Proceedings of the 1991 IEEE International Symposium
on Intelligent Control
13 - 15 August 1991, Arlington, Virginia, U.S.A.

Strategy Generation and Skill Acquisition for Automated Robotic Assembly Task

D.S.Ahn and H.S.Cho
Korea Advanced Institute of Science & Technology, P.O.Box 150 Chongryangri, Seoul, Korea

K.Ide and F.Miyazaki
Faculty of Engineering Science, Osaka University, Osaka, Japan

S.Arimoto
Faculty of Engineering, University of Tokyo, Tokyo, Japan

Abstract

This paper treats a practical method to generate task strategies applicable to chamferless and high-precision assembly. The difficulties in devising reliable assembly strategies results from various forms of uncertainty such as an imperfect knowledge of the parts being assembled and limitations of the devices performing the assembly.
Our approach to cope with this problem is to have the robot learn the appropriate control response to measured force signals, that is, the mapping relation between sensing data and corrective motion of robot, through iterative task execution.

In this paper, the strategy is acquired by using a learning algorithm and represented with a binary tree type database. Experiments are carried out by taking account of practical prodution facilities. It is shown by experimental results that an ideal mapping is acquired effectively by using the proposed method and the assembly task is carried out smoothly.

1. Introduction

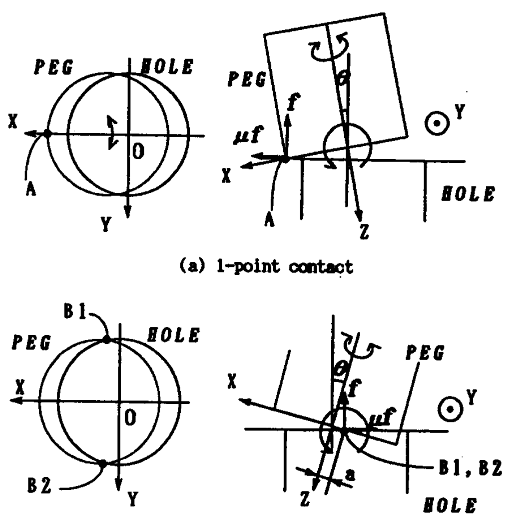
The peg-in-hole assembly problem is still the important subject of many investigations in the robotics field. In particular, the insertion of a peg into a chamfered hole can be readily carried out with the aid of RCC(Remote Center Compliance) installed in an industrial robot[1]. But in the case of nonchamfered and high-precision parts mating, only a few applications were implemented successfully by use of special hardware such as a vibratory equipment[2].
The purpose of this paper is to develop Automated Robotic Assembly System(ARAS) that guarantees successful assembly tasks for the nonchamfered and high-precision parts with the minimum use of the special hardware and commercially available components.
Since a human being can easily perform insertion of parts with small clearance with one's eyes closed, it seems evident that the human insertion is based on the reaction forces created by the contact of the parts.
In fact, in the case of chamfered parts, the assembly tasks have been successfully performed using RCC, under the condition that the peg is in or partly in the hole. In other words, RCC can be regarded as a kind of mapping which correlates force/moment signals with the desired motion. On the other hand, it is well known that chamferless insertion task requires multiple mappings which must be selected in accordance with the contact configurations[3]. In order to develop reliable assembly strategies for the insertion of a peg into a nonchamfered hole, it is necessary to identify contact configurations through which the parts must pass during assembly[4]. Another approach to this problem is to constrain the assembly to some allowable subset of contact configurations for which a solution can be found[5].

