Error Recovery Using Task Stratification and Error Classification for Manipulation Robots in Physical Distribution

Akira Nakamura, Kazuyuki Nagata, Kensuke Harada, and Natsuki Yamanobe

Abstract: In recent years, the techniques performed by manipulation robots in various fields have been researched. The authors have shown that manipulator tasks such as grasping and assembly can generally be constructed using several motion primitives which they call “skills,” and explained how most manipulator tasks can be composed of sequences of these skills. In this paper, the authors have further considered hierarchizing the manipulation tasks in order to facilitate their organization as such tasks continue to become more complex. This paper presents concrete processes of error recovery for manipulation robots used in physical distribution, and makes use of the concepts of task stratification and error classification. This paper also refers to the reusability of task planning which may become an important aspect of research on error recovery.

Keywords: error recovery, task stratification, error classification, physical distribution, manipulation skill

1. INTRODUCTION

In recent years, studies on robotic manipulation for performing required tasks have been conducted in various fields. We have done various research on robotic manipulation used to perform plant maintenance tasks and to produce industrial products [1-3,7-16]. In many cases, a maintenance task consists of a sequence of sub-tasks such as disassembly, inspection, replacement (if necessary), and assembly. Since each fundamental unit in this sequence is composed of actions such as sensing, modeling, planning, and execution, it is necessary for many basic units to be used repeatedly. Therefore, maintenance tasks tend to be complex, and it becomes necessary to devise composition rules for the entire work process.

By analyzing the assembly and disassembly sequences performed by humans, we found those tasks to be composed of several significant motion primitives. We called each motion primitive a “skill” and have shown that most maintenance tasks can be composed of a number of skills [1-3,7-16]. Given a hierarchy of robot controllers, control of a given skill is located between the control of the task, which is located in the highest layer, and the control of the servo controller of each joint, which is located in the lowest layer. Thus, if a program is created based on skills, the programmer can construct robotic tasks that are higher than the skill level layer. Making use of this hierarchy, we do not need to think about control at the servo level. This is one of the advantages of having tasks comprised of skills.

Ideally, a robotic task has to be successfully completed as planned. However, in actual tasks of complicated plant maintenance, it is not rare to have the execution of a task terminate before completion. Therefore, since error recovery is an important research theme for robots that need to perform actual tasks, various techniques have been reported so far [7-14]. Donald has studied error detection and recovery using the concept of a fine motion planning [7]. Recently, Pastor et al. has proposed using a skill library with failure detection [14]. However, most of the reports on error recovery are related to orbit adjustment and there have been few reports which can be applied to complicated error recovery in maintenance tasks.

We have explored error recovery in robotic tasks so that robots can be used for the complicated tasks of plant maintenance [16-18]. In our method, error recovery through a forward correction process is used by slightly correcting the preplanned task. Also, error recovery through a backward correction process is used for significantly correcting the preplanned task.

In particular, in error recovery using a backward correction process, the pose of the robot may return to certain steps by correcting some parameters through sensing, modeling or planning actions. Or, to further correct the robot’s motions, the path of the robot may retreat to a task in the upper layer. Our error recovery processes are conducted using the concepts of both task stratification and error classification [16].

In recent years, robotics research has begun to be conducted in various fields apart from the industrial applications that have been studied for a long time. We also target robots working in various fields. In this paper, we will consider robots working in physical distribution.
To automate the tasks in a logistics center, we will explain error recovery processes specifically for the task of repacking objects from a large box into a small box.

Based on the concept of error classification, our detailed recovery technique will be explained by addressing typical errors that are likely to happen. In this paper, we will also refer to the reusability of task planning which may become an important aspect of research on error recovery. The degree of difficulty of reusability is defined by investigating the contents of the path of recovery.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.

In this paper, a brief literature review such as the concept of skills and stratification of tasks is presented in the next section. Then, the classification of errors, error recovery in the task hierarchy and flow charts of the total process are presented in Section 3. In Section 4, error recovery of robots working in physical distribution is explained in as specific detail as possible.

