A reconfiguration system for a dual arm manipulator has been developed which can autonomously change its one arm configuration for use in maintenance and repair tasks for nuclear plant facilities. A connection mechanism and a control algorithm are proposed to easily execute the connection.

Development of Reconfiguration System

Hitachi has been working on a series of manipulators for use in maintenance and repair tasks for nuclear plant facilities. One of the basic problems for such manipulators is the confined and complicated workspace. A solution for this problem is to develop a manipulator that can autonomously change its end effectors, and its configuration, to fit the workspace.

The work to develop such a manipulator has been divided into four steps;

i) development of a dual-arm master-slave manipulator mechanism;
ii) development of autonomous task execution capability;
iii) development of a reconfiguration system, and
iv) evaluation of adaptability of the manipulator configurations to applied tasks.

In the first step, we have demonstrated some demolition tasks using a special hand with a plasma torch [11]. In the second, we achieved autonomous motion; the manipulation system autonomously decided a suitable path to the target that did not collide with the surroundings [2]. The manipulator moved the arms along the decided path, and removed a bolt from the mock-up of a recirculatory system valve of a nuclear power plant.

The results for the third step are presented in this work. The problems of this step are the development of the hardware of the reconfigurable arm, and its control system.

A cellular robot, CEBOT [3] and the modular manipulator systems [4], [5] are best known for their strategies which allow adaptation to various environments by dividing their structure into simple modules. CEBOT, a cellular robotic system, is a realization of the self-organization. Generally, in the case of self-organizing systems, high adaptability to unknown environments and to the various performance goals are required. To this end, CEBOT is a distributed intelligent system without a fixed supervisor. In the case of modular manipulator systems, the determination of the configuration is done by another system, and the configuration task itself is executed by a human.

In the case of our manipulator, the environment is radioactive, requiring the robot to be able to execute reconfiguration tasks by itself. But extensive adaptability is not required, because its performance goals and the environment are predictable. At this point CEBOT has higher adaptability capabilities than needed in this study (less appropriate to the heavy duty use). Consequently, we feel that using a fixed supervisor is a better strategy, because the problem of realization of distributed intelligence can be avoided, thereby reducing the time needed for adaptation.

Basically we feel that motion under constrained situations should be as simple as possible, so we propose a new connection mechanism that enables only one straightforward inlaying motion to connect and to lock the modular joint. Thus, for the control algorithm of connection, a simple compliance control suffices.

With regard to joint attachment or detachment, several research results have appeared [6], [7], where results for the peg in the hole problems [8], [9] or a kind of force control [10], [11] are especially useful for our purpose.

System Overview and Connection Mechanism

In the hardware shown in Fig. 1, the left arm is a reconfigurable arm and the right arm is a normal arm which is designed to execute the reconfiguration task. Although some features...
restrict the connection control design, this mechanism simplifies the control design as discussed later.

**Left Reconfigurable Arm**

The left reconfigurable arm consists of nine modular joints (Table I). The standard configuration is a PUMA-type, i.e., roll-pitch-pitch-roll-pitch-roll-gripper. In addition, this system has one extra pair of roll and pitch modular joints for redundancy, which can be used in case even more degrees of freedom are needed. So this system can achieve 625 configurations (64 combinations without using the extra pair).

Each modular joint comprises two links, an actuator, a joint angle sensor, a brake, a pair of limit sensors, a pair of connectors, and a communication unit. A DC motor serves as the actuator, electric switches as the limit sensors, a rotary encoder as the joint angle sensor, a harmonic drive gear as the reduction mechanism, and an electromagnetic off-brake or dynamic brake by shorting motor serve as braking mechanisms. The connector consists of an electrical connection part and a mechanical connection part. The former is a no insertion force type, and the latter is a ball lock type. The communication unit is a local communication controller which includes a CPU. The communication unit reads the joint status, including the joint angle, brake status, and CPU status, then transfers information to the controller and accepts the controller's command to change the brake status or to reset the encoder counter.

**Right Arm**

The right arm for reconfiguration is also a PUMA-type. While the configuration is fixed, the gripper is exchangeable. A special gripper is used for the reconfiguration, which is designed to release the lock of the modular joint connector by simply grasping it. This makes it easy to carry out the attachment and detachment of the modular joint, since no special motion is required to release or lock the connector under the constrained state. The joint is actuated by the DC motor using the harmonic drive gear; the joint angle is detected by the rotary encoder and the movable range is detected by the potentiometer. For attachment and detachment, a 6-axis force sensor is mounted on the wrist of the right arm.