ARAS, which we will develop in this paper, depends on the actual engineering practice. ARAS is established through practice in the real task environment which consists of an industrial robot, a force sensor, and small clearance parts(at a clearance level 0.01 mm). Namely, ARAS is constructed and executed according to a given assembly environment.
In general, there may be possibly two pieces of methodology to solve the assembly problem. One is the feedback gain method by servo control, based on explicit control law. Since they should be acquired in advance through the kinematic or dynamic analysis, this method is limited by its own form. The other, the logical branching method, represents corrections of motion by a set of IF-THEN rules(we call them skills) in which conditions of sensor signals are evaluated and appropriate actions are selected in accordance with the sensory information. Though this method produces slow, intermittent actions, it allows us to deal with the assembly problem without complete kinematic or dynamic analysis of task.
The method we will develop allows the robot by itself to learn IF-THEN rules through iterative implementation. The proposed learning mechanism dose not require an explicit representation of control strategies. The strategies for decision-making, that is, the set of IF-THEN rules are learned through iteratively collecting desired input-output pairs and generating the correspondences between input signals and output actions.
There are alternative approaches for a robot to learn skills of experts. One method is to transfer skills of the human expert to the robot manipulator by the off-line method.[6]
That is, we collect sample data

$$S = \{ (s_i, a_i), i=1,\cdots,n\} \qquad (1)$$

where s and a represent, respectively, the sensor information and the motion of the robot and n is the number of samples. Then, the data are analyzed by a computer in order to derive skills of the human expert. Finally the skills are transferred to a robot by programmable language of the robot. However, in skill representation or skill transfer, the capacity mismatch problem arises due to the difference in the physical capability of perception and action between a human expert and a robot.
Based on the above observation, our ARAS is structured not by skill transfer from a human expert to a robot with off-line method but by their own set of primitive features which can be directly acquired in an actual environment, even though some help of experts is needed in advance.
In summary, this paper will show a new approach to chamferless and high-precision robotic assembly. Our ARAS will not require complete analysis of assembly tasks and will need little special hardware such as vibratory equipments. The resulting ARAS have learning ability which is performed with little help of human experts. In

CH3019-7/91/0000-0128$01.00 ©1991 IEEE



(a) 1-point contact

(b) 2-point contact

Fig.1 Contact configuration

Table 1. A priori knowledge for mapping between sensor information and direction of positional error

Sensor Information	Contact configuration Direction of lateral error
$F_X < 0,\ M_Y < 0,\ \|F_X\| \gg \|F_Y\|$	One-point contact, +X
$F_X<0, F_Y<0, M_X>0, M_Y<0 \|F_X\| \approx \|F_Y\|, \|M_X\| \approx \|M_Y\|$	One-point contact, +X +Y
$F_Y < 0,\ M_X > 0, \|F_Y\| \gg \|F_X\|$	One-point contact, +Y
$F_X>0, F_Y<0, M_X>0, M_Y>0, \|F_X\| \approx \|F_Y\|, \|M_X\| \approx \|M_Y\|$	One-point contact, -X +Y
$F_X > 0,\ M_Y > 0,\ \|F_X\| \gg \|F_Y\|$	One-point contact, -X
$F_X>0, F_Y>0, M_X<0, M_Y>0, \|F_X\| \approx \|F_Y\|, \|M_X\| \approx \|M_Y\|$	One-point contact, -X -Y
$F_Y > 0,\ M_X < 0,\ \|F_Y\| \gg \|F_X\|$	One-point contact, -Y
$F_X<0, F_Y>0, M_X<0, M_Y<0, \|F_X\| \approx \|F_Y\|, \|M_X\| \approx \|M_Y\|$	One-point contact, +X -Y
Others	Two-point contact

particular, we will emphasize the strategies and skills obtained through ARAS are effective in a given actual environment by robotic assembly experiments even though they are constrained to a narrow class of assemblies.