2. STRATIFICATION OF TASKS

Let us first explain our concept of skills and stratification of tasks. In this section, although the tasks of maintenance robots are mainly taken into account, the tasks of manipulation robots operating in physical distribution may be considered to be similar. See References [1, 6, 17-18] for details.

A. Manipulation Skills

In assembly and disassembly tasks, the skills in which the contact states vary are particularly significant. In References [1, 3], we considered three skills, “move to touch,” “rotate-to-level” and “rotate-to-insert,” which all play an important part in such tasks.

(i) Move-to-touch Skill: This skill is the transition of a grasped object P in a constant direction that continues until contact with another object Q occurs (Fig. 1(a)).

(ii) Rotate-to-level Skill: This skill is the rotation around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 1(b)).

(iii) Rotate-to-insert Skill: This skill is the motion of rotating the object P obliquely into the hole in another object Q to accurately insert object P into the hole (Fig. 1(c)).

Fig. 1. Three fundamental skills.
B. **Hierarchy of Tasks**

Manipulation tasks composed of several skills have been considered previously as described in References [15, 16]. However, actual tasks composed of many skills are in fact more complex and a stratification of the tasks is preferable for efficient management and execution.

We have described the hierarchizing of manipulation tasks based on a bottom-up approach [15, 16]. If we ignore the servo layer, the skill layer, which consists of elements such as the move-to-touch and rotate-to-bite skills, is located in the lowest layer called the \( \text{task}^{(0)} \) layer. Each skill is performed using the processes of visual sensing, geometric modeling, planning and execution. One tier above the \( \text{task}^{(0)} \) layer is called the \( \text{task}^{(1)} \) layer. Similarly, the \( \text{task}^{(i+1)} \) is composed of sequences of \( \text{task}^{(i)} \) elements (Fig. 3). The top layer, where the error recovery loop is closed, is called \( \text{task}^{(\text{max})} \) and one tier above \( \text{task}^{(\text{max})} \) is called the project layer. The project layer is a high-ranking class that is not affected by the recovery loop. The project layer might also be hierarchized, but we will not discuss this here.

**C. Stratification of Maintenance Tasks**

Let us consider the typical tasks involved in the repair of a portable radio (Fig. 4(a)) as an example of stratification. This maintenance project involving component replacement for an electrical appliance is performed as shown in Fig. 4(b). The task sequence \{ \( \text{task}^{(2)}(1, i_1) \) \} of the case opening \( \text{task}^{(3)}(1) \) is shown in Fig. 4(c). If there are four Philips head screws at the rear, \( \text{task}^{(2)}(1, 1) \) is composed of the four tasks of loosening each of the four Philips head screws using a Phillips head screwdriver which can be described as \( \text{task}^{(3)}(1, 1, i_1) \) \((i_1 = 1, 2, 3, 4)\), and those skill sequences \{ \( \text{task}^{(0)}(1, i_1, i_2) \) \((i_0 = 1, 2, 3, 4)\) \} are shown in Fig. 2. These skill primitives are described in detail in References [15, 16]. Moreover, these skill sequences will be called the minimum traceable unit, which means the smallest unit in which it is necessary to return to the first node of a skill primitive sequence if an error occurs. After all the screws are extracted, the task of removing the case is performed as shown in Fig. 4(d). Then, the \( \text{task}^{(1)}(1, 2, 1) \) layer, which has no meaning, adds one tier below \( \text{task}^{(2)}(1, 2) \) to make the number of layers the same.

### 3. **Error Recovery in Stratificated Tasks**

Tasks will finish without any errors occurring in an ideal environment. In actual manipulation tasks, however, errors often occur from various causes. We have described our concept of error classification and process flow with error recovery in the task hierarchy [15, 16].

The error recovery technique that we have proposed is performed according to the process flow shown in Fig. 5. Figure 5(a) shows the total process flow with error recovery. Figure 5(b) shows a skill primitive sequence in a minimum traceable unit. Figure 5(c) shows a detailed process flow in each skill primitive. Moreover, Fig. 5(d) is a recovery process which returns to the upper task levels. In this section, the classification of errors, recovery path based on the classification and total formation of error recovery in the task are shown.

