

Integrated Task Planning based on Mobility of Mobile Manipulator (M2) Platform

TaeSeok Jin^{*}, Hyun-Sik Kim^{**} and Jong-Wook Kim^{***}

^{*} Dept. of Mechatronics Engineering, DongSeo University, Busan, 617-716, Korea

^{**} Dept. of Robot System Engineering, TongMyong University, Busan, 608-711, Korea

^{***} Dept. of Electronics Engineering, Dong-A University, Busan, 604-714, Korea

Abstract

This paper presents an optimized integrated task planning and control approach for manipulating a nonholonomic robot by mobile manipulators. Then, we derive a kinematics model and a mobility of the mobile manipulator(M2) platform considering it as the combined system of the manipulator and the mobile robot. To improve task execution efficiency utilizing the redundancy, optimal trajectory of the mobile manipulator(M2) platform are maintained while it is moving to a new task point. A cost function for optimality can be defined as a combination of the square errors of the desired and actual configurations of the mobile robot and of the task robot. In the combination of the two square errors, a newly defined mobility of a mobile robot is utilized as a weighting index. With the aid of the gradient method, the cost function is minimized, so the path trajectory that the M2 platform generates is optimized. The simulation results of the 2-link planar nonholonomic M2 platform are given to show the effectiveness of the proposed algorithm.

Key Words : Task Planning, mobile robot, Manipulator, Nonholonomic, Kinematics.

1. Introduction

While a mobile robot can expand the size of the workspace but does not work, a vertical multi-joints robot or manipulator can't move but it can do works. Therefore there has been an increase interest in multiple robots and combination robots to accomplish the given task in cooperation [1]. And at present, there has been a lot of research on the redundant robot which has more degrees of freedom than non combination robots in the given work space, so it can have optimal position and optimized job performance [2],[3]. While there has been a lot of work done on the control for both mobile robot navigation and the fixed manipulator motion, there are few reports on the cooperative control for a robot with movement and manipulation ability[1]. It is desirable to look at improvement in two areas: one is a case where multiple manipulators cooperate and perform the task in parallel, the other case is where a M2 platform performs the task in serial. In this paper, we define a M2 platform as a mobile robot combined with a vertical multi-joint robot and define a vertical multi-joint robot as a task robot. Different from the fixed redundant robot, the M2platform has the characteristic that with respect to the given working environments, it has the merits of abnormal 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 M2 platform with the transportation ability and

dexterous handling in hazardous working environments. This paper utilizes the M2 platform that is a combination in series of a mobile robot that has 3 degrees of freedom for efficient job accomplishment and a task robot that has 5 degrees of freedom. We have analyzed the kinematics and inverse kinematics to define the 'mobility' of the mobile robot as the most important feature of the M2 platform. We researched the optimal position and movement of 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 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 that we acquired by implementing the proposed algorithm through computer simulation and the experiment using M2platform are demonstrated.

2. Structure of M2 Platform

2.1 Configuration of M2 Platform

The robot that was used in our research is shown in Fig. 1. The M2 platform consists of the task robot with 5 degrees of freedom and the mobile robot with 2 degrees of freedom. The mobile robot has two DC servo motors for two different directions of movement. We used Atmel AVR128 microprocessor as the motor controller, so that we could control the three motors concurrently. And configured the motor-drive port with an IGBT in H-bridge form. We mounted the ROB3 with 5 joints as the task robot, and installed a gripper at the end-effector so it can grip things. In addition to that, we

Manuscript received May. 29. 2009; revised Jun. 30. 2009.

^{***} Corresponding Author : kjwook@dau.ac.kr

This study was supported by research funds from Dong-A University.

mounted the portable PC to be used as the host computer to monitor the controller of the M2 platform and to monitor the states of the robot. The block diagram of mobile robot configuration is shown in Fig. 2.



