Residual Reinforcement Learning for Robotic Assembly of Large-Scale Aerospace Components

Xiaoyou Duan[1], Guijun Ma[1*], Weibo Liu[2], Kaiqi Fang[1], and Yuzhe Wang[1]

[1] School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, P. R. China
mgj@hust.edu.cn

[2] Department of Computer Science, Brunel University of London, Uxbridge, UB8 3PH Middlesex, U.K.

Abstract. Robotic assembly of large-scale aerospace components demands millimeter-level accuracy under intermittent contacts, while collecting rich interaction data remains costly and risky. This paper presents a demonstration-guided residual reinforcement learning framework for precision assembly. A diffusion-based action-chunking policy trained from limited teleoperated demonstrations generates long-horizon nominal trajectories at a low frequency. A closed-loop residual policy optimized with PPO then adds per-step pose corrections to compensate for distribution shift and contact dynamics during the final mating phase. An action-hold sparse reward is introduced to promote stable mating rather than transient contact. Simulation experiments on KUKA KR210 industrial robot demonstrate that the proposed approach improves assembly success rate and efficiency compared with baselines, validating the effectiveness of combining imitation-based priors with closed-loop residual refinement.

Keywords: Residual reinforcement learning · Aerospace assembly · Imitation learning · Component docking · Sparse reward

1 Introduction

Automation and digitalization are increasingly transforming aerospace manufacturing, where large-scale assembly operations are being migrated from manual, skill-intensive procedures to robotized and sensor-rich work cells to improve productivity, quality consistency, and operator safety [6, 15]. Learning-based approaches have been explored in related robotic manufacturing tasks, such as quality modeling, parameter optimization, industrial time-series analytics and data-driven sensor calibration in robotic machining [9, 8, 11, 10, 4, 7]. However, robotic assembly of aerospace structural segments remains challenging. Heavy, compliant parts must be aligned to millimeter-level tolerances under intermittent contact, and collecting interaction data in production is costly and risky.

A natural approach is to leverage expert demonstrations via imitation learning (IL). Recent IL methods such as diffusion-based policies [3] and action-chunking [16] can capture long-horizon behaviors from limited demonstrations. However, demonstration-only policies often suffer from distribution shift.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

Reinforcement learning (RL) offers a complementary paradigm by optimizing a policy through interaction rewards, enabling locally corrective behaviors. However, training RL from scratch in long-horizon sparse-reward assembly is sample-inefficient and often requires extensive reward shaping [2]. Applying IL or RL alone to millimeter-tolerance assembly remains challenging.

These considerations motivate residual policy learning, which combines a base policy with a learned residual for task-specific corrections [12, 5]. By restricting exploration around demonstrated behavior, residual learning improves data efficiency while retaining long-horizon competence [1].

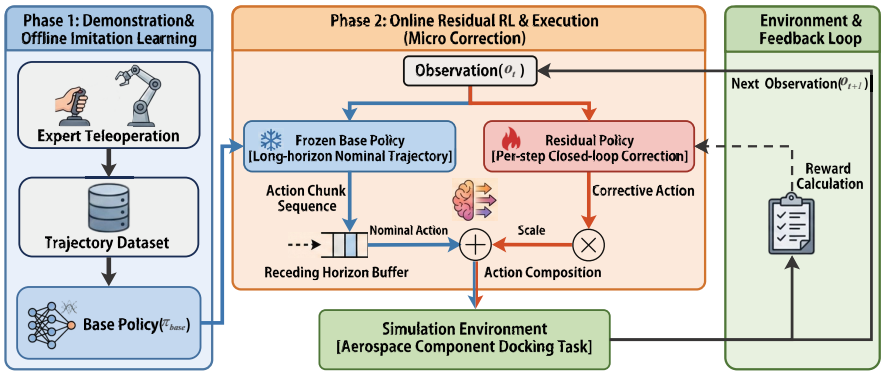

Fig. 1. Overview of the demonstration-guided residual reinforcement learning framework for large-scale aerospace component docking assembly.

