# Experimental Evaluation of Manipulator Teleoperation System Based on Trajectory Planning for Obstacle Removal Task in Nuclear Plant Decommissioning

Hashimoto, Tatsuya ; Tazaki, Yuichi ; Matsuda, Fumiya ; Kitajima, Katsumasa ; Nagano, Hikaru ; Yokokohji, Yasuyoshi

(Citation)
Journal of Robotics and Mechatronics,36(1):49-62

(Issue Date)
2024-02-20

(Resource Type)
journal article

(Version)
Version of Record



(URL)
https://hdl.handle.net/20.500.14094/0100488636

https://doi.org/10.20965/jrm.2024.p0049

Paper:

# Experimental Evaluation of Manipulator Teleoperation System Based on Trajectory Planning for Obstacle Removal Task in Nuclear Plant Decommissioning

**Tatsuya Hashimoto*, Yuichi Tazaki**, Fumiya Matsuda**, Katsumasa Kitajima*, Hikaru Nagano**, and Yasuyoshi Yokokohji****

*Mitsubishi Heavy Industries, Ltd.
1-1-1 Wadasaki-cho, Hyogo-ku, Kobe, Hyogo 652-8585, Japan
E-mail: {tatsuya.hashimoto.84@nu., katsumasa.kitajima.c2@}mhi.com
**Kobe University
1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan
E-mail: {tazaki@mech, 221t361t@stu, nagano@mech, yokokohji@mech}.kobe-u.ac.jp
[Received July 28, 2023; accepted December 11, 2023]

**In the teleoperation system of a dual-arm manipulator developed to retrieve fuel debris and reactor interior structures at the Fukushima Daiichi Nuclear Power Plant, we used software in which an obstacle avoidance function (trajectory planning) was implemented to conduct a mock test simulating obstacle removal operations in narrow spaces. The test results confirmed the validity of the obstacle avoidance function, the executability of a series of necessary tasks, and the improved operability. In addition, issues were identified using the test data.**

**Keywords:** robotic manipulator, teleoperation, nuclear power plant decommissioning, trajectory planning

## 1. Introduction (Background, Related Studies)

Since the retrieval of fuel debris at Fukushima Daiichi Nuclear Power Plant requires long operations in the high radiation-dose environment inside the primary containment vessel (PCV), high expectations are placed on the application of robotic technology [1]. The International Research Institute for Nuclear Decommissioning (IRID) has proposed an operational process for the retrieval of fuel debris using teleoperated manipulators [a].

Various computer-assisted teleoperation methods have been proposed thus far, and research is progressing to apply teleoperation to extreme environments, such as the maintenance of nuclear reactors, decommissioning of nuclear reactors, operations in outer space [2, 3], and the maintenance of underwater facilities [4]. Existing methods divide the tasks of recognition, judgment, and operation between a computer and human operator to reduce the recognition load of the operator and increase the efficiency and safety of the operation. Specific examples include automatic generation of operational procedures [5], automatic generation of valve turning [6,7], and automatic recognition and generation of the grasping and cutting operations of pipes [8, 9]. Additionally, methods have been investigated to automatically generate a trajectory to arrive at the instructed target position, rather than having the operator directly operate the end effector of the manipulator [10, 11]. However, most previous studies were based on operating in open spaces or environments with very few obstacles, and very few studies have explicitly considered obstacle avoidance in narrow environments, such as those expected in reactor decommissioning. To the best of our knowledge, no studies have conducted evaluation tests specifically premised on the retrieval of fuel debris.

To improve the safety and operational efficiency in the teleoperation of dual-arm manipulators in narrow spaces, we have been developing a teleoperation interface that automatically performs trajectory planning to support the operator. In our previous study, we conducted a mock test based on obstacle removal inside a collapsed nuclear reactor to quantitatively compare teleoperation using trajectory planning and the conventional method of direct manipulation of the end effector and confirmed that the proposed method is superior in terms of operational efficiency, safety, and operator load [12]. However, we found that because of the insufficient time resolution in trajectory planning, interference with the actual environment occurred when tracking the planned trajectory. Furthermore, we did not consider complex operations that require trajectory replanning.

The objectives of this study were to confirm the operability of a dual-arm manipulator teleoperating system that employs trajectory planning in reactor decommissioning tasks, and to identify issues to further improve the system. To this aim, we resolved the aforementioned technical issue of obstacle avoidance and conducted a mock test with a high degree of difficulty to simulate the actual environment. The contributions of this study are as follows. First, in addition to the original planner to generate the reference trajectory to the target pose (which is hereafter referred to as the "global planner"), a local trajectory planner that

plans tracking motions to the reference trajectory at a finer resolution was added. Consequently, the certainty of obstacle avoidance was improved, and it became possible for the operator to intervene when necessary, thus improving operational flexibility. Second, we expanded the control interface. Specifically, we made it possible to modify the planning model according to the environmental changes caused by cutting and grasping objects. We implemented a function that interactively aligned the actual environment and the model. Third, using the improved system, we conducted a mock test to identify issues for future improvements. It is highly significant to carry out mock tests since it is difficult with computer simulation-based virtual environments to faithfully replicate the quality of actual camera images or deviations between the environmental model and actual environment.

The remainder of this paper is organized as follows. In Section 2, we describe the operational process of debris retrieval and the associated technical problems. In Section 3, we describe the configuration and operational method of the teleoperation interface that employs trajectory planning. In Section 4, we describe the settings of the mock test. The results are discussed in Section 5. Section 6 presents the conclusions and future issues.

## 2. Operational Procedure of Scaled-Up Fuel Debris Retrieval and Technical Issue

Below, the procedure for obstacle removal during large-scale fuel debris retrieval is described. An outline of this process is shown in **Fig. 1**.

I. Advance planning of the task is performed using a simulator. When removing an obstacle inside the pedestal opening, the target for removal is determined, and the positions to be grasped with the manipulator hand or cut with the cutter are confirmed on the control interface of the simulator. The operator controls the robot model in the simulator environment and generates a trajectory to move to the vicinity of the removal target. To determine the obstacle-removal procedure, it is necessary to employ physical simulations or other means to find a procedure that does not induce collapse and is easy to operate.

II. The operator enters the onsite operation phase. For onsite operation, the obstacle-removal robot is first moved to the interior of the pedestal. By replaying the trajectory generated by the operator in advance, the obstacle-removal robot is controlled to track this trajectory and move to the vicinity of the target of removal while avoiding obstacles.

III. When the robot has moved to the vicinity of the removal target, of the two manipulators of the robot, the hand attached to the end of one manipulator is used to grasp the target and the cutter attached to the end of the other manipulator is used to cut the target. While grasping and cutting the removal target, the operator manually controls the hand or cutter to move it in small increments while monitoring the camera images. A separate investigation is required to determine the methods for confirming the grasping force, determining the cutting position, or adjusting the cutting speed.

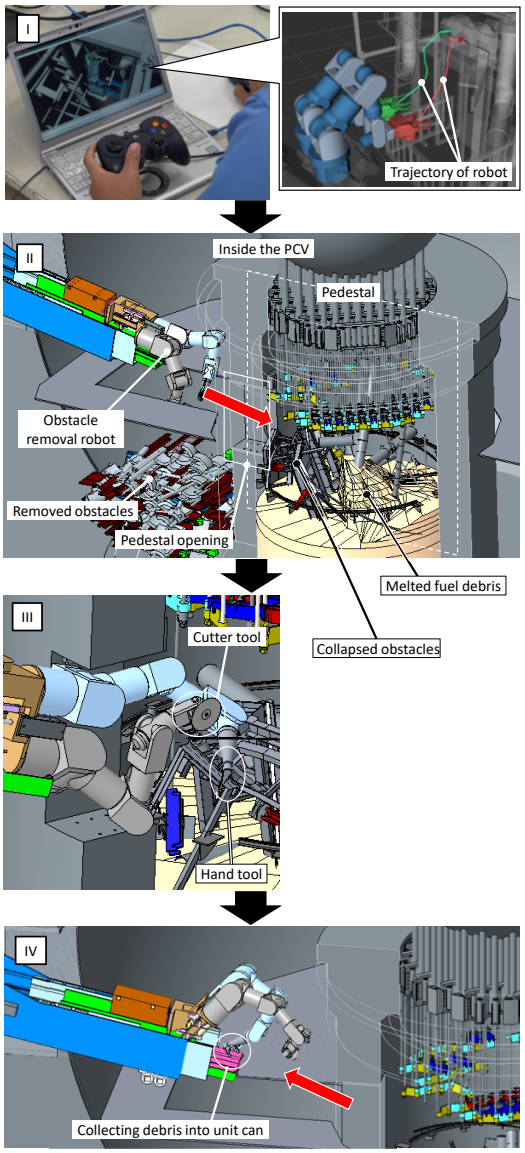



**Fig. 1.** Workflow for obstacle removal task.

IV. When the obstacle has been grasped and cut while maintaining the state in which the obstacle is grasped by hand, the robot moves out of the pedestal while avoiding the obstacles. If no fuel debris is attached to the obstacle or the structure is transported out of the pedestal, it is left behind on the floor of the PCV. If the obstacle includes debris, it is deposited in a unit can and transported outside of the PCV. Therefore, it is necessary to establish a method for determining whether debris is attached to obstacles.