2. Kinematic Analysis

When a peg contacts with the surrounding of a hole, contact configurations can be classified into two groups, 1-point contact and 2-point contact, depending on positional and angular error. Fig.1 shows the force and moment applied to the peg and the corresponding reaction forces at the contact points. In our models, we assume the parts to be infinitely rigid and massless, and we use the dry coulomb model to represent friction. We represent the normal component of the reaction force at a contact point, as f and static coefficient of friction as μ. We represent the angle of the peg's axis with respect to the hole's axis as θ. Assuming that there is only x-direction positional error and one-point contact, the resulting equilibrium relations are

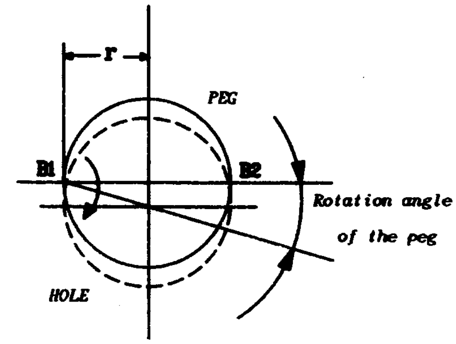


Fig. 2 Rotation of the peg in search stage

expressed in the coordinates of the peg frame as follows:

$$F_z + f\cos\theta + \mu f\sin\theta = 0 \tag{2}$$
$$F_x - f\sin\theta + \mu f\cos\theta = 0 \tag{3}$$
$$M_y - fr\cos\theta - \mu fr\sin\theta = 0 \tag{4}$$

Here Fi and Mi, i=x, y, and z, represent the components of the measured force and moment transformed by the peg coordinate frame, respectively and r is the radius of the peg. The similar analysis can be applied to other direction by transformation of coordinate frame. Table 1. shows the rough relation between the direction of positional error and the force sensor information, considering the various forms of uncertainty. Using the result in Table 1 as a priori knowledge, the peg can be approached to the neighborhood of the Goal Area[1], even though the searching task is not yet completed successfully. (we call it Sub Goal Area[2])

3. Mechanics of Rotation

The final purpose of ARAS is to acquire the task strategies generated through iterative learning. Since ARAS allows the robot to learn the appropriate action in response to the percepted signal, ARAS must have efficient appropriate actions.

Assuming that the positional error is extremely small, in case of 2-point contact, if we rotate the peg clockwise, the contact point B_1 becomes the instantaneous center and the instantaneous center moves along the edge of the hole. Therefore the peg can move the Goal Area[1] as shown in Fig.2. If we rotate the peg counterclockwise, the contact point B_2 becomes the instantaneous center and the peg can move the Goal Area, too. Of course, if the peg is connected with the robot in one rigid body, the movement as shown in Fig.2 will be impossible because the motion between the peg and the surrounding of the hole will become sliding motion. With the aid of compliant structure installed between the end effector of the robot and the peg, the motion between the peg and the surrounding of the hole becomes rolling motion. Therefore the contact point becomes the instantaneous center and the peg begins to rotate about the contact point. As the peg rotates the instantaneous center moves toward Goal Area.

[1] Goal Area is the region in manipulator space where the searching task is completed successfully.
[2] Sub Goal Area is the region in manipulator space where the peg can easily move into Goal Area with only rotation action.

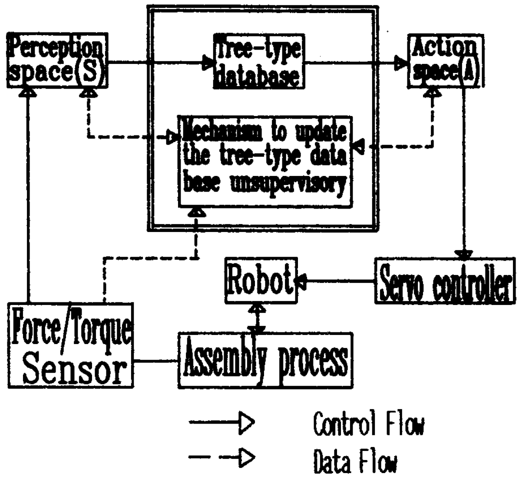


Fig. 3 Structure of ARAS

4. ARAS Structure

This paper proposes a new concept of robotic assembly scheme with learning ability, ARAS, which is shown in Fig.3 In this section, a method for learning strategies, i.e. mappings which relate sensor signals to appropriate actions of a robot is discussed. In paticular, this paper will emphasize the robot's self-acquisition of skills from the experience in actual task environments.