In this paper, a demonstration-guided residual RL framework is presented for robotic aerospace component assembly (Fig. 1). A frozen base policy generates nominal trajectories, while a residual policy provides closed-loop corrections. An action-hold sparse reward is designed to facilitate subsequent fastening operations. Simulation experiments demonstrate that residual corrections substantially improve assembly reliability and efficiency.

2 Methodology

In this section, a demonstration-guided residual RL framework is presented for aerospace component assembly, employing a dual-timescale control architecture as illustrated in Fig. 2. A frozen diffusion-based policy π_{base} generates nominal trajectory, while a residual policy π_{res} provides closed-loop corrections.

2.1 Task Formulation

The component assembly task is formulated as a finite-horizon episodic Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \gamma)$. At time step t, the robot receives

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

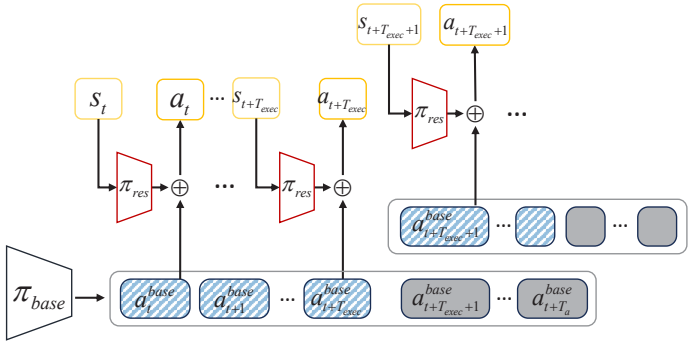

Fig. 2. Dual-timescale residual learning framework for large-scale component assembly.

an observation $\mathbf{o}_t$ (derived from state $\mathbf{s}_t$), executes an action $\mathbf{a}_t \in \mathcal{A}$, transitions to $\mathbf{s}_{t+1} \sim P(\cdot|\mathbf{s}_t, \mathbf{a}_t)$, and obtains reward $r_t = R(\mathbf{s}_t, \mathbf{a}_t)$. The goal is to learn a policy π maximizing the expected return:

$$J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T}\gamma^t r_t\right]. \tag{1}$$

The state $\mathbf{s} \in \mathcal{S} \subset \mathbb{R}^{20}$ comprises the end-effector state (position, orientation, linear and angular velocity) and the target component pose. The action $\mathbf{a} \in \mathcal{A} \subset \mathbb{R}^6$ is an end-effector pose increment, converted to joint commands via Damped Least-Squares (DLS) inverse kinematics while the policy focuses on generating task-level motion increments.

2.2 Diffusion Action-Chunking Base Policy

A base policy π_{base} is learned from expert demonstrations using diffusion-based imitation learning. Instead of predicting a single action, π_{base} outputs an action chunk of length T_a:

$$\mathbf{a}^{\text{base}}_{t:t+T_a-1} = \pi_{\text{base}}(\mathbf{o}_{t-T_o+1:t}) = \{\mathbf{a}^{\text{base}}_t, \ldots, \mathbf{a}^{\text{base}}_{t+T_a-1}\}, \tag{2}$$

where $\mathbf{o}_{t-T_o+1:t}$ denotes an observation history window. To balance long-horizon planning and responsiveness, a receding-horizon execution is adopted: only the first T_{exec} actions in the chunk are executed before replanning. After imitation learning, π_{base} is frozen and serves as a nominal trajectory generator during residual learning.

2.3 Residual Policy Learning

A residual policy π_{res} provides closed-loop corrections at every control step. The executed action is:

$$\mathbf{a}_t = \mathbf{a}^{\text{base}}_t + \alpha \cdot \pi_{\text{res}}(\mathbf{o}_t, \mathbf{a}^{\text{base}}_t), \tag{3}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

where $\mathbf{a}_t^{\text{base}}$ is extracted from the chunk sequence and $\alpha \in (0, 1]$ bounds the correction magnitude.

