Error Recovery Techniques Focused on Revival Process from Failures in Robotic Manufacturing Plants

Akira Nakamura
Department of Information Systems, Faculty of Engineering, Saitama Institute of Technology
1690 Fusaiji, Fukaya, Saitama 369-0293, Japan

Kensuke Harada
Robotic Manipulation Research Group, Systems Innovation Department
Graduate School of Engineering Science, Osaka University
1-3 Machikaneyama, Toyonaka 560-8531, Japan

Abstract

In recent years, working robots have been used in various fields, from manufacturing industries to domestic living spaces. An increasing number of difficult tasks are performed by robots, introducing a necessity for error recovery techniques. Our proposed error recovery technique is based on a method of returning to the process before the step in which the failure occurred and starting over. Of course, in practice, not only such a backward recovery but also a forward recovery that moves forward even after a failure occurs is used. In this study, we consider various paths from failure occurrence to recovery execution.

Keywords: error recovery, task stratification, error classification, robotic manufacturing plant

1. Introduction

In recent years, the speed of automated plants that use robots has increased. However, the possibility of errors occurring in automated operations has increased. Therefore, studying the recovery process after an error occurs remains important.[1]-[5]

We have studied the systematization of error recovery theory for several years and have previously proposed an error recovery method based on the concepts of task stratification and error classifications.[6]-[9] The main part of the method consists of the basic elements of sensing, modeling, planning, and execution sequences (Fig. 1). If an error occurs while working, the process proceeds to the recovery part,inwhich the cause of the error is estimated, the error is classified, the system is corrected, and the process is run with the corrected system and improved reliability.

Our proposed error recovery technique returns to the step before the one in which the failure occurred and starts from there. Of course, in practice, not only this type of backward recovery is used, but also forward recovery, which moves forward after a failure occurs. In this study, we consider various paths from failure occurrence to recovery execution.

The remainder of the paper is organized as follows. The concept of skills, which are motion primitives, is described in Section 2. The fundamental techniques for error recovery are presented in Section 3. Multiple possibilities of various recovery procedures are considered in Section 4, and examples of each are presented in Section 5. Section 6 concludes this paper.

2. Concept of skill

In this section, we briefly describe a skill, which we define as the unit of motion.[10]-[12]

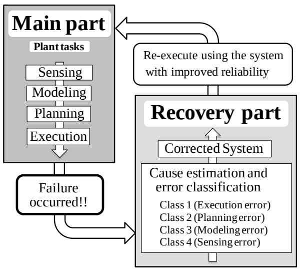


Fig. 1 Robot task system with an error recovery function

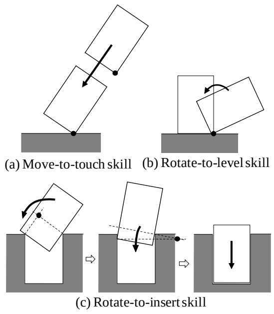


Fig. 2 Three fundamental skills

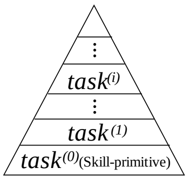


Fig. 3 Hierarchy of tasks

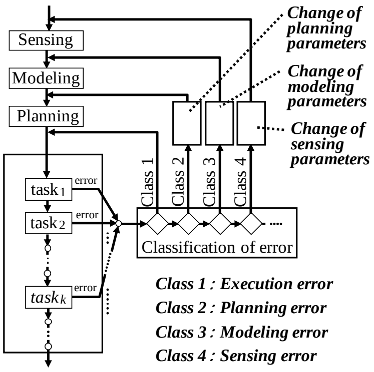


Fig. 4 Fundamental process flow with error recovery

2.1. *Skill primitives*

The three skill primitives, shown in Fig. 2 can be considered by analyzing human behavior. A task such as ‘assembly’ consists of these three skill primitives and the resemble skill primitives.

2.2. *Stratification of tasks*

The hierarchy of robotic manipulation tasks is illustrated in Fig. 3. Neglecting the servo layer, the first skill layer consists of movement units, such as the important skill primitives shown in Fig. 2.