4.1 Perception Space

In general,industrial robots can not perform chamferless assembly successfully only with its own capability due to the presence of various forms of uncertainties[7]. These uncertainties make a robot to deviate from the nominal motion and undesired geometrical contacts. These undesired geometrical contacts, after all, result in contact forces and moments. In order to succeed chamferless assembly tasks in spite of uncertainties, it is necessary to devise part mating strategies that relate the force signal to desired corrective motion.Even though sensor signals include its own uncertainties, ARAS can deal with these uncertainties.Our idea on dealing with the uncertainties starts with the fact that the uncertainties are bounded to certain region[8]. Therefore the signal region can be mapped into the appropriate corressponding action of the robot.Consequently task strategies are established by connecting primitive signals with the appropriate robot's action through iterative experiences in actual engineering environments.

Inertial effects which depend on the placement of the force sensor on the robot arm may cause robot to make an erroneous decision. Inertial forces varies depending on the magnitude of acceleration. To remove the difficulties caused by inertial effects, force/moment readings are conducted under static conditions.

4.2 Action Space

Our method doesn't need to collect sample data in advance. Actions in our method are not self-generated but provided in advance with the aid of experts. Then the actions are correlated with the percepted signals. The criterion in selecting an action is as follows:

(1) Reliability: Robot should exhibit the action reliably in spite of uncertainties of the robot control.
(2) Effectiveness: The action is assured to be effective by experts.
(3) Executability: The action is executable without any change of robot control hardware. It can be implemented with the original capability of a commercial robot.

4.3 Self-Organization of Database (Unsupervised Learning Database)

The key issue addressed in this paper is to associate sensor signals with appropriate actions of a robot and generate task strategies via iterative learning in actual physical environment. Task strategies can be represented as the mapping between the perception space and action space. Skills are denoted by terms of IF(state of sensor signal)THEN(desired corrective action). In this paper, we propose unsupervised learning database supported by self-organizing database[9] for representing and generating task strategies. Task strategies and skills must be discovered via iterative learning even though they are expressed implicitly. Unsupervised means here that whether the database is expanded or not is decided depending on the evaluation of a criterion which is established in advance by assistance of experts.

Let R^n be the set consisting of all ordered n- tuples of real numbers and let the perception space S and the action space A be n- and m- dimensional vector spaces, respectively, defined over a scalar number filed F. Then $S \subset R^n$, $A \subset R^m$. Any $s \subset S$ represents a signal in the perception space and is also an input to the learning machine. Any $a \in A$ represents an action in the action space and is also an output of the learning machine corresponding uniquely to a s. Let the descrimination function d be an mapping $S \times S \rightarrow real \geq 0$ that satisfies the following condition:

For any $(s,\tilde{s}) \in S \times S$, $d(s,\tilde{s})$ is positive whenever $a \neq \tilde{a}$
Here, a and $\tilde{a}$ are the outputs which s and $\tilde{s}$ correspond to, respectively.

Experience, as a collection of signal-action pairs, (s_i,a_i), can be organized in a tree type data base. This machine analyses the samples of $(s,a) \in S \times A$ with d and memorizes its results in a self-organizing way. The followings are the algorithm of unsupervised learning database:

Step 1:(Initialization) Set i=1. Let $(s[root], a[root])$ = (s,a). Here $s[\cdot]$ and $a[\cdot]$ are the memory variables assigned for respective nodes to memorize samples

$\forall(s,a) \in S \times A$

Step 2:(Finding terminal node) Increase i by 1, and put s_i in. After resetting a pointer n to the root node, repeat the following until the pointer arrives at some terminal node. If $d(s_i,s[n_l]) < d(s_i, s[n_r])$, $n = n_l$. Otherwise $n = n_r$. Here n_l and n_r mean the successor nodes of n.