Fig. 1 Structure of M2 Platform

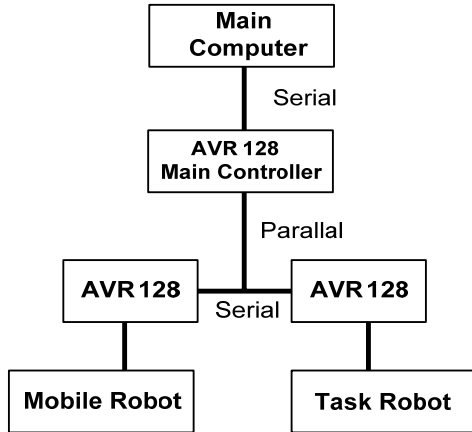


Fig. 2 Block diagram of M2 Platform

2.2 Forward Kinematics Analysis

We analyzed the forward kinematics to calculate the position in Cartesian coordinate system using the variables of the mobile robot [4]. The coordinate system and modeling for the forward kinematics is shown in Fig. 3.

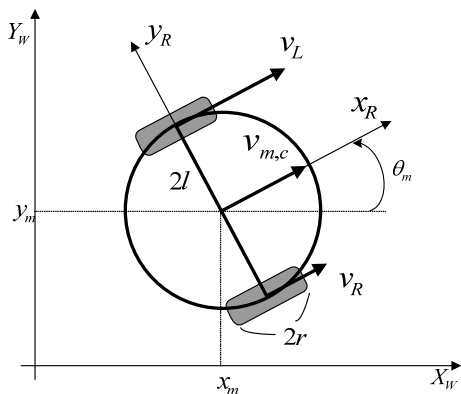


Fig. 3 Mobile robot modeling and coordinate system

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

$$\dot{p}_m = J(p_m)\dot{q}_m \quad (1)$$

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

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

2.3 Inverse Kinematics Analysis

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

$$\dot{q}_m = J(p_m)^+ \dot{p}_m = (J^T J)^{-1} J^T \dot{p}_m \quad (3)$$

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

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

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

And for the left wheel, as in (6).

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

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

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

Let us denote the wheel radius as r , the distance between wheel and the center as l . Then \dot{x}_m , \dot{y}_m , and $\dot{\theta}_m$ are calculated by (8),(9),(10) as follows and we can express the Jacobian matrix as (11).

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

$$\dot{y}_m = \frac{r}{2}(\dot{q}_{m,l} + \dot{q}_{m,r})\sin\theta_m \quad (9)$$

$$\dot{\theta}_m = \frac{r}{2l}(\dot{q}_{m,l} - \dot{q}_{m,r}) \quad (10)$$

$$\dot{p}_m = J\dot{q}_m \tag{11}$$

where $\dot{p}_m = [\dot{x}_m \ \dot{y}_m \ \dot{z}_m \ \dot{\theta}_m]^T$, $\dot{q}_m = [\dot{q}_{m,r} \ \dot{q}_{m,l} \ \dot{q}_{m,z}]$ and Jacobian matrix $J_{4 \times 3}$ can be defined by (12)[5].

$$J_{4 \times 3} = \begin{bmatrix} \frac{r}{2} \cos \theta_m & \frac{r}{2} \cos \theta_m & 0 \\ \frac{r}{2} \sin \theta_m & \frac{r}{2} \sin \theta_m & 0 \\ 0 & 0 & 1 \\ -\frac{r}{2l} & \frac{r}{2l} & 0 \end{bmatrix} \tag{12}$$

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 M2 platform, the position of base frame of task robot varies according to the movement of mobile robot. Therefore through inverse kinematics, the task planning has many solutions with respect to the robot movement. For the robot to perform the task efficiently first after defining the constraint condition, we must find the accurate solution to satisfy 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 M2 platform states as (13).

$$q = \begin{bmatrix} q_m \\ q_t \end{bmatrix} \tag{13}$$

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 M2 platform and consists of q_m representing the position and direction of mobile robot in Cartesian space and q_t , the joint variable to each n link of the task robot. Now to plan the task to minimize the whole movement of M2 platforms, a cost function, L , is defined as

$$L = \Delta q^T \Delta q = (q_f - q_i)^T (q_f - q_i) \\ = (q_{m,f} - q_{m,i})^T (q_{m,f} - q_{m,i}) + (q_{t,f} - q_{t,i})^T (q_{t,f} - q_{t,i}) \tag{14}$$

