研究解説

Obtaining Manipulation Skills from Observation

Jun TAKAMATSU*, Hirohisa TOMINAGA*, Koichi OGAWARA*,
Hiroshi KIMURA** and Katsushi IKEUCHI*

1. Introduction

To enable a robot to program automatically, we propose an enhanced assembly-plan-from-observation (APO) method. This method has three modules as shown in Fig. 1: observe a human performing an assembly task, recognize the task, and then generate a robot program to accomplish that same task.

In order to do so, the following needs must be satisfied: the observation module needs to have the ability to obtain the trajectory of a manipulated object; the recognition module needs to have the ability to extract the transition of topological contact relations, because an assembly task can be represented by that [9]; and the generation module needs to have the ability to move the assembly parts to achieve the same contact transition. [10] has proposed a method to compute these movements. But in an assembly task, the generation module needs to have not only position-control ability, but also force-control ability; in addition, the module needs to detect the execution errors by using the various sensors and to correct these errors. For more effective operation of these higher controls, a *manipulation skill* has been proposed [3]. Manipulation skills are basic operations to achieve an assembly task. So, the robot program that uses these skills is represented as a combination of manipulation skills. In this paper, these basic operations are called *sub-skills*.

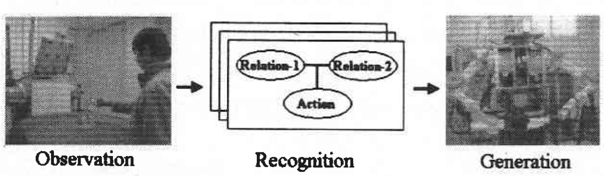

Fig. 1 Outline of APO

* Institute of Industrial Science, University of Tokyo
** Graduate School of Information Systems, University of Electro-Communications

2. Observation module

The trajectory of each object is recorded as a sequence of range data through a real-time stereo system with nine cameras. From the range data, using 3DTM object recognition system, the system extracts the trajectory of each object as shown Fig. 2 [11].

Fig. 2 3 DTM object recognition system

The trajectory information obtained is represented in the configuration space (C-space), a 6-dimensional space which represents both the position and the orientation of an object [2]. It observes configuration of objects at certain intervals throughout the system. This system represents object configurations as points in the C-space. As a result of the observation, a series of points, corresponding to the object configurations, is recorded in the C-space.

A constraint of a manipulated object constitutes a manifold, referred to as a C-obstacle surface, in a C-space. The configuration value obtained contains some observation errors. Due to these errors, observed points in the C-space jump around on the C-obstacle surface. In order to smooth out these observation errors, the system regards those points close enough to the surface as those on the constraining surface, and makes a smooth trajectory by connecting those observed points in the C-space. See Fig. 3 for an illustration of how to make the trajectory feasible [2].

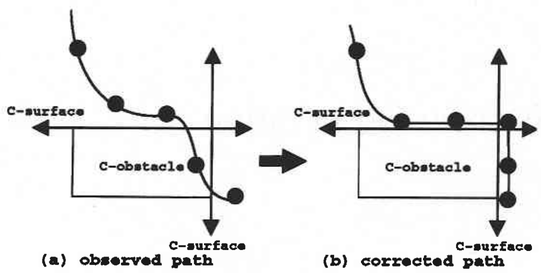

Fig. 3 Correcting path on C-surface

3. Recognition module

This section describes the recognition module. The assembly tasks can be represented by the transitions of the topological contact-relation [9]. For achieving the aimed transitions, the recognition module needs to obtain the feasible motion of a manipulated object. But the possible motion is represented as non-linear equations, so it is difficult to treat that. First, we will introduce the screw theory, by which the possible motion can be represented as linear equations approximately. Second, we will introduce the method to obtain features of the possible motion for assigning sub-skills using these equations.