---

**Fig. 4.** Tasks of replacing parts of a portable radio.

**Fig. 2.** These skill primitives are described in detail in References [15, 16]. However, actual tasks composed of many skills are in fact more complex and a stratification of the tasks is preferable for efficient management and execution.
Accurate expression:
\[
\begin{align*}
\odot 1 & : \text{Preprocessing} \\
\odot 2 & : \text{Execution} \\
\odot 3 & : \text{Postprocessing} \\
\odot 4 & : \text{Backward}
\end{align*}
\]

\[
\begin{align*}
\text{task}(0)(q_1) & = \text{task}(0)(q_0) \\
\text{task}(1)(q_1) & = \text{task}(1)(q_0) \\
\vdots & \notag
\end{align*}
\]

\[
\begin{align*}
\text{task}(k)(q_1) & = \text{task}(k)(q_0) \\
\text{task}(\text{max})(q_1) & = \text{task}(\text{max})(q_0)
\end{align*}
\]

\[
\begin{align*}
\text{task}(0)(q_0) & = (i_{\text{max}}, \ldots, i_1) \\
\text{task}(1)(q_0) & = (i_{0}, \ldots, i_{1}) \\
\text{task}(k)(q_0) & = (i_{0}, \ldots, i_{k}) \\
\text{task}(\text{max})(q_0) & = (i_{\text{max}}, \ldots, i_1)
\end{align*}
\]

(a) Total process flow

\[\text{Sensing (All-over task (max))} \rightarrow \text{Modeling (All-over task (max))} \rightarrow \text{Planning (All-over task (max))}\]

\[\text{Sensing (task (i_{max}))} \rightarrow \text{Modeling (task (i_{max}))} \rightarrow \text{Planning (task (i_{max}))} \rightarrow \text{Classification of error}\]

\[\text{Minimum traceable unit} \rightarrow \text{Classification of error} \rightarrow \text{Classification of error} \rightarrow \text{Stop the process}\]

\[\text{Start} \rightarrow \text{Sensing (All-over task (max))} \rightarrow \text{Modeling (All-over task (max))} \rightarrow \text{Planning (All-over task (max))}\]

\[\text{Sensing (task (i_{max}))} \rightarrow \text{Modeling (task (i_{max}))} \rightarrow \text{Planning (task (i_{max}))} \rightarrow \text{Stop the process}\]

\[\text{Minimum traceable unit} \rightarrow \text{Classification of error} \rightarrow \text{Stop the process}\]

Fig. 5. Process flow with error recovery (1/2)
Failures can be caused by several kinds of errors such as control errors, modeling errors and visual sensing errors. We group the error states into several classes according to possible causes. The classes of errors are described in detail in References [15, 16].

- **Execution error**: This is a mechanical error caused in the manipulator mechanism such as a gear backlash.
- **Planning error**: This is an error caused by inaccurate parameter values in planning.
- **Modeling error**: This is an error caused by differences between the real object and the geometric model in the software.
- **Sensing error**: This is an error occurring during visual sensing.

Merely remedying the causes of these errors does not always solve the problem. For instance, it may be necessary to return to a previous step when the working environment is greatly changed by the error.

### B. Error Recovery based on Classification

We will explain recovery paths through a *backward correction process*. The overall process flow is shown in Fig. 5 (a), and recovery through a backward correction process means a recovery path without involving a *forward correction process* which passes Class 0 in the section of supplementary classification of error in Fig. 5 (c). At the Confirmation step in Fig. 5(c), the result is judged as correct or failed by an automatic process or by a human operator. If the result is judged as failed, the path is bound for classification of error in Fig. 5(a) because a forward correction process is not involved. After passing through the...
section of classification of error, the path branches out into several routes as shown in Fig. 5(a). Error recovery is performed using the following error classification.