3. Error recovery

Errors can occur in equipment for various reasons. In this section, we outline the concept of error classification and the proposed error recovery method.[6]-[9]

3.1. *Error classification*

Errors can be divided into four types, based on their possible causes: execution, planning, modeling, and sensing errors (Fig. 4).[6]-[9]

3.2. *Error recovery based on classification*

When an error occurs, the cause is first estimated, and then a suitable correction based on the estimated cause is implemented. In many methods, the executive process returns to the previous step and the task restarts from that step (Fig. 4).[6]-[9] For small-scale errors, the process returns to the previous step in the lowest layer of the task hierarchy, while for large-scale errors, the process returns to the previous step in the high-ranking hierarchy layers (Figs. 4 and 5). The probability of the same error occurring is reduced because of the correction that has been executed.

4. Various recovery pathways

As described in Section 3, recovery reruns the process by returning to the step previous to that in which an error occurs. In other words, the fundamental recovery process is executed backward. However, backward recovery is not the only conceivable option; other recovery procedures, such as forward recovery, without backward recovery are also possible. The occurrence of an error may affect the surrounding environment, such as the destruction of the arrangement or shape of certain objects.

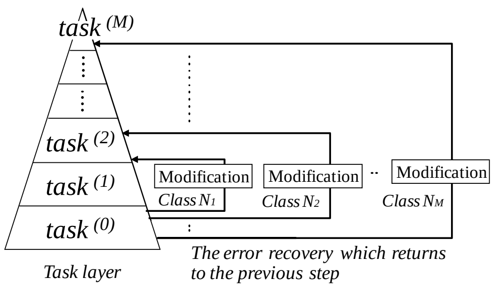


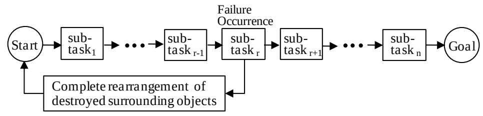
Fig. 5 The expression of task stratification and the process flow of the error recovery

Then, it could happen that the same recovery process as the previous one is not possible without modification when returning to the original process. In such cases, partial restructuring of planning is required. Of course, sensing and modeling are also necessary, even for local planning.

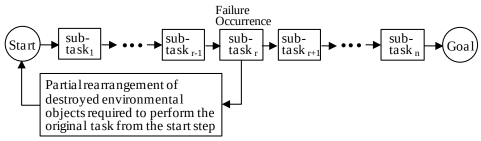
In the remainder of this section, we consider the degree of destruction of the environment surrounding the object and consider multiple possibilities for various recovery procedures.

4.1. *Recovery Formula I (RF-I): Exact original repetition*

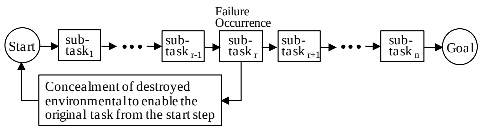
In this method, the environment is completely restored to its original state and work is resumed using the same process as the original starting point (Fig. 6 (a)). The process is performed by replacing the original object, or part of it, with a new object, if necessary. The same



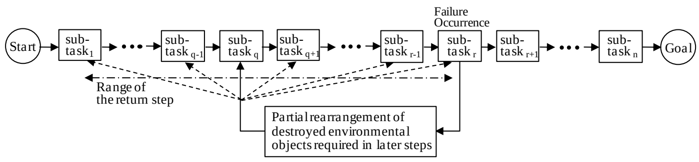
(a) (RF-I) Exact Original Repetition



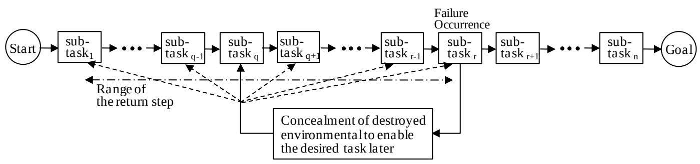
(b) (RF-II) Incomplete Original Reproduction from Start Step