3.1 Screw theory

The screw theory is employed for representing the possible motion [4]. For example, when two objects contact each other at a point, as shown in Fig. 4, the possible motion of the object B is constrained by the inequality (1). An equal part of the inequality (1) represents the motions which enable the contact relations to be maintained, while the greater part performs the detaching motions.

$$s_1t_4 + s_2t_5 + s_3t_6 + s_4t_1 + s_5t_2 + s_6t_3 \geq 0 \tag{1}$$

In the case of polyhedral objects, all kinds of contact relations can be represented by a combination of a vertex-face, a face-vertex, and an edge-edge contact. And these three contacts can be converted into constraint inequalities.

If a contact relation does not have any contacts shown in Fig. 5, we call *not-singular contacts*, the possible motion is constrained by the inequalities (2).

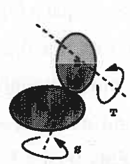

Fig. 4 contact with screw representation

$$\begin{pmatrix} a_{11} & \cdots & a_{16} \\ & \ddots & \\ a_{n1} & \cdots & a_{n6} \end{pmatrix} \begin{pmatrix} t_1 \\ \vdots \\ t_6 \end{pmatrix} \geq \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \tag{2}$$

If a contact relation has any of the contacts shown in Fig. 5, we call them *singular contacts*; the feasible motion is constrained by the inequalities (3).

$$\begin{array}{l} A_{11}\cdot T \geq 0 \;\; or \;\; \cdots \;\; or \;\; A_{1m_1}\cdot T \geq 0 \\ \vdots \\ A_{n1}\cdot T \geq 0 \;\; or \;\; \cdots \;\; or \;\; A_{nm_n}\cdot T \geq 0 \end{array} \tag{3}$$

$(A_{ij} \in 1 \times 6 \; matrix, T={}^t(t_1 \cdots t_6))$

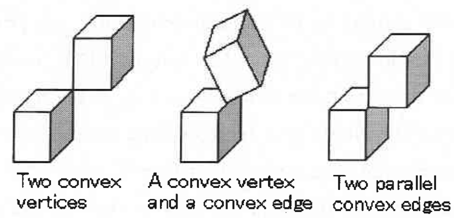

Fig. 5 Three singular contacts

3.2 Features of a possible motion

The previous APO system assigns the skill using features that consist of maintaining, detaching, and constraining degrees-of-freedom (DOF) in translation in *non-singular* contact [1]. We add three DOFs in rotation as shown in Fig. 6:

- Maintaining: The DOFs of axis directions to be able to rotate maintaining the contact relation.
- Detaching: The DOFs of axis directions not to be able to rotate maintaining the contact relation.
- Constraining: The DOFs of axis directions not to be able to rotate.

3.2.1 Extracting features in "non-singular"

Using [8], the number of rank r, range of dimension of face of PCC $\{6-r-p, \cdots, 6-r\}$, possible motion of translation, the number of both rotatable (means can rotate both clockwise and counter-clockwise) axis n_b, and the number of either rotatable (means can rotate either clockwise and counterclockwise) axis n_e are extract-

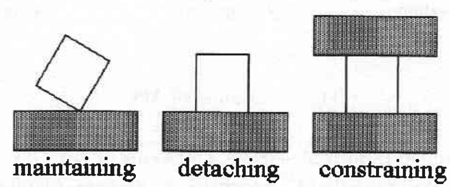

Fig. 6 Maintaining, detaching, and constraining DOFs in rotation

Table 1 Relation between feasible motion of translation and maintaining, detaching, and constraining DOF of translation

Translation	m_t	d_t	c_t	Translation	m_t	d_t	c_t
All	3	0	0	3D half	2	1	0
3D quarter	1	2	0	3D 1/8	0	3	0
2D space	2	0	1	2D half	1	1	1
2D quarter	0	2	1	Line	1	0	2
Half line	0	1	2	Point	0	0	3

ed from constraint inequalities (2). Maintaining m_t, detaching d_t constraining c_t DOFs of translation are obtained from Table 1 and maintaining m_r, detaching d_r, constraining c_r DOFs of rotation are from the equation (4).

$$m_r = 6 - r - m_t$$
$$d_r = 3 - m_r - c_r \qquad (4)$$
$$c_r = 3 - (n_b + n_e)$$

3.2.2 Extracting features in singular

In the case of singular, we assume that those singular DOFs are treated as DOFs having no singular contact. So, we can analyze contact relations by using the same method. In this case, we call these DOFs *singular maintaining*, *singular detaching*, and *singular constraining*.