| Class 1: | When it is judged to be an execution error, task\((i+1)_{(i,i)}\) is executed again without correcting the parameters. |
| Class 2: | When it is judged to be a planning error, task\((i+1)_{(i,i)}\) is executed again with a change in the planning parameters. |
| Class 3: | When it is judged to be a modeling error, task\((i+1)_{(i,i)}\) is executed again with a change in the modeling parameters. |
| Class T\((1)\): | When it is judged to be a sensing error, task\((i+1)_{(i,i)}\) is executed again with a change in the sensing parameters. |
| Class T\((2)\): | task\((i+2)_{(i,i)}\) is executed again after the execution of the necessary changes and returns to the start at one tier above the layer task\((i+1)_{(i,i)}\). |
| Class T\((i)\): | task\((i+max)_{(i,i)}\) is executed again after the execution of the necessary changes and returns to the start at \((max-1)\) tier above the layer task\((i+1)_{(i,i)}\). |
| Class T\((max+1)\): | When it is judged that too many changes will be required, the process being executed is aborted. |

For Class 1, Class 2, Class 3, or Class T\((1)\) errors, the process flow must return to the indicated step before the minimum traceable unit as shown in Fig. 5(b). Similarly, for Class T\((2)\), Class T\((3)\), ..., or Class T\((max)\) errors, the process flow must return to the starting point of the indicated upper task layer as shown in Fig. 5(c). See Reference [25] for details on the restoration technique for each class of error.

C. Total Process Flow with Error Recovery

We will explain the total flow of error recovery through both the forward and backward correction processes as shown in Fig. 5. Figure 5(b) illustrates the process flow of a minimum traceable unit that existed in Fig. 5(a). Figure 5(c) shows a detailed process flow in a skill primitive which is a component of Fig. 5(b).

In each task\(^{0}\)(i) in a minimum traceable unit, the sequence of preprocessing, execution and postprocessing is performed as shown in Fig. 5(b). In preprocessing and postprocessing, sensing and modeling are performed at each step and this creates additional processing overhead. However, if the movements of the successive skill primitives are within a small range, the sensing and modeling in the preprocessing of the next primitive task\(^{0}\)(i) might be able to be used together for the sensing and modeling in the postprocessing of task\(^{0}\)(i).

A detailed flow chart of each task\(^{0}\)(i) is shown in Fig. 5(c). In the Preprocessing step, visual sensing using a vision system and geometric modeling in computer software are performed. Planning for execution of the skill primitive is performed sequentially. This means that the auxiliary processes in which the indeterminate parameters necessary for operation of the manipulator are determined.

After the Execution step in which the skill primitive is performed, the Postprocessing step that is important for error recovery is performed. First, sensing, modeling and confirmation are performed. At the Confirmation step, the result is judged to be correct or failed by an automatic process or by a human operator. If the judgment is failure, the process proceeds to the Supplementary classification step in which the necessity of a manual operation is determined. To correct the robot’s motions at each step, a manual operation module for robot control has been inserted in the terminal processing of each primitive motion. If the processing can advance to the next primitive task\(^{0}\)(i) by means of the manual operation, the correction is performed. Contrary to recovery through the backward correction process, this is recovery through a forward correction process. Let us call this type of error Class 0 as shown in Fig. 5(c). On the other hand, if a minor correction is not judged to be effective, it is necessary to proceed to recovery through a backward correction process. The return route after leaving the Supplementary classification step from the backward direction is decided by passing the classification of error step in Fig. 5(a).

We will describe the relative advantages of both forward and backward correction in the recovery processes. For recovery through the forward correction process, there is little wasted time and the operator can easily understand the overall correction since the task sequence does not change. For recovery through the backward correction process, the possibility that controls can be done more accurately than the last time is large when the same skill primitive is executed again, since the appropriate correction to the proper values of the system parameters is done. The whole system model of the robot may become correct if the error recovery is performed repeatedly.