Step 3:(Evaluation) Move a robot as the action memorized in a[terminal node]and evaluate the action to judge a success or a failure with an evaluation function. If evaluation results in a success, go back to step 2. Otherwise go to step 4.

Step 4:(Selecting better action) Let a^1 = a[terminal node]. Choose $a \in A$ randomly, except a^j and execute the action a and evaluate it. Here a^j are the excuted actions. If successful, go to step 5. Otherwise, return to the original state and after letting a^j = a[just executed action], repeat step 4 until success.

Step 5:(Expanding the data base) Regard and establish new successor nodes as follows:

$(s[n_l],a[n_l])=(s[n],a[n]),(s[n_r],a[n_r])=(s_i,a)$

Finally go back to step 2.

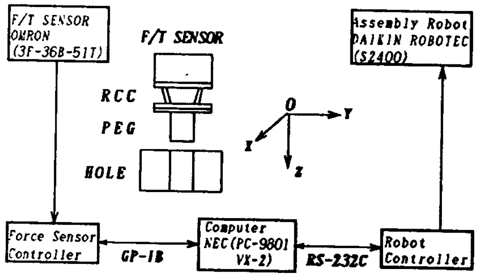


Fig.4 Structure of experimental system

5. Experiments

Actually, the assembly task is composed of two process; the searching process and the inserting process. The inserting process is quite different from the searching process. Our ARAS can be applied to both the searching process and the inserting process without the change of system structure. When the assembly task transits from the searching process to the inserting process, ARAS can be adapted only by changing the database. We have already shown that our ARAS is effective in case of inserting process[10]. In this paper, therefore, we focus on the searching process.

In addition, based on above discussions, we assume as follows:

(1) Positional error between a peg and a hole is within about 0.5 mm.

(2) Angular error is adjusted almost to zero by the setting of swivel table.

(3)Some compliance exists between the robot end effector and the peg

5.1 System Hardware

Fig.4 shows the structure of the robot system.It consists of a SCARA type robot,a 6-axis force sensor,a compliant structure(RCC), and a computer(PC-9801 VX2). The specification of hardware used in the experiment is shown in Table 2. The computer communicates with the force sensor by GP-IB and the robot by RS-232-C. ARAS and control schemes of circumferential equipments are implemented using C language. The diameter of the peg is 20 mm, clearance is 0.01 mm and both the peg and the hole are chamferless. The swivel table is used to adjust the angular error.

5.2 Preliminary Experiment

To acquire task strategy autonomously, ARAS requires some criteria for evaluating whether the executed action is successful or not. To obtain them, we made simulated experiments so as to complete the task successfully.Finally, to discriminate whether the peg is in the SGA or not, the following criterion is introduced:

$$\frac{|F_{after}| - |F_{before}|}{|F_{before}|} < -0.20 \qquad (5)$$

$$\text{or} \quad |F_{after}| < 1500(gf) \qquad (6)$$

where F_{before} and F_{after} represent, respectively, Fz signals before and after the executed action.

Next in order to discriminate whether the searching task is completed or not, we employ the following heuristic function based on the preliminary experiments:

$$\frac{|F_{after}| - |F_{before}|}{|F_{before}|} < -0.95 \qquad (7)$$

$$\text{or} \quad |F_{after}| < 50(gf) \qquad (8)$$

Table 2. Specifications of Experimental hardware

Manipulator (Daikin S2400)	
Type	SCARA
No. of axis	4
Pay load	3.5kgf(high speed)-5kgf(low speed)
Repeatability	± 0.05 mm
Actuators	AC servo motor
Force Sensor(Omron)	
No of axis	6
Regular load	Force 5 kgf moment 20kgf cm
Resolutions	0.1 % of regular load
Sampling time	5 m sec
RCC(Bando RCC-112-BS)	
Horizontal stiffness	7.4 kgf/cm
Rotational stiffness	890 kgf·cm/rad

