Proceedings of the
American Control Conference
Seattle, Washington • June 1995

Efficient Control of a Mobile-Task Robotic System

J.M. Lee[1], M.C. Lee[2], K. Son[2],D.S. Ahn[3], S.H. Han[4], and M.H. Lee[2]

[1]:Department of Electronics Engineering
Pusan National University
Pusan, 609-735, Korea
Fax:(8251) 515-5190
jmlee@hyowon.pusan.ac.kr

[2]:Department of Mechanical Enginering
Pusan National University

[3]:Pusan Fishery University

[4]:KyungNam University

Abstract

The control of a mobile-task robot is implemented with the suitable decoupling of the dynamic interactions between two local robots, *i.e.,* a task robot and a mobile robot. The mobile-task robot is a redundant robot formed by the serial connection of the two robots, and it is suitable for the multiple task execution without any constraints on the workspace. The decoupling process can be initiated when the motions and static forces required for each robot are described explicitly. In this paper, we demonstrate the performance of a redundant robot system controlled by two independent controllers which are coorperating to achieve a common task. By using the two independent controllers, we can achieve a task assigned at the end of the task robot with high speed and high precision. One typical example of task distribution is that fast and rough motions are mainly assigned to the mobile robot, while slow and accurate motions to the task robot. Forming a redundant manipulator through the serial connection of two robots, decoupling the dynamics of the two robots through the task distribution and controlling each of the local robot by an independent controller is a new approach for the design/control of high performance robotic systems.

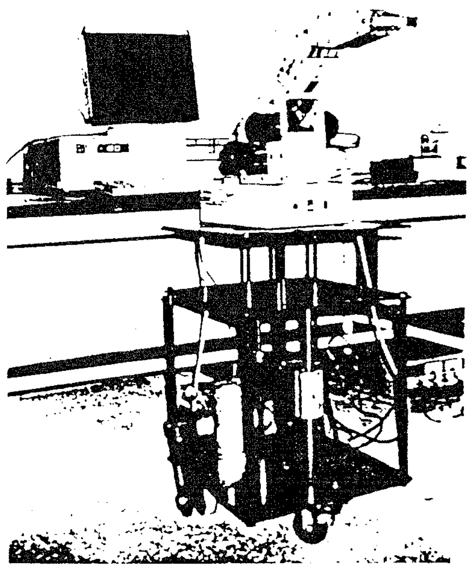
1. Introduction

A conventional manipulator has 6 joints for 6 d.o.f motion with the fixed base. With this 6 d.o.f manipulator, the end-effector assumed to be located at any position/orientation within its own workspace. However, practically there are some geometrical constraints, *i.e.,* singularity avoidance and/or obstacle avoidance, which may cause the manipulator not to be able to perform a given task or may cause the task execution performance to be very low. Especially, when the trajectory for a given task goes through the singular configuration, the positional control error becomes large and high joint torques are required, which cause the control to be unstable. To overcome these kinds of constraints, redundant manipulators are introduced by many of researchers [3,4]. In designing redundant manipulators, a lot of factors need to be added to the geometrical constraints : 1. The manipulator is contacting with the unknown environment during the guarded motion. 2. The manipulator is required to perform various tasks. 3.The manipulator should maintain high task execution performance in the sense of accuracy and speed. 4. The manipulator should be easy to be handled and maintenanced by the users. These kinds of redundant manipulators are implemented by the various types of manipulators, *i.e.,* by adding a joint (either a prismatic or a revolute joint) to the conventional robots or by forming the parallel manipulator structure [11], for example, Stewart Platform.

Recently instead of the fixed base robots, the moving base robots are attracting high interests, since these kind of robots have a lot of advantages compared to the fixed base robots. First of all, the moving base robots have unlimited work space as far as the base can be moved arbitrarily. And also, by the movement of the base, the singularity of the robot can be avoided. However, in the behind of these advantages, we have additional job that is to control the base of the robot, *i.e.,* to control a mobile robot which supports the base of the robot When we use this kind of moving base robot, the upper robot which performs a task can be termed as "Task Robot" and the lower robot which moves the base of the task robot can be termed as "Mobile Robot". The coordination between these two local robots is a critical point of the control of this mobile-task robot system. There are lots of types of mobile-task robot systems conceptually [1,6]. Forming the redundant robot by the serial connection of the two independent robot is a new trial for the redundant robot design. This saves much cost in designing redundant robots. Also, we can change the task robot for a specific type of task easily. We are going to show the design procedure for the mobile-task robot and finally to demonstrate the performance of the mobile-task robot system.