Here, $q_i = [q_{m,i} \ q_{t,i}]^T$ represents the initial states of the M2 platform, 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, equation (15) must be satisfied. In (15), we denote as $R(\theta_{m,f})$ and $f(q_{t,f})$, respectively, the rotational transformation to $X-Y$ plane and kinematics equation of task robot[10].

$$X_{t,d} = R(\theta_{m,f})f(q_{t,f}) + X_{m,f} \tag{15}$$

where $X_{t,d}$ represents the desired position of task robot, and $X_{m,f}$ is the final position of 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 (16).

$$L = \{X_{t,d} - R(\theta_{m,f})f(q_{t,f}) - X_{m,i}\}^T \{X_{t,d} - R(\theta_{m,f})f(q_{t,f}) - X_{m,i}\} \\ + \{q_{t,f} - q_{t,i}\}^T \{q_{t,f} - q_{t,i}\} \tag{16}$$

In (16), $\theta_{m,f}$ and $q_{t,f}$ which minimize the cost function L must satisfy the condition in (17).

$$\nabla L = \begin{bmatrix} \frac{\partial L}{\partial \theta_{m,f}} \\ \frac{\partial L}{\partial q_{t,f}} \end{bmatrix} = 0 \tag{17}$$

Because the cost function is nonlinear, it is difficult to find analytically the optimum solution that satisfies (17). So in this papers, we find the solution through the numeric analysis using the gradient method described by (18).

$$\begin{bmatrix} \theta_{m,f(k+1)} \\ q_{t,f(k+1)} \end{bmatrix} = \begin{bmatrix} \theta_{m,f(k)} \\ q_{t,f(k)} \end{bmatrix} - \eta \nabla L \Big|_{\theta_{m,f(k)}, q_{t,f(k)}} \tag{18}$$

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 (19).

$$q_f = \begin{bmatrix} q_{m,f} \\ q_{t,f} \end{bmatrix} = \begin{bmatrix} X_{t,d} - R(\theta_{m,f})f(q_{t,f}) \\ q_{t,f} \end{bmatrix} \tag{19}$$

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. 4, the mobile robot transforms input into output in the form of rotation of the wheels and then has a different position in 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 direction, and that represents the adaptability of robots configuration in a given path.

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

the control. The robot satisfies (20),(21) with remarked kinematics by denoting left, right wheel velocities ($\dot{q}_{m,l}, \dot{q}_{m,r}$) and linear velocity and angular velocity (v_m, ω).

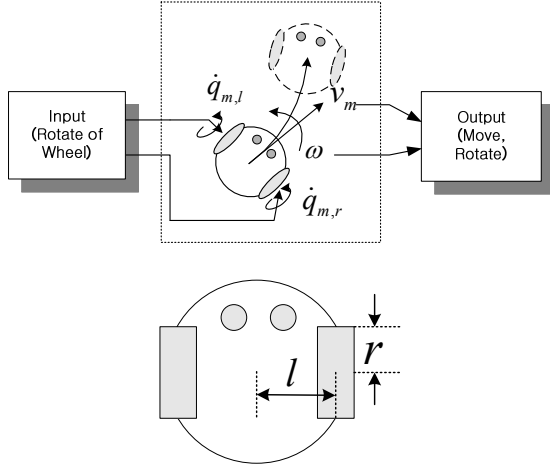


Fig. 4 Task of mobile robot

$$v_m = r \frac{\dot{q}_{m,l} + \dot{q}_{m,r}}{2} \quad (20)$$

$$\omega = \frac{r}{l} \frac{\dot{q}_{m,l} - \dot{q}_{m,r}}{2} \quad (21)$$

Rewriting the equation (20), (21), we get the equations (22) and (23).

$$\dot{q}_{m,r} = \frac{v_m + \omega l}{r} \quad (22)$$

$$\dot{q}_{m,l} = \frac{v_m - \omega l}{r} \quad (23)$$

Mobility is the output to input ratio with a unity vector, $\|v_m\|=1$, or $\dot{q}_{m,l}^2 + \dot{q}_{m,r}^2 = 1$ and the mobility v_m in any angular velocity ω is calculated by equation (24).

$$v_m = r \sqrt{\frac{1}{2} - \frac{\omega^2 l^2}{r^2}} \quad (24)$$

When the mobile robot has the velocity of unity norm, the mobility of mobile robot is represented as Fig. 5. It shows that the output, v and ω , in workspace for all direction inputs that are variance of robot direction and movement. For any input, the direction of maximum movement is current robot direction when the velocities of two wheels are same.