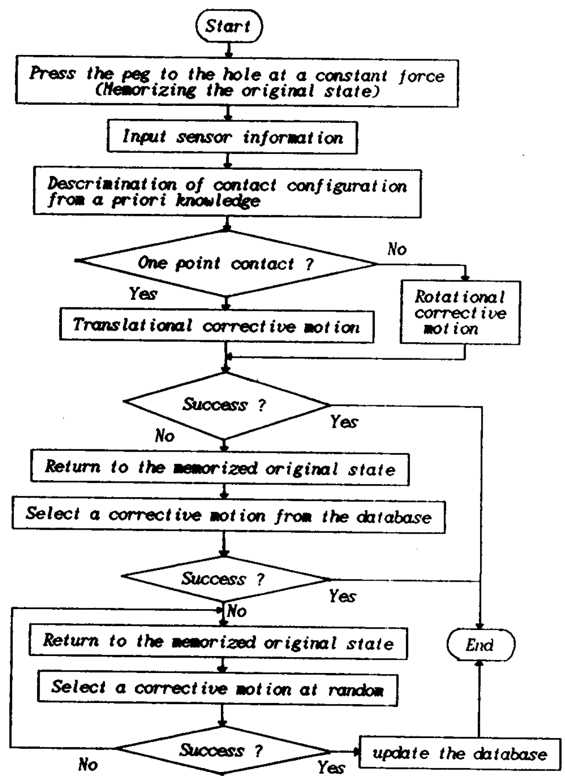


Fig.5 Flowchart of experiment

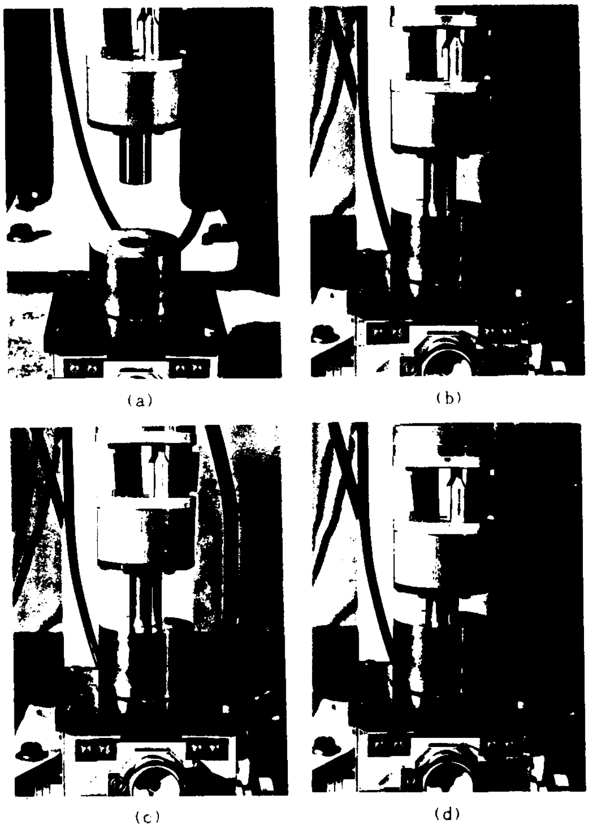


Fig.6 Photo of the assembly task
(a) Peg and block with hole are lying on swivel table
(b) Robot contacts the peg with the surrounding surface of the hole
(c) Robot brings the peg over the hole through corrective actions ·
(d) Robot inserts the peg into the hole

5.3 ARAS

ARAS accepts only static force/moment signals when Fz comes to about 400 gf. Each of perception signals contains 6 components of forces and torques, i.e, F_x, F_y, F_z, M_x, M_y, and M_z from the force sensor.
Since our assembly tasks are performed by SCARA robot, some simple motion, e.g. straight line or simple rotation is sufficient to maintain the same contact configuration during the searching process. The action space is composed of 8 translational movement and 2 rotational movement about Z axis which correct the angular error as discussed in Section 3. Each of A_i's in Fig.7 represents a direction of translational corrective action.Step distance of translational movement is 0.05 mm and that of rotational movement is 1 degree.The discrimination function d is calculated based on Eucledian distance on perception space.