2. Kinematicsof a Mobile-Task Robot

To achieve a unified control of the mobile-task robot for a common task, the whole system should



Pic. 1. Mobile-Task Robot Systemby Serial Connection of Two Independent Robots

be analyzed and controlled in a common frame, since we serially connected a mobile robot and a task robot which have their own controllers individually. Picture 1 shows the mobile-task robot system developed by this research.

Fig. 1 shows the link frames assigned onto the mobile-task robot system and a world frame assigned onto the ground.

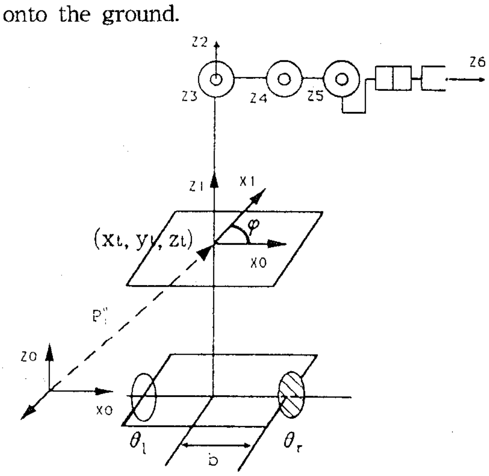


Fig. 1. Link Frames of Mobile-Task Robot System

As it is shown in Fig. 1, this mobile robot can be viewed as a 3 d.o.f robot (x, y and z directional linear motions). Even though each joint has its own controller, x-y planar motion is governed by the motions of two wheels - left and right wheels governing x-y planar motion -. This is a nonholonomic system for which a closed form of kinematic equatiions can not be obtained. Instead, the joint angle for each wheel is obtained for each of the positional variation from the world frame [12] using the following constraint equation.

$$\dot{y}_t \cos\varphi - \dot{x}_t \sin\varphi = 0 \qquad (1)$$
$$\dot{x}_t \cos\varphi + \dot{y}_t \sin\varphi + b\,\dot{\varphi} = r\,\dot{\theta}_r \qquad (2)$$
$$\dot{x}_t \cos\varphi + \dot{y}_t \sin\varphi - b\,\dot{\varphi} = r\,\dot{\theta}_l \qquad (3)$$

where r is the radius of wheels.

We assigned a frame {1} at the end-plate of the mobile robot, which will be used as a base for the task robot. The positioning vector from the world frame to the frame {1}, P_1^0=[Px Py Pz]^T, will be represented as functions of joint variables, θ_1, θ_2, and θ_3, in a complex forms. We are not dealing these issues here, since those are analyzed completely in the reference [10].

We used a conventional 5 axis manipulator, Model: Pro-Arm RS2200, as our task robot which is attached onto the end-plate of the mobile robot.. The link parameter table is shown in the Table 1. In the table, the joint limit for individual joint is shown in the last column.

Joint	α	a	d	θ	Joint Limts
1	0	0	16cm	θ_1	-120° $<\theta_1<$120°
2	90°	20cm	0	θ_2	-18° $<\theta_2<$126°
3	0	15cm	0	θ_3	-100° $<\theta_3<$-10°
4	0	0	0	θ_4	-90° $<\theta_4<$ 90°
5	90°	0	9cm	θ_5	-720° $<\theta_5<$720°

Table 1. Denavit-Hartenberg Parameters and Joint Limits of Task Robot

From the table, forward kinematic equations can be derived. That is, a point in the Cartesian space, X=[Px Py Pz α β γ]^T, can be represented as functions of joint variables, Θ= [θ_1 θ_2 θ_3 θ_4 θ_5]^T, and generally represented by

$$X = T_5^0\,(\Theta) \qquad (4)$$

The orientation is represented as [α β γ] based upon the Euler angles [5]. From the equations of (4), the Jacobian matrix can be also derived.

