Automating Tolerance-Critical Window Installation via Installer-in-the-Loop Interactive Reinforcement Learning

Zekai Jin[1], Huiguang Wang[1], Jiaduo Xing[2,3] and Yi Shao[1]

[1]Department of Civil Engineering, McGill University, Montreal, Quebec, Canada
[2]Department of Electrical Engineering, McGill University, Montreal, Quebec, Canada
[3]Advanced Micro Devices, Inc.

zekai.jin@mail.mcgill.ca, huiguang.wang@mail.mcgill.ca, jiaduo.xing@mail.mcgill.ca, yi.shao2@mcgill.ca

Abstract -

Automating tolerance-critical construction assembly, particularly prefabricated window installation, remains a persistent challenge because of millimeter-scale clearances, complex multi-surface friction, and the risk of damaging high-value hardware. In the final insertion stage, small pose errors can rapidly develop into jamming configurations that are difficult for conventional open-loop planning to resolve. Although Reinforcement Learning (RL) provides a data-driven means of acquiring contact-rich recovery behaviors, practical deployment is constrained by unsafe exploration and poor sample efficiency. This paper presents a construction-oriented simulation testbed and a sample-efficient Installer-in-the-Loop RL framework. We develop a contact-rich physics simulation that reproduces clearance-limited insertion and emergent jamming dynamics. We further propose a training protocol that integrates control sharing, in which a human installer overrides unsafe motions, with a contact-stabilized warm-start procedure. To address sparse success signals and multimodal recovery behaviors, we introduce a Generative Action-Sequence Policy based on Flow Matching and temporal abstraction. Experimental results under strictly enforced 2 mm clearances show that the proposed method achieves a 93.3% success rate. Ablation studies indicate that temporal abstraction is critical for stable credit assignment in sparse-reward settings, preventing the myopic behaviors that cause standard RL approaches to fail.

Keywords -

Construction Robotics; Interactive Reinforcement Learning; Installer-in-the-Loop; Window Installation; Generative Policy; Human-Robot Collaboration

1 Introduction

The construction industry faces persistent productivity stagnation, escalating material costs, and a severe shortage of skilled labor [1]. At the same time, demand for high-performance building envelopes and energy-efficient retrofits has increased pressure to adopt robotic automation [2]. Among on-site assembly tasks, prefabricated window installation is a strong candidate for automation because it is repetitive, physically demanding, and commonly performed under constrained site conditions. Yet despite its apparent similarity to standard industrial pick-and-place workflows, its physical execution is substantially more complex.

The core difficulty is the "last centimeter" problem. Installers must manipulate heavy, fragile payloads with millimeter-level precision into constrained rough openings. Unlike free-space transport, this terminal insertion phase is governed by uncertain multi-surface frictional interactions. As-built geometric variations in the window buck can reduce effective clearances to sub-millimeter levels. Under such tight tolerances, small pose errors can rapidly escalate into binding, wedging, or jamming states [3]. Current manual practice relies on the "tacit dexterity" of skilled workers, especially their ability to sense resistance and execute subtle contact-adaptive corrections, such as wiggling, pivoting, or partial retraction, to resolve jamming. This heuristic expertise is difficult to encode in rigid position controllers and costly to scale [4].

Recent construction robotics has largely followed two paradigms: large-scale specialized platforms, such as gantry systems, and general-purpose manipulators [5, 6]. These systems are effective for transport and coarse alignment but remain limited during contact-rich insertion. Impedance or admittance control can regulate contact forces, yet often lacks the strategic reasoning needed for non-convex jamming configurations. When a window becomes wedged, simply "pushing compliantly" may intensify the jam; successful recovery instead requires coordinated spatial adjustments that analytical controllers struggle to generate reliably under uncertainty [7].

Data-driven Reinforcement Learning (RL) offers a promising route for acquiring contact-aware policies directly from interaction, but standard algorithms such as SAC and PPO face two major barriers in construction

assembly. First, **Safety and Cost:** RL depends on extensive trial-and-error exploration. In a tolerance-critical task involving glass and finished surfaces, a single unsafe action can cause hardware damage, making "tabula rasa" learning infeasible [8]. Second, **Sample Efficiency:** Sparse-reward, long-horizon insertion may require millions of interaction steps, which is impractical for physical deployment. Moreover, friction-induced jamming often admits multimodal solutions: pivoting left or right may both be valid, while their average is not. Standard unimodal Gaussian policies therefore represent such recovery distributions poorly.