The above constitutes the flowchart of the obstacle removal task.

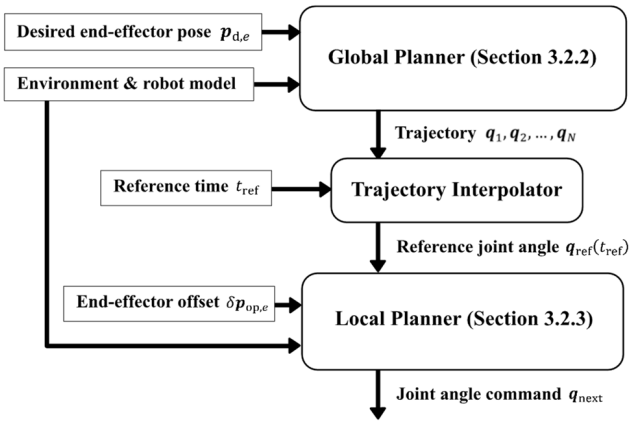



**Fig. 2.** Configuration of trajectory planner.

## 3. Manipulator Teleoperation Interface Based on Trajectory Planning

### 3.1. Configuration of Trajectory Planner

The configuration of the trajectory planner is shown in **Fig. 2**. Integration with a graphical user interface and the control method are described in Section 3.3. The global planner calculates the robot's trajectory starting from its current state to the target state while avoiding obstacles, based on the geometric model of the environment and robot, the robot's kinematic model, and the instructed target position of the end effector. The trajectory planned by the global planner is transmitted to the trajectory interpolator. Here, the robot poses at the reference time are computed using linear interpolation, and the reference poses are input to the local planner. The offsets of the end effector's target position and pose based on the operator's operation are input. The local planner determines the joint angle command to the robot for the next time step, which tracks the reference pose and operator controls while avoiding interference with the environment.

### 3.2. Trajectory Planning Algorithm

#### 3.2.1. Outline

Here, we outline the trajectory generation algorithm used in this study. The objective of trajectory generation is to generate a trajectory that moves the end effector to the target position without interfering with the environment. Although a study on trajectory generation that allows contact with the environment exists [13], we believe that interference must be avoided at all times because of the extremely high risk of unexpected contact with the environment during reactor decommissioning operations.

Although the present method is similar to that proposed in [14], it does not employ the prioritized optimization proposed there; instead, it employs a multi-objective optimization based on weights, which is simpler. The main improvement lies in the use of a two-step strategy for trajectory planning consisting of global and local trajectories. Global trajectory planning involves planning a trajectory from the initial position to the target position using a coarse resolution (20–30 divisions). Local trajectory planning consists of planning the immediate movements that follow the global trajectory while avoiding obstacles using a finer resolution (100 ms period). The reliability of obstacle avoidance can be improved by adopting this two-step strategy. Specifically, whereas it was not possible to avoid obstacles using global trajectory planning alone in a previous study [12], which made it necessary to set a margin of approximately 50 mm in the robot model, we were able to eliminate this margin by adding a local trajectory planner. Consequently, it became possible to pass through narrow environments that were not passable using only the global trajectory planner. Furthermore, this made it possible for the operator to intervene while trajectory tracking was being performed when it was necessary to execute unexpected trajectory modifications because of model errors or errors originating in the registration.

#### 3.2.2. Global Trajectory Planning

In the trajectory planning method, a trajectory of the robot is expressed as a discrete time-series of poses. The discrete time steps consist of $N$ steps from 0 to $N$. Denoting by $n$ the robot's degrees of freedom, the generalized coordinate vector of the robot at discrete time $k$ can be expressed as $q_k \in \mathbb{R}^n$. Global trajectory planning can be formulated as the problem of determining an $N$-step time series that minimizes the following cost function:

$$\begin{aligned}
&\text{find } q_1, q_2, \ldots, q_N, \\
&\text{minimize } J = \sum_{k=1}^{N} \sum_{e} \left\| w_{\text{reach}} \left( p_e \left( q_k \right) - p_{\text{d},e} \right) \right\|^2 \\
&\qquad + \sum_{k=0}^{N} \sum_{i,j} \left\| w_{\text{avoid}} y_{\text{avoid},i,j} \left( q_k \right) \right\|^2 \\
&\qquad + \sum_{k=0}^{N} \left\| w_{\text{range}} y_{\text{range}} \left( q_k \right) \right\|^2 \\
&\qquad + \sum_{k=0}^{N-1} \left\| w_{\text{smooth}} \left( q_{k+1} - q_k \right) \right\|^2 .
\end{aligned}$$

Because the initial pose $q_0$ is known, it is not included in the decision variables. The first term of the cost function expresses the deviation from the target end effector position, where $p_e(q)$ is the position of end effector $e$ in the generalized coordinates $q$, and $p_{\text{d},e}$ is the target position of the end effector. The second term represents the cost of interference with the environment. The nearest neighboring point of link $i$ to obstacle $j$ is denoted $r_{i,j}(q)$, the nearest neighboring point of obstacle $j$ to link $i$ is denoted $s_{i,j}(q)$, and the normal vector (outward) of the obstacle shape at $s_{i,j}(q)$ is denoted $\eta_{i,j}(q)$. The interference depth between the link and obstacle can then be expressed as

$$y_{\text{avoid},i,j}(q) = \max \left( 0, \eta_{i,j}(q)^T \left( r_{i,j}(q) - s_{i,j}(q) \right) \right).$$

The third term represents the cost of the deviation from the joint movable range. By denoting $q_{\min}$ and $q_{\max}$ as the

lower and upper limits of the movable range, respectively, the deviation from the movable range can be expressed as

$$y_{\text{range}}(q) = \max(0, -q + q_{\min}, q - q_{\max}).$$

The fourth term represents the cost of joint movement. This is used to smooth the planned trajectory. $w_{\text{reach}}$, $w_{\text{avoid}}$, $w_{\text{range}}$, and $w_{\text{smooth}}$ are the weighting coefficients of the corresponding costs. The weights are set according to the priority by which the costs are to be reduced; the larger the weighting coefficient, the higher the priority. In this study, the following weights were set:

$$w_{\text{avoid}}, w_{\text{range}} \gg w_{\text{reach}} \gg w_{\text{smooth}}.$$

#### 3.2.3. Local Trajectory Planning

Local trajectory planning determines the immediate changes required for the robot to move from its present state toward the target state along a global trajectory. Let $t_{\text{ref}}$ denote the reference time of the reference trajectory generated by global trajectory planning. $q_{\text{now}}$ denotes the robot's current target state held by the local trajectory planner, and $q_{\text{next}}$ denotes the target state after updating. Local planning can be expressed as the following cost function minimization:

$$\begin{aligned}
&\text{find } q_{\text{next}} \\
&\text{minimize } J; \\
&J = \left\| w_{\text{track}} \left( q_{\text{next}} - q_{\text{ref}} \left( t_{\text{ref}} \right) \right) \right\|^2 \\
&\quad + \sum_e \left\| w_{\text{op}} \left( p_e(q_{\text{next}}) - \left( p_e(q_{\text{ref}} \left( t_{\text{ref}} \right)) - \delta p_{\text{op},e} \right) \right) \right\|^2 \\
&\quad + \sum_{i,j} \left\| w_{\text{avoid}} y_{\text{avoid},i,j} \left( q_{\text{next}} \right) \right\|^2 \\
&\quad + \left\| w_{\text{range}} y_{\text{range}} \left( q_{\text{next}} \right) \right\|^2 \\
&\quad + \left\| w_{\text{smooth}} \left( q_{\text{next}} - q_{\text{now}} \right) \right\|^2
\end{aligned}$$

The first term is the cost of tracking the reference trajectory obtained by global trajectory planning, where $t_{\text{ref}}$ is the reference time, and $q_{\text{ref}}(t_{\text{ref}})$ is the target pose at the reference time computed by interpolating the reference trajectory. The second term is the cost of tracking the target pose commanded by the operator, where $\delta p_{\text{op},e}$ is the correction to the end effector position commanded by the operator via intervention control. The third and fourth terms are the costs to avoid interference and the movable range cost, respectively, as in global trajectory planning. Finally, the fifth term is the cost of the difference in target poses before and after updating, and has the effect of minimizing pose changes. The weighting coefficients are set according to the guidelines used in global trajectory planning.

