




Digital Twin and Deep Reinforcement Learning-Driven Robotic Automation System for Confined Workspaces: A Nozzle Dam Replacement Case Study in Nuclear Power Plants

Su-Young Park¹ · Cheonghwa Lee² · Suhwan Jeong¹ · Junghyuk Lee¹ · Dohyeon Kim¹ · Youhyun Jang³ · Woojin Seol³ · Hyungjung Kim⁴ · Sung-Hoon Ahn^{1,5} 

Received: 2 August 2023 / Revised: 16 December 2023 / Accepted: 22 December 2023 / Published online: 18 March 2024

© The Author(s) 2024

Abstract

Robotic automation has emerged as a leading solution for replacing human workers in dirty, dangerous, and demanding industries to ensure the safety of human workers. However, practical implementation of this technology remains limited, requiring substantial effort and costs. This study addresses the challenges specific to nuclear power plants, characterized by hazardous environments and physically demanding tasks such as nozzle dam replacement in confined workspaces. We propose a digital twin and deep-reinforcement-learning-driven robotic automation system with an autonomous mobile manipulator. The study follows a four-step process. First, we establish a simplified testbed for a nozzle dam replacement task and implement a high-fidelity digital twin model of the real-world testbed. Second, we employ a hybrid visual perception system that combines deep object pose estimation and an iterative closest point algorithm to enhance the accuracy of the six-dimensional pose estimation. Third, we use a deep-reinforcement-learning method, particularly the proximal policy optimization algorithm with inverse reachability map, and a centroidal waypoint strategy, to improve the controllability of an autonomous mobile manipulator. Finally, we conduct pre-performed simulations of the nozzle dam replacement in the digital twin and evaluate the system on a robot in the real-world testbed. The nozzle dam replacement with precise object pose estimation, navigation, target object grasping, and collision-free motion generation was successful. The robotic automation system achieved a 92.0% success rate in the digital twin. Our proposed method can improve the efficiency and reliability of robotic automation systems for extreme workspaces and other perilous environments.

Keywords Robotic automation · Autonomous mobile manipulator · Deep reinforcement learning · Digital twin · Nuclear power plants · Confined workspace

1 Introduction

Digital twin (DT) and deep reinforcement learning (DRL) have emerged as pivotal technologies for advanced robotic automation systems (RASs) in extreme environments [1–3]. For instance, nuclear power plants need sophisticated RAS for tasks such as maintenance, inspection, and repair, which are often executed under harsh conditions with elevated radiation exposure and geometrically confined and complex structures [4]. Tasks in nuclear power plants are hazardous and difficult, with potential radiation exposure that could lead to serious health problems or even death for human workers. Moreover, the tasks are technically challenging and require specialized skills, experience, and precise tools. The challenge lies in ensuring safe, accurate, and efficient completion of these tasks.

✉ Hyungjung Kim
hyungjungkim@konkuk.ac.kr

✉ Sung-Hoon Ahn
ahnsh@snu.ac.kr

¹ Department of Mechanical Engineering, Seoul National University, Seoul 08826, Republic of Korea

² Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, Republic of Korea

³ Digital Solution Section, Korea Hydro and Nuclear Power Co., Ltd, Gyeongju-si 38219, Republic of Korea

⁴ Department of Industrial Engineering, Konkuk University, Seoul 05029, Republic of Korea

⁵ Institute of Advanced Machines and Design, Seoul National University, Seoul 08826, Republic of Korea

With nuclear power technology now classified as green technology in the European Union Taxonomy, the importance of deploying RAS in nuclear power plants has become more pronounced than ever. This shift towards recognizing nuclear energy as a sustainable energy source underscores a crucial precondition, which is the assurance of safety. In this context, the introduction of RAS in nuclear power plants offers various advantages. First, RAS minimizes the risk of workplace accidents and harmful radiation exposure for human workers [5], which mitigates environmental risks associated with nuclear power plants. Additionally, robots can operate continuously without breaks, resulting in increased productivity and efficiency in power plant operations. This contributes to a reduction in carbon emissions and other environmental impacts related to electricity generation. Furthermore, RAS decreases the reliance on human labor, which can lead to improved social outcomes, such as increased employment opportunities in other sectors, enhanced working conditions, and reduced income inequality [6]. Considering these aspects, the introduction of RAS in nuclear power plants can be viewed as an embodiment of advanced technological integration, which, from a long-term perspective, qualifies as a form of green technology that protects the environment [7–10].

DT technology, in conjunction with RAS, is crucial in extreme environments such as those of nuclear power plants, manufacturing facilities, and other hazardous workspaces [11, 12]. DT serves as a virtual representation of a physical system or process, enabling engineers and operators to simulate and test various scenarios without endangering human lives or incurring expensive equipment damage. DT technology, when integrated with RAS, allows for optimized performance, proactive issue resolution, and real-time control, enhancing safety and environmental sustainability in extreme industrial environments.

DRL has emerged as a promising approach for resolving intricate decision-making problems in the robotics industry. It enables robots to learn new skills and adapt to dynamic environments, resulting in advancements in automation, productivity, and safety [13, 14]. Applying DRL in robotics enables robots to execute complex tasks with enhanced precision and efficiency, reducing human error and increasing overall productivity. It holds significant potential across various fields such as healthcare, manufacturing, exploration, and disaster response, ultimately contributing to an improved quality of life, increased productivity, and a safer society [15].

Numerous studies are being conducted to enhance the reliability and efficiency of autonomous mobile manipulators (AMMs) [16]. To mitigate frequent hardware damage during robot development, algorithms and systems related to self-collision avoidance between the autonomous mobile robot (AMR) and the robot manipulator have been investigated

[17, 18]. Furthermore, the development of algorithms and systems for collision avoidance in dynamic and uncertain environments is underway, enabling AMMs to operate over an extensive range [19]. Deep-learning algorithms have recently been integrated into the complex control systems of AMMs, enabling them to perform more diverse and intricate tasks, like peg-in-hole and spraying [20].

As the application of AMMs expands, efforts have been made to employ them as maintenance robots in facilities with confined structures, such as nuclear power plants [21]. Typical tasks required in large and structurally confined facilities involve avoiding collisions between structures and robot hardware, necessitating the use of end-effectors capable of stably performing specific tasks. Continuum robots are flexible to work in complex and confined structures, but they are limited by the inevitable contact with structures [22]. Unmanned aerial vehicles (UAVs), such as drones, can maneuver without contact in complex structural environments but they cannot physically control targets [23, 24].

AMMs can significantly address these challenges; however, there are still some limitations when performing tasks in confined and complex structures. Although numerous studies focused on autonomous driving based on simultaneous localization and mapping and navigation from a mobility perspective [25], further research is needed on manipulators that can organically respond to the dynamic three-dimensional (3D) environment, particularly for executing delicate tasks in intricate spaces. Most vision sensor-based perception algorithms rely on pre-defined augmented reality tags for accurate calibration between the target and the robot location, or when the target is initially present within the field of view (FOV) of the camera [26]. However, these conditions are inefficient in actual workspaces owing to their large and complex structures. Although preliminary studies have been conducted on trajectory generation in confined spaces, they often do not consider the geometry of robot manipulators with intricate shapes, which are mostly limited to simulation environments [27]. To overcome these limitations, a stable, robust, and integrated DT platform is needed. An integrated DT platform should enable the development and evaluation of state-of-the-art (SOTA) perception, decision, and control algorithms, ensuring secure operation of robots without collisions in confined workspaces.

We propose a DT and DRL (DT-DRL) approach for a RAS capable of robust and flexible operations within a confined chamber including a narrow passage, targeting steam generator compartments in nuclear power plants. The hybrid visual perception system precisely estimates the target pose using deep object pose estimation (DOPE) and iterative closest points (ICP). After finding the optimal base pose by using an inverse reachability map (IRM), a DRL algorithm, along with a novel and intuitive reward shaping strategy, trains the robot manipulator of the

AMM for collision-free trajectory generation in confined workspaces. This DT-DRL approach is implemented in real-world robotic automation systems using the robot operating system (ROS) and has proven successful. The contributions of the study include the following:

- (1) **End-to-end DT for robotic automation in confined and hazardous workspaces:** The study develops a high-fidelity DT capable of virtualizing actual environments including geometrically confined structures. The DT also enables the development and evaluation of SOTA RAS without the need for any third-party platforms. Notably, the DT delivers hyper-realistic rendering for visual perception systems and also offers training acceleration, facilitating efficient trajectory generation for complex robots. This end-to-end DT can drastically reduce the development time and cost of RAS for demanding perilous tasks, where conducting actual pre-evaluation poses challenges.
- (2) **Hybrid vision algorithm-driven enhancement of 6D pose estimation accuracy:** The study introduces a hybrid visual perception system that combines deep-learning-based estimation algorithm DOPE with a mathematical registration algorithm ICP, focusing on improving the accuracy of 6D pose estimation. The fusion approach significantly surpasses the performance of single-algorithm approaches, leading to an increased success rate of tasks. This novel fusion approach can estimate the precise pose of a target object in various environments that are not defined in advance, improving perception performance, which is essential for RAS.
- (3) **DRL-driven collision-free trajectory generation in a confined chamber:** The study presents a novel and intuitive reward shaping of DRL algorithm for collision-free trajectory generation in a confined chamber. A centroidal waypoint with a high positive reward guides the robot manipulator to a safe path. The method has a higher task success rate than conventional or distance-based DRL motion-planning algorithms in confined chambers, including narrow passages. This method can be universally used in various confined workspaces where the joints of the robot manipulator are constrained.
- (4) **Nozzle dam replacement task in nuclear power plants using the DT-DRL for RAS:** This study demonstrates the feasibility of DT-DRL for RAS through a nozzle dam replacement task in nuclear power plants. RAS fully developed in the DT environment is used to automate the nozzle dam replacement task in nuclear power plants, and empirically evaluate it even in the real-world system.

The remainder of this paper is organized as follows. Section 2 explores the research focusing on robots in confined workspaces and DRL applications for AMMs. Section 3 provides the background of the systems used in this study. Section 4 introduces the proposed DT-DRL system for RAS, with specific emphasis on the development of a hybrid perception algorithm employing DOPE and ICP, DRL-based algorithms using IRM and a centroidal waypoint, along with ROS-based control systems. Section 5 provides both qualitative and quantitative experimental evaluations of the DT and real-world environments. Section 6 presents conclusions and future research prospects.

2 Related Works

2.1 Digital Twin with Robotic Perception, Decision, and Control System

The DT concept, originally articulated by Grieves [28], entails creating a digital mirror of a physical entity to facilitate its study and comprehension. Over time, the complexity and scope of this notion have evolved significantly, as numerous researchers explored its applications. A pivotal study by Tao et al. [29] underscored the essential function of DT in Industry 4.0, amalgamating technologies such as simulation, the Internet of Things, and Big Data to enhance product quality while minimizing development expenses. The application of DTs in the realm of robotics, specifically in perception, decision, and control, has attracted significant attention from the research community. For instance, Niki et al. [30] proposed a DT-based method for enhancing robotic perception. Their approach used a high-fidelity DT to mimic the environment, providing a training platform for machine-learning algorithms and significantly improving the ability of the robot to understand its surroundings. In terms of decision making, a study by Lee et al. [31] introduced a DT-driven approach for complex task planning and execution in robotic construction. By creating a virtual replica of the physical world, their system tested various decision-making strategies in a risk-free environment, thereby enhancing the efficiency and robustness of the robotic system. Regarding control, Yang et al. [32] made a significant contribution using DTs for predictive control in robotic systems. Their approach leveraged the real-time data from the DT to predict the future state of the robot and adjust its control strategy accordingly. This work demonstrated the potential of DTs to significantly improve the operational efficiency and adaptability of robotic systems.

Despite these substantial strides in research and application, the current literature reveals the lack of comprehensive studies that delve into the intricate intersection of DT technology, visual perception systems, and

GPU-based training acceleration in robotics. Recognizing this gap, this study aimed to meticulously investigate these themes, and contribute to a more holistic understanding of this rapidly evolving field.

2.2 Robots in Confined Workspaces

Extensive studies have been conducted on different types of robots, such as AMRs, UAVs, continuum robots, and robot manipulators, for their potential to operate in obstructed workspaces [33]. Each type of robot has unique advantages and limitations, and various methods have been investigated to overcome these limitations. Compact and lightweight AMRs were proposed for inspection and maintenance tasks in cluttered workspaces [16, 17]. These AMRs could navigate through narrow passages and confined spaces; however, their mobility was limited, which posed a disadvantage in complex environments with obstacles.

Another approach used lightweight and easy-to-operate UAVs for inspection and maintenance tasks in obstructed workspaces, such as inspecting large structures [34]. However, UAVs have limitations in terms of precise operation, making handling targets in narrow or obstructed environments challenging. Furthermore, UAVs cannot provide tactile feedback, which may not be suitable for tasks requiring contact with surfaces or objects.

Many continuum robots have been proposed for use in narrow workspaces. For instance, a snake robot was developed for inspecting and repairing pipes [35]. The design of the snake robot enabled it to move around obstacles and through confined spaces, making it suitable for use in restricted environments. However, controlling the movement of the snake robot is challenging owing to its unstable movement, and it may not be ideal for carrying heavy payloads or performing tasks that require precise manipulation of objects.

The emerging field of AMMs combines the mobility of AMRs with the dexterity of robot manipulators. Their mobility enables them to perform tasks in hard-to-reach locations and various translations and orientations, which can increase the efficiency and safety of industrial processes by reducing the need for human intervention in hazardous areas [36]. However, the size and weight of AMMs can limit their mobility in certain environments, and advanced navigation and control systems are required to ensure safe and accurate operation, increasing system complexity. These challenges can be addressed by leveraging new technologies such as DT and DRL, which have the potential to make AMMs the most practical robots in industrial fields.