The residual policy is trained using Proximal Policy Optimization (PPO). The PPO loss combines a clipped surrogate objective with value-function regression and entropy regularization:

$$L_{\text{PPO}}(\theta) = -L^{\text{CLIP}}(\theta) + c_v\, \mathbb{E}_t[(V_\theta(\mathbf{o}_t) - \hat{V}_t)^2] - c_e\, \mathbb{E}_t[\mathcal{H}(\pi_\theta)], \tag{4}$$

where L^{CLIP} is the clipped probability ratio objective, V_θ is the value network, $\hat{V}_t$ is the GAE target, and $\mathcal{H}(\cdot)$ denotes entropy.

To discourage the residual policy from deviating excessively from the base prior, the PPO objective (4) is augmented with ℓ_1 and ℓ_2 regularization on the residual action mean:

$$L_{\text{total}} = L_{\text{PPO}} + \lambda_1 \|\mathbf{a}^{\text{res}}\|_1 + \lambda_2 \|\mathbf{a}^{\text{res}}\|_2^2, \tag{5}$$

where λ_1 and λ_2 are regularization coefficients.

To prevent early-stage RL updates from destabilizing the nominal behavior, two complementary mechanisms are adopted. First, residual scaling via α in (3) restricts correction magnitude. Second, near-zero initialization is applied to the residual policy output layer, such that $\mathbf{a}_t^{\text{res}} \approx 0$ at the start of training. These choices encourage the policy to initially follow π_{base} faithfully, then gradually learn local, closed-loop corrections in contact-sensitive phases as training progresses.

2.4 Action-Hold Sparse Reward for Stable Assembly

A sparse success reward with an action-hold requirement is employed to enforce stable assembly rather than transient success. Define a per-step success predicate

$$C_{\text{succ}} = (\|\mathbf{p}_{ee} - \mathbf{p}_{\text{goal}}\| < \epsilon_p) \wedge (\angle(\mathbf{R}_{ee}, \mathbf{R}_{\text{goal}}) < \epsilon_q)\,, \tag{6}$$

where ϵ_p and ϵ_q are position and orientation tolerances. The reward is given only when the success condition holds for at least K_{hold} consecutive steps:

$$r_t = \begin{cases} 1, & \text{if } C_{\text{succ}}^{(k)} \text{ holds for } k \geq K_{\text{hold}} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

This design encourages policies that reach and maintain the assembly configuration, naturally supporting subsequent fastening operations.

2.5 Training Procedure

The overall training proceeds in two stages. In Stage 1, the diffusion action-chunking policy π_{base} is trained on expert demonstrations to model the conditional distribution of action sequences given observation histories; after convergence, π_{base} is frozen. In Stage 2, trajectories are rolled out using the composed action (3) with receding-horizon chunk execution for the base action and per-step closed-loop residual corrections. Collected transitions are used to optimize π_{res} with the PPO objective (5) under the sparse action-hold reward (7).

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

3 Experiments

3.1 Simulation Environment and Control Stack

All experiments are conducted in Isaac Gym physics simulator with KUKA KR210. The task is to assemble a large structural segment to a fixed base component under millimeter-level clearance. The simulator runs at 120 Hz, while the controller outputs actions at 10 Hz. To improve data efficiency, 64 parallel environments are run on an NVIDIA RTX 4090 GPU [14], as shown in Fig. 3.

The policy outputs end-effector pose increments, tracked by a low-level PD controller for joint-space trajectory tracking [13]. For real-world deployment, a compliant controller can be added; in this work, we focus on policy evaluation.

3.2 Demonstration Collection

Expert demonstrations are collected via teleoperation using a 3Dconnexion SpaceMouse. 80 successful trajectories are recorded, each with an average length of about 100 control steps. The initial assembly configuration is randomized to improve generalization.

3.3 Implementation Details

The base imitation policy is implemented as a diffusion policy with a conditional U-Net denoiser. Given an observation history window, it predicts an action chunk of length $T_a = 32$. A receding-horizon execution is used where only the first $T_{\text{exec}} = 8$ actions of each predicted chunk are executed before replanning.

The residual policy is implemented as a stochastic Gaussian MLP with a separate actor and critic. The actor is a 2-layer MLP with 512 hidden units and SiLU activation, outputting the mean of a diagonal Gaussian distribution; a learnable log-standard-deviation parameter (initialized to -3) controls exploration. The output layer uses small-gain initialization with zero bias. A residual scaling factor $\alpha = 0.1$ bounds the correction magnitude during training.