#### 3.2.4. Computation Algorithm

Because both the global and local trajectory planning described above constitute nonlinear least-squares problems, it is possible to determine a locally optimal solution using an iterative numerical method. Although our previous study [12] employed the Gauss–Newton algorithm, the computational cost was $O((Nn)^3)$. Therefore, the generation of a long trajectory entailed a high computational cost. Here, we employed differential dynamic programming (DDP), a nonlinear optimal control method that reduces the computational cost. Therefore, the time required for global trajectory planning can be substantially reduced. Specifically, in the global trajectory planning used for "(ii) move into opening" (described later), the computation time using the Gauss–Newton algorithm was 30.6 s, whereas it was 10.9 s using DDP.

### 3.3. Control Interface

We developed a control interface in which the aforementioned global and local trajectory planners were implemented. The operational flow using the control interface and examples of control screens are shown in **Fig. 3**. The elements of the operational flow are described in detail below. The letter headings below correspond to the figure symbols.

(a) Selection of the environmental model: The operator selects the structures that exist in the environment. In the model, the structures are divided into parts, such as opening, structure A, structure B, and so on, which can be selected via check box. There are two types of check boxes: "display" to display the model on the screen and "interference" to activate collision detection. Checks are usually entered in both boxes, although the check is removed from the "interference" box to inactivate collision detection when the target of removal is being grasped or cut, thus allowing the hand or cutter to make contact.

(b) Registration: The operator adjusts the relative positions and poses of the robot against the environmental model, or of the environmental model against the robot, so that the environmental model and point clouds acquired by the range sensors mounted on the robot match. This adjustment can be performed in two ways: by automatic processing using the iterative closest point algorithm or by manual processing based on visual observation. The detailed task flow of the registration is shown in the upper-right panel of **Fig. 3**. The operator displays the point cloud and target model (environmental model that should be matched) on the control interface screen and uses the mouse to drag arrows or rings to translate and/or rotate the target so that its 3D shape aligns with the point cloud. Through this procedure, the deviation of the point cloud from the environmental model is input into the system through an intuitive operation.

(c) Selection of the control mode: There are three major methods for moving the robot model on the control interface, one of which the operator chooses. The first is to generate a new trajectory by setting the target position of the end effectors, computing a collision-free trajectory towards the target, and then executing

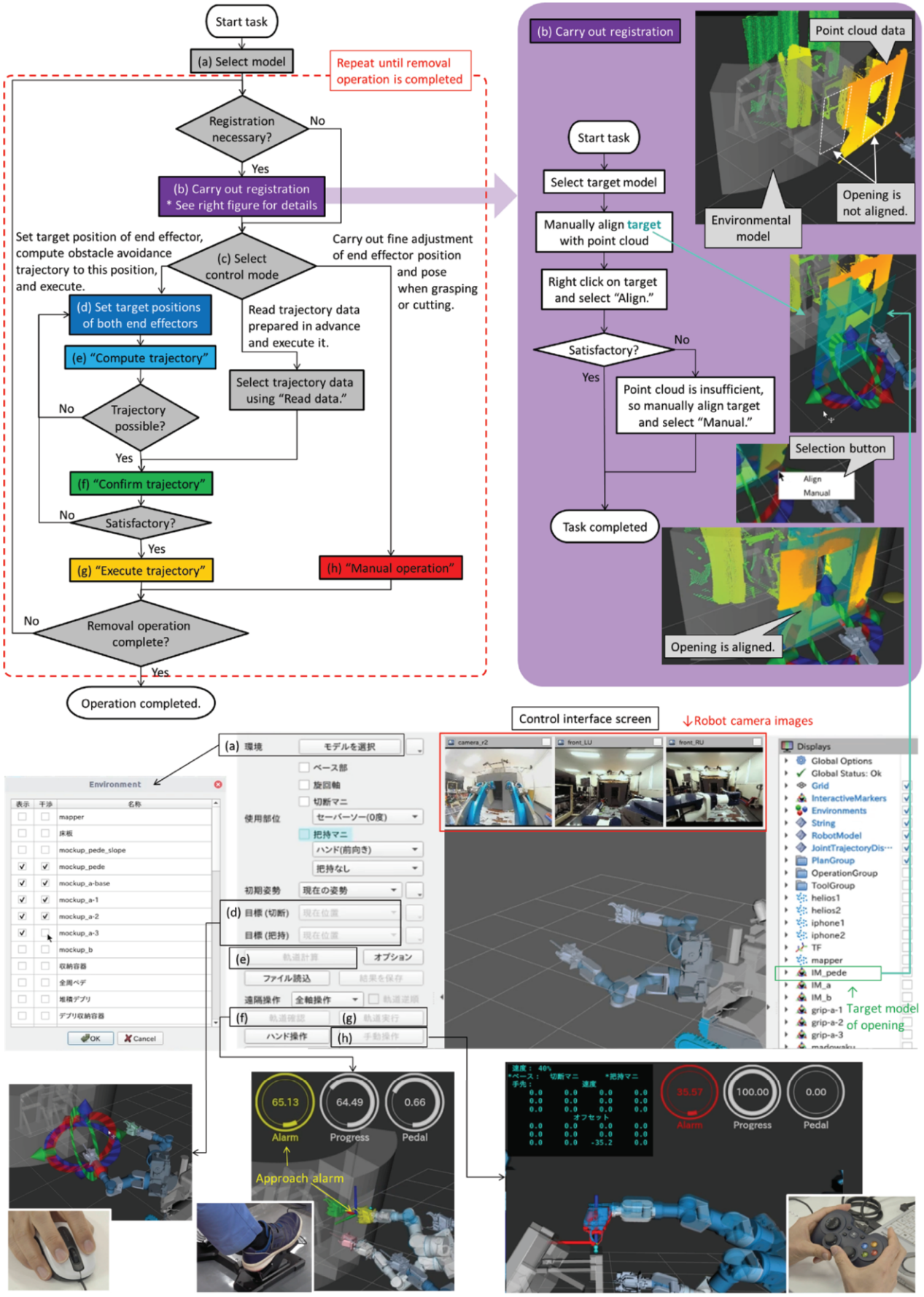



**Fig. 3.** Workflow and control screen of control interface.

it. The second step is to replay a trajectory generated in advance, which is performed by retrieving and executing the trajectory data. Third, when performing a complex task (grasping or cutting) in which trajectory planning cannot be used, the end effector's position and pose are finely adjusted while observing camera images. The first and second are described in detail in (d)–(g) below, and the third is described in (h).

(d) Setting the target positions of the two end effectors: The target of the end effector is displayed on the control interface and the operator translates and/or rotates it. This operation is performed separately for the grasping manipulator (mounted with a hand at the end) and the cutting manipulator (mounted with a cutter at the end).

(e) Trajectory computation: Trajectory data that avoid obstacles are generated by the global trajectory planner. The trajectory data consist of an array of connected joint-axis values spaced at a coarse time resolution from the initial pose to the target pose. If the computation does not converge in approximately 60 s, it implies that it is not possible to generate an obstacle-avoidance trajectory, and the target position of the end effector must be revised.

(f) Confirmation of trajectory: Using the robot model on the control interface, the trajectory data are checked to determine whether it can be executed. First, the operator either selects the trajectory data output using trajectory computation just performed or loads it from a file stored in advance. When the operator presses down on the foot pedal, the robot model begins to move at a speed corresponding to the depressed amount of the pedal. Concurrently, local trajectory planning is automatically performed as a background process that prevents any penetration with obstacles when the discrete trajectory data are executed. During the operations of "confirmation of trajectory," "execute trajectory," and "manual operation," the latter two of which are described below, the robot part which is closest to the environmental model on the control interface lights up, which alerts the operator of the risky parts (approach alarm). The part is displayed in yellow when it approaches 100 mm or less and in red when it approaches 50 mm.

(g) Execute trajectory: When the operator wishes to execute the trajectory data confirmed in the above trajectory confirmation step with the real-world robot, he or she presses the "execute trajectory" button and presses down on the foot pedal. The results of the local trajectory planning performed for trajectory confirmation are not stored, and local trajectory planning is performed once again at the trajectory execution stage. The operator must cautiously monitor the part indicated by the approach alarm in camera images while executing the trajectory. At this point, the operator can intervene to correct the end effector position using a gamepad. Using the trajectory data or the operator's intervention control as inputs, the local trajectory planner successively transmits the collision-free pose (array of joint-axis values) to the robot.

(h) Manual operation: This involves finely adjusting the position and pose of the end effector when grasping or cutting. Using a gamepad, the operator controls the $X$, $Y$, and $Z$ positions, and the roll, pitch, and yaw poses in the end-effector coordinate system. At this stage, local planning is constantly performed with its target end-effector positions set as the current positions, and thus there is no risk that the elbow or shoulder of the manipulator will collide with the surrounding environment.

The above operations are repeated until the obstacle has been removed and the robot has returned to the starting position.

## 4. Planning of Mock Test

A mock test was performed to verify the effect of the proposed method on manipulator teleoperation using the control interface described above.