2.3 Deep Reinforcement Learning for Autonomous Mobile Manipulators

Considerable research has been conducted on the efficacy of DRL in training AMMs to master collision avoidance. An early study employed deep deterministic policy gradient (DDPG) to guide an AMM around obstacles in its environment [20]. The AMM leveraged sensor data to perceive its surroundings and predict potential collisions in real time, effectively avoiding obstacles while efficiently pursuing its intended destination. The application of DRL in complex environments has also been explored. Xia et al. [37] employed soft actor-critic (SAC), a model-free algorithm, in AMMs operating in cluttered environments. The authors demonstrated how SAC, integrated with a meticulously designed reward function, could facilitate effective navigation in spaces populated with static and dynamic obstacles. DRL has shown promise in assisting AMMs in executing complex tasks. Ander et al. [38] implemented a twin delayed DDPG (TD3) algorithm, a variant of DDPG, to instruct an AMM in precise object manipulation tasks, effectively circumventing the inherent difficulties linked to conventional control algorithms. In multi-robot scenarios, DRL has been of considerable interest as well. Chen et al. [39] used a multi-agent DDPG (MADDPG) to optimize the coordination between a group of AMMs, resulting in improved performance in task completion time and collision avoidance. Additionally, Sun et al. [40] showcased the versatility of DRL by employing it to develop an adaptive trajectory generation system for AMMs using an asynchronous advantage actor-critic (A3C), demonstrating its potential to adapt to changes in the dynamics of the robot or its environment.

Despite the advantages of using machine-learning-based approaches for AMMs, there are still limitations. One such challenge is the requirement for extensive training data to learn collision-free C-spaces or collision avoidance policies, leading to long training times. Previous DRL-based algorithms were limited to implementing complex RAS or evaluating simulation-to-reality (Sim2Real) transferability. As a result, a stable virtual environment capable of evaluating any RAS and an algorithm that can automate new tasks rapidly and flexibly are necessary.

3 Backgrounds

The RAS constructed in this study comprises perception, decision, and control systems. For a better understanding, this section aims to provide an explanation of the main algorithms.

3.1 Perception algorithms

3.1.1 Deep Object Pose Estimation (DOPE)

The proposed approach uses DOPE algorithm, which combines visual geometry group 19 deep-learning models, belief maps, and vector fields to estimate the 6D pose of objects in images or video frames. In the initial step, a convolutional neural network (CNN)-based detector generates belief maps to identify locations of objects in an image. The subsequent step involves another CNN model using these belief maps to regress the 6D pose of the object, represented as $x, y, z, \rho, \phi, \psi$ (Eq. 1).

$$DOPE(I_{RGB}) = 6D\ pose(x, y, z, \rho, \phi, \psi) \tag{1}$$

where I_{RGB} denotes the RGB image, and x, y, z and ρ, ϕ, ψ represent the 3D position coordinates of the object and the Euler angles for rotation, respectively. The DOPE algorithm demonstrates efficiency and robustness in challenging scenarios, including cluttered backgrounds and occlusions [41], and finds use in applications such as robot manipulators, augmented reality, and autonomous systems.

However, its reliance on well-defined environments limits its flexibility and scalability in dynamic or unknown settings. Performance may decline due to limitations in training data, occlusions, and lighting changes. In scenarios where the environment frequently changes or is unknown, the lack of adaptability becomes a significant disadvantage. Additionally, as an algorithm highly dependent on the pixel resolution of RGB cameras, it also has the drawback of significantly reduced accuracy for objects at a distance. To overcome these challenges, this study proposes an innovative solution based on deep neural networks (DNN) and robust mathematical methods [42, 43].

3.1.2 Iterative Closest Point (ICP)

ICP algorithm is widely used for registering and aligning two or more point cloud data (PCD) to estimate their relative transformation (Eq. 2).

$$\operatorname{argmin}_{R, T} \sum_{i=1}^N \left\| P_i - (RP'_i + T) \right\|^2 \tag{2}$$

where the variables R and T represent the rotation matrix and translation vector, respectively, optimized to find the transformation between two point clouds. P_i denotes a point in the source point cloud, and P'_i represents its corresponding point in the target cloud.

ICP operates by iteratively identifying and minimizing the distance between corresponding points until the error is below

a threshold τ , achieving accurate PCD alignment. ICP is versatile, handling rigid and non-rigid alignments, and manages noise, outliers, and missing data effectively. While computationally efficient for various applications, the accuracy of ICP depends on the initial transformation estimate. If the initial estimate is incorrect, the resulting estimate can be significantly erroneous, particularly in point clouds with significant shape differences or occlusions [44].

3.2 Decision and Control Algorithms

3.2.1 Inverse Reachability Map (IRM)

IRM algorithm aims to identify the optimal mobile base pose for the AMM, thereby narrowing the solution space for DRL in achieving collision-free trajectories [45]. IRM computes the optimal base pose for a robot, ensuring the tool center point (TCP) reaches a specified target point (Eq. 3).

$$RM_{AMM_{base}}^{TCP} = (x, y, z, \rho, \phi, \psi) \tag{3}$$

where $RM_{AMM_{base}}^{TCP}$ is a reachability map containing the target pose of the robot manipulator TCP relative to the base pose of the AMM in Cartesian coordinates. The inverse transformation of this map, $(RM_{AMM_{base}}^{TCP})^{-1}$, computes the inverse reachability map $IRM_{AMM_{base}}^{TCP}$, containing multiple base poses of the AMM (Eq. 4).

$$(RM_{AMM_{base}}^{TCP})^{-1} = IRM_{AMM_{base}}^{TCP} \tag{4}$$

The search space B , including each grid cell b_i , is defined before IRM computation (Eq. 5).

$$B = [b_0, \dots, b_i, \dots, b_T] \tag{5}$$

where b_i contains the set of (x, y, ψ) the base pose that the robot manipulator can reach the target point p_{target} . The feasible base poses of the AMM are stored in each grid cell b_i including the translations x, y , and the orientation yaw ψ . These feasible base poses are computed using $IRM_{AMM_{base}}^{TCP}$ (Eq. 6).

$$b_{ij} = \left\{ p_{target} * (IRM_{AMM_{base}}^{TCP})_j \right\}_i \tag{6}$$

where b_{ij} means the set of $b_i(x, y, \psi)$.

To determine the optimal base pose, the algorithm identifies the maximum norm of feasible base poses $\max(\| b \|)$ and normalizes it as d_i , resulting in a feasibility score (Eq. 7).

$$d_i = \begin{cases} \frac{\|b_i\|}{\max(\|b\|)}, & \text{if } \|b_i\| \geq 1 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

The normalized values are stored in an array D (Eq. 8).

$$D = [d_0, \dots, d_i, \dots, d_T] \tag{8}$$

This process assists in selecting the optimal base pose for the AMM. However, while IRM is effective in determining the best base pose, it does not directly compute collision-free trajectories as it lacks integrated collision avoidance strategies during the planning phase.

DRL agents aim to maximize accumulated rewards over time, using DNNs to represent policies that map states to actions a . The agent iteratively improves its decision-making by interacting with the environment, observing the state s , receiving rewards r , and updating its DNN parameters θ and policy π (Fig. 1).

PPO, a robust DRL algorithm, optimizes the policy using a surrogate loss function, advantage function, and an clipping ratio ϵ for stable and efficient updates [46]. The surrogate loss function L enables effective balance between exploration and exploitation (Eq. 9).

$$L(s, a, \theta_k, \theta) = \hat{\mathbb{E}} \left[\min \left(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s, a), \text{clip} \left(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_k}(a|s)}, 1 - \epsilon, 1 + \epsilon \right) A^{\pi_{\theta_k}}(s, a) \right) \right] \tag{9}$$

3.2.2 Deep Reinforcement Learning (DRL) and Proximal Policy Optimization (PPO)

DRL combines deep-learning with Markov decision processes (MDP) principles, providing a framework for agents to learn optimal decision-making policies through environmental interaction and feedback in the form of rewards [13].

where $\hat{\mathbb{E}}[\cdot]$ represents taking the expected value, which means averaging the expression inside the function over a specific probability distribution, $\pi_t(\cdot|s)$ represents the probability of selecting a given action state under the policy, t denotes the step of the policy, $\pi_{t+1}(\cdot|s)$ denotes the updated policy, θ is a set of parameters that govern the action selection based on the current state s , and θ_k is a set of fixed parameters that determine the action selection in the current policy iteration and remains constant during the policy update process. $\text{clip}(\cdot)$ is used to restrict the policy update to

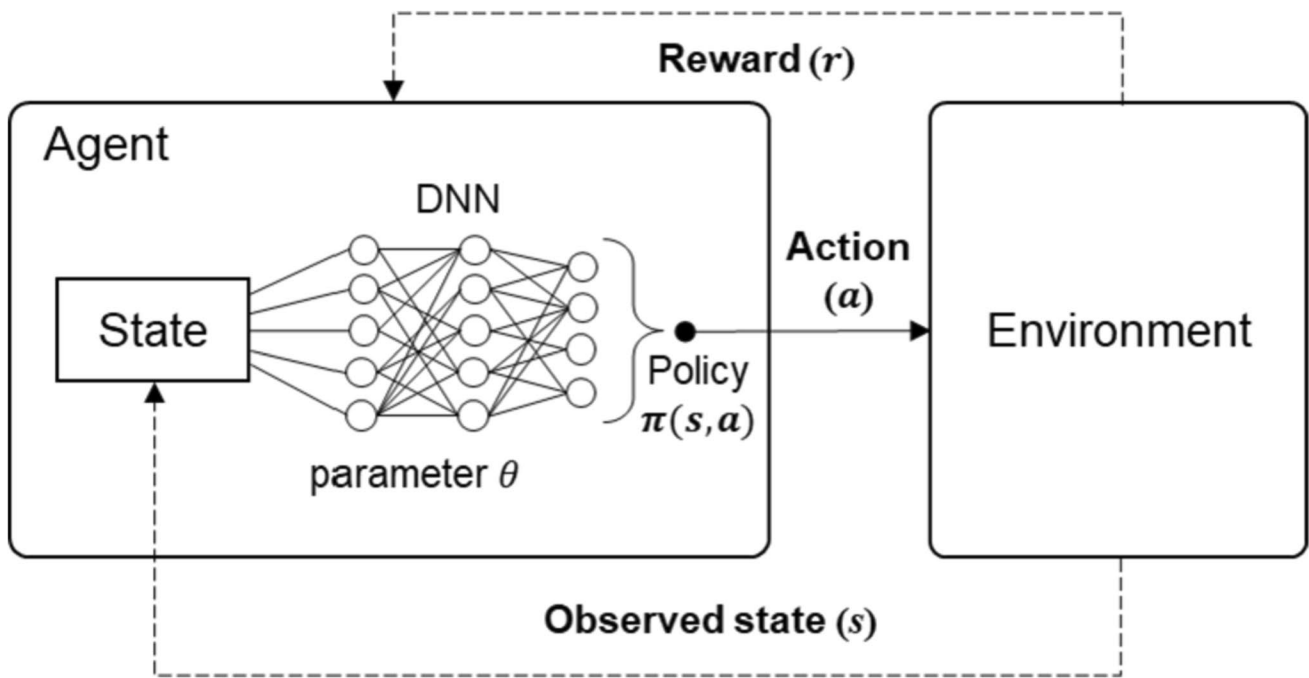


Fig. 1 Basic principle and parameters of DRL

ensure that the new policy does not deviate too much from the old policy. To ensure that policies are updated within a trust region and to prevent drastic and unstable policy changes, PPO optimizes this surrogate loss function.

The advantage function $\hat{A}^{GAE(\gamma, \lambda)}$ quantifies each impact of an action on performance (Eq. 10).

$$\hat{A}^{GAE(\gamma, \lambda)} = \sum_{l=0}^{\infty} (\gamma \lambda)^l \delta_{t+l}^V \quad (10)$$

where λ denotes a parameter used in algorithms that employ generalized advantage estimation, such as PPO. It determines the trade-off between bias and variance in estimating the advantages of policy updates. A higher λ assigns more weight to future rewards, leading to a higher emphasis on long-term planning, while a lower λ places more importance on immediate rewards and focuses on short-term gains. λ affects the balance between exploration and exploitation during policy updates in PPO. δ_{t+l}^V represents the temporal difference error, which is the difference between the estimated value V of the current state and the value of the next state. The exponent l represents the time steps into the future,

indicating how far the estimator looks ahead to calculate the advantage.

In PPO, one of the crucial hyper-parameters is ϵ , which controls the level of clip or proximity during the policy update step. It limits the maximum change allowed in policy parameters and maintains the stability of the learning process. By constraining the policy update within a certain range determined by ϵ , PPO balances exploration and exploitation, avoiding large policy changes that could hinder the stability of the learning process. The selection of ϵ impacts the trade-off between learning speed and policy stability, with a smaller value encouraging more cautious updates and a larger value allowing more exploration and potentially faster learning.