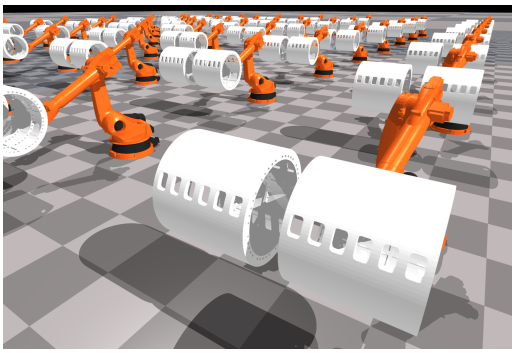

Fig. 3. Parallel simulation of the large-scale component assembly task in Isaac Gym. 64 environments run simultaneously on GPU to accelerate reinforcement learning training.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

3.4 Baselines and Ablation

Two baseline methods and one key ablation are considered. DP (Diffusion Policy, IL-only) represents the frozen diffusion action-chunking policy executed without residual corrections. PPO from Scratch is trained from random initialization without demonstrations. As an ablation, Residual-Chunk (C-Resi) applies the residual policy only at chunk boundaries rather than at every control step, reducing closed-loop feedback within each executed chunk.

4 Results and Discussion

4.1 Comparison with Baselines

Table 1 reports the main comparison results. The diffusion policy (IL-only) provides long-horizon motion but degrades in the final phase due to compounding errors during open-loop execution. PPO from scratch achieves the lowest success rate, indicating trial-and-error learning alone is insufficient. Residual learning substantially improves both reliability and efficiency by enabling closed-loop corrections while anchoring exploration around the base policy. The per-step residual achieves 98.4% success rate and the lowest average steps, demonstrating the effectiveness of combining a long-horizon prior with closed-loop refinement.

Table 1. Comparison of success rates for large-scale component assembly.

Method	Success Rate	Avg. Steps
DP (IL-only)	42.1%	153.3
PPO from Scratch	13.6%	214.5
Residual-Chunk (C-Resi)	91.1%	125.7
Residual RL (Ours)	**98.4%**	**93.2**

4.2 Ablation: Per-step vs. Chunk-level Residual Corrections

Fig. 4 compares per-step residual (Resi) with chunk-level residual (C-Resi), where the residual is updated only at chunk boundaries. Per-step residual demonstrates rapid learning and stable convergence, approaching asymptotic performance within fewer environment steps. In contrast, chunk-level residual eventually reaches above 90% success but requires more training interactions and exhibits higher variance. This behavior is consistent with the task requirements: the policy must react to contact events at the control-step timescale during the final mating phase. This ablation confirms that per-step closed-loop feedback is critical for contact-rich assembly tasks.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

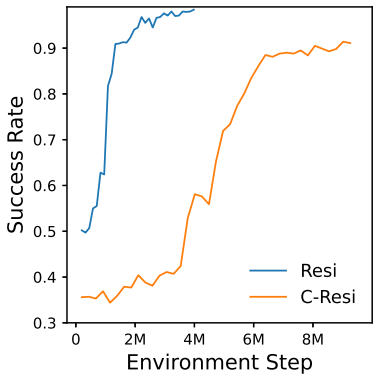

Fig. 4. Evaluation success rate during residual PPO training. Per-step residual (Resi) converges faster and more stably than chunk-level residual (C-Resi), which reaches $> 90\%$ success only after more interaction steps and with higher training variance.

5 Conclusion

This paper presented a demonstration-guided residual RL framework for large-scale aerospace component assembly. A frozen diffusion-based base policy generates nominal trajectories, while a PPO-trained residual policy provides closed-loop corrections at each control step. An action-hold sparse reward encourages stable mating. In Isaac Gym experiments with a KUKA KR210 robot, the proposed method achieves 98.4% success rate, substantially outperforming IL-only and RL-from-scratch baselines. The ablation confirms that per-step residual is crucial for contact-rich tasks.