3. Decoupling of Mobile Robot and Task Robot

The motion at the end-effector of the mobile-task robot can be described in terms of the motions of the two robots:

$$\dot{X} = \dot{X}_m + J\,\dot{q}_t \qquad (5)$$

where $\dot{X}_m$ is the Cartesian space motion of the mobile robot, J is the Jacobian matrix, and $\dot{q}_t$ is the generalized joint velocities of the task robot.

The task denoted by $\dot{X}$ can be achieved either by only $\dot{X}_m$ or by only $J\dot{q}_t$. It is very natural question that how the coordination of the two robots can be achieved for a common goal. There are three ways of coordination: 1. Using the mobile robot and the task robot continuously for a task excution. 2. Using the mobile robot to position the task robot to execute a task well. 3. Fixing the pose of the task robot suitable for a given task and leaving the task execution to the mobile robot. This task distribution scheme is relatively well discribed in [3]. Through the task distribution, the trajectory of the base of the task robot are determined. For obtaining the dynamics of the task robot using recursive Newton-Euler method, this pre-determined velocity (and acceleration) of the base of the task robot can be directly plugged into the velocity propagation equations of the first joint of the task robot:

$$ {}^{1}v_{1} = {}^{1}_{0}R\,({}^{0}v_{0} + {}^{0}\omega_{0} \times {}^{0}P^{0}_{1}) \quad (6)$$

$$ {}^{1}\omega_{1} = {}^{1}_{0}R\,{}^{0}\omega_{0} + \dot{\Theta}_{1}\,{}^{1}\hat{z}_{1} \quad (7)$$

where $[\,{}^{0}v_{0}^{T}\ {}^{0}\omega_{0}^{T}\,]^{T}$ is the task assigned to the mobile robot, ${}^{0}P^{0}_{1}$ is the position vector from the end-plate of the mobile robot to the first joint frame of the task robot, and ${}^{1}_{0}R$ is the rotation matrix of frame {1} w.r.t frame {0}.

For obtaining the dynamics of the mobile robot, the reactive forces calculated from the pre-determined trajectory of the task robot can be directly used. That is,

$$ {}^{0}f_{0} = {}^{0}_{1}R\,{}^{1}f_{1} \quad (8)$$

$$ {}^{0}n_{0} = {}^{0}_{1}R\,{}^{1}n_{1} + {}^{0}P_{1} \times {}^{0}_{1}R\,{}^{1}f_{1} \quad (9)$$

$$ \tau_{0} = {}^{0}n_{0}^{T}\,{}^{0}\hat{z}_{0} \quad (10)$$

where $[\,{}^{1}f_{1}^{T}\ {}^{1}n_{1}^{T}\,]^{T}$ is the 6 dimensional force at the first joint of the task robot, τ_0 is the joint torque required at the last joint of the mobile robot.

Therefore, even though we combined the two local robots and formed a redundant robot, we can implement high speed control of the redundant robot, as far as two controllers are used with the dynamic decoupling between two robots as in our scheme.

4. Composition of a Mobile-Task Robot

The control system has a hierarchical architecture ; 1. A main controller governs both of the mobile and task robots for the higher level processes, for example, trajectory planning, obstacle aviodance and cooperation (task distribution) between two robots. 2. Each joint controller governs the joint control. The control architecture of mobile-task robot system is shown in Fig. 2.

The main controller (IBM-PC) calculates the joint angles - using the inverse kinematic equations - for each joint of the mobile-task robot for the desired position/orientation of the end-effector, and sends the values to the joint controllers. Note that the end-plate of the mobile robot is the same as the base of the task robot. The joint limits and task execution performance of the task robot are monitored by the main controller to access the coordination of the mobile robot if necessary. The main controller also performs the trajectory planning of a given task considering the capability of joint controllers as explained in the previous section. The variational values of the joint angles are finally sent to the joint controllers for the instantaneous velocity command.