### 4.1. Prerequisites of Mock Test

We established the prerequisites for the mock test. First, we assume that a 3D model of the environment has been obtained in advance based on information obtained from a preliminary survey or other means. Furthermore, because our purpose was to evaluate the operational efficiency or safety of a general-purpose teleoperation system, we do not discuss specific engineering methods or the mechanisms of grasping and cutting objects in detail.

### 4.2. Dual-Arm Robot

A comparison of the obstacle-removal robot under consideration for use at the Fukushima Daiichi Nuclear Power Plant and the dual-arm robot used in this study is shown in **Fig. 4**. Because the link configuration of the dual-arm robot [b] was similar to that of the obstacle-removal robot, we decided that it could be used in the present mock test. The linear movement of the telescopic arm of the obstacle-removal robot was substituted by the forward-backward movement of the crawler, and the telescopic arm's approach diagonally from above was simulated by setting the mockup at an inclination angle of 15°.

Nine cameras were mounted on the robot. Two cameras were mounted on each end tool (the hand and cutter tools). A pan-tilt-zoom camera was mounted at the top of the robot's body, whereas four fixed cameras were mounted on the body to squarely face and monitor the four corners of the pedestal opening.

Among the end tools, the claws of the hand tool could be opened or closed via teleoperation. The operator grasps the removal target by gradually closing their claws while

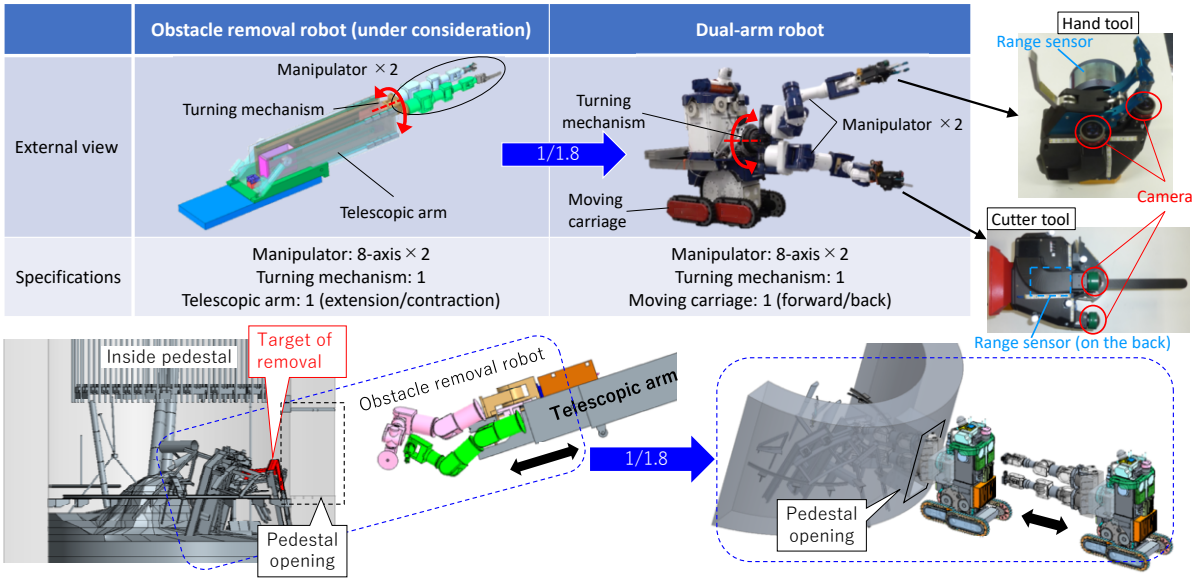



**Fig. 4.** Comparison of obstacle-removal robot (currently under consideration) and dual-arm robot used in the mock test.

monitoring the camera images. Although a saber saw blade was attached to the cutter tool, actual cutting task of the removal target was not carried out in the mock test. Instead, it was determined that cutting was successfully executed when it was possible to confirm in the camera images that the blade had made contact with the section to be cut.

### 4.3. Mock Test

In the above-described dual-arm robot, the dimensions of the manipulator were scaled down to 1/1.8 of those of the obstacle-removal robot. Therefore, we used a 1/1.8-scale model of the pedestal opening, which allowed us to verify the degree of narrowness and operability of the actual obstacle removal task environment.

To design the mockup, we obtained a 3D damage model [c] of the pedestal interior reconstructed from images obtained from an investigation of the interior of the PCV of the Fukushima Daiichi Nuclear Power Plant conducted in July 2017, and fabricated the mockup to simulate this as closely as possible. Of the inside structures of the mockup, the large structures were constructed from wood, while the small parts with complex shapes were fabricated with a 3D printer, and attached using magnets. Magnetic attachment gave seamless appearance to the mockup, while enabling the robot to detach the target part by simply grasping and lifting it with the hand tool and transition smoothly to transferring the grasped object.

### 4.4. Subject Operators

To verify the range of the operator's load, recognition capability, and judgment, we recruited four subjects with varying levels of experience as operators. Although all four subjects had experience operating a robot, three (subjects A, B, and D) had extensive experience operating manipulators, with careers of ten years or more, whereas one (subject C) had virtually no experience operating a manipulator. A discussion of the test results, including the level of experience of the participants, is presented later.

The subject operators were first given a preliminary training session to familiarize themselves with teleoperation of the robot using the control interface and took part in the test described below only after they were able to carry out teleoperation of the tasks involved.

### 4.5. Operational Procedure

In the mock test, the subject operators were asked to execute a series of tasks, which consisted of starting from the "start position" set in advance, "move into the opening," "grasp and cut the removal target," "carry the removal target out of the opening and deposit it in a unit can," and return to the "start position." Before doing so, the subject operators used the control interface to produce a set of trajectory data, as "advance planning," in another room, following which they moved to the robot test site to carry out the task using the actual teleoperation system. This arrangement was based on using the system in an actual debris removal task, where we expect advance planning and actual operations to be performed in response to an environment that changes with daily work progress.

The subject operators were asked to undertake three different tests, which are described later, where advance planning and actual operation alternated each other, in the manner of Test 1 ("advance planning" → "actual operation") → Test 2 ("advance planning" → "actual operation") → . . . .

Although the positions and poses of the control interface (environmental model in the simulator) and mockup (physical setup) were matched as closely as possible in the test, it was still necessary to sporadically perform registration because of the misalignment arising from the crawler movement of the robot. Thus, in addition to the mandatory

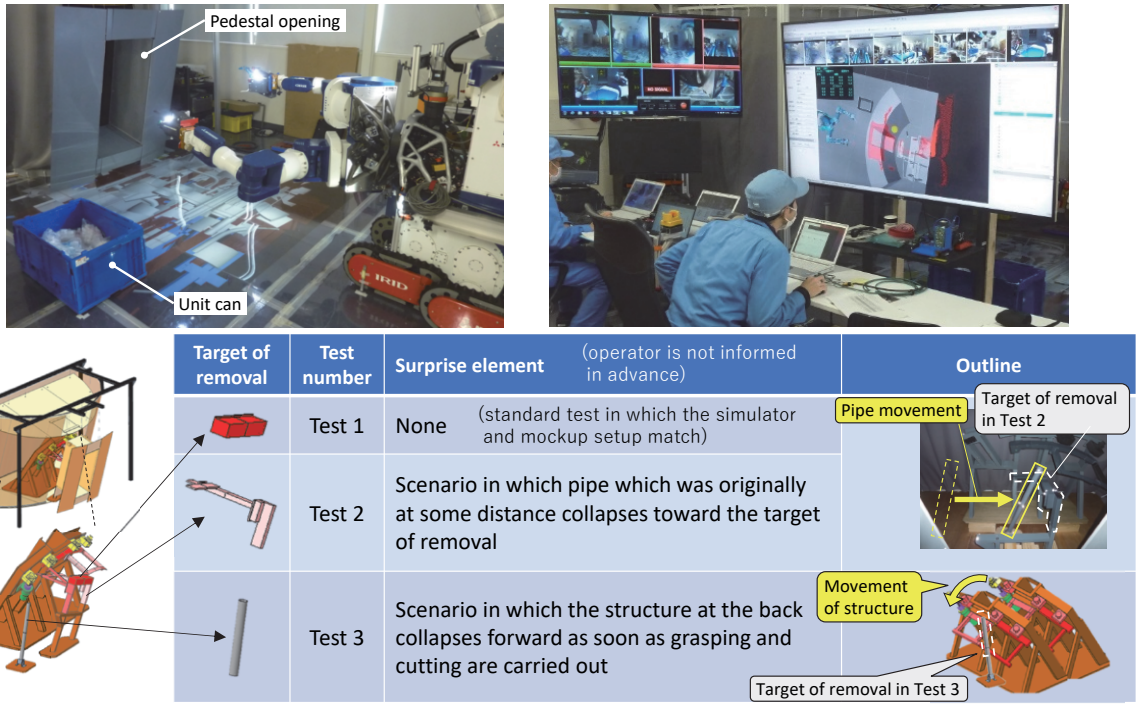



**Fig. 5.** Targets of removal and outlines of Tests 1–3.

initial registration at the start position, subsequent registrations were left to the judgment of each subject operator (common to Tests 1–3).