In the realm of robot control systems, PPO stands out as a robust DRL algorithm with multiple benefits. First, it can handle continuous action spaces, making it ideal for controlling the precise joint movements of the robot manipulators. By using PPO, robot manipulators can learn policies that enable smooth and continuous motions, empowering them to navigate confined and complex environments without collision. The stability and reliability of PPO during the learning

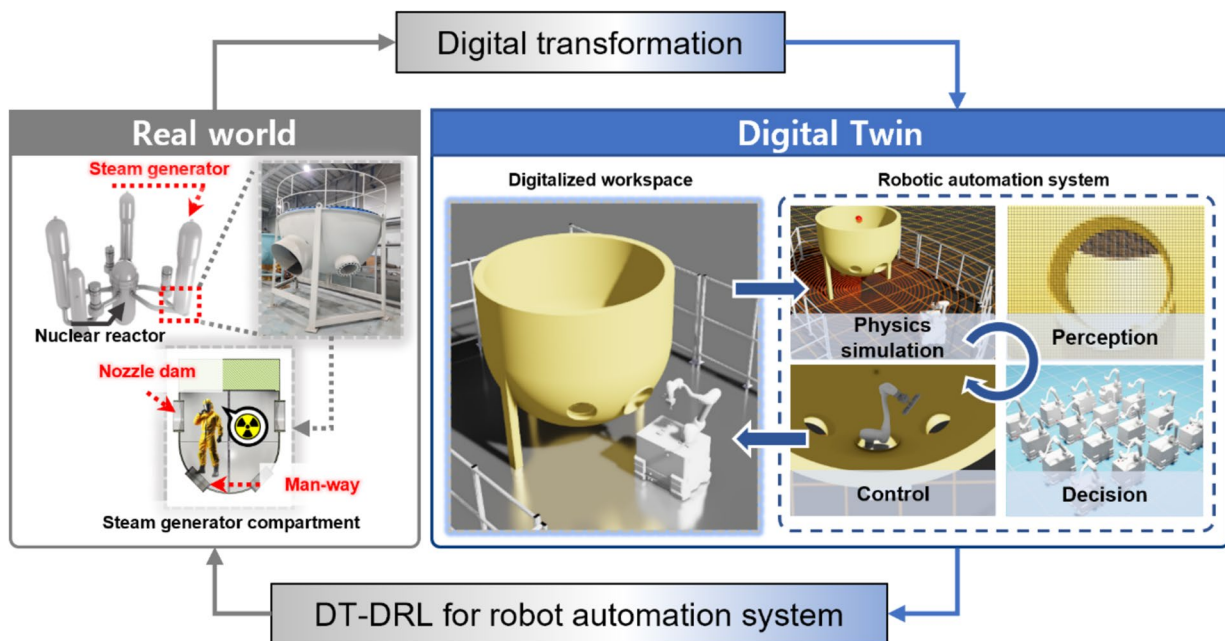


Fig. 2 System overview

process are crucial for robot manipulators, facilitating safe exploration of the environment while gradually enhancing collision-free trajectory generation capabilities. This feature is particularly valuable when dealing with confined and complex environments that require accurate and safe trajectory generation.

4 Methods

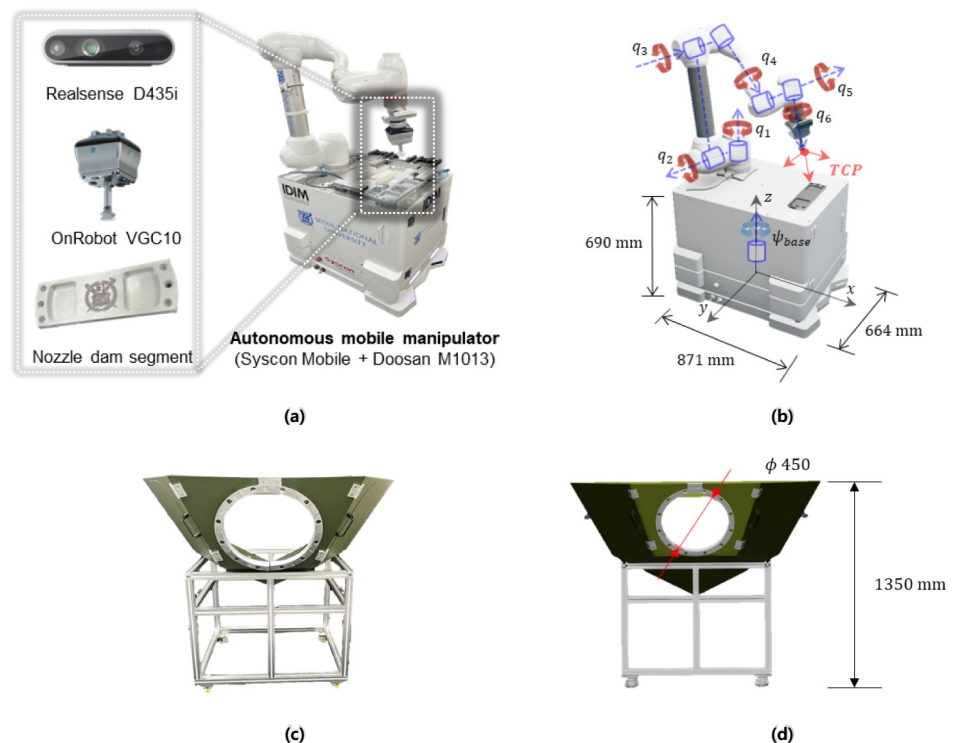
This study aims to develop an innovative RAS using DT-DRL for maintenance tasks in a challenging confined chamber, focusing on the nozzle dam replacement tasks in nuclear power plants. This task involves workers entering a confined steam generator compartment through a narrow man-way with high levels of radiation. Owing to the high risk to workers, primarily caused by excessive radiation exposure and the heavy weight of the nozzle dam, this task is often classified as an aversion task. Therefore, it is necessary to create a DT environment and develop RAS within it to transform high-risk tasks typically performed by human workers into automated tasks performed by the RAS. The proposed DT-DRL for RAS comprises three subsystems (Fig. 2): (1) a digital transformation to the DT, explained in Sect. 4.1, (2) a DT-based visual

perception system, described in Sect. 4.2, and (3) a DRL-based trajectory, described in Sect. 4.3. This DT-DRL for RAS enables precise modeling, realistic perception, autonomous learning, and seamless integration between virtual and real-world environments. The study aims to improve the efficiency, accuracy, and safety of RAS in challenging industrial fields, where conducting empirical evaluations is difficult for real-world robot systems.

4.1 Digital Transformation of Actual System to High-Fidelity DT

All the hardware components are virtualized in the DT environment, including an AMM (Syscon© AMR and Doosan© M1013 robot manipulator), RGB-D camera (Realsense© D435i), suction gripper (OnRobot© VGC10), simplified steam generator compartment model with a single port (man-way), and nozzle dam segment (Fig. 3). The actual steam generator was simplified considering the specifications of the robot hardware. Considering the reachable workspace of the robot manipulator and the structure and size of the simplified steam generator, this structure was suitable for the motion generation problem in a confined workspace that we were trying to solve with our proposed method.

Fig. 3 Real robot system and hardware models for the nozzle dam task, **a** the hardware components of the robot system in the real-world, **b** the virtual hardware components of the robot system in the DT, **c** the simplified steam generator compartment model in the real-world, **d** the virtual steam generator compartment model in the DT



An NVIDIA Omniverse Isaac Sim was used as the DT platform in this study; it is an end-to-end platform that provides a realistic physical environment as well as perception, decision, and control systems, which are the components of RAS [47]. This DT platform offers a highly realistic environment for modeling and evaluating complex robot systems, with the added benefit of integrating the NVIDIA Isaac Gym [48] for training and optimizing intelligent agents. By interacting with the parallelized simulation environment, these agents can learn and adapt optimal behaviors, making this platform a powerful tool for developing and validating the proposed RAS. For these reasons, our DT platform is particularly proper to address the challenges associated with single-port enclosure tasks, such as the nozzle dam replacement of nuclear power plants.

4.2 Hybrid Visual Perception System for the Precise 6D Pose Estimation

We propose a hybrid algorithm that integrates the DOPE and ICP algorithms to estimate a more precise 6D pose of the target object. DOPE employs deep-learning models and belief maps to estimate the initial pose of the object, whereas ICP facilitates the iterative refinement of point cloud alignment (Fig. 4). By combining the strengths of both algorithms, we aim to overcome individual limitations and achieve enhanced accuracy and robustness in 6D pose estimation.

This hybrid visual perception system offers several advantages over using either algorithm individually. DOPE provides an accurate initial pose estimation by harnessing the power of deep-learning and belief maps. This estimation

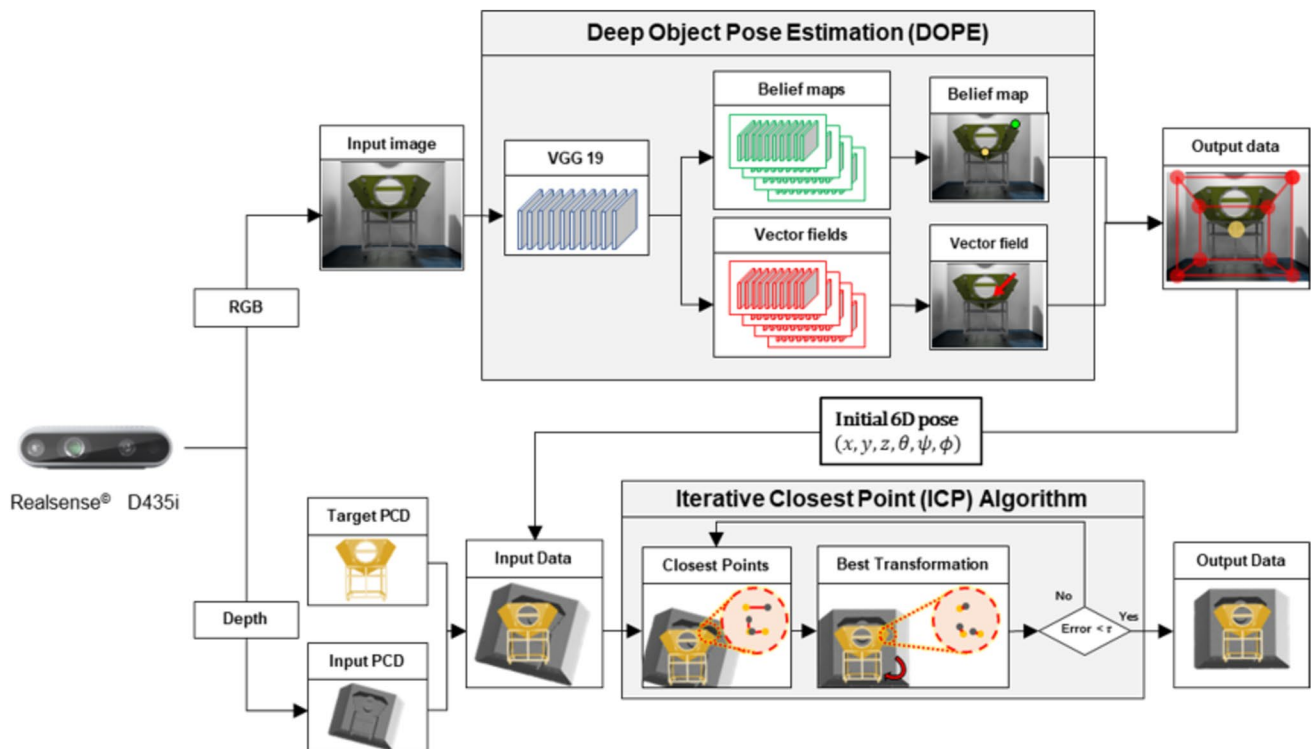


Fig. 4 Diagram of hybrid visual perception system

serves as a reliable foundation for subsequent ICP alignment, reducing the reliance on potentially inaccurate initial guesses and improving overall alignment accuracy. Incorporating ICP allows for iterative refinement, enabling fine-grained adjustments to align point clouds and augment the accuracy and precision of the pose estimation. By synergizing the strengths of both algorithms, the hybrid approach attains superior robustness, accuracy, and performance compared to employing either algorithm alone.

Where the mobile bases of the AMMs should be located can be defined using the hybrid visual recognition system. The final base position p^* to which the mobile base should move can be determined by adding b_{opt} obtained by IRM to the absolute coordinates of the steam generator p_{SG} (Eq. 11).

$$p^* = p_{SG} + b_{opt} \quad (11)$$

where p_{SG} is the absolute steam generator pose estimated by the combined algorithm with DOPE and ICP, b_{opt} is the relative optimal pose of AMM obtained by IRM.

4.3 Hybrid Decision System Combining IRM and PPO for Collision-Free AMM Trajectory

We used a hybrid decision system to generate the collision-free trajectory τ^* of the robot manipulator to the target point (Eq. 12).

$$\tau^* = [p_0, \dots, p_i, \dots, p_T] \quad (12)$$

where p_0 is the initial point, p_i is the i -th point, and p_T is the final point of TCP. Note that p_T should be within 10 mm from the target point p^* , and all points p_i ($0 \leq i \leq T, i \in \mathbb{Z}$) should be collision-free. The initial point p_0 is determined by IRM, and the rest of the trajectory $[p_1, \dots, p_i, \dots, p_T]$ is determined by the trained policy of PPO.

Controlling the mobile base and the 6-DOF robot manipulator simultaneously theoretically allows for more degrees of freedom and potentially more flexible movements in confined workspaces. However, in practice, the accumulation of errors and instability in the hardware can render it unsuitable for performing complex tasks accurately. To address this issue, we adopted a different approach by fixing the mobile base at a specific location and controlling the robot manipulator separately. This separation of control aims to enhance precision and stability in the trajectory generation of the mobile manipulator and execution.

This hybrid decision system that combines IRM and PPO generated collision-free trajectories for the AMM operating in a confined chamber, including narrow passages (Algorithm 1). First, the target points p_{target} that the robot manipulator TCP needs to reach and search space B are set. The algorithm then iterates through the IK computation for every grid cell (lines 1–4). The base pose b_{opt} with the highest number of feasible solutions is selected as the final pose (line 5). Once the selected optimal base pose b_{opt} is applied to the pose of the AMM, the PPO algorithm trains the collision-free trajectory generation of the robot manipulator. The PPO policy $\pi(a|s)$ selects an action a , which is assigned to the robot manipulator TCP p , and this process is repeated to obtain state information s and reward r (lines 6–11). In the case of a collision, the entire environment is reset, and a penalty is applied (lines 12–15). When the robot manipulator TCP passes through a waypoint, a high reward is applied only once (lines 16–18). This is to prevent the TCP from repeatedly passing by the waypoint intentionally. In addition, the distance between p and p_{target} is given as a penalty, so that learning can converge quickly (line 19). This iterative process trains the robot manipulator to generate final trajectories τ^* in confined workspaces without collisions (lines 20–22).