**Connection Mechanism**

The connection mechanism is shown in Fig. 2. Fig. 2(a) represents the state when the modular joint is not grasped, while Fig. 2(b) shows when it is grasped. As in Fig. 2(a), the sleeve is pushed leftward by the spring-A, pushing down the locking ball. This motion puts the connector in the locked state. The sleeve also pushes down the pin, which pushes down the electrical lock lever and closes the arrayed electrical contact points. In Fig. 2(b), when grasping the modular joint, the finger pulls the sleeve rightward, and the locking ball is released; this causes the connector to be mechanically unlocked and at the same time opens electrical contact points.

**Computer Control System**

The control system consists of a work station, and two sets each of arm controllers and motor drivers. The arm controller incorporates four one-board micro-computers, with I/O boards for the motor driver, the sensors, and the CPUs installed in the
modular joints. Those one-board-micro-computers take partial charge of the control. Their roles and CPU types are listed in Table II.

The motor driver regulates the error of the motor current relative to the reference data, unless the back electromotive force is less than full power (88 V) of the driver. In the right arm controller, there are A/D boards for the 6-axis-force-sensor, counter boards for the encoder, and D/A boards for the motor drivers. In the left arm controller, there are communication boards and D/A boards for the 6-axis-force-sensor, counter boards for the encoder, and D/A boards for the motor drivers.

The supervisor resides in the work station which communicates with an operator to schedule the task order, generates the path plan, and transfers the reference data to each arm controller.

### Controlling Reconfiguration

The reconfiguration tasks are decomposed into inlaying and extracting since the connection mechanism frees the manipulator from extra motion for locking and unlocking. Although simplified, the task still requires some method of force control. This section presents such a force controller design of the right arm that executes reconfiguration. During the reconfiguration the right arm must manipulate various work pieces, i.e., the modular joints, and the posture of manipulation is varied to gain access to the left arm with various configurations. We propose a design in which controller parameters can be chosen easily with no need for detailed kinematic analyses.

### Actuator Control

This manipulator has a joint angle controller instead of a joint force controller, where the actuators are driven by current regulating servo modules. There are three reasons for this.

i) In this system, collision avoidance is the most important problem. In Cartesian space, it is impossible to judge whether the manipulator collides with its surroundings or not, because it has many singular postures. Thus, judgment for collision is made in the joint angle space.

ii) This manipulator includes actuators with reduction gears. Because of friction and backlash, it is impossible to accurately control the output torque of each joint, unlike manipulators with direct drive motors.

iii) This manipulator has no need to move fast, unlike industrial manipulators, and dynamic force interactions induced by the movement of each joint are not so large, because of the actuator system. The inertias of the fast rotating components (a motor rotor, inner rings of bearings, etc.) which are converted into low speed axes are listed in Table III as well as the inertias of all other components. The inertias of the other components are less influential than that of the fast rotating inertia in the link dynamics. Consequently, a simple joint angle control scheme using PID control suffices for this system.

### Compliance Control for Reconfiguration

Reconfiguration is a task with kinematic interactions. In previous work regarding collision with the environment, it has been pointed out that the environment (i.e., the left arm in our case) accepts force inputs and determines motion commands in response, so the environment can be called an admittance and the manipulator should behave as an impedance [11] that accepts motion as inputs and determines output forces in response. But it is not an easy task to control the output force and torque adequately in our manipulator, because of its architecture; i.e., friction and backlash of each joint are unavoidable. So conversely we regarded the environment as the impedance, and the manipulator as the admittance. We feel this is a better strategy in our case, because of the following two reasons:

i) The environment is the reconfigurable arm all of whose joints include gears. These gears have some compliance, because of the inherent character of the harmonic drive system. So the environment can absorb the motion and it is possible to describe the environment as the impedance.

ii) Since the controller of the manipulator that executes reconfiguration is a conventional position control type, it is easy to modify the controller to an admittance type that accepts the force (and the torque) and decides its motion by only adding a 6-axis force sensor to the link on which the end-effector is mounted.