To address these gaps, this paper presents a sample-efficient Installer-in-the-Loop framework. As a specialized Interactive Reinforcement Learning paradigm, it combines human expertise with generative control. Rather than excluding the worker, we formalize the installer as both a safety supervisor and a source of high-quality recovery data. By integrating control sharing with a Generative Action-Sequence Policy, the robot can learn robust jamming recovery strategies safely and efficiently.

The main contributions of this paper are summarized as follows:

1. **Construction-Oriented Simulation Testbed:** We establish a physics-based simulation in MuJoCo that explicitly models the contact dynamics, stick-slip friction, and jamming phenomena of window installation under 2 mm clearance.
2. **Safety-Aware Installer-in-the-Loop Protocol:** We propose a control-sharing mechanism in which human interventions serve a dual purpose: ensuring operational safety during exploration and providing hard-negative samples that enable the policy to learn recovery from near-failure states.
3. **Generative Action-Sequence Policy:** We instantiate a multimodal policy using Flow Matching and Action Chunking. This architecture captures diverse human recovery strategies, such as pivoting, while enforcing temporal consistency. Crucially, chunking enables multi-step value propagation over action sequences, improving credit assignment and training stability in sparse-reward, long-horizon tasks.

2 Related Work

This section reviews robotic window installation research across three domains: building-envelope automation hardware, data-driven construction control, and Human-in-the-Loop (HIL) learning. We identify gaps in multimodal jamming recovery, temporal coherence, and operational safety that motivate the proposed generative framework.

2.1 Building-Envelope Installation Automation

Robotic automation for building-envelope installation has progressed from basic material handling toward autonomous placement of prefabricated components [2]. Existing approaches differ mainly in operational scale and control complexity, which determine their ability to handle geometric variability and tolerance-critical insertion.

Large-scale Specialized Systems: Early work primarily addressed the transport of heavy envelope elements across large workspaces. Examples include gantry-based facade installation platforms and cable-driven parallel robots for curtain wall assembly [9, 10]. These systems are effective for payload transport and coarse positioning. However, recent studies highlight an accuracy–workspace tradeoff: structural compliance and dynamic oscillations often restrict on-site positioning precision to the centimeter range [5]. Consequently, such systems typically stop before final insertion, leaving tolerance-critical mating between the window panel and opening to manual labor.

Manipulator-based Precision Assembly: To improve dexterity, another line of work investigates general-purpose manipulators integrated with perception pipelines [6, 11]. These systems can perform free-space alignment effectively, but reliability often degrades once sustained contact begins. Window installation is dominated by multi-surface, asymmetric friction between the frame and buck, where small pose errors can rapidly escalate into binding or wedging [3]. Under these conditions, traditional impedance or force-feedback control can limit contact forces but lacks the strategic reasoning needed to coordinate corrective sequences, such as wiggling or pivoting, that are required for recovery [12]. Even small perception delays can desynchronize state estimation from emergent frictional dynamics, causing failures that local feedback cannot resolve [7]. Our work addresses this gap by treating insertion not as trajectory tracking, but as contact-rich recovery requiring learned reactive policies.

2.2 Reinforcement Learning in Construction Robotics

Reinforcement learning (RL) provides a general paradigm for training autonomous systems through interaction, allowing policies to capture causal relationships among observations, actions, and contact outcomes. This capability is particularly attractive for construction, where analytical contact models are often brittle because of site variability and uncertainty.

Recent applications of deep RL in construction have shown promising results. For example, Huang et al. [11] applied RL to window panel handling and highlighted the efficiency gains of virtual demonstrations. Apolinarska et al. [13] showed that RL could compensate for tight ma-