Since the task robot is activated by 5 stepping motors with a control unit, the main controller sends

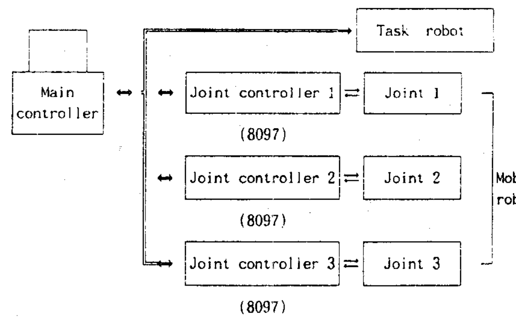


Fig. 2. Architecture of Mobile-Task Robot Control System

to the task robot the step numbers for each joints for each task point through the bi-directional data bus. The mobile robot is composed of three DC servo motor controllers for the x, y and z directional linear motions. Each motor is controlled to track the velocity value - sent from the main controller through the bi-directional data bus by every T_s msec - according to the PD algorithm.

The joint angle values read by the encoders are sent back to the main controller to monitor the control performance of the mobile robot and also to modify the task trajectory based upon the tracking errors. Therefore, the main controller forms a control loop for the whole mobile robot movement.

The mobile robot system is composed of four major parts: a control box unit, the mechanical robot, the interface unit, and the main controller. In Fig. 3, the composition of the mobile robot is represented as a block diagram.

The driving mechanism is composed of two parts:1. Two DC servo motors which drive left and right

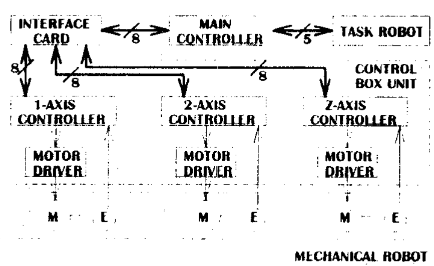


Fig. 3. Block Diagram of Mobile Robotic System

wheels individually to obtain the motion on the x-y plane. 2. One motor which performs the z directional linear motion through a screw gear. The detailed mechanical analysis of the mobile robot is given in the reference [10]. Each of the three joint controllers are implemented on each single board and plugged into a mother board designed for the connections of the joint controllers to the IBM PC bus with high expandability [6-9].

As it is shown in Fig. 3, a motor driver is provided for each motor and an incremental encoder is used for sensing the joint values of each motor. The joint controllers share the bus interface card for the data communication in between the main controller and joint controllers. In addition to these parts, power units for motor drivers and digital circuitry are included in the mobile robot control system [13].

5. Preliminary Experimental Results

To demonstrate the task distribution between the two robots, we selected a task drawing a big rectangular shape which can not be done by only the task robot. To draw a line from the left end to the right end (It is out of the work space of the task robot, unless the base is moved.) on the wall, the base of the task robot needs to be moved to a suitable position. For this purpose, the mobile robot carries the task robot which is attached on the end-plate of the mobile robot to a suitable position. In detail, we used the condition number (which is defined as the ratio of the largest eigenvalue to the smallest eigenvalue of Jacobian matrix, J of the task robot) as a cost function to guide the task distribution between the task robot and the mobile robot.

To make the problem simple, we operated the mobile-task robot within relatively slow speed range so that the dynamic effects can be ignored in the control. The experimental procedure can be summarized as follows:

1) Obtain the values of [θ_1 θ_2 θ_3 θ_4 θ_5] from the desired position/orientation of the end-effector, [Px Py Pz α β γ].
2) For the obtained configuration, the manipulability ellipsoid [2,3] can be determined for the x-z planar motion. From the ellipsoid, we obtain the value of condition number (C.N.). If this value is smaller than the reference value (it is determined heuristically), the task robot performs the task, *i.e.*, moving the end-effector to the position/orientation.
3) If the C.N. is larger than the reference value, the robot configuration assumed to be bad to perform the task. Therefore, the mobile robot moves the base of the task robot. At this moment, to keep the end-effector of the task robot at the same position / orientation (stationary state), the motion of task robot should compensate for the motion of its base generated by the mobile robot. This coordinative control mode should be closely controlled.