4. Designing sub-skills

In the previous section, we defined three DOFs of translation and rotation. The change of contact relations lead to changing those DOFs. There are some transitions of DOFs as shown in Fig. 7. Among those possible transitions, the following three transitions occur toward the direction of movements: *maintaining to detaching maintaining to singular maintaining*, and *maintaining to singular detaching*. These three transitions are important in the design of sub-skills.

4.1 Maintaining to detaching in translation

The motion as shown in Fig. 8 leads to the transition from maintaining to detaching in translation. We call this motion *make-contact in translation.*

In order to implement the make-contact-in-transition sub-skill,

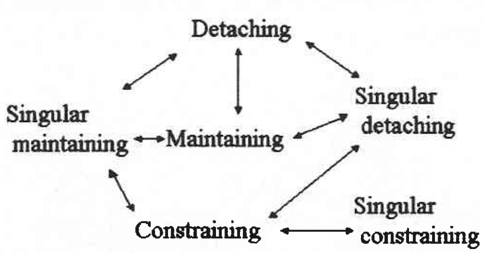

Fig. 7 possible transitions of DOFs

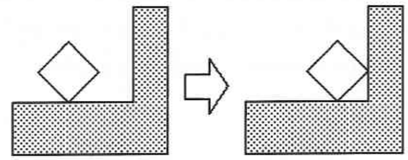

Fig. 8 Make-contact in translation

we use force sensors to detect when the manipulated object makes contact with the environmental object, and the force value increases beyond a threshold. The system moves the manipulated object to the detaching direction of the next state until it makes contact.

4.2 Maintaining to detaching in rotation

The motion as shown in Fig. 9 leads to the transition from maintain to detaching in rotation. We call this motion *make-contact in rotation.*

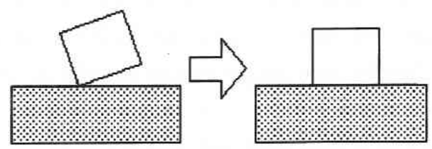

Fig. 9 Make-contact in rotation

To implement this sub-skill, we use a force sensor as well as make-contact-in-transition. The system can determine the direction of the rotation using the demonstration of an operator, and determine a rotation center by the analysis in C-space in advance. It rotates the manipulated object using this information until the force sensors detect the contact.

4.3 Maintaining to singular maintain in translation

The motion as shown in Fig. 10 leads to the transition from maintain to singular maintain in translation. We call this motion *slide in translation.*

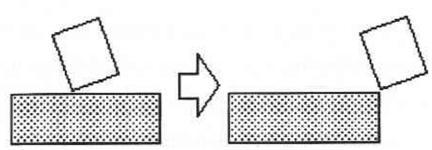

Fig. 10 Slide in translation

When a manipulated object loses contact with an environmental object, the force value decreases beyond a threshold value. Using the decrement of the force value, the system detects the point of singular contact. The direction in which the manipulated object is to be moved is acquired from the demonstration. The system moves the manipulated object toward the direction until contact is lost.

4.4 Maintaining to singular maintain in rotation

The motion as shown in Fig. 11 leads to the transition from maintain to singular maintain in rotation. We call this motion *slide in rotation*.