Algorithm 1 Hybrid algorithm combining IRM and DRL for collision-free trajectory

Input: Target point p_{target} , Search space B

Output: Optimal base pose b_{opt} , Collision-free trajectory τ^*

Procedure:

- 1: **IRM** to find the optimal base pose:
 - 2: **for** B **do**
 - 3: Compute feasible $b_{ij} = f_{IK}(p_{target})$ of AMM without collision
 - 4: Compute the feasibility score (D_i) **for** the base poses (b_i)
 - 5: **Select** the optimal base pose (b_{opt})

 - 6: **DRL** to generate a collision-free trajectory of the robot manipulator:
 - 7: Initialize the current state (s) **with** the selected base pose (b_{opt})
 - 8: Initialize an empty trajectory (τ)

 - 9: **while not** reaching the target point **do**
 - 10: Take an action based on the policy ($\pi(a|s)$)
 - 11: Update the state and joint values of the manipulator based on the action

 - 12: **If** collision detection:
 - 13: Initialize the state (s) **and** TCP (p) of the robot manipulator
 - 14: Reset the episode
 - 15: **return** collision penalty ($-r_{collision}$)

 - 16: **If** passing the waypoint:
 - 17: **If** the first time to pass:
 - 18: $r \leftarrow r + r_{waypoint}$

 - 19: $r \leftarrow r - r_{dist}$

 - 20: Append the current state (s) **and** TCP (p) to the trajectory (τ)
 - 21: **return** r

 - 22: **Select** the collision-free trajectory (τ^*)
-

4.3.1 IRM-Driven Optimal Base Pose Estimation

In this study, we set the search space range x_{\min} as -0.20 m, x_{\max} as 0.20 m, y_{\min} as -0.30 m, y_{\max} as -0.10 m, based on the origin of the steam generator. The grid cells were divided by $50 \times 50 \times 1$, then the x, y interval of grid cell $x_{\text{step}}, y_{\text{step}}$ were set as 8.00 mm and 4.00 mm, respectively. We set the range of yaw values ψ in each grid cell ψ_{\min} as -90.0° , ψ_{\max} as 90.0° , and the interval ψ_{step} as 0.50° . Consequently, the dimension of search space B and feasibility score set D were both 2500. To find the optimal base pose b_{opt} , base poses b and target point p_{target} we used the objective function (Eq. 13). We set the target point to $p_{\text{target}}(x, y, z, \theta, \phi, \psi) = (-0.43 \text{ m}, 0.18 \text{ m}, 1.08 \text{ m}, -122^\circ, -33.2^\circ, 71.1^\circ)$, and that point is where the nozzle dam segment was to be attached.

$$b_{\text{opt}} = \underset{b \in B}{\operatorname{argmax}} D(b|p_{\text{target}}) \quad (13)$$

The grid cells are visualized with different colors based on the feasibility score D , and the grid cell with the highest number of IK solutions (blue) was chosen as the optimal base pose b_{opt} (Fig. 5(a)). The color of each grid point varied depending on the relative number of reachable base poses, indicating relatively few (green) or no solutions (red), respectively. Finally, the base pose of the mobile base was fixed to one of its optimal base poses in terms of the absolute coordinate system (Fig. 5(b)). The optimal base pose was selected as $b_{\text{opt}}(x, y, \psi) = (-0.32 \text{ m}, -1.64 \text{ m}, 22.5^\circ)$.

4.3.2 DRL-Driven Collision-Free Trajectory Generation with a High-Rewarded Waypoint

After finding the optimal base pose using IRM, PPO trains the policy to find the collision-free trajectory τ^* . Distance-based DRL methods, which use the distance between the TCP of the robot manipulator and the target point, are commonly employed in DRL-based motion generation. However, it is difficult to create collision-free paths in narrow passages and confined workspaces using distance-based reward shaping only. To address this issue, the centroidal waypoint p_c with a high reward value was applied. This waypoint has a higher positive reward than the penalty for collisions and it makes the training converge; it generates collision-free trajectories. Therefore, the robot manipulator can pass through the narrow passage without collision according to the trajectory including the waypoint p_w (Eq. 14).

$$\tau_{\text{waypoint}} = [p_0, \dots, p_i, \dots, p_w, \dots, p_T]. \quad (14)$$

where p_w is the centroidal waypoint, and p_w is located inside the midpoint of the entrance. We set $p_w(x, y, z) = (1.05 \text{ m}, 0 \text{ m}, 1.15 \text{ m})$ relative to the origin of the absolute coordinate system. Next, the components of MDP, which serves as the fundamental framework for DRL, are defined. These components include the state observation s , action a , reward r , and the reset condition for initializing episodes. The current state observation s_t includes various elements such as the current time step t , current joint angles q_{current} , joint velocities \dot{q} , pose of the TCP p_{TCP} , pose of the

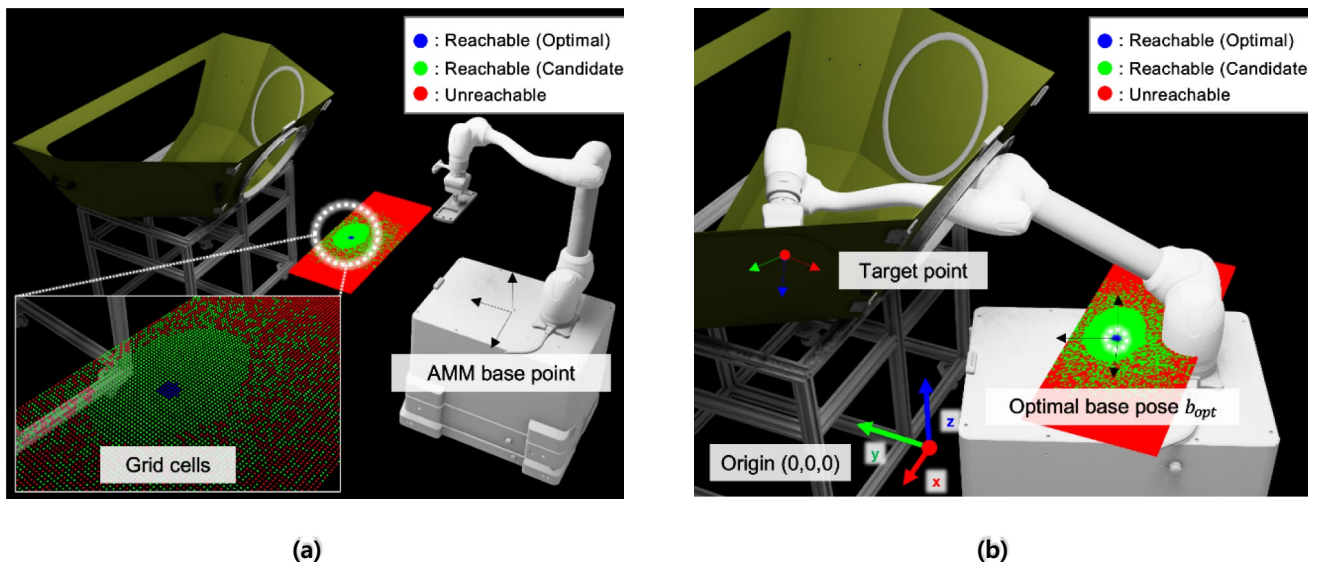


Fig. 5 IRM-driven optimal base pose estimation of the AMM in the DT environment, **a** generated IRM with grid cells in front of the steam generator compartment and **b** a collision-free IK solution of the AMM on the optimal base pose

target point p_{target} , collision buffer C_o , and translation of a waypoint $p_{waypoint}$ (Eq. 15).

$$s_t = [t, q, \dot{q}, p_{TCP}, p_{target}, C_o, p_{waypoint}] \in \mathbb{R}^{29}. \quad (15)$$

The next step involves setting a as the change of joint values q per unit time dt (Eqs. 16–18). By adding Δq to the current joint values of the robot manipulator q_t , the PPO policy enables direct control of the joints of the robot manipulator to the next joint states q_{t+1} .

$$q = [q_1, q_2, q_3, q_4, q_5, q_6] \in \mathbb{R}^6. \quad (16)$$

$$a_t = \Delta q = [\Delta q_1, \Delta q_2, \Delta q_3, \Delta q_4, \Delta q_5, \Delta q_6] \in \mathbb{R}^6. \quad (17)$$

$$q_{t+1} = q_t + a_t. \quad (18)$$

where the range of the action a was within $\pm 1.19^\circ/\text{frame}$ considering the performance of the actual robot manipulator. Since the maximum angular velocity of each joint of the robot manipulator we used was $120^\circ \sim 220^\circ/\text{s}$, it is theoretically possible to rotate between $2.00^\circ \sim 3.67^\circ/\text{frame}$.

The rewards, r_1 and r_2 , represent penalties for the Euclidean distance between the TCP and the target point at each time step and collision occurrences, respectively. r_3 represents a positive reward for passing by the waypoint (Eqs. 19–21).

$$r_1 = - \| p_{target} - p_{TCP} \|. \quad (19)$$

$$r_2 = \begin{cases} -1, & \text{if collision occurs} \\ 0, & \text{otherwise} \end{cases} \quad (20)$$

$$r_3 = \begin{cases} +1 & \text{if away point passage} \\ 0, & \text{otherwise} \end{cases} \quad (21)$$

To prevent excessive bias toward a single reward parameter, each r is multiplied by a scale factor w . The values of w_1, w_2, w_3 are set to 2, 80, and 100, respectively (Eq. 22).

$$r_t(s_t, a_t) = w_1 r_1 + w_2 r_2 + w_3 r_3. \quad (22)$$

To reset an episode, conditions such as the distance between the TCP and the target point being less than 10.0 mm, collisions occurring, or exceeding the maximum time step are defined.

Next, the hyper-parameters are tuned (Table 1). The action and critic network of each PPO algorithm had three hidden layers, and the value of ϵ was set to 0.20 to ensure stable learning. In particular, λ was set to 0.85 to increase the exploration component of the PPO policy, enabling the performance of a wider range of actions. This set of hyper-parameters encourages passing the waypoints located within

Table 1 Hyper-parameters for the PPO algorithm

Hyper-parameters	Values
Learning epochs	8
Mini batches	8
Learning rate	0.001
Number of hidden layers of actor network	3
Number of hidden layers of critic network	3
Discount factor (γ)	0.99
GAE lambda (λ)	0.85
Ratio clip (ϵ)	0.20
Max time step	200
Total time step	5000
Number of training agents	512

a narrow single-port enclosure. A single episode consisted of 200 time steps, and 5000 time steps was executed during a single training.

The agent was trained in parallel using Isaac Sim and *OmniIsaacGymEnvs* [49], which enables the use of Isaac Gym within Isaac Sim. The training was conducted with 512 individual agents, and the information generated by each agent was used to update the global network and facilitate rapid learning through information sharing (Fig. 6).

5 Experiments and Results

5.1 Case study: Nozzle Dam Replacement Task

As a case study, a nozzle dam replacement task was performed using the integrated DT (Fig. 7). At the initial position, RGB data and PCD are obtained through the RGB-D camera, and (1) the DOPE algorithm is used to detect the target model, that is, the steam generator compartment model and approximately estimate its 6D pose. The approximately extracted 6D pose and (2) PCD obtained by the camera are then used as the input data for the ICP algorithm. Using ICP, a more precise 6D pose is obtained, enabling the estimation of the relative translation and orientation between the target object and the AMM. (3) The AMM navigates to the optimal base pose selected through IRM. (4) The suction gripper picks up the nozzle dam segment and (5) based on the collision-free motion generation policy trained by DRL, the robot manipulator can pass through the narrow passage, that is, the man-way. Once the TCP of the robot manipulator reaches the target point inside the confined steam generator chamber, (6) the nozzle dam segment is attached to it, and (7) the robot manipulator returns to the initial home position.