① Experimental Operation
The mobile-task robot performs a task to draw a rectangular shape formed by the path (Px, Py, Pz) = (0,44,43)→(30,44,43)→(30,44,0)→(-30,44,0)→(-30,44,43) →(0,44,43) with the same orientation variables, $\beta=0^\circ$ and $\gamma=90^\circ$, while a gripper is holding a pen rigidly.

② Simulation
For the given reference value, C.N.* = 1.37, we checked the configuration of the task robot and modified the configuration by the coordination of the mobile robot during the task execution along the rectangular path. As it is shown in Fig. 5, the C.N. can be kept below the reference value, 1.37 for the whole work path with the cooperation between two local robots.

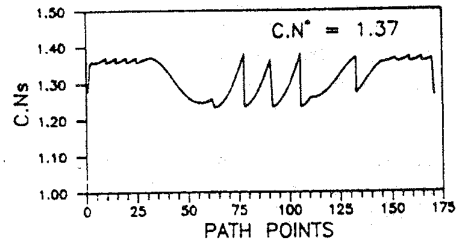


Fig. 5. Variation of C.N. on the Whole Work Path at C.N.* = 1.37

③ Experimental Results
The task described in the simulation sub-section is performed by the mobile-task robot to evaluate the control accuracy of the mobile robot and to demonstrate the coordination scheme of the mobile-task robot.

(1) Experimental Condition

- The center position of the end-plate of the task robot: (0,0,678)(mm) with reference to the world frame.
- Task path of the end-effector of the task robot : A rectangular shape formed by the four points,(X,Y,Z)=(-30,0,43),(30,0,43),(30,0,0)and(-30,0,0).
- Initial posture of tool: α = 90°, β = 0°, γ = 90° (Euler angle)
- Distance between path points : 1.25 (cm)

The passive compliance is added for normal direction to the drawing surface, Y-axis, to prevent the pen from tilting out from the gripper, and to protect the pen. In this experiment, the performance of the task robot is checked with the motions on the X-Z plane.

(2) Results

We checked the cooperation points where the cooperation between the mobile and task robots are required by the simulation and represented in Fig. 6 as numbers from 1 to 12. During the experiment, the coordinative control is required and performed at the same points (point 1 to 12).

Fig. 7 and Fig. 8 show that the positional control errors of x direction, and norm of errors for each of

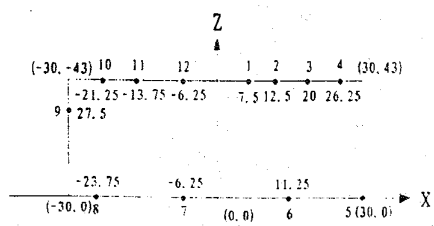


Fig. 6. Cooperation-occurring Points of Two Robots on the X-Z plane

the mobile, the task and the mobile-task robot at the cooperation points (1 to 12), individually.

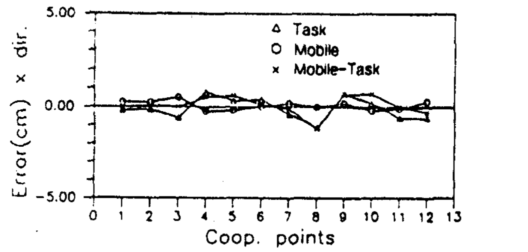


Fig. 7. Tracking Errors on the Cooperation Points of Mobile-Task Robot in the X-axis Direction

(3) Discussion

We demonstrated that the coordination between the two local robots can be performed to maintain the configuration of the task robot

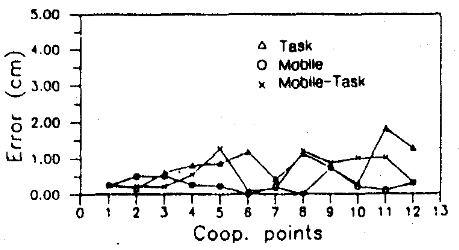


Fig. 8. Tracking Errors on the Cooperation Points of Mobile-Task Robot

to be efficient to execute the task. The C.N. is used as a cost function to guide the cooperation between the two robots. By the experiment, we recognized the positional tracking error of the mobile robot is kept within a few mm through the whole task. This accuracy is not good enough for the commercial robot. However, as it can be noticed from Fig. 7-8, the position error of the mobile-task robot is mostly caused by the task robot which is designed for commercial purposes.