### Table II

<table>
<thead>
<tr>
<th>CPU</th>
<th>MC68020</th>
<th>MC68881</th>
<th>CPU2</th>
<th>MC68020</th>
<th>MC68881</th>
<th>CPU3</th>
<th>MC68020</th>
<th>MC68881</th>
<th>CPU4</th>
<th>MC68000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU1</td>
<td>Cooperative control of CPU2 - CPU4</td>
<td>Communication with supervisor workstation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU2</td>
<td>MC68020</td>
<td>MC68881</td>
<td>Servo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU3</td>
<td>MC68020</td>
<td>MC68881</td>
<td>Motor control</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU4</td>
<td>MC68000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table III

| Components' Inertia of Each Arm Joint for Reconfiguration Tasks at the Initial Posture |
|-----------------------------------------------|-------|------|
| first rotating components [kgm²] | other components [kgm²] |
| roll | pitch | roll | pitch |
| 206.4 | 206.4 | 115.9 | 47.9 |
| 3.4 | 2.97 | 0.024 |

Fig. 3 shows the block diagram of the controller. The interaction force is sensed by the 6-axis force sensor. The sensor can only detect the forces and torques at the sensor point, so the detected values are converted to the values on the connection surface by using its configuration data. It is divided by K, which is the given compliance, and then fed back to accommodate the interaction force with the desired position. This is represented by

\[
X_{i+1} = (1 - \varepsilon)X_i + \varepsilon \left( \frac{F_k}{K} + X_d \right). \tag{1}
\]

Here \(x\) is the position (and posture) vector of the connection surface of the right arm and \(x_d\) is its desired position vector. Modifying this equation, we can get
The term in the bracket \{ \} represents the position error, and \Delta \alpha represents the movement during the sampling interval, so the parameter \( \varepsilon \) changes the motion. In this regard, the parameter \( \varepsilon \) works as the mass and changes the scale of the time response.

Next we consider stability. For simplicity, we use the one-dimensional system in Fig. 4. We assume that the position controller essentially makes the arm a first-order delay system whose time constant is \( \frac{1}{\alpha} \). This system is written as follows using the discrete-time state variable \( x_k \) and sampling interval \( T \):

\[
X_{k+1} = e^{\frac{T}{\alpha} \alpha} x_k + (1 - e^{\frac{T}{\alpha} \alpha}) X_p.
\]  

where \( x_k \) is the free position (and posture) vector of the connection surface of the reconfigurable arm. Substituting (4) and (5) into (3) gives

\[
X_{k+1} = e^{\varepsilon \alpha} X_p + (1 - e^{\varepsilon \alpha}) (1 - \varepsilon + \varepsilon^2 \alpha) x_k + (1 - e^{\varepsilon \alpha})(1 - e^{\varepsilon \alpha}) (1 - \varepsilon + \varepsilon^2 \alpha) x_k \]

which shows that if \( T/\alpha \) is large enough, the system stability is determined by

\[
(1 - \varepsilon) + \varepsilon^2 \alpha < 1
\]

Even if \( T/\alpha \) is not so large, it is easy to show that

\[
(1 - \varepsilon) + \varepsilon^2 \alpha < 1
\]

guarantees system stability. But if \( T/\alpha \) is too small, the system approaches the stability boundary and the compliance controller degrades in performance. Moreover, in that case the desired position \( x_d \) becomes meaningless. Equation (7) and Fig. 5 show that if the environment is too stiff (\( \varepsilon \) is big), the system becomes unstable. When we determine the controller parameters \( K \) and \( \varepsilon \), (7) suggests the appropriate strategy.
The rise time is about 0.27 s, and the error is 0.0013 rad.

from the detected joint angles. To eliminate the gravity effect, gravity compensation is also accomplished using the joint angles and the configuration data.

**Experimental Results**

Fig. 6 shows a step response of the fourth joint, which provides an example of the character of the joint angle controller. The time constant \( \tau \) is about 270 ms. On the other hand the sampling interval \( T \) of the compliance controller is 50 ms; so \( \tau \) is not large (we approximated this system with an ideal discrete system as in prior analyses). Compliance control parameters were chosen according to the strategy indicated by (7), and tuned experimentally; Table IV lists the parameters. From (7) and the value for the parameter \( \epsilon \), it is easy to see that all the poles of the system are near unity. The reconfiguration was successful over 100 trials, where a connection took approximately one minute to complete. Fig. 7 shows typical interaction force and torque responses. The interaction forces and torques settled to values realizing static compliance.