All experiments were conducted in the DT environment based on Isaac Sim. High-performance hardware and software were used to enable rapid computation of high-fidelity

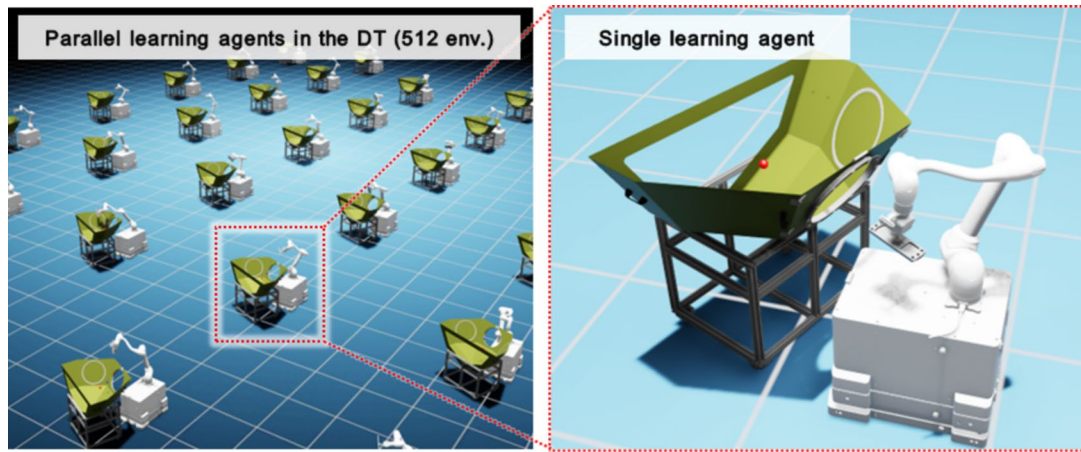


Fig. 6 Parallel learning in DT using *OmniIsaacGymEnvs*

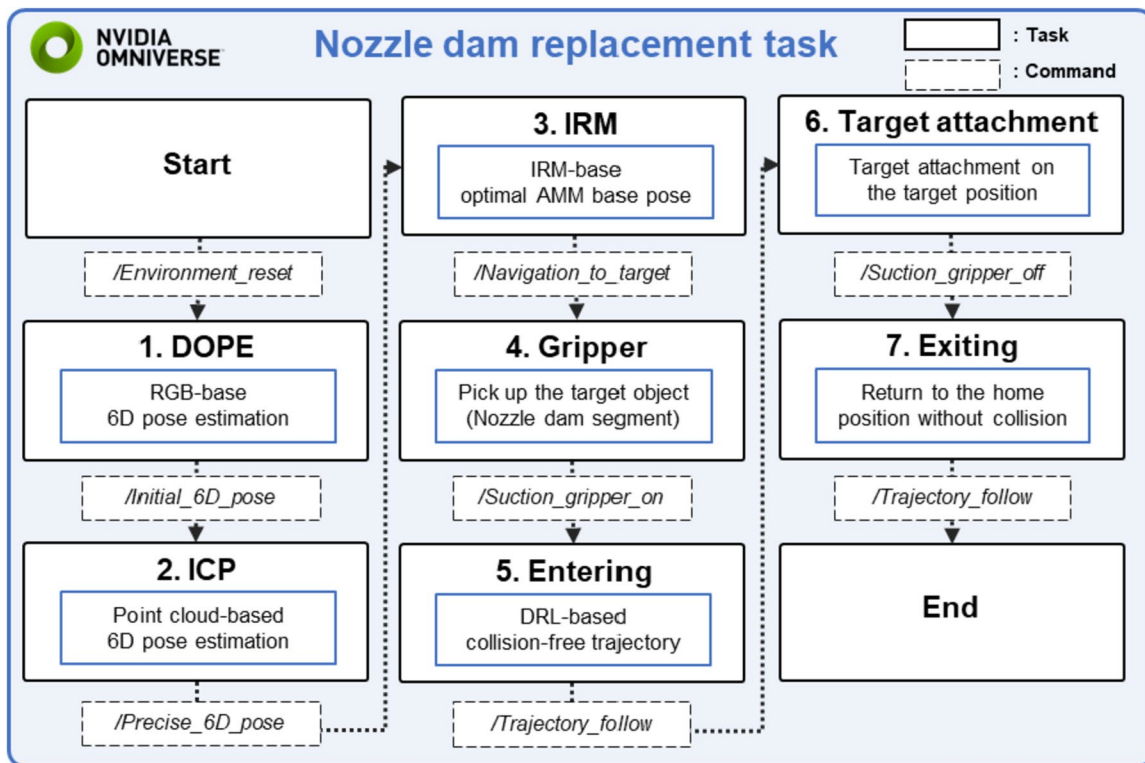


Fig. 7 System workflow of the nozzle dam replacement task

physics data and 3D visualization (Table 2). The DT based on Isaac Sim is built on Ubuntu; the algorithmic systems and actual hardware systems communicate through ROS.

5.2 Evaluation of the Hybrid Visual Perception System

To assess the performance of the hybrid visual perception system, experiments were conducted in both the DT and

real-world environments. Our goal was to perform more accurate pose estimation with a perception system that connects the two algorithms in series, rather than using each of the DOPE and ICP algorithms individually. In the first step, a realistic RGB-D camera was implemented in the DT environment. To increase the similarity with the real-world environment, the camera resolution in the DT was adjusted to match the specifications of the actual

Table 2 Hardware and software specification of a workstation for the DT

Component		Specification
Hardware	CPU	Intel Core i9-9900KF
	Memory	39.1 GB DDR4 RAM
	Graphics card	NVIDIA GeForce RTX 2080 Ti
	Storage	512 GB SSD
Software	Operating system	Ubuntu 18.04 LTS
	Graphic driver	525.60.11
	CUDA	12.0
	ROS	Melodic Morenia
	Isaac Sim	2022.2.0

camera. The RGB frame resolution was set to 1920×1080 pixels with a FOV of $69.0^\circ \times 42.0^\circ$, while the depth channel resolution was set to 1280×720 pixels with a FOV of $87.0^\circ \times 58.0^\circ$. The RGB data and PCD obtained from the

camera in the DT environment were fed into the hybrid perception algorithm. Following the algorithm flow, the obtained PCD (cyan) was registered with the PCD of the target object (yellow), resulting in a successful transformation (Fig. 8).

In the real-world environment, the same camera captures RGB data and PCD, and the hybrid algorithm is applied, resulting in a successful PCD transformation and smooth registration (Fig. 9).

To provide a more quantitative comparison, the accuracy of the DOPE, ICP, and hybrid algorithms was measured in the DT environment (Table 3). When using the DOPE and ICP algorithms individually, an average translation error of 0.41 m and 1.74 m, respectively, along with orientation errors of 8.33° and 102° , were observed. In contrast, when using the hybrid algorithm that combines DOPE and ICP, an average translation error of 0.11 m and an average orientation error of 7.00° were observed. The translation accuracy improved by 73.0% and 93.6%

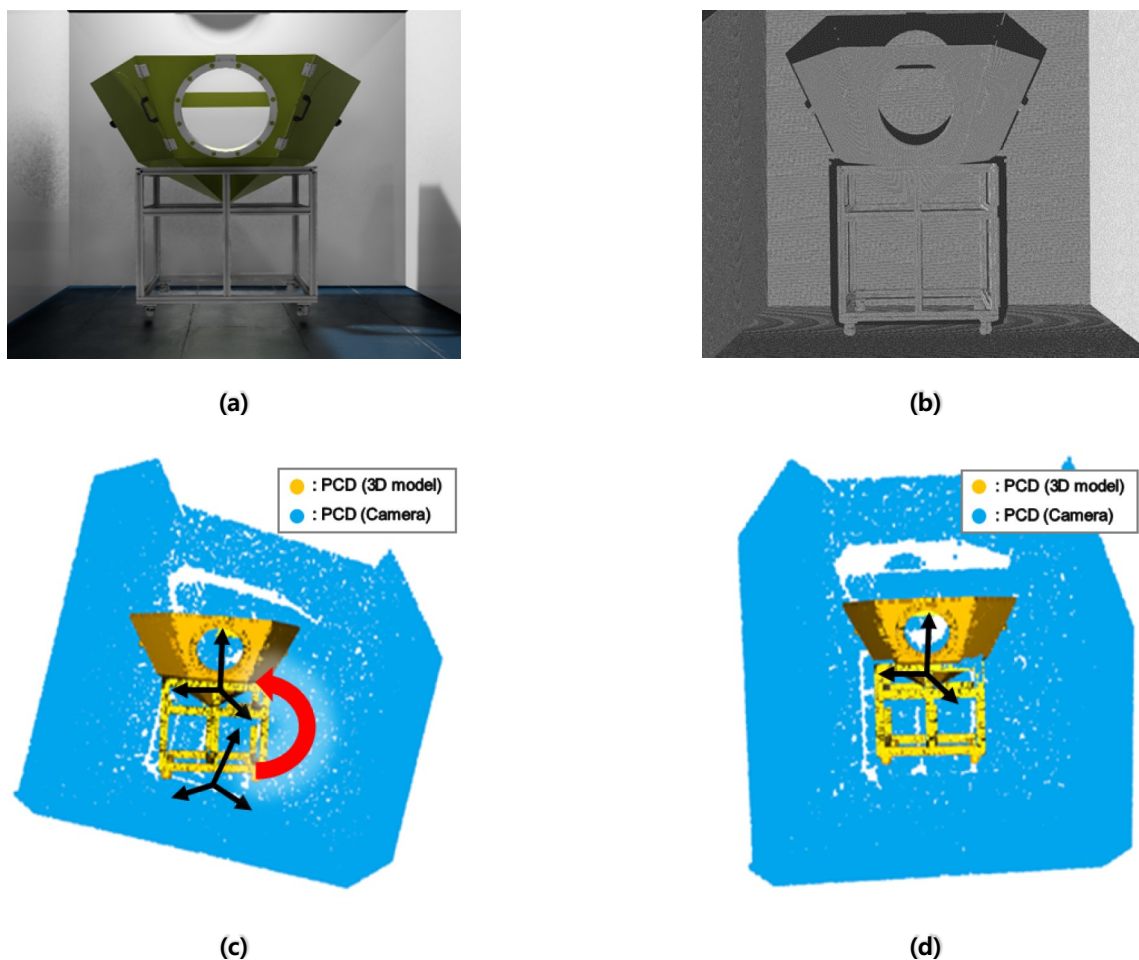


Fig. 8 Evaluation of the perception system in the DT **a** virtual model of steam generator compartment in the DT, **b** raw PCD obtained from RGB-D camera, **c** obtained and target PCD before hybrid perception algorithm, **d** PCD after hybrid perception algorithm

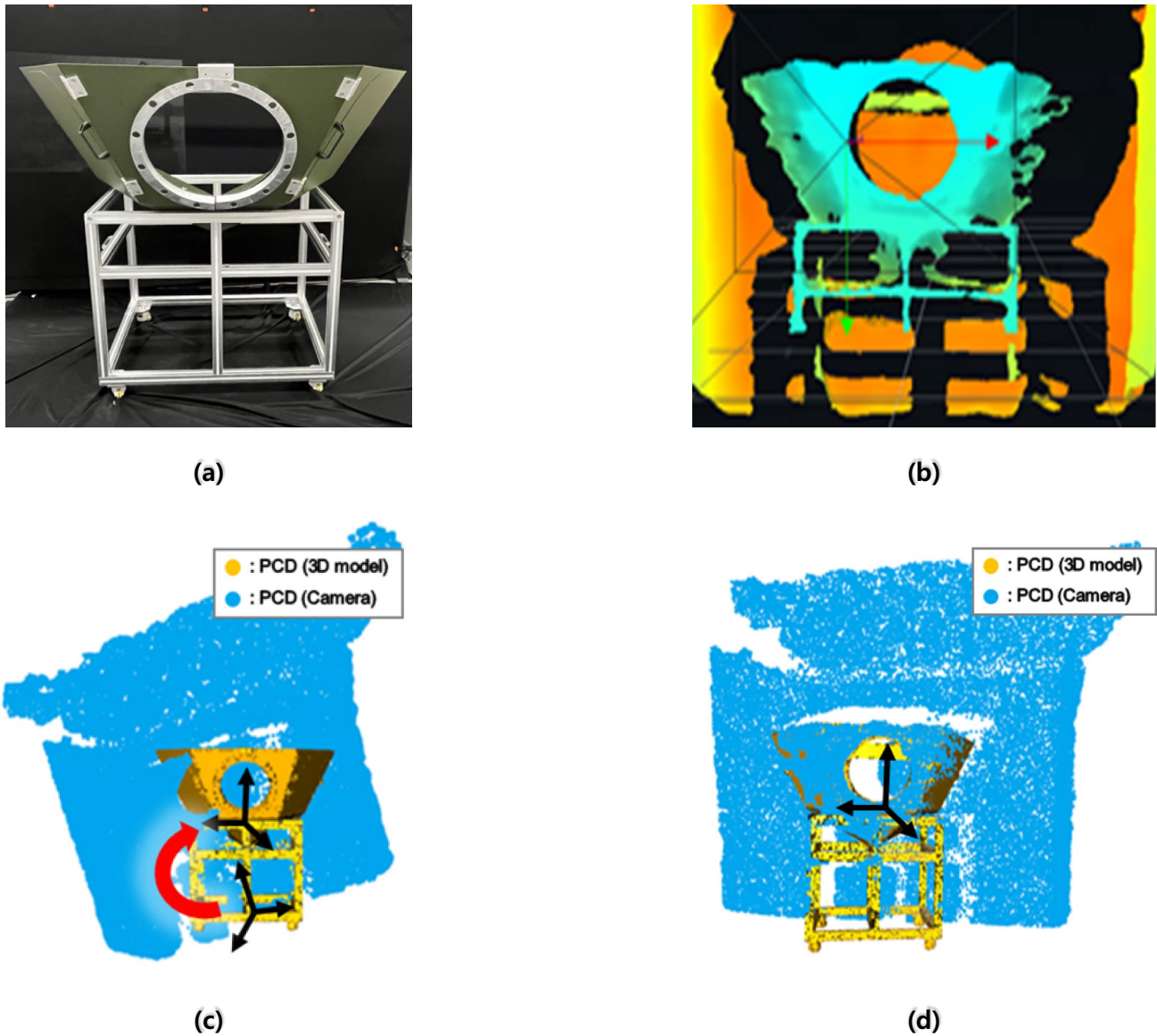


Fig. 9 Evaluation of the perception system in the real-world, **a** real model of steam generator compartment, **b** raw PCD obtained from RGB-D camera, **c** obtained and target PCD before hybrid perception algorithm, **d** PCD after hybrid perception algorithm

Table 3 Performance comparison of perception algorithms

$n = 20$		DOPE	ICP	Hybrid (DOPE + ICP)
Translation [m]	Mean	0.41	1.74	0.11
	Std	0.19	0.59	0.05
	Min	0.14	0.94	0.05
	Max	0.84	2.85	0.24
Orientation [°]	Mean	8.33	102	7.00
	Std	5.17	55.5	3.68
	Min	2.38	7.59	2.58
	Max	21.9	236	15.2

compared to DOPE and ICP alone, respectively. The orientation accuracy improved by 16.0% and 93.1% compared to DOPE and ICP alone, respectively.