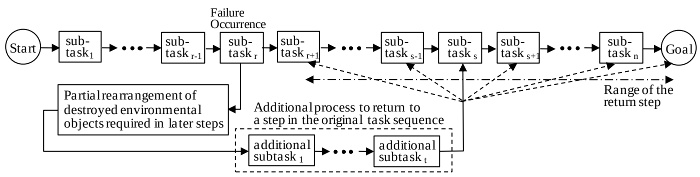
(c) (RF-III) Original Repetition from Start Step with concealment



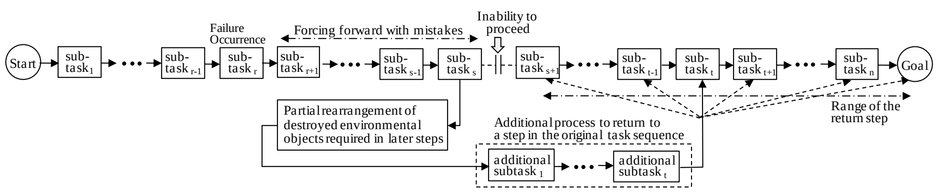
(d) (RF-IV) Incomplete Original Reproduction from Step along the Way



(e) (RF-V) Original Repetition from Step along the Way with concealment



(f) (RF-VI) Recovery through a different process



(g) (RF-VII) Continuation of work, including consideration of any recovery process that may be required later

Fig. 6 Various error recovery paths

approach applies to all recovery formulae outlined in this section (RT-II to RT-VII). Note that any replacement results in additional costs.

4.2. *Recovery Formula II (RF-II): Incomplete original reproduction from start step*

In this method, the environment is not restored to its original state, but is restored to a level that does not cause any problems for subsequent processes; the process is restarted from the original starting point using the same process (Fig. 6 (b)).

4.3. *Recovery Formula III (RF-III): Original repetition from start step with concealment*

In this method, the surrounding environment that has been broken from the occurrence of an error is concealed, and it is superficially restored; work resumes from the original starting point using the same process (Fig. 6 (c)). By concealing the broken environment, no problems occur in subsequent operations. However, if the concealed area reappears for any reason, the appearance of the object may be spoiled. For customer products, for example, this may result in a decrease in value.

4.4. *Recovery Formula IV (RF-IV): Incomplete original reproduction from step along the way*

In this method, the environment is not restored to its original state, but is restored to a level that does not cause any problems for subsequent processes; the process is resumed from a point in the middle of the original process, using the same process (Fig. 6 (d)).

4.5. *Recovery Formula V (RF-V): Original repetition from step along the way with concealment*

In this method, the surrounding environment that has been damaged by an error occurrence is concealed, restoring it superficially; work is resumed from a step in the middle of the original process, using the same process (Fig. 6 (e)). The recovery work procedure and commercial value are subject to the same concerns as those in RF-III.

4.6. *Recovery Formula VI (RF-VI): Recovery through different processes*

In this method, the environment is not returned to its original state, but instead is transformed into a state in which there are no problems with subsequent work. Then, a process that is different from the original process is executed to reach the goal (Fig. 6 (f)). Partial planning becomes necessary for the process to either achieve its goal or to return to a previous step in the middle of the original process; sensing and modeling accompany local planning. The final target state is the same as that of the original. Therefore, if the originally assumed process is restored to the middle, no new planning is required from the middle of the recovery process.

4.7. *Recovery Formula VII (RF-VII): Continuation of work, including consideration of any recovery process that may be required later*

In this method, work is continued for as long as possible, even if an error occurs. Corrections are made to the environment when the work becomes unsustainable, thereby continuing the work (Fig. 6 (g)). Local planning must be performed to execute the modification; sensing and modeling are performed for local planning. Such a forward recovery process is symmetrical to the backward recovery methods outlined in RF-I to RF-VI.

5. Illustrative examples of various recovery pathways

This section provides examples of each method outlined in Section 4.

5.1. *RF-I: Exact original repetition*

An example of RF-1 can be found in “Recovery Type I” in reference [9], where error recovery is considered in an assembly task, in which a hook is stuck to a plate by screws.