### 4.6. Test Content

We prepared three test assignments for the mock test, each based on a different theme, to investigate the subject operators' judgments and the time it took to carry out the assigned tasks. The targets for the removal and outlines of Tests 1–3 are shown in **Fig. 5**. The details of this process are described below.

#### 4.6.1. Test 1

Test 1 did not include the surprise element that was introduced in Tests 2 and 3, as described below, and was set up as a standard test where the environment underwent no change from that in advance planning. The target of removal was the rectangular parallelopiped part at the top of the gate-like structure immediately inside the opening. To remove this part, the operator sets the grasping point of the hand tool at the center and the cutting points of the cutter tool at the two ends.

#### 4.6.2. Test 2

In Test 2, there was only one cutting point for the cutting tool, so the removal task itself required fewer steps and was simpler than in Test 1. However, a pipe that was standing at some distance from the back in the simulator environment during advance planning was moved in the mockup for actual operation to a position where it obstructed the grasping of the target of removal. This is a surprise element based on a scenario in which the environment has changed from that in the advance planning stage, which may very likely occur onsite. The objective of this test was to determine whether the subjects noticed the environmental change and to examine their subsequent decisions.

#### 4.6.3. Test 3

Test 3 consisted of a task to remove a pipe standing inside the opening, where the removal task itself required fewer steps, and the target was also smaller than either Test 1 or 2, making this the easiest test. However, the large structure at the back falls forward when the pipe is grasped and cut during actual operation. This is a surprise element based on a scenario in which the environment changes during operation, which may occur onsite. To simulate this situation in the test, we interrupted the test and moved the structure while preventing the subjects from observing it. The objective of this test was to determine whether the subjects correctly recognized the environmental change and to examine their subsequent decisions.

## 5. Results of Mock Test

The results of the three tests are presented and discussed below.

### 5.1. Results of Test 1

In Test 1, all subject operators were able to successfully complete the set of tasks without any problems such

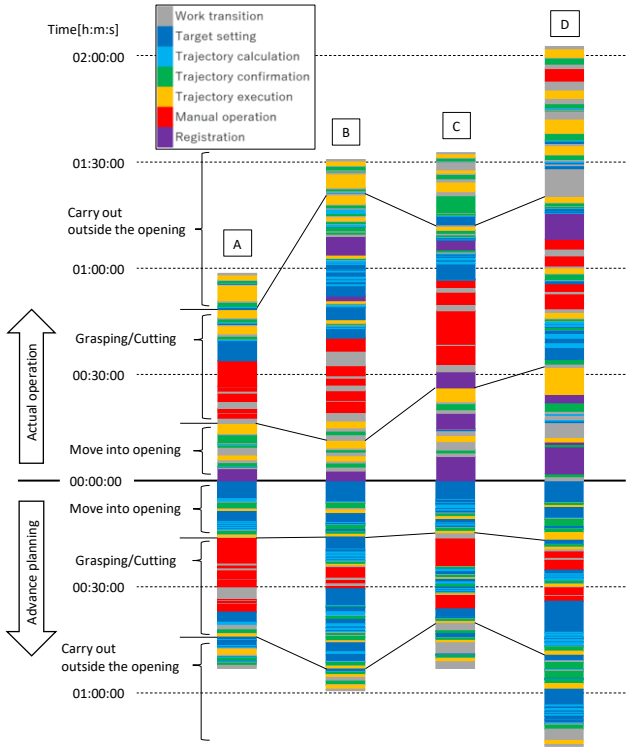



**Fig. 6.** Task times of four subjects in Test 1.

as collisions. The subject operators performed teleoperation while monitoring the actual situation using camera images, supplementing the simulator images on the control interface for the blind areas of the cameras.

The task times of the four subjects in Test 1 are shown in **Fig. 6**. The vertical axis represents the passage of time ([h]:[min]:[s]), where the tasks involved in advance planning are shown below the 00:00:00 line, and those undertaken during actual operation are stacked above. The colors indicate the various task items, allowing us to determine the time required to perform the various tasks and the order of execution. For reference, the operation was divided into the task phases of "move into opening," "grasping and cutting," and "carry out of the opening," and shown separated by the black line segments. If we first look at the advance planning stage, we see that all four subjects completed the task in approximately an hour. Looking next at the results of the actual operation, the fastest subject, A, took approximately an hour, subjects B and C about an hour and a half, and subject D close to two hours to complete the operation. One reason for the considerably longer time required by subject D was the high frequency of trajectory executions (yellow) and their relatively long times. This indicates that subject D proceeded cautiously while monitoring the camera images when moving the robot, dividing the trajectory into smaller steps to proceed slowly.

Looking at the contents of "move into opening," we find that none of the subjects proceeded directly from the start position to the target in a single step; instead, all of them divided the task into two or three steps (yellow). This is because performing registration using point-cloud data at intermittent intervals resolves the positional deviations caused by crawler movement and leads to the successful execution of subsequent tasks. In Test 1, subjects C and D judged that a misalignment had occurred after the first relocation and performed registration (purple). The operators intervened with correctional maneuvers whenever it became necessary to make unexpected trajectory corrections owing to model or registration errors, even when tracking the trajectory. Thus, the subjects undertook the tests while using the operational knowledge gained from the preliminary training session.

The transition of tasks during subject B's actual operation in Test 1 is shown in **Fig. 7**, along with the corresponding photos, while the proximity between the environment and the robot is shown in **Fig. 8**. Proximity shows the changes in the distance between the robot model and the closest point of the environmental model on the control interface, where the abscissa represents the passage of time and the ordinate is the proximity. The black solid curves represent the phases in which the real robot is being driven, and the gray broken curves represent the phases in which the operator is performing a task on the control interface. Looking at the changes in proximity from the starting position, it can be seen that, at "(i) move to front of opening," the robot moves from the starting position to the vicinity of the opening to drastically reduce the proximity. Subsequently, the proximity falls to 0.04 m at "(iii) approach target," where 0.04 m is the margin set in trajectory planning to which the robot is allowed to approach. The proximity falls to zero from "(iv) grasp" to "(vi) lift grasped object"; this is because the collision avoidance function was deactivated during grasping and cutting and the setting was changed to allow the robot to approach to within 0.04 m. From "(vi) lift grasped object," the grasped target is considered to be part of the robot, and the margin of 0.04 m set in trajectory planning is reactivated, and maintained up to "(xiii) store." A similar pattern was observed in Tests 1 to 3 for all four subjects, confirming that the collision avoidance function based on local trajectory planning worked as expected.

Tasks (vii) to (x) indicated by the red box in **Fig. 7** correspond to the operation shown in the figures to the right bordered in red, in which the target grasped by the hand tool is lifted, while the cutter tool end is turned to face upward so that the range sensor can acquire point-cloud data of the grasped condition. In this step, a model of the grasped object (a model of the obstacle separated at the cut section) is prepared in advance, and the operator must align the model of the grasped object with the point cloud data. Although subject B successfully turned the cutter tool to face upward in (vii), he was unable to obtain sufficient point-cloud data using the range sensor and repeated the tasks of setting the target, computing the trajectory, confirming the trajectory, and executing it, corresponding to tasks (viii), (ix), and (x) by trial and error. Consequently, the operations from (vii) to (x) took approximately one-third of the total time. This was because it was not possible to create a pose with which the grasped condition could be checked in advance