In the case of ICP, the translation and orientation errors were significantly larger than those of the other two algorithms. This was the result of the initial transformation failure, which is a main weakness of ICP. However, when the approximate 6D pose of the target object obtained through DOPE was used as the initial transformation of ICP, the accuracy was greatly improved. The vulnerability of DOPE to the noise of RGB data was overcome by using PCD-based registration of ICP and the estimation accuracy was improved. Based on the result, the hybrid algorithm

successfully addressed the limitations of DOPE and ICP and provided a significant improvement in 6D pose estimation accuracy.

Combining DOPE and ICP in a serial manner significantly improved the accuracy of 6D pose estimation, but also introduced some challenges. In particular, due to the high computational complexity of the DOPE algorithm, the average inference speed was only 1.32 frames per second. This processing speed presents a disadvantage in terms of real-time performance, especially when fast responses are required. However, in our application, where precision and accuracy of target object pose estimation are of utmost importance, the accuracy improvement achieved through the hybrid algorithm is more important than real-time performance. In situations where the accuracy of pose estimation is critical to the results, such as delicate or high-risk robotic tasks, slower inference speeds may be justified. Therefore, despite low real-time performance, a hybrid approach integrating DOPE and ICP provides a valuable solution where accuracy is paramount.

5.3 Hybrid Algorithm for Collision-Free Trajectory Generation Using IRM and PPO

To assess the efficacy of the collision-free trajectory generation policy, the PPO algorithm was implemented in the DT environment after 5000 training time steps. Five different target points were tested to evaluate the trajectory generation policy of the robot manipulator (Fig. 10). The TCP trajectories are recorded and displayed in a 3D Cartesian space (Fig. 10(a)). From the top-view, all trajectories can be observed to depart from the home point, pass through the waypoint (green circle), and arrive at the target point (Fig. 10(b)). From the isometric-view, the TCP can be seen to pass through the center of the narrow man-way (Fig. 10(c)). From the front-view, the TCP successfully reaches the designated target points through the waypoint (Fig. 10(d)). At the end of every TCP path, it reached the target by bending, which is because the geometrical complex collision avoidance between the robot manipulator and the confined steam generator was also considered. Consequently, it can be concluded that the waypoint-based policy has effectively learned how to plan collision-free trajectories.

To determine the impact of the waypoint on the training results, separate training sessions were conducted with and without the waypoint. The policies are displayed in 3D Cartesian space in Fig. 11. The policy without the waypoint avoided collisions but tended to linger around the vicinity of the target point, outside the steam generator compartment

(Fig. 11(a-b)). The policy without the waypoint failed to enter the man-way and reach the target point (Fig. 11(c)). However, the policy with the waypoint successfully generated trajectories that passed by the centroidal waypoint, enabling it to reach the target point (Fig. 11(d)).

Examining the accumulated rewards, it can be observed that in both cases, the reward values converged to a specific value after 2000 time steps (Fig. 12). The policy without the waypoint (blue line) converged to a reward value near -50 after approximately 1000 time steps, resulting from the penalty incurred for the distance, as it did not incur any penalty for collision or obtain any positive reward from the waypoint. Conversely, the policy with the waypoint (red line) maintained an accumulated reward of -100 from around 500 time steps and converged to close to 0 at 1500 time steps. This suggests that the centroidal waypoint-based DRL policy explores the high-reward waypoint within the narrow man-way through extensive exploration, even in the presence of collisions, and learns to minimize the distance to the target point. These results demonstrate the successful development of a DRL-based collision-free trajectory generation policy in the single-port enclosure.

To compare the performance of the DRL-based collision-free trajectory generation policy with and without the waypoint to conventional trajectory generation algorithms, sampling-based algorithms, rapidly exploring random trees connect (RRTConnect) and probabilistic roadmap star (PRMStar) [50] were evaluated. The results demonstrate that the success rate of the DRL policy with the waypoint was relatively higher than that of the other algorithms (Fig. 13). Particularly, the RRTConnect algorithm, which efficiently and rapidly generates paths in complex spaces using random trees, had an average success rate of 32.0% (green bar). The PRMStar algorithm (blue bar) had an average success rate of 4.00%. RRTConnect depends on sampling, and if sufficient sampling is not performed in the initial phase, performance in a geometrically confined environment is significantly reduced. PRMStar specializes in path optimization; more sampling was required and the success rate of path planning in a high-dimensional space was lower. In contrast, the policy trained with the waypoint (cyan bar) had a 100% success rate in all cases, while the policy trained without the waypoint (black bar) failed in all attempts. These results confirm that the DRL-based collision-free trajectory generation policy trained with the waypoint is the most suitable algorithm for performing the nozzle dam replacement task in the steam generator compartment of the single-port enclosure structure.

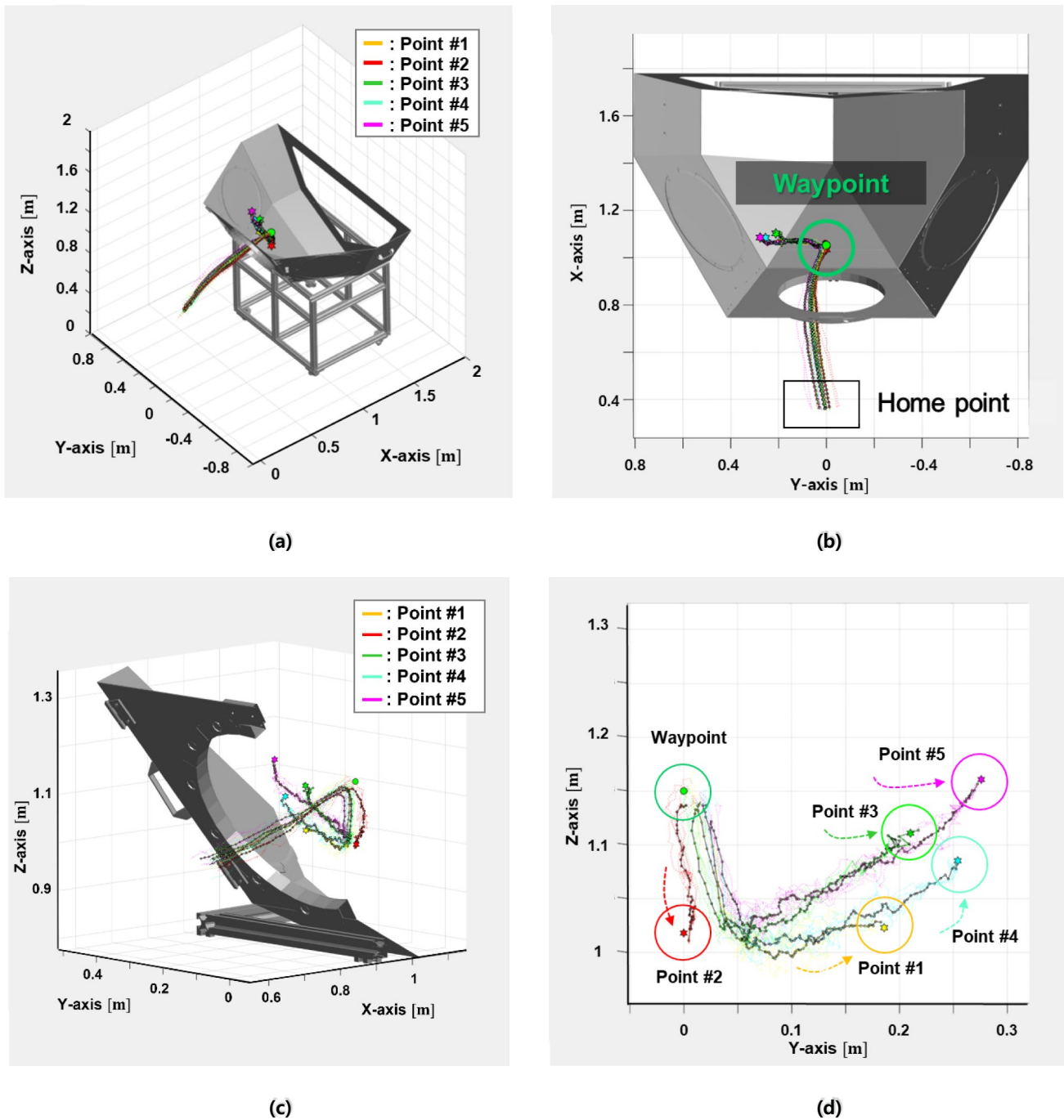


Fig. 10 3D trajectories of the TCP of the robot manipulator toward different target points in the Cartesian workspace **a** TCP trajectories of the robot manipulator, **b** top-view of the TCP trajectories, **c** isometric-view of the TCP trajectories, **d** rear-view of the TCP trajectories

5.4 Evaluation of the Integrated System in the DT and the Real-World Environment

5.4.1 Experiment in the DT

All the developed components were integrated to experiment on the nozzle dam replacement task in the DT environment

(Fig. 14). The image information of the surroundings was captured using a virtual RGB-D camera, and the hybrid perception system was used to estimate the precise 6D pose of the steam generator compartment (Fig. 14(a)). Using the estimated 6D pose as a reference coordinate, the AMR was navigated to the optimal base pose using IRM (Fig. 14(b)). Once the target location was reached, the suction gripper was

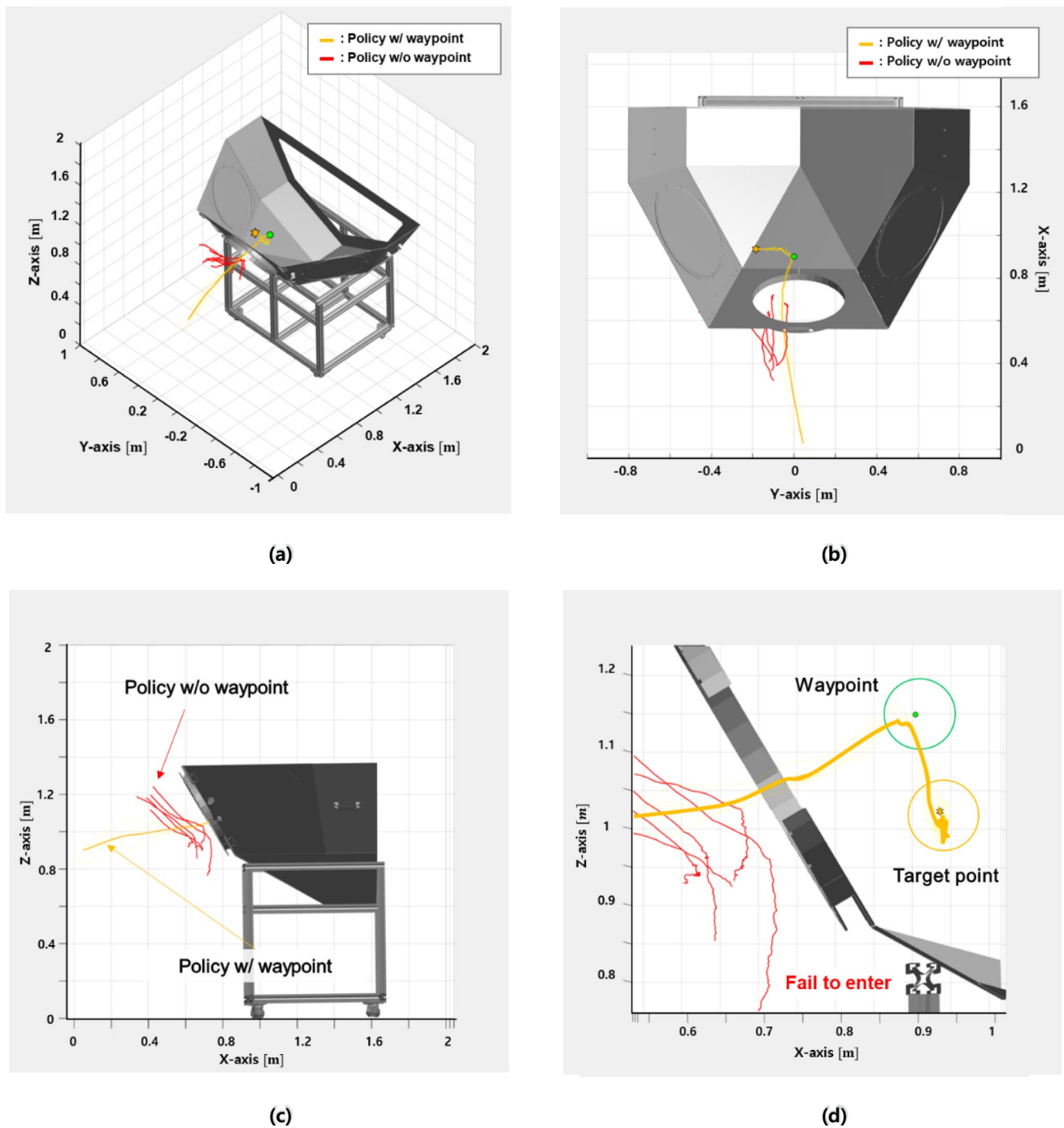


Fig. 11 3D trajectory comparison of DRL policies with and without the waypoint **a** TCP trajectories of the robot manipulator trained with and without the waypoint, **b** top-view of the trajectory, **c** side-view of

the trajectories, **d** difference of the trajectories between policies with and without the waypoint

employed to pick up the nozzle dam segment (Fig. 14(c)). The TCP of the robot manipulator navigated through the narrow port of the steam generator compartment using the DRL-based trajectory generation policy (Fig. 14(d-e)), passing through the waypoint to reach the target point

(Fig. 14(f)). Subsequently, the nozzle dam segment was attached, and the TCP returned to its initial home point following the reverse trajectory (Fig. 14(g-h)). This entire process was executed within 60 s in the DT environment.