Future work will focus on real-world validation with compliant control and force sensing, and extending the framework to multi-stage assembly sequences.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Grant 62503185; in part by China Post-Doctoral Science Foundation under Grant 2024M750991; and in part by the Post-Doctoral Project of Hubei Province of China under Grant 2024HBBHCXA010.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article

References

1. Ankile, L., Simeonov, A., Shenfeld, I., Torne, M., Agrawal, P.: From imitation to refinement-residual rl for precise assembly. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 01–08. IEEE (2025)
2. Chen, L., Shen, B., Hong, J.: A multi-task deep reinforcement learning framework based on curriculum learning and policy distillation for quadruped robot motor skill training. Systems Science & Control Engineering **13**(1), 2498914 (2025)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40

3. Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. The International Journal of Robotics Research **44**(10-11), 1684–1704 (2025)
4. Fang, J., Wang, Z., Liu, W., Zeng, N., He, Y., Cao, Y., Chen, L., Liu, X.: Learning with noisy labels for industrial time series outlier detection: A transformer-embedded contrastive learning framework. IEEE Transactions on Industrial Informatics (2025), early access, DOI: 10.1109/TII.2025.3616850
5. Johannink, T., Bahl, S., Nair, A., Luo, J., Kumar, A., Loskyll, M., Ojea, J.A., Solowjow, E., Levine, S.: Residual reinforcement learning for robot control. In: 2019 international conference on robotics and automation (ICRA). pp. 6023–6029. IEEE (2019)
6. Lettera, G., Natale, C.: An integrated architecture for robotic assembly and inspection of a composite fuselage panel with an industry 5.0 perspective. Machines **12**(2), 103 (2024)
7. Luo, X., Li, Z., Yue, W., Li, S.: A calibrator fuzzy ensemble for highly-accurate robot arm calibration. IEEE Transactions on Neural Networks and Learning Systems **36**(2), 2169–2181 (2025)
8. Ma, G., Wang, Z., Liu, W., Yang, Z., Huang, D., Ding, H.: Closed-loop parameter optimization for robotic machining using physics-informed machine learning and multiobjective optimization. IEEE Transactions on Automation Science and Engineering **22**, 22410–22422 (2025)
9. Ma, G., Wang, Z., Yang, Z., Chen, R., Liu, W., Zhang, Y., Yan, S.: A novel pairwise domain-adaptation-assisted dual-task learning approach to coprediction of robotic machining efficiency and quality in new parameter spaces. IEEE Transactions on Industrial Informatics **21**(7), 5150–5159 (2025)
10. Ma, G., Yang, X., Xu, S., Cheng, C., He, X.: Ermn: An enhanced meta-learning approach for state of health estimation of lithium-ion batteries. Journal of Energy Storage **72**, 108628 (2023)
11. Qiao, X., Xu, C., Wang, Y., Ma, G.: Sdi: A sparse drift identification approach for force/torque sensor calibration in industrial robots. Neurocomputing **620**, 129292 (2025)
12. Silver, T., Allen, K., Tenenbaum, J., Kaelbling, L.: Residual policy learning. arXiv preprint arXiv:1812.06298 (2018)
13. Song, Y., Zhang, B., Wen, C., Wang, D., Wei, G.: Model predictive control for complicated dynamic systems: a survey. International Journal of Systems Science **56**(9), 2168–2193 (2025)
14. Xue, Y., Li, M., Arabnejad, H., Suleimenova, D., Jahani, A., Geiger, B.C., Boesjes, F., Anagnostou, A., Taylor, S.J.E., Liu, X., Groen, D.: Many-objective simulation optimization for camp location problems in humanitarian logistics. International Journal of Network Dynamics and Intelligence **3**(3), 100017 (2024). https://doi.org/10.53941/ijndi.2024.100017
15. Yingke, Y., Dongsheng, L., Yunong, Z., Jie, W., Lei, X., Zhiyong, Y.: Robotic compliant assembly for complex-shaped composite aircraft frame based on gaussian process considering uncertainties. Chinese Journal of Aeronautics **37**(10), 471–482 (2024)
16. Zhao, T.Z., Kumar, V., Levine, S., Finn, C.: Learning fine-grained bimanual manipulation with low-cost hardware. In: Proceedings of Robotics: Science and Systems. Daegu, Republic of Korea (2023)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_40