3.3 Desired Configuration of a Task Robot for a Given Task

A task can be described by a set of artificial constraint [11] which can be controlled. For a given set of motion components, we can defined a desired manipulability ellipsoid based upon the task requirements, i.e., either force control or motion control is required along each direction. TOMM [16] 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.

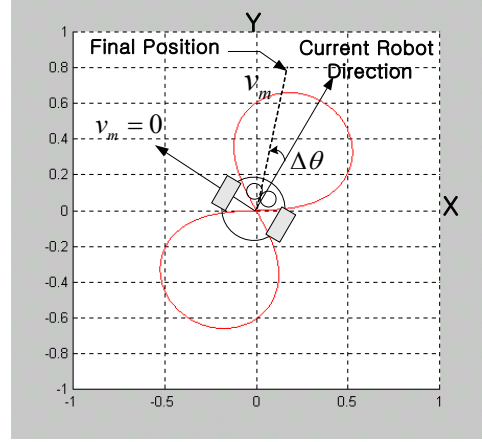


Fig. 5 Motion generation efficiency

3.4 Assigning of Weighting Value Using Mobility

From the mobility, we can know the mobility of robot in any direction, and the adaptability to a given task in the present posture of mobile robot. If the present posture of mobile robot is adaptable to the task, that is, the mobility is large to a certain direction, we impose the lower weighting value on the term in the cost function of equation (25) to assign large amount of movement to the mobile along the direction. If not, by imposing the higher weighting value on the term we can make the movement of mobile robot small. Equation (25) represents the cost function with weighting value

$$L = \{X_{t,d} - R(\theta_{m,f})f(q_{t,f}) - X_{m,i}\}^T W_m \{X_{t,d} - R(\theta_{m,f})f(q_{t,f}) - X_{m,i}\} + \{q_{t,f} - q_{t,i}\}^T W_t \{q_{t,f} - q_{t,i}\} \quad (25)$$

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 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 Cartesian coordinate system.

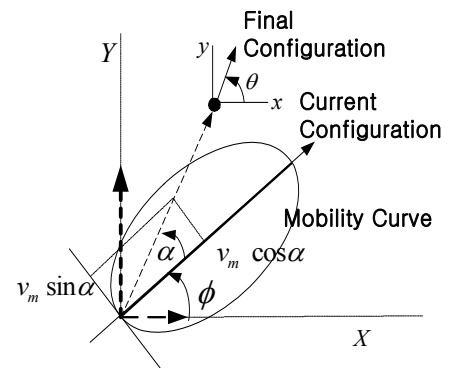


Fig. 6 Decomposing mobility

As in Fig. 6, we can decompose mobility into two components for the X and Y axes in the Cartesian coordinate system. Where α previously remarked the relationship between mobility and weighting value must be satisfied. So, we impose weighting value on the mobile robot as equation (26).

$$W_m = \begin{bmatrix} \omega_x & 0 & 0 & 0 \\ 0 & \omega_y & 0 & 0 \\ 0 & 0 & \omega_z & 0 \\ 0 & 0 & 0 & \omega_\theta \end{bmatrix} \quad (26)$$

where $\omega_x = \frac{1}{v \cdot \cos(\phi) \cos(\alpha) + e}$, $\omega_y = \frac{1}{v \cdot \sin(\phi) \sin(\alpha) + e}$,
 $\omega_z = \frac{k_1}{(z_d - f_z(q_t))^2}$, and $\omega_\theta = 1$.

The z directional portion of the mobile robot movement is controlled by the weighting value of ω_z , which implies that if the current height is far from the desired, the mobile robot takes care of controlling more ω_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. And we also assign the weighting value W_i to be fixed in an identity matrix so that importance varies relative to the weighting value of the mobile robot. Therefore, by considering mobility of the task direction of the mobile robot, the robot can accomplish a given task more efficiently through the application of a weighting value in the cost function. ω_θ is a dependent variable to ω_x , ω_y . Therefore, it is kept constraint.