Fig. 12 Accumulated reward of DRL with the waypoint (red line) and without the waypoint (blue line)

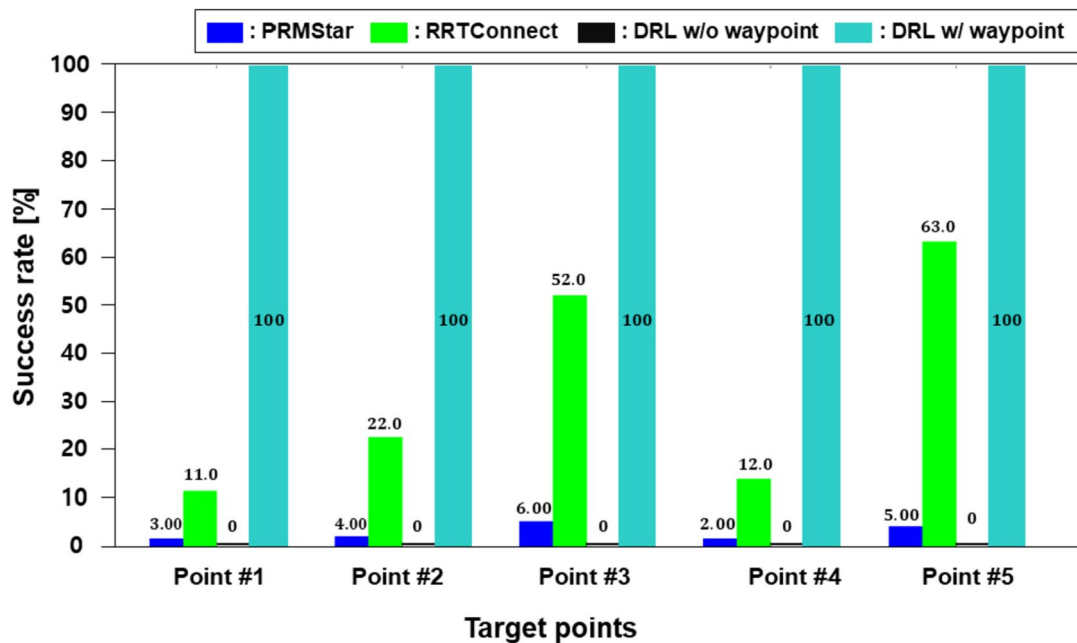
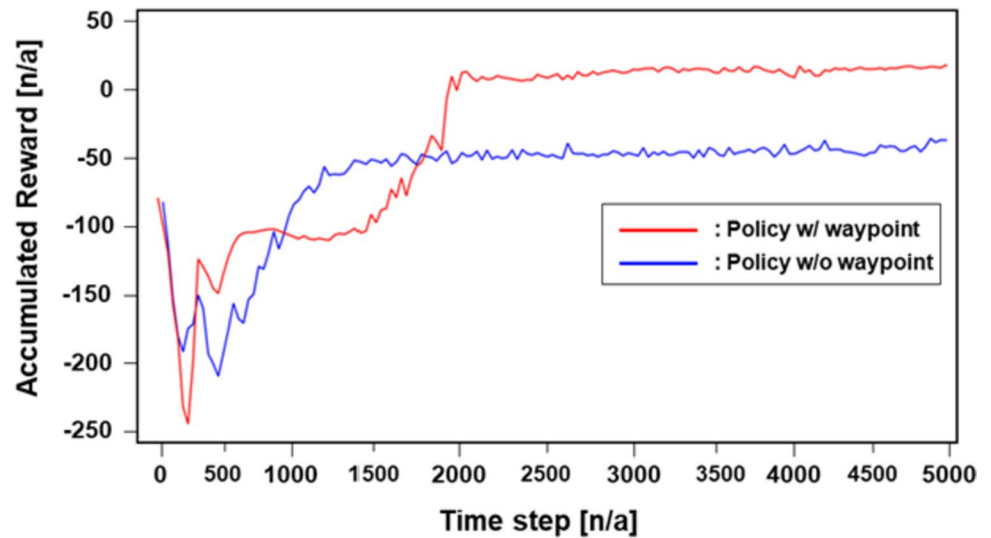


Fig. 13 Comparison of success rates between conventional and DRL-based trajectory generation

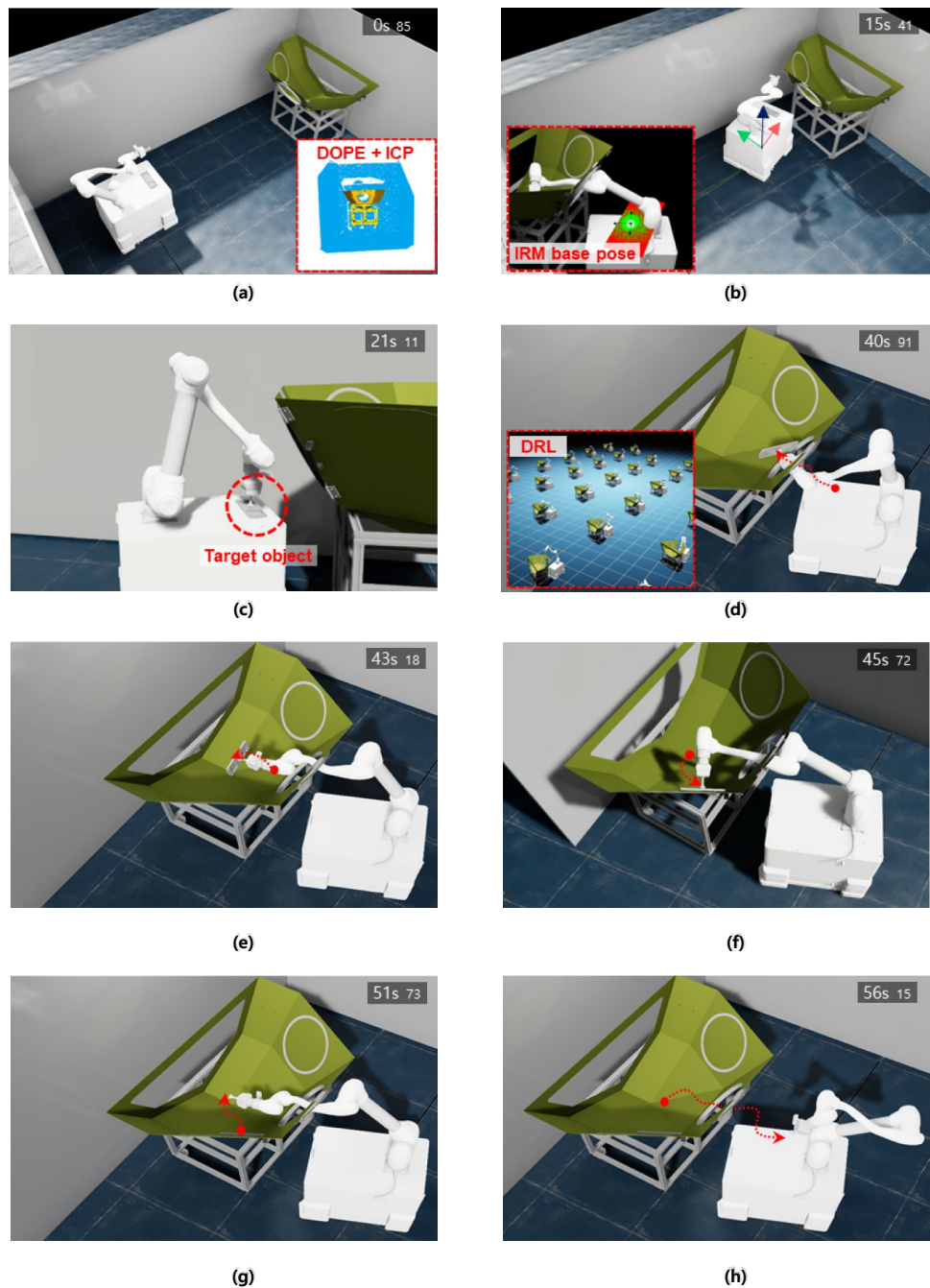
5.4.2 Experiment in the REAL-WORLD

To evaluate the performance in the real-world environment, all tasks conducted in the DT environment were repeated using ROS (Fig. 15). The RGB and PCD data obtained from the forward-facing RGB-D camera attached to the robot manipulator at the initial position were used by the hybrid perception system to estimate the precise 6D pose (Fig. 15(a)). The AMR moved to the optimal base pose computed by IRM (Fig. 15(b)). Next, the nozzle dam segment was picked up using the suction gripper (Fig. 15(c)). The DRL-based trajectory generation policy was used to

navigate the TCP of the robot manipulator through the narrow man-way of the simplified steam generator compartment (Fig. 15(d)), passing through the waypoint to reach the target point (Fig. 15(e)). After attaching the nozzle dam segment to the target point (Fig. 15(f)), the TCP returned along the same path to the initial home point (Fig. 15(g-h)). The entire process was executed within 130 s.

The outcomes of the experiments conducted in both the DT and real-world environment were quantitatively analyzed (Table 4). We calculated the error rates for perception and navigation, as well as success rates for trajectory generation and task completion. The translation error rate is determined

Fig. 14 Evaluation of the nozzle dam replacement task in the DT **a** initialization and 6D pose estimation of the steam generator compartment with hybrid perception system, **b** automation driving of the AMR toward the IRM-driven optimal base pose, **c** picking up the nozzle dam segment, **d** entering the TCP into the narrow man-way with DRL-based policy, **e** passing through the waypoint by the TCP, **f** reaching target point of the TCP and attaching nozzle dam segment, **g** returning to the home point, **h** completion of the task



by computing the absolute difference between the ground-truth Euclidean distance d and the estimated distance d^* , then dividing it by d (Eq. 23). Similarly, the orientation error rate is calculated by taking the absolute difference between the ground-truth yaw angle ψ , and the measured yaw angle ψ^* , and dividing the outcome by ψ (Eq. 24). Finally, the success rate is calculated by dividing the number of successful attempts by the total number of trials (Eq. 25). In the case of perception and navigation system, only translation x , y and orientation ψ values were used to control the AMM, so

translation z , and orientation ρ, θ were excluded from the errors.

$$(\text{Translation error rate}) = \|d^* - d\| / d \times 100\%. \quad (23)$$

$$(\text{Orientation error rate}) = \|\psi^* - \psi\| / \psi \times 100\%. \quad (24)$$

$$(\text{Success rate}) = (\text{number of success}) / (\text{number of trial}) \times 100\%. \quad (25)$$

Fig. 15 Evaluation of the nozzle dam replacement task in the real-world environment **a** initialization and 6D pose estimation of the steam generator compartment with hybrid perception system and the real RGB-D camera, **b** automation driving of the AMR toward the IRM-driven optimal base pose, **c** picking up the nozzle dam segment using the real suction gripper, **d** entering the TCP into the narrow man-way with DRL-based policy, **e** passing through the waypoint by the TCP, **f** reaching target point of TCP and attaching the nozzle dam segment, **g** returning to home point, **h** completion of the task

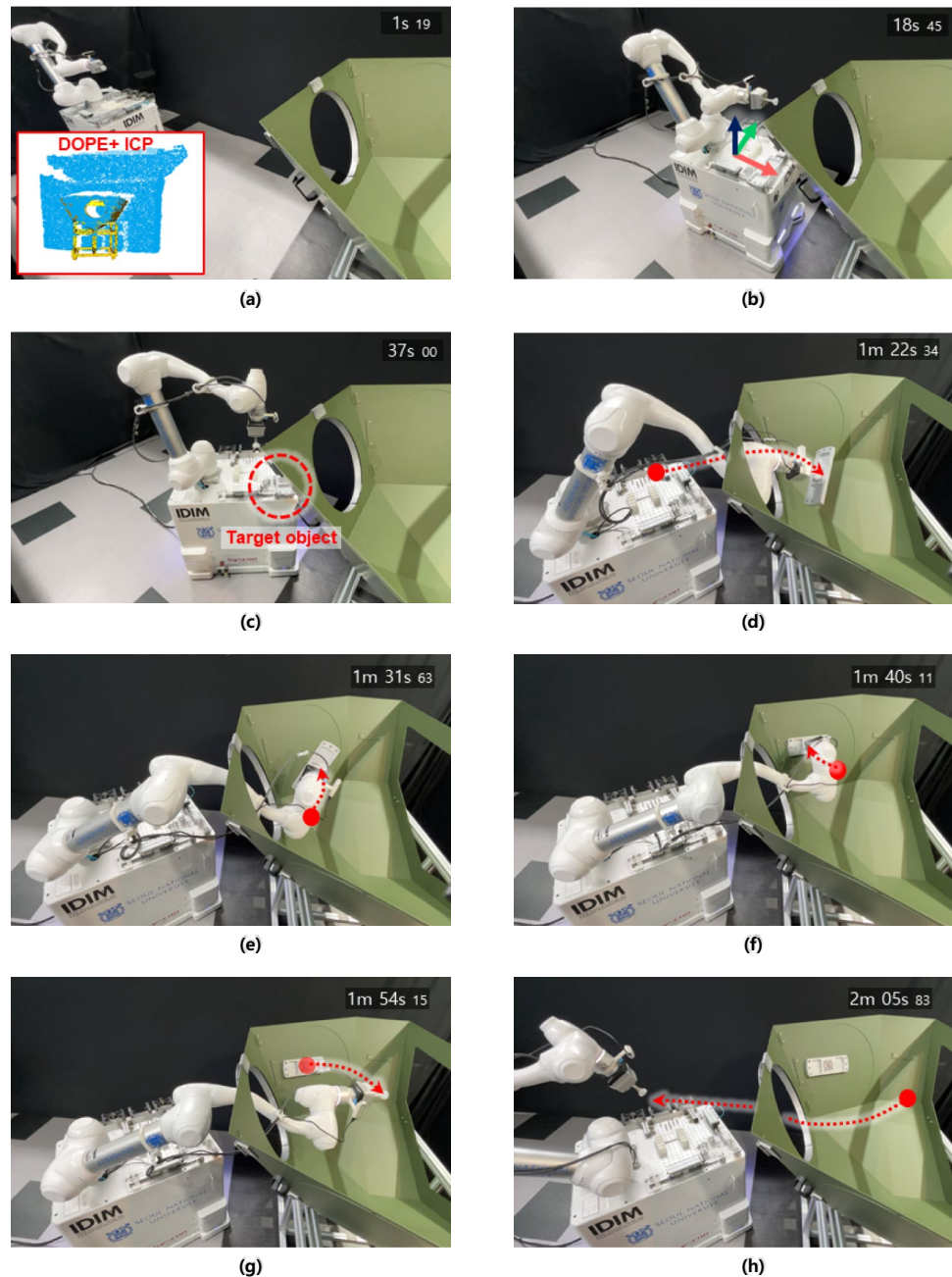


Table 4 Experimental results of the DT and the real-world systems

Metrics		Digital twin [%]	Real-world [%]
Perception error rate	Translation	4.66	–
	Orientation	2.16	–
Navigation error rate	Translation	0.95	–
	Orientation	4.49	–
Trajectory generation success rate		100	74.0
Task success rate		92.0	74.0