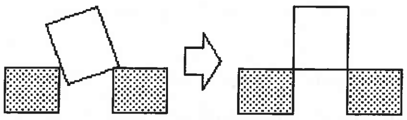

Fig. 11 Slide in rotation

In order to achieve this motion, two controls should be done. One is the control to maintain the contacts, while the other is to change the orientation of the manipulated object. Therefore, we decompose the motion into two parts: at the first the system moves the manipulated object to a detaching direction at a slight distance, although one of two contacts is lost by this first motion; then the system rotates the object around the contact point until the manipulated object again makes contact with the environmental object at two points.

4.5 Assigning a slide sub-skill

The transition of maintaining to singular detaching in the moving direction usually leads to the same transition in another direction. Therefore, it is difficult to assign a slide sub-skill to it. In the end of a slide sub-skill, the number of *restricted DOF* [3] increases. If the possible motion is constrained by the inequalities (3), restricted DOF is equal to the rank of matrix $({}^{t}A_{11}\cdots{}^{t}A_{1m_1}\cdots{}^{t}A_{nm_n})$.

If the number of restricted DOFs in translation increases, a slide in translation sub-skill is assigned; if the number of restricted DOFs in rotation increases, a slide in rotation sub-skill is assigned.

4.6 Maintaining to singular detaching

The motion as shown in Fig. 13 leads to the transition from maintain to singular detaching in translation. This motion looks like the motion combining make-contact and slide in translation. But because this motion can perform the similar method of make-contact, we do not treat this motion.

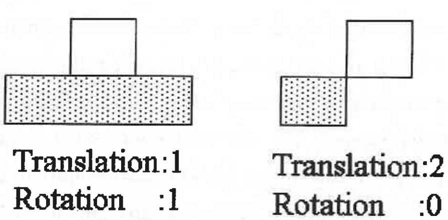

Fig. 12 Restricted DOF

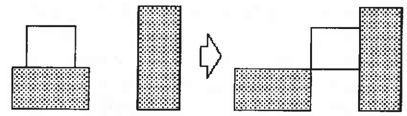

Fig. 13 maintaning to singular detaching

4.7 Execution error

A not-aimed movement leads to an execution error. In short, these three transitions of DOFs are relative to an execution error. In particular, *a detaching to maintaining* transition is very important. We will introduce the method to detect error contact-relations.

For example, consider the case as shown in Fig. 14. There are four vertex-face contacts. Each contact inequality corresponding to each contact is obtained. Several contacts can be removed if, and only if, the answer satisfying inequalities (5) is not empty.

$$\left.\begin{array}{l} f_{m1} > 0 \\ \vdots \\ f_{ms} > 0 \end{array}\right| \quad \textit{corresponding to several contacts}$$
$$\left.\begin{array}{l} f_{n1} = 0 \\ \vdots \\ f_{nt} = 0 \end{array}\right| \quad \textit{another} \qquad (5)$$

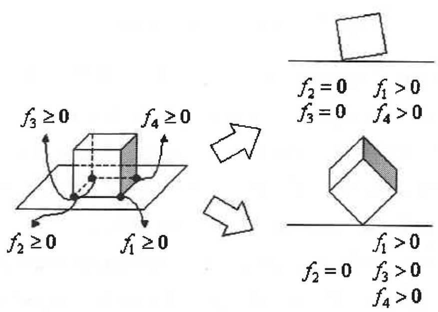

Fig. 14 execution error

5. Precondition for an another contact transition

5.1 Singular maintaining to detaching

As shown in Fig. 15, this transition appears in a slide sub-skill. Making singular contact correctly is very difficult, but a small execution error does not disable to realize the transition of the contact relations.

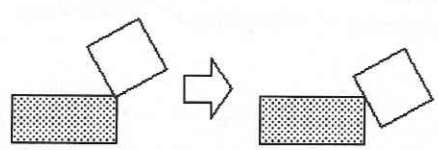

Fig. 15 Singular maintaining to detaching

5.2 Singular maintaining to constraint

As shown in Fig. 16, this transition also appears in a slide sub-skill. However, in this case, realizing the contact relation requires the precise position control in this dimension.