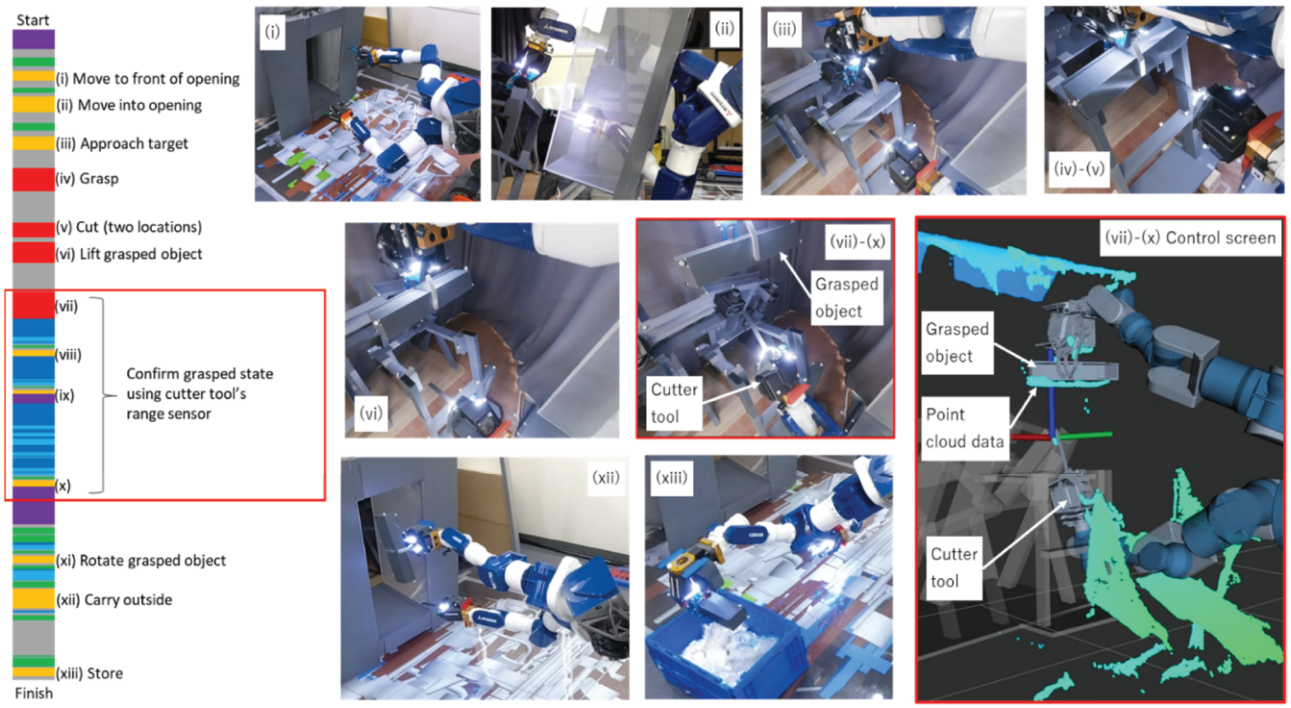



**Fig. 7.** Task sequence of subject B in Test 1.

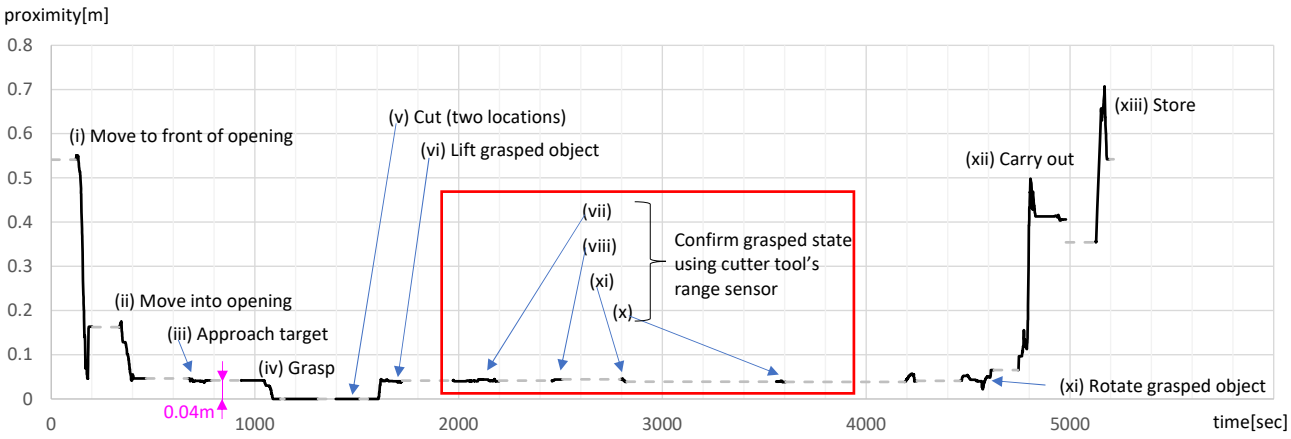



**Fig. 8.** Changes in proximity between environment and robot in Test 1 (subject B).

planning, as it was only possible to acquire the point-cloud data using the range sensor during actual operation, which made it necessary to carry it out on a trial-and-error basis. If it is possible in the future to incorporate a range sensor model into the robot in the simulation, it would make it possible to substantially reduce the task time inside the red box. If it is not possible to acquire sufficient point-cloud data, mechanisms such as setting a larger margin may be needed.

### 5.2. Results of Test 2

When Test 2 was conducted under the conditions described earlier, in which the environment changed from that in the advance planning stage, environmental recognition and judgment differed among the subjects. Subjects A, C, and D noticed that the pipe had moved before entering the opening. Whereas subjects A and C approached the removal target while avoiding the pipe, subject D judged that the pipe would obstruct their approach to the removal target and decided to remove it. Consequently, subject A caused the hand to collide with the removal target while avoiding the pipe. Subject C avoided the pipe and successfully retrieved the removal target, whereas subject D successfully retrieved the pipe. Among the four subjects, only subject B failed to notice that the pipe had moved and approached the removal target as planned in advance; consequently, the hand tool collided with the pipe. Whenever a collision occurred, the experimenter interrupted the recording process and notified the subject operator that a collision had occurred. The mockup was restored, after which the subject was asked to continue the task. The recording of the task time was also resumed.

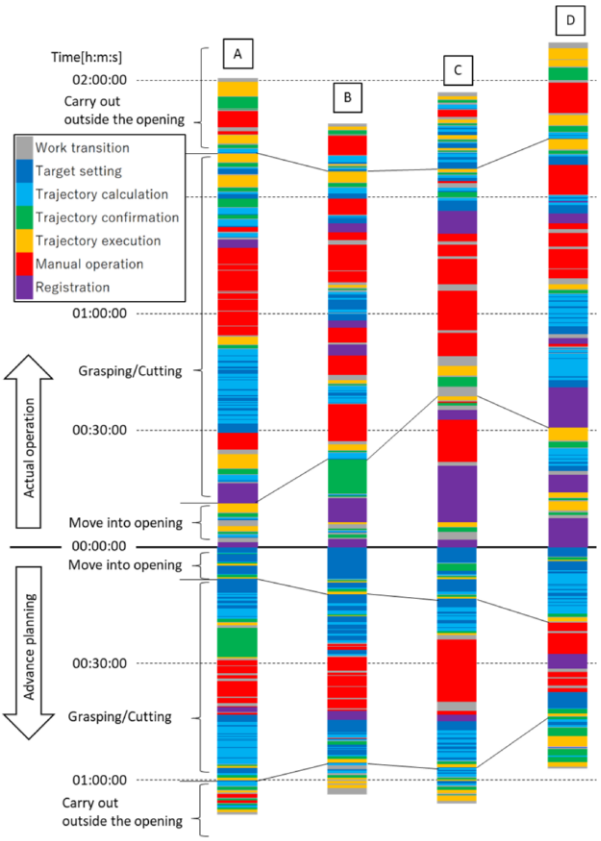



**Fig. 9.** Task times of four subjects in Test 2.

The task times of the four subjects in Test 2 are shown in **Fig. 9**. For advance planning, all four subjects completed the required tasks in approximately an hour. In the actual operation, subject A was the earliest to move into the opening and manually approach the target (red). However, he caused the hand tool to push the target down during the grasping task, from which it took some time to recover; thus, he took approximately the same time as the other subjects. As collisions occurred during the grasping task, the collision avoidance function was turned off.

Subject B proceeded without noticing that the pipe had been moved, which caused the hand tool to collide with the pipe. Because the position of the pipe model was not corrected, the local trajectory planner computed and issued a command (joint-axis value to avoid obstacles) based on the erroneous judgment that the path was clear, which resulted in a collision with the pipe. After the collision, subject B was notified of the situation by the experimenter and retreated out of the opening, where he reassessed the entire interior of the opening and registered the environmental model, including the pipe. He then decided to remove the pipe, which obstructed the approach to the target, and retrieved the pipe. To prevent such failures, it will be necessary for the system to continually acquire point-cloud data of the real environment and issue a warning when it differs substantially from the current model.

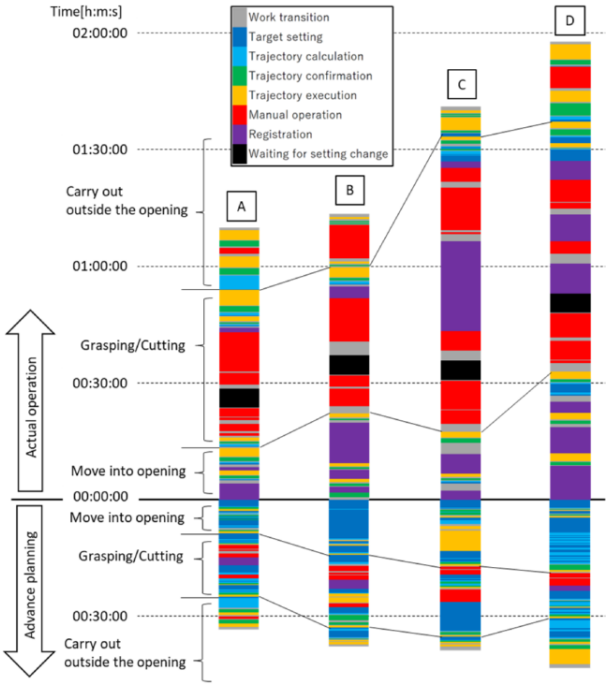



**Fig. 10.** Task times of four subjects in Test 3.

### 5.3. Results of Test 3

When Test 3 was performed under the conditions previously described, all subjects successfully completed the task. Compared to Test 2, in which it was possible to check the overall situation before grasping and cutting, in Test 3, the environment changed when the pipe was being grasped; thus, the robot could not move freely, while the environment must be recognized from information obtained from a narrow visual field. The subjects avoided the collapsing structure and retrieved the target by comparing camera images and point-cloud data to visualize the interior state of the mockup.