4. TASKS IN DISTRIBUTION SYSTEMS

To make our error recovery process more widely applicable beyond maintenance, in this section we will take into account the error recovery of robots playing an active part in physical distribution. Therefore, let us take up the tasks involved in repacking objects from a large box into a small box at a distribution center. And we will choose a typical task of picking up an indicated object using the parallel jaw gripper on the robot.\[16\]
Fig. 6. Picking and placing task using a gripper

Furthermore, this section describes the main errors in the repacking tasks. And we will explain the process of recovery from each of those errors. In addition, we will describe the reusability of task planning. This refers to how many control programs can be used in a recovery process after an error occurs. Reusability will become an important aspect of research on error recovery.

A. Sequence of Repacking Tasks

Figure 6 shows the processes of picking up and placing objects such as PET bottles using a manipulator with a gripper. The flow of the skill sequence is as shown in Fig. 7, if there is no error. These primitive motions are as follows.

**Figure 7. Task sequence of picking and placing a bottle**

- **Skill₁** Move-to-approach: The robot gripper moves to the starting point of the approach motion.
- **Skill₂** Pre-grasp: The robot gripper opens to grasp the target object.
- **Skill₃** Approach: The robot gripper moves to the grasping point at low speed.
- **Skill₄** Grasp: The robot gripper grasps the target object.
- **Skill₅** Lift-up: The robot lifts the grasped object.
- **Skill₆** Approach: The robot gripper moves to a specific point in a reference frame of a large box.
- **Skill₇** Move-between-reference-frames: The robot gripper moves to a specific point in a reference frame of a small box.
- **Skill₈** Move-to-destination: The robot gripper moves to the destination point.
- **Skill₉** Bring-down: The robot lowers the grasped object.
- **Skill₁₀** Hand-open: The gripper opens to place the object.
- **Skill₁₁** Leave: The robot gripper moves to the safe area.
- **Skill₁₂** Home: The robot gripper returns to the starting point for the next approach motion.

B. Candidate Errors in Repacking Tasks

We take into account the main types of errors in the tasks of grasping, carrying and packing.

1. **Errors when grasping and lifting**
   1. a) An error in which the gripper does not reach a PET bottle in the step of “Approach” performed in Skill₁ (Fig. 8 (a)).
   1. b) An error in which the open gripper is not positioned around the cap of a PET bottle in the step of “Approach” performed in Skill₁ (Fig. 8 (b)).
   1. c) An error in which a PET bottle cannot be extracted due to tight packing in the step of “Lift-up” performed in Skill₅ (Fig. 8 (c)).

2. **Errors when carrying**
   2. a) An error in which a PET bottle is dropped in the step of “Departure,” “Move-between-reference-frames” and “Move-to-destination” in Skill₁ (i=6, 7, 8) (Fig. 9).

3. **Errors when packing**
   3. a) An error in which parallel movement to the bottom of the box in which there is a gap between the target position in the step of “Bring-down” performed in Skill₆. No movement of another bottle is performed (Fig. 10 (a)).
   3. b) An error in which the grasped PET bottle pushes the surrounding PET bottles in the box in the step of “Bring-down” performed in Skill₆ (Fig. 10 (b)).
   3. c) An error in which (3. a) and (3. b) occur at the same time (Fig. 10 (c)). Gaps occur both between the positions of the grasped bottle and the...
surrounding bottles.

(3. d) An error in which the task of “Bring-down” stops before completing (Fig. 10 (d)) because the available space is too small.

C. Error Recovery Processes

The process of recovery from each error will be explained and the reusability of planning will be slightly discussed. The following bracket numbers and letters coincide with the bracket numbers and letters describing the candidate errors in the previous section. Corrections are performed based on definite or indefinite causes derived from the classification of error, and the process restarts from the corresponding step.

(1) Cases of (1. a), (1. b) and (3. a)

| Class 1: Execution is simply repeated. |
| Class 2: The process is re-executed from the planning step after the modification of the planning techniques. For example, the motion techniques of lifting or lowering while slightly shaking are used. |
| Class 3: The process is restarted from the modeling step after modification of the geometric models of the bottles or box. |
| Class T: The process is re-executed from the sensing step after modification of the coordinate system. |

Reusability is high for only minor modifications of the numbers, unless another motion technique is used in Class 2.