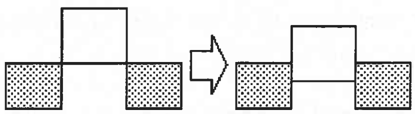

Fig. 16 Singular maintaining to detaching

5.3 Singular detaching to constraint

In this case, as shown in Fig. 17, realizing the contact relation requires making singular contact correctly in this dimension also. But in this case, it is easy to *pass through* singular contact. The slide sub-skill can be added the next sub-skill.

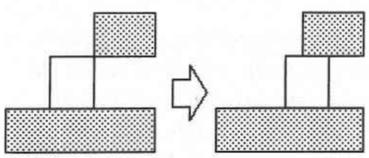

Fig. 17 Singular detaching to constraint

6. Examples

For this experiment, we constructed a test bed as shown in Fig. 18 that consists of a dual arm with a pair of dextrous hands and a real-time stereo system.

Consider the peg-in-hole operation shown in Fig. 19. In the first transition, a maintaining DOF in translation changes to a detaching DOF. A make-contact in translation sub-skill is assigned.

In the second transition, maintaining DOFs in translation and rotation change to singular maintaining DOFs. A restricted DOF in translation increases. A slide in translation sub-skill is assigned.

In the third transition, a singular maintain DOF in translation changes to a detaching DOF, so a small error does not disable the completion of the task.

In the fourth transition, a maintaining DOF in translation changes to a detaching DOF. A make-contact in translation sub-skill is assigned.

In the fifth transition, a maintaining DOF in rotation changes to a singular maintaining DOF and a restricted DOF in rotation increases. A slide in rotation sub-skill is assigned.

In the sixth transition, singular maintaining DOFs in translation and rotation change to constraining DOFs, so precise position control in these dimensions is needed.

In the seventh transition, a maintaining DOF in translation

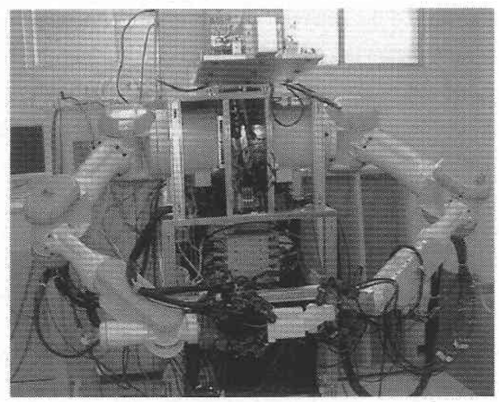

Fig. 18 A test bed

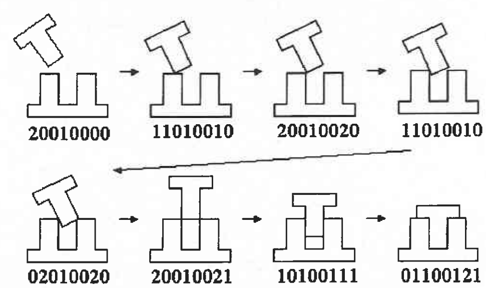

Fig. 19 Maintaining, detaching, and constraining DOFs of translation and rotation, and Restricted DOFs in translation and rotation