The task times of the four subjects in Test 3 are shown in **Fig. 10**. For advance planning, all four subjects completed the required tasks within 30–45 min. They were able to complete the tasks in a shorter time because fewer steps were involved than in Tests 1 and 2. In the actual operation, subjects A and B took approximately 1 h and 10 min, whereas subjects C and D took close to two hours, with a difference of approximately 30 min between the two groups. This difference arose because after the structure had been moved by the experimenter (black), subjects A and B decided to forgo registration based on point-cloud data, whereas subjects C and D carried out registration, which was evident from the longer time spent by the latter two on registration (purple).

Although subjects A and B completed the tasks in a shorter time, they were at a higher risk of colliding with the structure when carrying the target object out because they did not register the moved structure. In reactor decommissioning operations, the procedure performed by subjects C and D is preferable from the standpoint of safety, although it is more time-consuming.

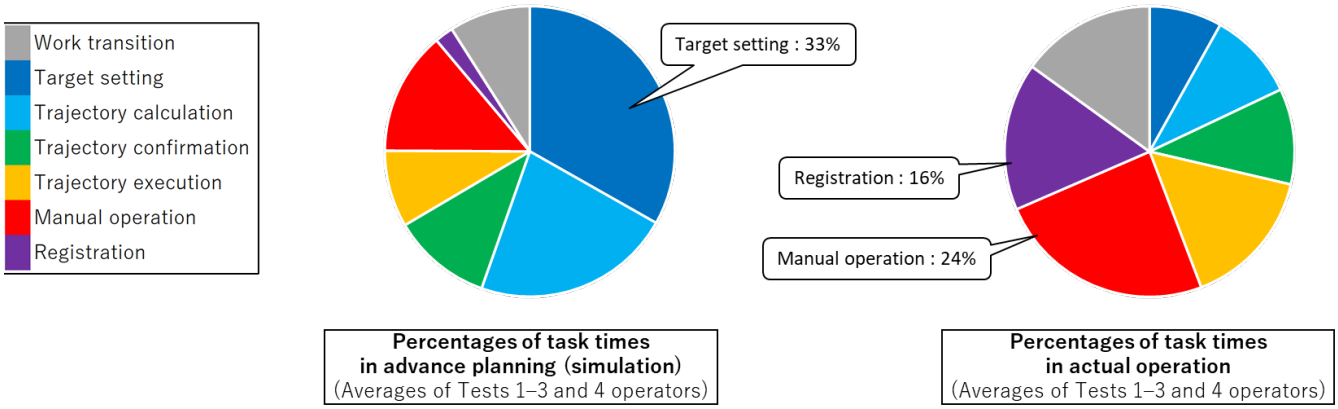



**Fig. 11.** Percentages of task times in Tests 1–3.

### 5.4. Discussion

In this section, we analyze and discuss the test results described above. The percentages of task times in Tests 1–3 are shown in **Fig. 11**. The two graphs present the average percentages of task times of the four subjects for Tests 1–3 in the advance planning and actual operation.

First, it can be seen that "target setting" took up one-third of the task time in advance planning. "Target setting" consists of using the mouse to move or rotate the end-effector target on the control interface. The subjects commented that it was "difficult to make fine adjustments." This is because mice are unsuitable for controlling miniscule amounts. If there is a function to automatically align the end-effector target to squarely face the removal target, this should contribute to reducing the task time of "target setting."

Looking at the actual operation, it can be seen that "manual operation" and "registration" each took up a high proportion of the task time. As in "target setting" above, "registration" requires mouse control to move or rotate the target, where it is difficult to make fine adjustments. If the function of automatically aligning the target to the point-cloud data is improved, the task time could be reduced. Moreover, as noted in Section 5.1, assessing the grasp state of the target object using the range sensor attached to the cutter tool required difficult operations. In actual operations, robots are deployed in highly radioactive environments, where minimizing the onsite operation time extends the robot's life. Thus, in long-term debris retrieval operations, it is important to reduce trial and error in actual operations by making preparations in the advance planning stage.

Although the four subjects in the present tests had varying manipulator operation experience, the operating times of the subjects with more experience were not necessarily shorter. This may be because the use of the teleoperation interface reduced the difference in experience, and the subjects' personalities, such as cautiousness, had a large effect on the operating times.

## 6. Conclusion

We developed a system to assist in the teleoperation of a redundant degree-of-freedom manipulator under narrow conditions with a restricted field of view, such as obstacle removal or debris retrieval. Specifically, we developed software to automate the collision avoidance of the manipulator and implemented a control interface, including a control screen. Four subject operators participated in a series of tests in which they used the control interface to teleoperate a dual-arm robot and remove obstacles from a mockup of the Fukushima Daiichi Nuclear Power Plant Unit 3. We confirmed that using the collision avoidance function of the system, a series of tasks could be conducted in a narrow environment, such as a pedestal opening. Although it was necessary in our previous study [12] to set a margin of approximately 50 mm for the robot model to avoid obstacles, we were able to reduce this margin to close to zero by adding a local trajectory planner and thus pass through narrow environments that were unpassable in the previous study. We confirmed that the system was able to deal with deviations (errors) between the environmental model and the actual site and sudden unexpected events using the registration function.

The issues identified in the mock test and their measures are summarized in **Table 1**. To follow up on this study, we plan to continue developing the teleoperation support system by incorporating these measures and verifying and further improving the system so that it can be deployed at the actual site.

#### Acknowledgments

This paper includes results of research and development which MHI, as a member of International Research Institute for Nuclear Decommissioning (IRID), has implemented by the subsidy of the projects of Ministry of Economy Trade and Industry (METI) on the Decommissioning and Contaminated Water Management. The authors are grateful to Dr. Tamio Arai (IRID) for his helpful discussions.

**Table 1.** Issues and countermeasures.

| Extracted issue | Phase in which issue was identified (test no.) | Measure |
|---|---|---|
| Hand tool collided with target in "manual operation." | Actual operation (2) | Monitor obstacles as a background process even when the collision avoidance function has been deactivated, such as during grasping, and issue a warning when collision is likely. |
| Failure to notice change of environment from advance planning (pipe moved). | Actual operation (2) | Have the system continually acquire point-cloud data of the actual environment, and issue a warning when it differs drastically from current model. |
| "Target setting" was time consuming. | Advance planning (1–3) | Select a control device suitable for minute control. Add function to automatically align end-effector target to a position squarely facing the target. |
| "Manual operation" and "registration" were time consuming. | Actual operation (1–3) | Select a control device suitable for minute control. Improve function to automatically align target with point-cloud data. |
| Confirmation of grasped state using the range sensor was time-consuming (part of "registration"). | Actual operation (1–3) | Prepare a pose to check the grasped state in advance planning by incorporate the range sensor model into the robot in the simulation. |

## References:


[1] Y. Yokokohji, "The Use of Robots to Respond to Nuclear Accidents: Applying the Lessons of the Past to the Fukushima Daiichi Nuclear Power Station," Annual Review of Control, Robotics, and Autonomous Systems, Vol.4, pp. 681-710, 2021. https://doi.org/10.1146/annurev-control-071420-100248

[2] P. Schmaus, D. Leidner, T. Krüger et al., "Preliminary Insights from the METERON SUPVIS Justin Space-Robotics Experiment," IEEE Robotics and Automation Letters, Vol.3, No.4, pp. 3836-3843, 2018. https://doi.org/10.1109/LRA.2018.2856906

[3] P. Schmaus, D. Leidner, T. Krüger et al., "Knowledge Driven Orbit-to-Ground Teleoperation of a Robot Coworker," IEEE Robotics and Automation Letters, Vol.5, No.1, pp. 143-150, 2020. https://doi.org/10.1109/LRA.2019.2948128

[4] G. R. Cho, G. Ki, M.-J. Lee et al., "Experimental Study on Tele-Manipulation Assistance Technique Using a Touch Screen for Underwater Cable Maintenance Tasks," J. of Marine Science and Engineering, Vol.9, No.5, Article No.483, 2021. https://doi.org/10.3390/jmse9050483

