurement by means of a variable resistor, the error can be ignored. consequently the final position of the end-effect a is placed at (1.2, 1.5, 0.8) on the object.

Figure 17 and 18 express the velocity profile of the right and left wheels, respectively.

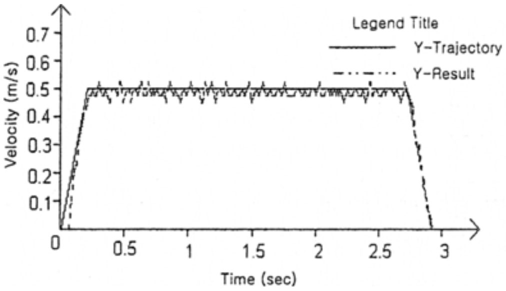


Fig. 17 Velocity profile of right wheel

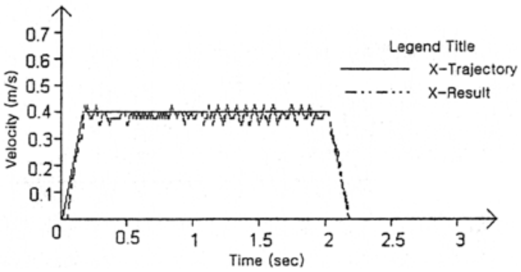


Fig. 18 Velocity profile of left wheel

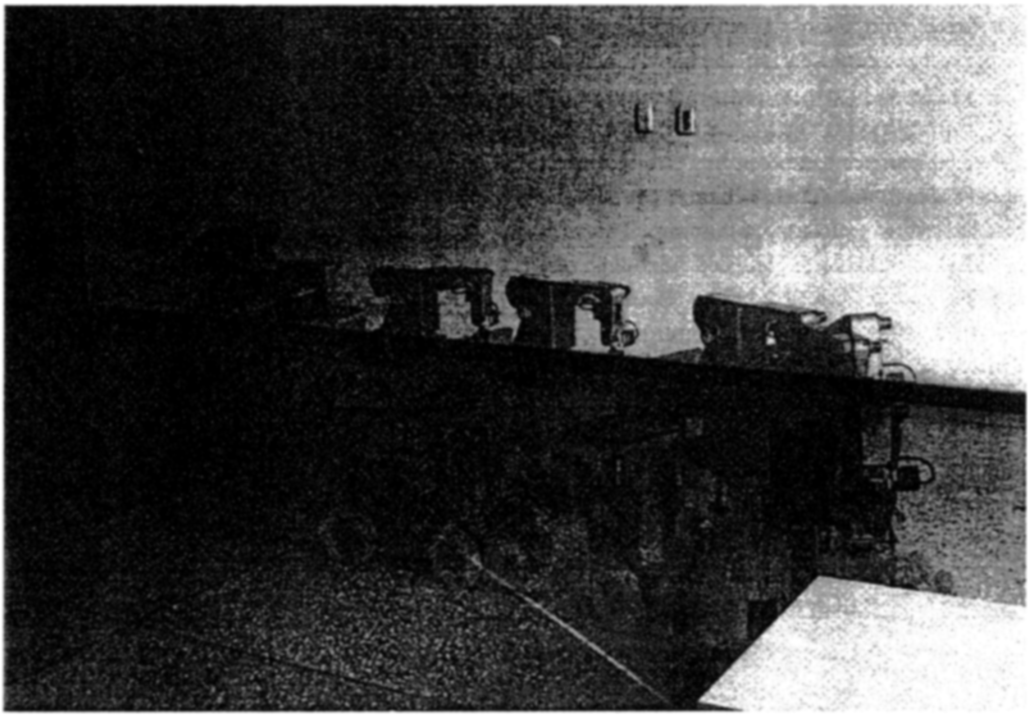


Fig. 19 Response of robot posture

Also, to show the reliability of this system and the algorithm, a second experiment is performed with the same assumptions for the first experiment. The mobile manipulator is setup for an initial position, (0, 0., 0.5). From this location, the mobile manipulator is ordered to bring the object to (3, 3, 0.5). The mobile manipulator performed the task successfully and placed the end-effector at (2.72, 2.85, 0.43) as desired. The experimental result is shown by the photograph in Fig. 19.

6. Conclusions

A new redundancy resolution scheme for a mobile manipulator was proposed in this paper. While the mobile robot is moving from one task (starting) point to the next task point, the task robot is controlled to have the posture proper to the next task, which can be pre-determined based upon TOMM (Jang Myung Lee, 1998). In effect, work efficiency was improved on the basis of an optimum sufficient condition that is defined as the discrepancy to a desired configuration. The configuration discrepancies of the two local robots are combined into a cost function through the weighting values representing the newly defined mobility of the mobile robot. Minimization of the cost function following the gradient method leads a mobile manipulator to an optimal configuration at the new task point. These schemes can also be applied to robot trajectory planning. The efficiency of this scheme is verified through real experiments with PURL-Ⅱ. The experimental results were shown by the photograph in Fig. 15. In further worh, it is necessary that a proper control algorithm be developed to improve control accuracy as well as efficiency in utilizing redundancy. Also, a precise position control scheme that can calibrate the position error in velocity control needs to be developed.