3.5 Mobile Robot Control

When the M2 platform 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. 7 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_i, 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, the direction of mobile robot at the desired position θ are noted [7,13]. These are explained in Fig. 7.

Fig. 8 represents 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 [14]. For this, the relationship of α and e , θ are described in equation (27),(28),(29).

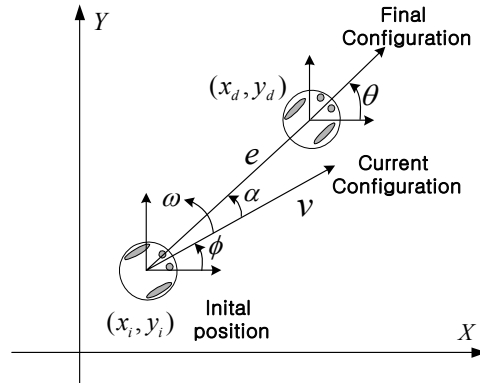


Fig. 7 Position movement of mobile robot by imposed weighting value

$$\dot{e} = -v \cos \alpha \quad (27)$$

$$\dot{\alpha} = -\omega + \frac{v \sin \alpha}{e} \quad (28)$$

$$\dot{\theta} = \frac{v \sin \alpha}{e} \quad (29)$$

A Lyapunov candidate function is defined as in equation (30).

$$V = V_1 + V_2 = \frac{1}{2} \lambda e^2 + \frac{1}{2} (\alpha^2 + h\theta^2) \quad (30)$$

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 equation (32) in terms of time, we can acquire the result as in equation (31).

$$\dot{V} = \dot{V}_1 + \dot{V}_2 = \lambda e \dot{e} + (\alpha \dot{\alpha} + h\theta \dot{\theta}) \quad (31)$$

Let us substitute equation (31) into the corresponding part in equation (31), it results in equation (32).

$$\dot{V} = -\lambda e v \cos \alpha + \alpha \left[-\omega + \frac{v \sin \alpha}{e} \cdot \frac{(\alpha + h\theta)}{e} \right] \quad (32)$$

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 equation (33),(34).

$$v = \gamma (e \cos \alpha), \quad (\gamma > 0) \quad (33)$$

$$\omega = k\alpha + \gamma \frac{\cos \alpha \sin \alpha}{\alpha} (\alpha + h\theta), \quad (k, h > 0) \quad (34)$$

Therefore, using this controller for the mobile robot, \dot{V} approaches to zero as $t \rightarrow \infty$; e and α also approach almost to zero as shown in (35).

$$\dot{V} = -\lambda (\gamma \cos^2 \alpha) e^2 - k\alpha^2 \leq 0 \quad (35)$$

4. Experiments

For verifying the proposed algorithm, simulations were

performed with M2 Platform. In 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 . Fig. 8 (a) represents the cost function in the $q_1 - q_2$ space; the area of dark has low value and the tracking trace to the optimal value of $q_1 - q_2$. Dynamic configurations of the task robot during the tracking are illustrated by Fig. 8 (b). There may be a lot of methods that enables the task robot to move the end-effector to a desired point through the cooperative control of the task robot and a mobile robot. However, the proposed algorithm is unique, that guarantees the minimum movement to a desired configuration.