![Fig. 6. Step response of fourth joint. The rise time is about 0.27 s, and the error is 0.0013 rad.](image)

<table>
<thead>
<tr>
<th>Parameters of Compliance Control</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>K</strong> [N/m/(Nm/rad)]</td>
<td>( \epsilon )</td>
</tr>
<tr>
<td>along with x-axis</td>
<td>4900</td>
</tr>
<tr>
<td>along with y-axis</td>
<td>4900</td>
</tr>
<tr>
<td>along with z-axis</td>
<td>4000</td>
</tr>
<tr>
<td>around x-axis</td>
<td>100</td>
</tr>
<tr>
<td>around y-axis</td>
<td>100</td>
</tr>
<tr>
<td>around z-axis</td>
<td>3.0</td>
</tr>
</tbody>
</table>

We have presented the development of a dual arm manipulator that can reconfigure one arm autonomously. The reconfigurable arm consists of eight modular joints and a modular gripper. The developed reconfiguration system has two features. One is concerned with the connection mechanism and the other is concerned with the control scheme during the attachment and the detachment.

The proposed connection mechanism causes the grasping motion of the modular joint to release the electrical and mechanical locks. Thus, the mechanism frees the system from the special locking or unlocking motion under the constrained situation (modular joint in contact with the left reconfigurable arm), and makes the connection control simple. Using this connection mechanism, the attachment and detachment can be decomposed into only the inlaying or extracting motion.

On the other hand, regarding the connection control scheme, impedance control is known to be convenient since it does not require calculation of the inverse Jacobian. But this is less critical with recent advances in microcomputing. Moreover, because our manipulator uses gear reduction for power, side effects caused by friction and backlash occur. And while these effects make impedance control difficult to realize, position control and position control-based compliance control is easier to realize. Thus we feel these control schemes are more feasible than impedance control for gear equipped manipulators like the one in this study. In the case of reconfigurable manipulators, it is possible to consider the environment as the impedance that accepts movement and decides its output forces (unlike that of a tracking task).

The compliance parameters were experimentally chosen for this study, but analysis using a simple kinematic model suggests that increasing stiffness of the left arm makes the system unstable, and increasing the time constant of the position controller or decreasing the sampling interval of the compliance control brings the system to an overdamped situation. Experimental results imply that the chosen parameters drove the system near the overdamped situation, causing a long time (about one minute) to
The analysis suggests that shortening the time constant of the position control is necessary to improve (shorten) this connection time. In addition, we are investigating a form of adaptive control which modifies compliance parameters and estimates $Z$ (environmental compliance) from the sensed forces and displacement.

References


Takaharu Fukuzaki was born in 1947. He received the B.S. and Ph.D. degrees in nuclear engineering in 1971 and 1987, respectively, from the University of Tokyo, Tokyo, Japan. He is currently a Senior Researcher at Energy Research Laboratory, Hitachi, Ltd. Since 1971 he has been engaged in research on computer applications to support operators and engineers of nuclear power plants. His research interests include the improvement of online computer systems for core performance evaluation to realize high availability and flexibility of reactor operations.

Yoshiaki Ichikawa was born in 1955. He received the B.S. degree in mechanical engineering in 1979 and the Ph.D. degree in 1987 from the University of Tokyo, Tokyo, Japan. He is currently a Senior Researcher at Energy Research Laboratory. Since 1979 he has been working on applications of AI technologies and modern control theories to maintenance robots and plant control systems. From 1987 to 1988 he was a Visiting Scientist at Field Robotics Center of Carnegie Mellon University. His current research interests include self-reproducing systems, evolutionary engineering, and cooperative distributed control systems.

Masanori Suzuki received the B.D. and M.E. degrees in electrical engineering from Osaka University in 1976 and 1978, respectively. Since 1978, he has been engaged in the development of remote maintenance robots for nuclear facilities. He is currently a Senior Researcher in Energy Research Laboratory, Hitachi, Ltd. His research interests include robot planning, collision avoidance, reconfiguration arm systems, and remote maintenance robots for reactor vessels.

Shinji Naito was born in 1959. He received the B.S. degree in mechanical engineering in 1982 from the University of Tohoku, Sendai, Japan. Since 1982 he has been a Research Engineer at Energy Research Laboratory, Hitachi, Ltd., working mainly on the developments of wall walking robots, manipulators, and inspection robots for nuclear plant maintenance.

Nobuyoshi Iwatsuka was born in 1963. He graduated from Taira Technical High School in 1981, Taira, Fukushima, Japan. Since 1981 he has been a Research Engineer at Energy Research Laboratory, Hitachi, Ltd., working mainly on the designs of manipulators and mobile robots for automated inspection.