The results show that in the DT environment, the perception and navigation systems had translation error rates of 4.66% and 0.95%, respectively. The orientation error rates were 2.16% and 4.49%, respectively. The trajectory generation success rate and task success rates were 100% and 92.0%, respectively. However, the accumulated error of the perception and navigation system led to an inaccurate base positioning of the AMM and an 8.00% task failure rate in the DT environment.

In the real-world, due to the lack of precise ground-truth data about perception and navigation, only the trajectory generation success rate was evaluated, which is equivalent

to the task success rate. The trajectory generation success rate was 74.0%, and the task success rate was 74.0%, which were lower than those of the DT environment. This can be attributed to the presence of various noise factors that affect the perception and navigation systems, leading to increased trajectory generation errors and task failures. Nevertheless, all the algorithms and systems demonstrated robust and reliable operation. The encountered problems could be addressed with better-performing sensors, hardware, and additional sensors. Consequently, the DT-DRL system was successfully implemented, enabling the development and evaluation of all robot automation tasks without the need for actual hardware.

6 Conclusion and Future Works

This study successfully developed and evaluated a DT-DRL system for RAS to operate in a narrow passage, particularly for a nozzle dam replacement task in a steam generator compartment of nuclear power plants. All the SOTA perception, decision, and control algorithms were integrated and evaluated in the high-fidelity DT environment. The perception system for robots to recognize target objects and estimate their precise 6D pose can be developed and evaluated in the DT. The DRL with IRM-based algorithm can also be employed to train robots with various tasks such as collision-free trajectory generation policies. We demonstrated that our high-fidelity DT imitates a task environment that is almost like the real one and can train and evaluate all components of robotic automation in a short time. Although the steam generator compartment in this study was simplified, the potential benefits of the DT-DRL for RAS, particularly for AMMs, are significant. It can lead to the development of more efficient, reliable, and safe RAS that can considerably reduce the risk of accidents and damage to the robots and facilities while improving productivity. Moreover, the use of AMMs can help minimize the environmental impact of hazardous workspaces, such as nuclear power plants, by reducing the need for human workers to operate in such environments. Future research will focus on improving the perception algorithm for more accurate 6D pose estimation and developing the DT-DRL for RAS that can handle more complex tasks, such as bolting and peg-in-hole operations, in more intricate structures. The oscillation of the path by the damping effect of the robot manipulator will be stabilized by reward shaping and a more stable inverse kinematics algorithm.

Acknowledgements This work was supported by K-CLOUD project (No. 2020-Tech-06) of KOREA HYDRO & NUCLEAR POWER CO., LTD (KHNP) of the Republic of Korea and the National Research Foundation of Korea (NRF) grants funded by the Korean

government (MSIT) [grant numbers NRF-2021R1A4A2001824, NRF-2021R1C1C2008026, and NRF-2021R1A2B5B03087094].

Author contributions S-YP and CL, these authors contributed equally to this work. S-HA and HK, these authors are equally corresponding to this work. S-YP: Conceptualization, Implementation, System Integration, Experiments, Writing—Original Draft, Visualization; Cheonghwa Lee: Implementation, Experiments, Writing—Original Draft, Visualization; SJ: Implementation, Experiments; JL: Implementation; D-HK: Experiments; YJ: Conceptualization, Writing – Review; WS: Conceptualization, Writing – Review; HK: Conceptualization, Writing—Original Draft, Visualization; S-HA: Supervision, Funding acquisition, Writing—Review & Editing.

Funding Open Access funding enabled and organized by Seoul National University.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ye, Z., et al. (2023). A digital twin approach for tunnel construction safety early warning and management. *Computers in Industry*, *144*, 103783.
- Bouman, A., et al. (2020). Autonomous spot: Long-range autonomous exploration of extreme environments with legged locomotion. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE.
- Ibarz, J., et al. (2021). How to train your robot with deep reinforcement learning: Lessons we have learned. *The International Journal of Robotics Research*, *40*(4–5), 698–721.
- Ou, Y., et al. (2022). An overview on mobile manipulator in nuclear applications. 2022 IEEE International Conference on Real-time Computing and Robotics (RCAR), IEEE.
- Iqbal, J., Tahir, A.M., & R. ul Islam. (2012). Robotics for nuclear power plants—challenges and future perspectives. 2012 2nd international conference on applied robotics for the power industry (CARPI), IEEE.
- Saxena, A., et al. (2022). Technologies empowered Environmental, Social, and Governance (ESG): An Industry 4.0 landscape. *Sustainability*, *15*(1), 309.
- Ahmed, A.A., Nazzal, M.A. & B.M. Darras. (2021). Cyber-physical systems as an enabler of circular economy to achieve sustainable development goals: A comprehensive review. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 1–21.

8. Kim, H., et al. (2023). Smart factory transformation using Industry 4.0 toward ESG perspective: a critical review and future direction. *International Journal of Precision Engineering and Manufacturing-Smart Technology*, 1(2), 165–185.
9. Qin, Z., et al. (2021). Advancement of mechanical engineering in extreme environments. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 8, 1767–1782.
10. Lee, J., Dong, H., et al. (2023). Cyber-physical systems framework for predictive metrology in semiconductor manufacturing process. *International Journal of Precision Engineering and Manufacturing Smart Technology*, 1, 107–113.
11. Chen, Z., et al. (2022). Patrol robot path planning in nuclear power plant using an interval multi-objective particle swarm optimization algorithm. *Applied soft computing*, 116, 108192.
12. Zhu, D., et al. (2020). Robotic grinding of complex components: A step towards efficient and intelligent machining—challenges, solutions, and applications. *Robotics and Computer-Integrated Manufacturing*, 65, 101908.
13. Nguyen, H., La, H., (2019). Review of deep reinforcement learning for robot manipulation. 2019 Third IEEE International Conference on Robotic Computing (IRC), IEEE.
14. Lu, F., et al. (2023). Energy-efficient multi-pass cutting parameters optimisation for aviation parts in flank milling with deep reinforcement learning. *Robotics and Computer-Integrated Manufacturing*, 81, 102488.
15. Miao, X., et al. (2022). Vibration reduction control of in-pipe intelligent isolation plugging tool based on deep reinforcement learning. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 9(6), 1477–1491.
16. Zhou, Z., et al. (2022). Learning-based object detection and localization for a mobile robot manipulator in SME production. *Robotics and Computer-Integrated Manufacturing*, 73, 102229.
17. Jang, K., et al. (2021). Reactive self-collision avoidance for a differentially driven mobile manipulator. *Sensors*, 21(3), 890.
18. Chiu, J.-R., et al. (2022). A collision-free mpc for whole-body dynamic locomotion and manipulation. 2022 International Conference on Robotics and Automation (ICRA), IEEE.
19. Lim, J., et al. (2021). Designing path of collision avoidance for mobile manipulator in worker safety monitoring system using reinforcement learning. 2021 IEEE International Conference on Intelligence and Safety for Robotics (ISR), IEEE.
20. Fan, Q., et al. (2021). Base position optimization of mobile manipulators for machining large complex components. *Robotics and Computer-Integrated Manufacturing*, 70, 102138.
21. Kim, S., et al. (2010). Application of robotics for the nuclear power plants in Korea. in 2010 1st International Conference on Applied Robotics for the Power Industry, IEEE.
22. Wang, M., et al. (2021). Design, modelling and validation of a novel extra slender continuum robot for in-situ inspection and repair in aero-engine. *Robotics and Computer-Integrated Manufacturing*, 67, 102054.
23. Kim, J., et al. (2019). Unmanned aerial vehicles in agriculture: A review of perspective of platform, control, and applications. *Ieee Access*, 7, 105100–105115.
24. Kwak, J., et al. (2022). Autonomous UAV target tracking and safe landing on a leveling mobile platform. *International Journal of Precision Engineering and Manufacturing*, 23(3), 305–317.
25. Patle, B., et al. (2019). A review: On path planning strategies for navigation of mobile robot. *Defence Technology*, 15(4), 582–606.
26. Qi, S., et al. (2021). Review of multi-view 3D object recognition methods based on deep learning. *Displays*, 69, 102053.
27. Karur, K., et al. (2021). A survey of path planning algorithms for mobile robots. *Vehicles*, 3(3), 448–468.
28. Grieves, M. (2014). Digital twin: Manufacturing excellence through virtual factory replication. *White paper*, 1, 1–7.
29. Tao, F., et al. (2018). Digital twin in industry: State-of-the-art. *IEEE Transactions on industrial informatics*, 15(4), 2405–2415.
30. Kousi, N., et al. (2019). Digital twin for adaptation of robots' behavior in flexible robotic assembly lines. *Procedia manufacturing*, 28, 121–126.
31. Lee, D., et al. (2022). Digital twin-driven deep reinforcement learning for adaptive task allocation in robotic construction. *Advanced Engineering Informatics*, 53, 101710.
32. Li, H., et al. (2022). A framework and method for human-robot cooperative safe control based on digital twin. *Advanced Engineering Informatics*, 53, 101701.
33. Seo, T., et al. (2019). Survey on glass and façade-cleaning robots: Climbing mechanisms, cleaning methods, and applications. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 6, 367–376.
34. Jawad, R., et al. (2021). Autonomous mobile robot for visual inspection of MEP provisions. *Journal of Physics*, 2070, 012199.
35. Kelasidi, E., et al. (2019). Path following, obstacle detection and obstacle avoidance for thrusted underwater snake robots. *Frontiers in Robotics and AI*, 6, 57.
36. Sato, S., Song, T., & Aiyama, Y. (2021). Development of tele-operated underfloor mobile manipulator. *Journal of Robotics and Mechatronics*, 33(6), 1398–1407.
37. Xia, F., et al. (2020). Relmogen: Leveraging motion generation in reinforcement learning for mobile manipulation. arXiv preprint [arXiv:2008.07792](https://arxiv.org/abs/2008.07792).
38. Iriondo, A., et al. (2023). Learning positioning policies for mobile manipulation operations with deep reinforcement learning. *International Journal of Machine Learning and Cybernetics*, 14(9), 3003–3023.
39. Chen, W., et al. (2021). MADDPG Algorithm for Coordinated Welding of Multiple Robots. 2021 6th International Conference on Automation, Control and Robotics Engineering (CACRE).
40. Sun, C., et al., (2018). Fully Autonomous Real-World Reinforcement Learning for Mobile Manipulation. arXiv, 2021.
41. Tremblay, J., et al. (2018). Deep object pose estimation for semantic robotic grasping of household objects. arXiv preprint [arXiv:1809.10790](https://arxiv.org/abs/1809.10790)
42. Luh, Y.-P., et al. (2020). A smart manufacturing solution for multi-axis dispenser motion planning in mixed production of shoe soles. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 7, 769–779.
43. Jiang, N., Xu, J., & Zhang, S. (2020). Event-triggered adaptive neural network control of manipulators with model-based weights initialization method. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 7, 443–454.
44. Zhang, J., Yao, Y., & Deng, B. (2021). Fast and robust iterative closest point. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7), 3450–3466.
45. Makhai, A., Goins, A.K. (2018). Reuleaux: Robot base placement by reachability analysis. 2018 Second IEEE International Conference on Robotic Computing (IRC), IEEE.
46. Schulman, J., et al. (2017). Proximal policy optimization algorithms. arXiv preprint [arXiv:1707.06347](https://arxiv.org/abs/1707.06347)
47. Isaac, S. (2023) Extensions API. [cited 2023 16 March]; Available from: <https://docs.omniverse.nvidia.com/py/isaacsim/index.html>.
48. Makoviychuk, V., et al. (2021). Isaac gym: High performance gpu-based physics simulation for robot learning. arXiv preprint [arXiv:2108.10470](https://arxiv.org/abs/2108.10470).
49. Serrano-Munoz, A., et al. (2022) skrl: Modular and flexible library for reinforcement learning. arXiv preprint [arXiv:2202.03825](https://arxiv.org/abs/2202.03825).
50. Kavradi, L. E., et al. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE transactions on Robotics and Automation*, 12(4), 566–580.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.