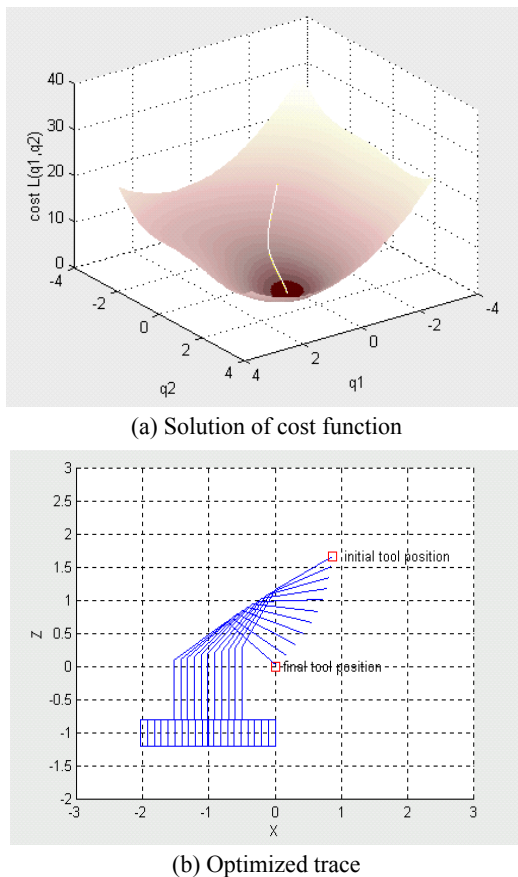
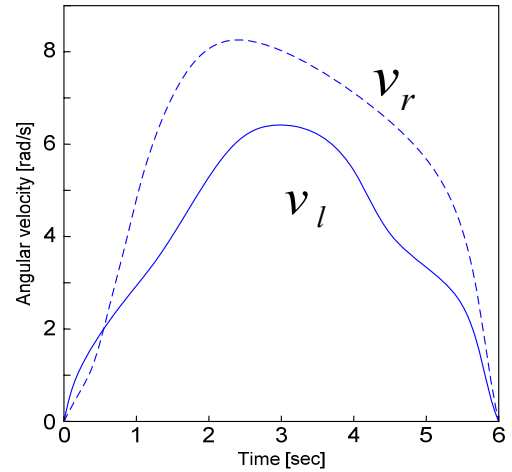


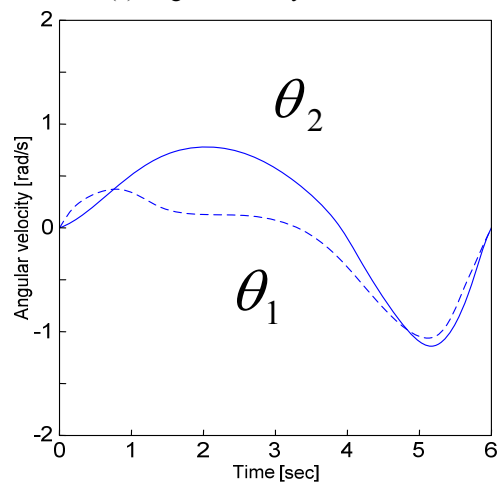
Fig. 8 Tracking trace to the optimal position

Fig. 8 (a) is the cost-function $q_1 - q_2$ on side view. The optimum tracking by the gradient method where initial position of mobile robot is $(-0.5, 0.3)$, initial position of task robot is $(60^\circ, 30^\circ)$, final position of mobile robot is $(-1.5373, 0.0921)$, and final position of task robot $(36.2151^\circ, 79.2887^\circ)$. (b) is the task of moving a point of action by the optimal position planning where $(0.86, 1.66) \rightarrow (0, 0)$. (d) is initial and final position of the robot.

Fig. 9 shows angular velocities of wheels and joints 3, 4. simulation results were given to show the effectiveness of the proposed method.



(a) Angular velocity of wheel



(b) Angular velocity of joint 1, joint 2.

Fig. 9 Simulation results for tracking trace

5. Conclusions

A new redundancy resolution scheme for a M2 platform is 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[15]. In effect, work efficiency was improved on a basis of optimum sufficient condition that is defined as a 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 M2 platform an optimal configuration at the new task point. These schemes can be also applied to the robot trajectory planning. The efficiency of this scheme is verified through the real experiments with M2 Platform. In further study, it is necessary that a proper control algorithm should be developed to improve the control accuracy as well as efficiency in utilizing redundancy.