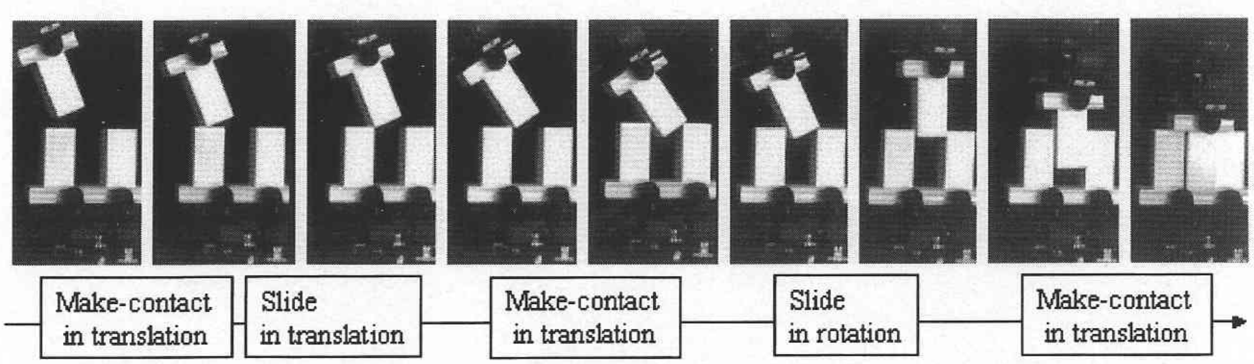

Fig. 20 peg-in-hole task

changes to a detaching DOF. A make-contact in translation sub-skill is assigned.

Fig. 20 represents the sequence of the robot executing a peg-in-hole task using a sequence of assigned sub-skills. We confirmed that these sub-skills work effectively.

7. Conclusions

In this paper, we proposed a system which has the ability to observe a human motion, divide the trajectory obtained from the observation into several states according to the contact relation, and assign a sub-skill to each contact transition. We proposed a new method to classify the contact relations, implemented the sub-skills, and verified the behavior of the system with the sub-skills. Our enhanced system has the advantages of both the contact-state-based system (the APO system) and the trajectory-based system.

(Manuscript received, March 13, 2000)

References

1) K. Ikeuchi and T. Suehiro, "Toward an Assembly Plan from Observation Part I: Task Recognition With Polyhedral Objects," *IEEE Trans. Robotics and Automation*, Vol. 10, no. 3, pp. 368-384, June 1994.

2) G. V. Paul and K. Ikeuchi, "A Quasi-Linear Method for Computing and Projecting onto C-Surfaces: Planar Case," *IEEE Int. Conf. on Robots and Autoomation*, pp. 2032-2037, 1997.

3) T. Suehiro, "Study of an advanced manipulation system," *Researches of the Electrotechnical Laboratory*, No. 912, June, 1990. (in Japanese).

4) B. Roth, "An Extension of Screw Theory," Journal of Mechanical Design, Vol. 103, pp. 725-735, 1981.

5) J. Miura and K. Ikeuchi, "Task-Oriented Generation of Visual Sensing Strategies in Assembly Tasks," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 2, Feb, 1998.

6) H. W. Kuhn and A. W. Tucker, "Linear Inequalities and Related Systems," *Annals. of Mathematics Studies*, Vol. 38, 1956.

7) J. Xiao and L. Zhang, "Contact Constraint Analysis and Determination of Geometrically Valid Contact Formations from Posible Contat Primitives," *IEEE Trans. on Robotics and Automation*, Vol. 13, No. 3, pp. 456-466, June 1997.

8) J. Takamatsu, H. Kimura, and K. Ikeuchi, "Classifying Contact States for Recognizing Human Assembly Tasks," *IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pp.177-182, Aug 1999.

9) S. Hirai, H. Asada, and H. Tokumaru, "Kinematic Analysis of State Transitions in Assembly Operations and Automatic Generation of Transition Network," *SICE*, Vol. 24, No. 4, pp. 84-91, 1988. (in Japanese)

10) B. J. McCarragher and H. Asada, "A Discrete Event Approach to the Control of Robtic Assembly Tasks," *IEEE Int. Conf. on Robotics and Automation*, pp. 331-336, 1997.

11) M. D. Wheeler and K. Ikeuchi, "Sensor Modeling, Probalilistic Hypothsis Generation, and Robust Localization for Objext Recognition," *IEEE trans. Pattern Analysis and Machine Intelligence*, Vol. 17, pp. 252-265, 1995.