5.4 Experimental Procedure

Fig.5 shows the flow chart of ARAS and and Fig.6 shows the sequence of photographs taken during the experiment. When the contact force between the peg and the hole comes to a certain threshold, the searching process starts. First of all contact configuration is discriminated and the

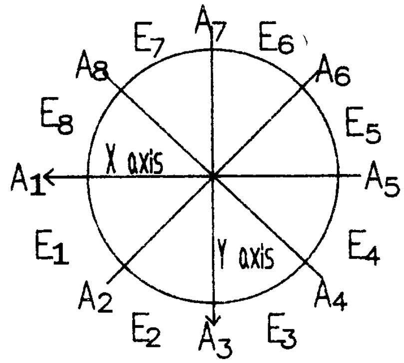


Fig.7 Direction of positional error and corrective action

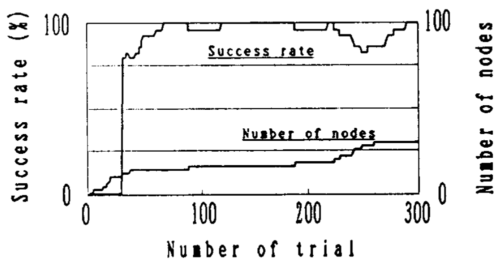


Fig. 8 Trasition of the success rate and number of nodes during the learning process for searching stages

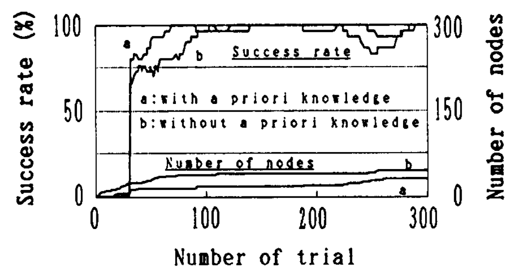


Fig. 9 Comparision of the success rate and the number of nodes with/without a priori knowledge for searching stages

corrective action of the robot is selected using the priori knowledge given in Table 1. If the action does not lead the peg to Sub Goal Area or Goal Area, another action is determined according to the strategy constructed by ARAS iteratively.

6. Experimental Results and Discussion

To confirm the learning effect of ARAS,we assume that the direction of the positional error is limited to 8 different type as shown in Fig.7.E_i's in Fig.7 represent the directions of positional error.The initial positional error at every trial is given randomly. Fig.8 shows the transition of the success rate and the number of nodes during the learning process with a priori knowledge. The success rate is computed as the rate of successful times in 30 trials right before the action. The success rate comes to nearly 100% after about 100 trials. Fig.9 shows the comparison of

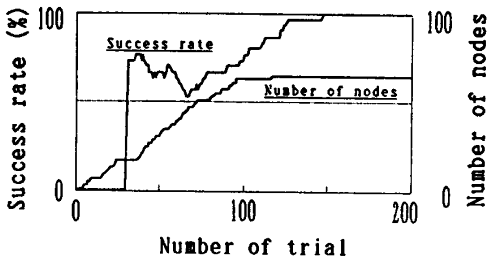


Fig.10 Transition of the success rate and the number of nodes in the case that clearance is changed from 10μm to 50μm at 100th trial