5.2. *RF-II: Incomplete original reproduction from start step*

An example of RF-II can be considered in that presented for RF-I, because the outlined example does not consider whether the environment was destroyed. If the environment was destroyed, it is simply restored to such a level that the recovery task can be performed without

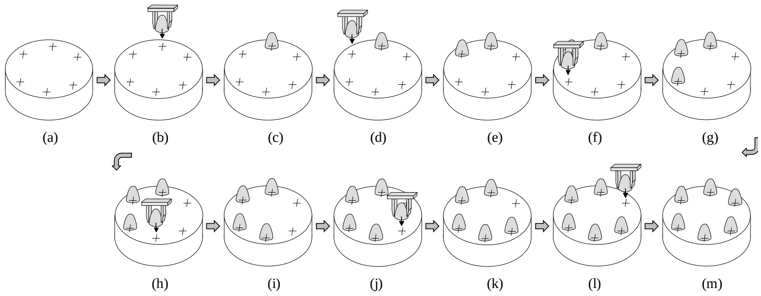


Fig. 7 Cake-making task

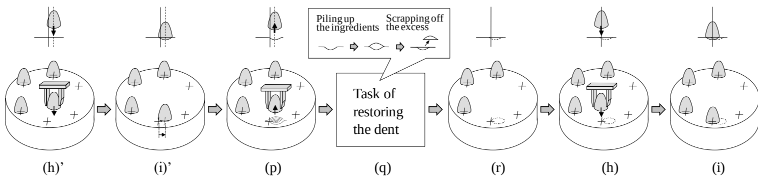


Fig. 8 Error and recovery process in cake-making task

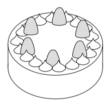


Fig. 9 Whole Cake with Cream Decoration

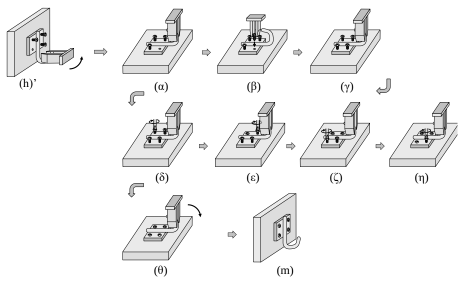


Fig. 10 The recovery task performed by laying the object on its side

problems, and then the original task is rerun from the start. Additionally, reference [7] provides an example of recovery for an error that damages the placement of objects around it.

5.3. *RF-III: Original repetition from start step with concealment*

Examples of RF-III are shown in conjunction with R-V, which are described later.

5.4. *RF-IV: Incomplete original reproduction from step along the way*

An example of RF-IV can be found in "Recovery Type II" in reference [9].

5.5. *RF-V: Original repetition from step along the way with concealment*

Consider the cake-making task shown in Fig. 7. To obtain a cake that is cut with equal divisions from a whole cake, decorations, such as strawberries, must be placed at the correct positions in terms of the angles and position of the cutting device. Let us consider the error in which the position of the decoration is shifted to the side, as shown in Fig. 8. In this case, the following recovery task is performed: the decoration is lifted, the dent in the cream is filled and finished with a scraper, and the decoration is placed back on the cake. Note that the repair task becomes more difficult if the top surface of the cake is decorated with cream before the strawberries are placed on top, as shown in Fig. 9.

The error recovery approach of RF-V is characterized by concealing the failure such that it is not visible. If the process returns to the starting step and the same process can be re-executed, the approach corresponds to RF-III. In the cake-decorating example, implementing RF-III is easy if the placement of the first strawberry fails; however, if the placements of the second or subsequent strawberry fail, returning to the starting state is difficult.

5.6. *RF-VI: Recovery through different processes*

Let us consider an example of RF-VI using reference [9], which considers an assembly task in which a hook is stuck to a plate by screws. The recovery task is performed by laying the object on its side so that the temporarily inserted screw could be stabilized, as shown in Fig. 10. In RF-VI, a process different from the original process is executed to reach the original goal. Note that this method cannot always be used, as problems could arise, such as the inability to lay the object on its side.

5.7. *RF-VII: Continuation of work, including consideration of any recovery process that may be required later*

An example of RF-VII can be found in "Recovery Type III" in reference [9].