(3) Case of (2. a)

In most cases, picking up a bottle in the box is changed to lifting the bottle that has dropped to the bottom of the path, and Class T is chosen since it is difficult to use the path of Class 1, 2, 3 and T. The skill sequence of the picking and placing tasks is executed from the beginning. Therefore, reusability is not high.

(4) Cases of (3. b) and (3. c)

The original task is completed, the process returns to the start of the flow, and then the task to move the bottle shifted to the correct position is executed. When the movement of many bottles is needed, it is desirable to consider whether or not the paths converge efficiently. Reusability is low, since the planning of additional tasks begins.
5. Conclusion

In this paper, we have described our error recovery technique based on specific actual operation tasks using robots that perform packing in physical distribution. We use the concepts of stratification of tasks and classification of errors as techniques to facilitate the construction of recovery processes. In this paper, we have mentioned the reusability of task planning which is an important theme in research on error recovery. Reusability is an indicator of the degree of simplicity or difficulty of the error recovery process.

In the future, we will further study optimum selection techniques for error recovery using a forward correction process or a backward correction process, suitable ways of judging whether an error occurred or not, optimum adjustment methods for the error recovery parameters in the backward correction process, and a fully automatic method for confirming the achievement of various tasks. We will attempt to apply our method of error recovery to real robot systems.

References


Akira Nakamura graduated from Keio University, Yokohama, Japan, in 1986. He received the ME and PhD degrees in electrical engineering from Keio University, in 1988 and 1991, respectively. He is currently a senior researcher at National Institute of Advanced Industrial Science and Technology (AIST). His research interests include robot planning, vision and control system. He is a member of IEEE, the Robotics Society of Japan (RSJ), the Society of Instrument and Control Engineers (SICE), the Japan Society of Mechanical Engineers (JSME), the Japan Society for Aeronautical and Space Sciences (JSASS), and the Institute of Electronics, Information and Communication Engineers (IEICE).
Kazuyuki Nagata received his B.S., and Ph.D. in Engineering from Tohoku University, Japan in 1986, and 1999, respectively. He joined Tohuku National Industrial Research Institute (TNIRI) at former AIST of MITI, Japan in 1986, he was assigned to Electrotechnical Laboratory (ETL) in 1991, and assigned to Planning Headquarters of AIST in 2001. He is currently a Senior Research Scientist with Intelligent Systems Research Institute of AIST. His current research interests include control of robotic manipulations, grasp planning, tactile sensing, and haptic interface. He is a member of the Japan Society of Mechanical Engineers (JSME), the Robotics Society of Japan (RSJ), and the Society of Instrument and Control Engineers (SICE).

Kensuke Harada received his B.S., M.S., and Doctoral degrees in Mechanical Engineering from Kyoto University in 1992, 1994, and 1997, respectively. He worked as a Research Associate at Hiroshima University from 1997 to 2002. From 2002, he has been working at the National Institute of Advanced Industrial Science and Technology (AIST). For one year from 2005 to 2006, he was a visiting scholar at the computer science department of Stanford University, Currently, he is a leader of the vision and manipulation research group. His research interest includes mechanics and control of robot manipulators and robot hands, biped locomotion, and motion planning of robotic systems. He is a member of IEEE, JSME, RSJ, and SICE.

Natsuki Yamanobe received her M.E., and Ph.D. on Precision Machinery Engineering from the Graduate School of Engineering, the University of Tokyo in 2004 and 2007 respectively. Since 2007, she has been a Research Scientist at the Intelligent Systems Research Institute at Japan’s National Institute of Advanced Industrial Science and Technology. Her research interests include robotic assembly, grasp planning, and human interface for tele-operation. She is a member of IEEE, JSME, JSPE, and RSJ.

Available online at: http://tsest.org/index.php/TCMS/article/view/289
Download full text article at: http://tsest.org/index.php/TCMS/article/download/289/166