References

Francois G. Pin, 1994, "Using Minimax Approaches to Plan Optimal Task Commutation Configuration for Combined Mobile Platform -Manipulator System," *IEEE Transaction on Robotics and Automation*, Vol. 10, No. 1, pp. 44 ~53

Tsuneo Yoshikawa, 1985, "Manipulability of Robotic Mechanisms," *The International Journal of Robotics Research*, Vol. 4, No. 2, pp. 3~9

Stephen L. Chiu, 1998, "Task Compatibility of Manipulator Postures," *The International Journal of Robotics Research*, Vol. 7, No. 5, pp. 13 ~21

Mark W. Spong, 1989, *Robot Dynamics and Control*, pp. 92~101, John Wiley & Sons.

Jin-Hee Jang, and Chang-Soo Han, 1997, "The State Sensitivity Analysis of the Front Wheel Steering Vehicle: In the Time Domain," *KSME International Journal*, Vol. 11, No. 6, pp. 595 ~604

Keum-Shik Hong, Young-Min Kim, and Chiutai Choi, 1997, "Inverse Kinematics of a Reclaimer: Closed-Form Solution by Exploiting Geometric Constraints," *KSME International Journal*, Vol. 11, No. 6, pp. 629~638

N. Hare and Y. Fing, 1997, "Mobile Robot Path Planning and Tracking an Optimal Control Approach," *International Conference on Control, Automation, Robotics and Vision*, pp. 9~11

Jae-Kyung Lee and Hyung Suck Cho, 1997, "Mobile Manipulator Motion Planning for Multiple Task Using Global Optimization Approach," *Journal of Intelligent and Robotics System*, pp. 169~190

Sam-Sang You, 1996, "A Unified Dynamic Model and Control System for Robotic Mainpulator with Geometric End-Effector Constraints," *KSME International Journal*, Vol. 10, No. 2, pp. 203~212

Sam-Sang You, and Seok-Kwon Jeong, 1998, "Kinematics and Dynamic Modeling for Holonomic Constrained Multiple Robot System through Principle of Workspace Orthogonalization," *KSME International Journal*, Vol. 12, No. 2, pp. 170~180

M. T. Mason, 1981, "Compliance and Force Control for Computer Controlled Manipulators," *IEEE Transaction on Systems, Man, Cybernetics*, Vol. 11, No. 6, pp. 418~432

Sukhan Lee and Jang M. Lee, 1988, "Task-

Oriented Dual-Arm Manipulability and Its Application to Configuration Optimization," *Proceeding 27th IEEE International Conference on Decision and Control*, Austin, TX

M. Aicardi, 1995, "Closed-Loop Steering of Unicycle-like Vehicles via Lyapunov Techniques," *IEEE Robotics and Automation Magazine*, Vol. 10, No. 1, pp. 27~35

James C. Alexander and John H. Maddocks, 1998, "Shortest Distance Path for Wheeled Mobile Robot," *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 5, pp. 657~662

F. L. Lewis, 1993, *Control of Robot Manipulators*, pp. 136~140, Macmillan Publishing.

Jang Myung Lee, 1998, "Dynamic Modeling and Cooperative Control of a Redundant Manipulator Based on Decomposition," *KSME International Journal*, Vol. 12, No. 4, pp. 642~658

J. H. Ju, Jang M. Lee, 1994, "Implementation of a Redundant Robot through the Serial Connection of a Mobile Robot and Task Robot," *Proceeding of the Asian Control Conference Tokyo* (*ASCC '94*), pp. 583~586

Alessandro De Luca, 1997, "Nonholonomic Behavior in Redundant Robots Under Kinematic Control," *IEEE Transactions on Robotics and Automation*, Vol. 13, No. 5, pp. 776~782

David Lim, Homayoun Seraji, 1997, "Configuration Control of a Mobile Dexterous Robot: Real-Time Implementation and Experimentation," *The International Journal of Robotics Research*, Vol. 16, No. 5, pp. 601~618

Homayoun Seraji, 1998, "A Unified Approach to Motion Control of Mobile Manipulator," *The International Journal of Robotics Research*, Vol. 17, No. 2, pp. 107~118

Jea H. Chung, 1998, "Interaction Control of a Redundant Mobile Manipulator," *The International Journal of Robotics Research*, Vol. 17, No. 12, pp. 1302~1309

Tsuneo Yoshikawa, 1990, *Foundation of Robotics*, MIT press, pp. 133~135