6. Conclusion

If an error occurs during work, the process advances to a recovery part. Many types of recovery processes can be selected using our proposed error recovery method, which is based on both task stratification and error classification. In this paper, we considered the various paths from failure occurrence to recovery execution, and not only recovery of backward type but also recovery of forward type were shown.

There are various recovery routes, and various methods are conceivable for which route to choose. In future work, we would like to determine the optimal recovery process. The method of process selection will change depending on the type of robotic system considered; however, deriving a systematized method remains important.

References

1. B. R. Donald, Planning multi-step error detection and recovery strategies, Int. J. Robot. Res., 9(1) (1990) 3-60.
2. T. Niemueller, G. Lakemeyer and S. S. Srinivasa, A Generic Robot Database and its Application in Fault Analysis and Performance Evaluation, in Proc. IEEE/RSJ Int. Conf. on Intell. Robots Syst., (Vilamoura, Portugal, 2012), 364-369.
3. E. D. Lello, M. Klotzbucher, T. D. Laet and H. Bruyninckx, Bayesian Time-Series Models for Continuous Fault Detection and Recognition in Industrial Robotics Tasks, in Proc. IEEE/RSJ Int. Conf. on Intell. Robots Syst., (Tokyo, Japan, 2013), 5827-5833.
4. E. Krabbe, E. Kristiansen, L. Hansen and D. Bourne, Autonomous Optimization of Fine Motions for Robotic Assembly, in Proc. IEEE Int. Conf. Robot. Autom., (Hong Kong, China, 2014), 4168-4175.
5. A. S. Wang and O. Kroemer, Learning Robust Manipulation Strategies with Multimodal State Transition Models and Recovery Heuristics, in Proc. IEEE Int. Conf. Robot. Autom., (Montreal, Canada, 2019), 1309-1315.
6. A. Nakamura, K. Nagata, K. Harada, N. Yamanobe, T. Tsuji, T. Foissotte and Y. Kawai, Error recovery using task stratification and error classification for manipula-tion robots in various fields, in Proc IEEE/RSJ Int. Conf. on Intell. Robots Syst., (Tokyo, Japan, 2013), 3535-3542.
7. A. Nakamura, K. Nagata, K. Harada and N. Yamanobe, Technique of Recovery Process and Application of AI in Error Recovery Using Task Stratification and Error Classification, J. Robotics, Networking and Artificial Life, 5(1), 2018, pp. 56-62.
8. A. Nakamura, N. Yamanobe, I. R. Alpizar, K. Harada and Y. Domae, Cost-oriented Planning for Error Recovery in an Automation Plant, J. Robotics, Networking and Artificial Life, 6(4), 2020, pp. 225-230.
9. A. Nakamura, N. Yamanobe, I. R. Alpizar, K. Harada and Y. Domae, Selection of Optimal Error Recovery Process using Evaluation Standards in Automated Plants, J. Robotics, Networking and Artificial Life, 8(3), 2021, pp. 211-217.
10. T. Hasegawa, T. Suehiro and K, Takase, A robot system for unstructured environments based on an environment model and manipulation skills, in Proc. IEEE Int. Conf. Robot. Autom., (Sacramento, USA, 1991), 916-923.
11. A. Nakamura, T. Ogasawara, T. Suehiro and H. Tsukune, Fine motion strategy for skill-based manipulation, Artificial Life and Robotics, Springer, 1(3), 1997, pp. 147-150.
12. A. Nakamura, K. Kitagaki and T. Suehiro, Using simplified geometric models in skill-based manipulation, Advanced Robotics, 18(8), 2004, pp. 835-858.

Authors Introduction

Prof. Akira Nakamura

He received the Ph.D. degree in Electrical Engineering from Keio University in 1991. From 2021, he has been working as a Professor at Faculty of Engineering of Saitama Institute of Technology. His research interests include robot planning, vision and control system.

Prof. Kensuke Harada

He received his Doctoral degrees in Mechanical Engineering from Kyoto University in 1997. From 2016, he has been working as a Professor at Graduate School of Engineering Science, Osaka University.