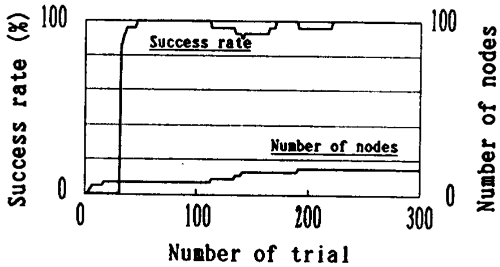


Fig.11 Transition of the success rate and the number of nodes in the case that initial angular error(θ_x) is changed as follows:0.0 → 0.2(changed at 100th trial) → 0.1degree(changed at 200th trial)

the success rate and the number of nodes with/without a priori knowledge. Even without using a priori knowledge, the success rate comes to nearly 100 %. Though ARAS with the priori knowledge is more effective than ARAS without it, the difference is small. After all our ARAS leads to success even in case of no a priori knowledge. The node number in figure's represents the amount of generated nodes in database. About 30 byte memory size is enough for each node to contain 6-component of perception signal data, action data, and data for the connection of binary tree type database. Therefore the memory quantity for the database and the time to handle the database is negligible.

In actual assembly tasks, there must be a variation in the part size or the angular error. To investigate if ARAS is still effective,even in case of the change of part size,we used pegs with different size. The clearance is changed from 0.01 mm to 0.05 mm at 100th trial. Fig. 10 shows that ARAS is still effective to the variation of the part size. Fig.11 shows success rate vs trial number while angular error of x component is varying from 0 to 0.1 through 0.2 degree.This result indicates that the strategy established up to 200 th trial is not updated at all after 200 th trial.That is,ARAS can be thought to have a kind of interpolation ability for the angular error.

In summary ARAS is effective for the acquisition of the strategy in a given actual assembly task and somewhat robust to the variation of angular error or part size.

7.Conclusions

We have discussed the assembly system which can acquire the mapping relations between the force information from sensor and the corrective actions of the robot through the iterative learning in actual task environments.

It is important to select inputs and outputs of learning machine based on the effectiveness in given tasks. We have introduced the rotational corrective motion in the search task, in addition to the conventional translational corrective motion and have confirmed the effectiveness of the rotational action.

Our proposed system have been applied to the search stage of assembly task in actual environments to acquire the task strategy autonomously. The experimental results have showed that the proposed system is effective for acquiring the task strategy and somewhat robust to the variation of part size or angular error.

Acknowledgement

This research was supported in part by NON DIRECTED RESEARCH FUND, Korea Research Foundation, 1990.

References

[1] D. E. Whitney. "Quasi-Static Assembly of Compliantly Supported Rigid Parts", Robot Motion, MIT Press, pp.439-471,1981.

[2] K.W.Jeong and H.S.Cho."Development of a pneumatic vibratory wrist for robotic assembly", Robotica, Vol.7, pp.9-16,1989.

[3] H.Asada. "Teaching and Learning of Compliance Using Neural Nets", Proc. IEEE Int. Conf. Robotics and Automation, pp.1237-1244,1990.

[4] H.Asada and H.Hirai, "Towards a Symbolic-Level Force Feedback:Recognition of Assembly Process States", Proc. 5-th Int. Symp. of Robotics Research, Tokyo,1989.

[5] M.E.Caine, et. al. "Assembly Strategies for Chamferless Parts", Proc. IEEE Int. Conf. Robotics and Automation, pp.472-477,1989.

[6] H.Asada and H.Izumi, "Automatic Program Generation from Teaching Data for the Hybrid Control of Robots. IEEE Trans. on Robotics and Automation, Vol. 5, No.2, pp.163-173,1989

[7] S. N. Gottschlich and A. C. Kak. "A Dynamic Approach to High-precision Parts Mating", IEEE Int. Conf. on Robotics and Automation, pp.1246-1253,1988.

[8] D.K.Pai and M.C.Leu. "The effect of a Robot Manipulators Uncertainty and Compliance on Task Feasibility", Japan-USA Symp. on Flexible Automation, pp.979-984,1990.

[9] H. Suzuki. "Study of Universal Learning Machine Based on a Self-organization Algorithm", Chap. 1, Ph.D Thesis in Osaka University,1988.

[10] D.S.Ahn, et al. "Automated Robotic Assembly with Unsupervised Learning Machine", Proc. 28th SICE, pp.1385-1388,1989.