References

- [1] H.G. Tanner and K.J. Kyriakopoulos, "Nonholonomic Motion Planning For Mobile Manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1233-1238, 2000.
- [2] G. Foulon, J.Y. Fourquet and M. Renaud, "Planning Point to Point Paths for Nonholonomic Mobile Manipulators," *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 374-379, 1998.
- [3] Stephen L. Chiu., "Task Compatibility of Manipulator Postures," *The International Journal of Robotics Research*, Vol. 7, No. 5, pp. 13-21, 1998.
- [4] Mark W. Spong., *Robot Dynamics and Control*, John Wiley & Sons, pp. 92-101, 1989.
- [5] Zacharias, F., Borst, C., Beetz, M., Hirzinger, G., "Positioning mobile manipulators to perform constrained linear trajectories," *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 2578-2584, 2008.
- [6] Keum-Shik Hong, Young-Min Kim, and Chiutai Choi., "Inverse Kinematics of a Reclaimer: Closed-Form Solution by Exploiting Geometric Constraints," *KSME International Journal*, vol. 11, no. 6, pp. 629-638, 1997.
- [7] N. Hare and Y. Fing., "Mobile Robot Path Planning and Tracking an Optimal Control Approach," *International Conference on Control, Automation, Robotics and Vision*, pp. 9-11, 1997.
- [8] Dong Xu., Dongbin Zhao., Jianqiang Yi., Xiangmin Tan., "Trajectory Tracking Control of Omnidirectional Wheeled Mobile Manipulators: Robust Neural Network-Based Sliding Mode Approach," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* vol. 39, Issue 3, pp. 788-799, 2009.
- [9] Dong Hun Shin., Hamner, B.S., Singh, S., Myung Hwangbo., "Motion planning for a mobile manipulator with imprecise locomotion," *Intelligent Robots and Systems, 2003. (IROS 2003). Proc. 2003 IEEE/RSJ International Conference on*, vol. 1, pp. 847-853, 2003.
- [10] Abdessemed, F., "Trajectory generation for mobile manipulators using a learning method," *Control & Automation, 2007. MED '07. Mediterranean Conference on*, pp. 1-6, 2007.
- [11] M. T. Mason., "Compliance and Force Control for Computer Controlled Manipulators," *IEEE Transaction on Systems, Man, Cybernetics*, vol. 11, no. 6, pp. 418-432, 1981.
- [12] Dong Xu., Dongbin Zhao, Jianqiang Yi, Xiangmin Tan, "Robust adaptive tracking control of omnidirectional wheeled mobile manipulators," *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp.3598-3603, 2007.
- [13] M. Aicardi., "Closed-Loop Steering of Unicycle-like Vehicles via Lyapunov Techniques," *IEEE Robotics and*

Automation Magazine, vol. 10, no. 1, pp.27-35, 1995.

- [14] James C. Alexander and John H. Maddocks., "Shortest Distance Path for Wheeled Mobile Robot," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 5, pp. 657-662, 1998.
- [15] Jang Myung Lee., "Dynamic Modeling and Cooperative Control of a Redundant Manipulator Based on Decomposition," *KSME International Journal*, vol. 12, no. 4, pp. 642-658, 1998.



Tae-Seok Jin

He received the Ph.D. degrees from Pusan National University, Busan, Korea, in 2003, in electronics engineering.

He is currently a full-time lecturer at DongSeo University. From 2004 to 2005, he was a Postdoctoral Researcher at the Institute of Industrial Science, The University of Tokyo, Japan. His research interests include network sensors fusion, mobile robots, computer vision, and intelligent control. Dr. Jin is a Member of the KIIS, ICROS, and RSJ.



Hyun-Sik Kim

He received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from Pusan National University, Korea in 1994, 1996, and 2001, respectively. He had been a Senior Researcher at the Agency for Defense Development.

He is now an Assistant Professor of Dept. of Robot System Engineering in Tongmyong University. His research interests include intelligent control, evolutionary algorithms, robotics, underwater vehicle control, target tracking, and underwater target classification.



Jong-Wook Kim

He received the B.S., M.S., and Ph.D. degrees from the Department of Electronics and Electrical Engineering at Pohang University of Science and Technology (POSTECH), Pohang, Korea, in 1998, 2000, and 2004, respectively.

Currently, he is an Assistant Professor in the Department of Electronics Engineering at Dong-A University, Busan, Korea. His current research interests are numerical optimization methods, robot control, intelligent control, diagnosis of electrical systems, and system identification.