[5] M. D. Johnston and K. J. Rabe, "Integrated Planning for Telepresence with Time Delays," Proc. of the 2nd IEEE Int. Conf. on Space Mission Challenges for Information Technology (SMC-IT'06), 2006. https://doi.org/10.1109/SMC-IT.2006.39

[6] N. Alunni, C. Phillips-Grafflin, H. B. Suay et al., "Toward a User-guided Manipulation Framework for High-DOF Robots with Limited Communication," Proc. of the 2013 IEEE Conf. on Technologies for Practical Robot Applications (TePRA), 2013. https://doi.org/10.1109/TePRA.2013.6556356

[7] C. Phillips-Grafflin, H. B. Suay, J. Mainprice et al., "From Autonomy to Cooperative Traded Control of Humanoid Manipulation Tasks with Unreliable Communication," J. of Intelligent and Robotic Systems, Vol.82, pp. 341-361, 2016. https://doi.org/10.1007/s10846-015-0256-5

[8] M. Bandala, C. West, S. Monk et al., "Vision-Based Assisted Tele-Operation of a Dual-Arm Hydraulically Actuated Robot for Pipe Cutting and Grasping in Nuclear Environments," Robotics, Vol.8, No.2, Article No.42, 2019. https://doi.org/10.3390/robotics8020042

[9] S. D. Monk, A. Grievson, M. Bandala et al., "Implementation and Evaluation of a Semi-Autonomous Hydraulic Dual Manipulator for Cutting Pipework in Radiologically Active Environments," Robotics, Vol.10, No.2, Article No.62, 2021. https://doi.org/10.3390/robotics10020062

[10] D. Kent, C. Saldanha, and S. Chernova, "A Comparison of Remote Robot Teleoperation Interfaces for General Object Manipulation," Proc. of the 2017 12th ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI), pp. 371-379, 2017.

[11] M. Rubagotti, T. Taunyazov, B. Omarali, and A. Shintemirov, "Semi-Autonomous Robot Teleoperation with Obstacle Avoidance via Model Predictive Control," IEEE Robotics and Automation Letters, Vol.4, Issue 3, pp. 2746-2753, 2019. https://doi.org/10.1109/LRA.2019.2917707

[12] N. Mizuno, Y. Tazaki, T. Hashimoto, and Y. Yokokohji, "A Comparative Study of Manipulator Teleoperation Methods for Debris Retrieval Phase in Nuclear Power Plant Decommissioning," Advanced Robotics, Vol.37, No.9, pp. 541-559, 2023. https://doi.org/10.1080/01691864.2023.2169588

[13] M. D. Killpack, A. Kapusta, and C. C. Kemp, "Model Predictive Control for Fast Reaching in Clutter," Autonomous Robots, Vol.40, pp. 537-560, 2016. https://doi.org/10.1007/s10514-015-9492-6

[14] Y. Tazaki and T. Suzuki, "Constraint-Based Prioritized Trajectory Planning for Multi-Body Systems," IEEE Trans. on Robotics, Vol.30, No.5, pp. 1227-1234, 2014. https://doi.org/10.1109/TRO.2014.2320794


## Supporting Online Materials:


[a] International Research Institute of Technology (IRID), "Development of technology for further increasing the scale of retrieval of fuel debris and internal structures." https://irid.or.jp/wp-content/uploads/2021/01/2019002entoridasikibonosaranarukakudaif.pdf [Accessed December 15, 2023]

[b] "Development of decontamination equipment for the upper floors of the Fukushima Daiichi Nuclear Power Plant" (in Japanese). https://irid.or.jp/wp-content/uploads/2015/12/20151216_01.pdf [Accessed December 15, 2023]

[c] "Fukushima Daiichi Nuclear Power Plant Unit3, 3D reconstruction from investigation video inside primary containment vessel" (in Japanese). https://photo.tepco.co.jp/date/2018/201804-j/180426-02j.html [Accessed December 15, 2023]



**Name:**
Tatsuya Hashimoto

**Affiliation:**
Senior Deputy Manager, Equipment Designing Section, Nuclear Plant Component Designing Department, Nuclear Energy Systems, Mitsubishi Heavy Industries, Ltd.

**Address:**
1-1-1 Wadasaki-cho, Hyogo-ku, Kobe, Hyogo 652-8585, Japan


**Brief Biographical History:**
2006-2009 Hitachi, Ltd.
2009- Mitsubishi Heavy Industries, Ltd.

**Name:**
Yuichi Tazaki

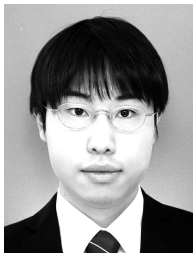
**Affiliation:**
Associate Professor, Department of Mechanical Engineering, Graduate School of Engineering, Kobe University

**Address:**
1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan


**Brief Biographical History:**
2007-2009 Research Fellow, Japan Society for the Promotion of Science
2008 Guest Scientist, Honda Research Institute Europe
2009-2016 Assistant Professor, Nagoya University
2016- Associate Professor, Kobe University

**Main Works:**
• Autonomous vehicles, mobile robots, and humanoid robots.

**Membership in Academic Societies:**
• Institute of Electrical and Electronics Engineers (IEEE)
• Institute of Systems, Control and Information Engineers (ISCIE)
• The Robotics Society of Japan (RSJ)

**Name:**
Fumiya Matsuda

**Affiliation:**
Department of Mechanical Engineering, Kobe University

**Address:**
1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan

**Name:**
Katsumasa Kitajima

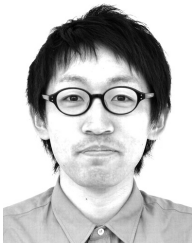
**Affiliation:**
Mitsubishi Heavy Industries, Ltd.

**Address:**
1-1-1 Wadasaki-cho, Hyogo-ku, Kobe, Hyogo 652-8585, Japan


**Brief Biographical History:**
2018- Mitsubishi Heavy Industries, Ltd.

**Main Works:**
• "Robust Localization Using Sensor Fusion for Industrial Mobile Robot," SICE SI2020, pp. 1128-1133, 2020.
• "Visual Localization Based on Dynamically Controlled Keyframe Intervals," IPSJ SIG Technical Report, Vol.2020-CVIM-221, No.45, 2020 (in Japanese).
• "Robust and Accurate Self-Localization Method Under Varying Lighting Conditions," Mitsubishi Heavy Industries Technical Review, Vol.58, No.1, 2021.

**Membership in Academic Societies:**
• The Robotics Society of Japan (RSJ)

**Name:**
Hikaru Nagano

**ORCID:**
0000-0001-5230-6288

**Affiliation:**
Assistant Professor, Graduate School of Engineering, Kobe University

**Address:**
1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan


**Brief Biographical History:**
2015 Received Ph.D. degree in Engineering from Nagoya University
2015-2018 Assistant Professor, International Research Institute of Disaster Science, Tohoku University
2018 Assistant Professor, Graduate School of Information Sciences, Tohoku University
2018- Assistant Professor, Graduate School of Engineering, Kobe University

**Main Works:**
• Haptic interface, human-machine interface, and virtual reality.

**Membership in Academic Societies:**
• The Robotics Society of Japan (RSJ)
• The Japan Society of Mechanical Engineers (JSME)
• The Virtual Reality Society of Japan (VRSJ)
• Institute of Electrical and Electronics Engineers (IEEE)


**Name:**
Yasuyoshi Yokokohji

**ORCID:**
0000-0001-8869-7102

**Affiliation:**
Professor, Department of Mechanical Engineering, Graduate School of Engineering, Kobe University

**Address:**
1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan


**Brief Biographical History:**
1988-1992 Research Associate, Kyoto University
1992-2009 Associate Professor, Kyoto University
2009- Professor, Kobe University.

**Main Works:**
• "Assembly Challenge: A Robot Competition of the Industrial Robotics Category, World Robot Summit – Summary of the Pre-Competition in 2018," Advanced Robotics, Vol.33, No.17, pp. 876-899, 2019.
• "The Use of Robots to Respond to Nuclear Accidents: Applying the Lessons of the Past to the Fukushima Daiichi Nuclear Power Station," Annual Review of Control, Robotics, and Autonomous Systems, Vol.4, pp. 681-710, 2021.
• "Jigless Assembly of an Industrial Product by a Universal Robotic Hand Mounted on an Industrial Robot," Robotica, Vol.41, Issue 8, pp. 2464-2488, 2023.

**Membership in Academic Societies:**
• The Japan Society of Mechanical Engineers (JSME)
• The Robotics Society of Japan (RSJ)
• The Society of Instrument and Control Engineers (SICE)
• Institute of Systems, Control and Information Engineers (ISCIE)
• The Virtual Reality Society of Japan (VRSJ)
• Institute of Electrical and Electronics Engineers (IEEE), Senior Member