We would like to note one of the interesting point obtained by the experiment: The positional control accuracy of the task robot is decreased with the increase of the C.N. Therefore, we can conclude that the C.N. value is well suited for our demonstration. To improve the control accuracy of the mobile-task robot, we do need to develop a control algorithm which suits with the dynamics of the mobile robot and compensates for the disturbances from the wheels. The task robot needs to be replaced by a high accurate robot to achieve high speed/high accuracy motions as intended.

6. Conclusion

In this paper, we demonstrated a way of forming redundant manipulators by the simple serial connection of two robots: a mobile robot which can provide 3 dimensional linear motion through two wheels and one screw-gear, and a task robot which can be selected and attached at the end-plate of the mobile robot for a specific purpose. For this, we designed and implemented a mobile robot system and tested the performance of the mobile robot in the mobile-task robot system. For the mobile robot, 3 individual joint controllers are designed and implemented. Each controller is governed by the 16 bits micro-controller, Intel N8097BH.

To perform the task more efficiently, we selected the manipulability as a cost function, and optimized the configuration of the task robot during the task execution. The results were satisfactory even though a couple of improvements are required. For the control of the mobile robot, we implemented the PD control algorithm which was suitable to overcome the small distortions, for example, the friction of mechanical joints and the friction of ball-caster with the surface. However, if the task robot is changed during the control process, the distortion might be too large for the control system to compensate for. We do need to develop more robust control algorithm to make the mobile robot suitable for the base of various task robots.

7. References

[1] H. Seraji, "Motion Control of Mobile Mainipulators," Proc. IEEE Conf. on Intelligent Robots and Systems, pp.2056-2063, 1993.
[2] T. Yoshikawa, "Manipulability of Robotic Mechanisms," Proc. 2nd Int. Symp. on Robotics Research, Kyoto, Japan, 1984.
[3] S.Lee and J.M.Lee, "Multiple Task Point Control of a Redundant Manipulator," IEEE Conf. on Robotics and Automation, May 1990.
[4] K.K.Degaldo, M.Long, and H.Seraji, "Kinematic Analysis of 7-DOF Manipulators," Int. Journal of Robotics Research, Vol.11, No.5, 1992.
[5] J.J. Craig, Introduction to Robotics - Mechanics and Control, 2nd edition, Addison-Wesley Publishing Company, New York, 1986.
[6] M.W.Spong and M.Vidyasagar, Robot Dynamics and Control, John Wiley & Sons, New York, 1990.
[6] D.G.Bihn, "Universal Six-Joint Robot Controller," Control System Magazine, pp.31- 35, 1988.
[7] Intel Corporation, 16-Bit Embedded Controllers, Embedded Applications - microprocessor and Peripheral Handbook, Vol.2, O.N. 270250-005, 1990.
[8] Intel Corporation, C-96 User's Guide for Dos System, O.N.167632-002, 1987.
[9] Intel Corporation, MCS-96 Macro Assembler User's Guide, O.N. 122048-001, 1983.
[10] JinHwa Ju and Jang M. Lee, "Implementation of a Redundant Robot Through the Serial Connection of a Mobile Robot and a task Robot,", Proc. of The First Asian Control Conference, 1994.
[11] J-P.Merlet, "Direct Kinematics and Assembly Modes of Parallel Manipulators," Int. Journal of Robotics Research, Vol.11, No.2, 1992.
[12] Yoshio Yamamoto and Xiaoping Yun, "Modeling and Compensation of the Dynamic Interaction of a Mobile Manipulator," IEEE Coference on Robotics and Automation, pp.2187-2192. 1994.
[13] C.S.G LEE and P.R.Chang, "Efficient Parallel Algorithm for Robot Inverse Dynamics Computation," IEEE Transaction on Systems, Man and cybernetics, pp.532-542, 1986.