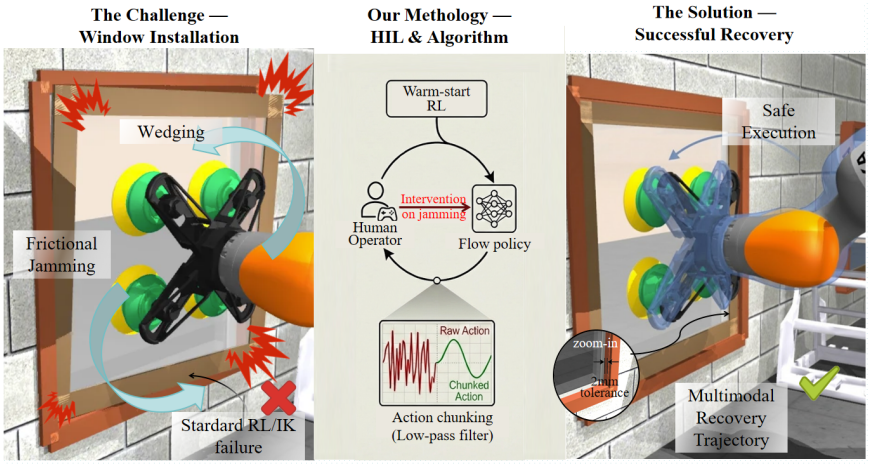

Figure 1. Installer-in-the-Loop framework. Human interventions and a generative action-sequence policy enable multimodal recovery strategies, such as pivoting and wiggling, for high-precision insertion under 2 mm clearance.

chining tolerances in timber joint assembly when classical planning failed. In the domain of safety, Duan and Zou [14, 15] advanced safety-constrained RL for human-robot collaboration. However, these studies primarily address collision avoidance and workspace safety, whereas window installation requires the agent to exploit contact dynamics to resolve jamming within millimeter-level clearances.

Two limitations remain in construction RL. First, **Sample Efficiency and Safety:** Standard RL algorithms, such as SAC and PPO, rely on extensive stochastic exploration [8]. In tolerance-critical window installation, one unsafe action can cause glass breakage or deep jamming. This makes "tabula rasa" learning prohibitively costly for deployment. Second, **Temporal Coherence:** Most RL baselines use step-wise atomic actions, which can cause high-frequency control jitter in friction-dominated tasks as the robot oscillates between conflicting commands. Although Sun et al. [16] recently applied imitation learning with action chunking to rebar cage assembly, temporal abstraction in online RL for jamming recovery remains underexplored in construction robotics [17].

2.3 Human-in-the-Loop and Generative Policies

To address the safety and efficiency bottlenecks of pure RL, Human-in-the-Loop (HIL) frameworks leverage expert knowledge to guide learning [18].

Interactive Learning: Early approaches, including DAgger and its variants, allowed experts to provide corrective labels on states visited by the robot [19, 20]. More recently, the HIL-SERL framework [21, 22] proposed a system-level recipe that combines offline demonstration bootstrapping with online safety interventions. In this paradigm, the human acts as a safety shield and overrides the robot only when it enters hazardous states. This control-sharing mechanism is especially valuable for construction assembly because it transforms rare, high-risk failure modes, which would otherwise terminate the process, into high-quality training signals [23].

Generative Policy Representations: A subtle but critical challenge in jamming recovery is the *multimodality* of valid solutions. When a window is jammed, effective recovery may involve twisting left *or* twisting right. A standard unimodal policy, such as the Gaussian policy used in most RL methods, tends to average these modes, producing a "push straight" action that can worsen the jam. To address this issue, recent robot learning studies have adopted generative models, such as Diffusion Policies [24, 25] or Implicit Behavioral Cloning [26], to represent complex action distributions. However, diffusion models typically suffer from slow inference, limiting their applicability to high-frequency contact-control loops.

Gap Identification: Our work integrates these advances. We adopt an intervention-based HIL protocol for safety [21] while replacing the standard policy architecture with a **Flow Matching** network [27]. This combination enables the policy to capture multimodal human recovery strategies while maintaining the inference speed required for real-time control. Such integration remains underexplored in construction robotics, particularly for tolerance-critical, contact-rich installation tasks.

3 Methodology

We propose a sample-efficient **Installer-in-the-Loop (IIL)** RL framework for high-precision, contact-rich construction assembly. Unlike standard RL benchmarks that assume simplified physics, the proposed framework targets the specific challenges of window installation: millimeter-scale clearances, stochastic friction, and operational safety. The framework consists of three integrated modules: (1)

a contact-rich physics testbed; (2) a safety-aware human-robot interaction protocol; and (3) a generative action-sequence policy. Figure 2 illustrates the overall architecture.

3.1 High-Fidelity Simulation Benchmark

To reduce the gap between simulation and physical deployment, we developed a contact-rich physics environment in MuJoCo [28]. The workcell contains a 7-DoF KUKA LBR iiwa 14 manipulator equipped with a Robotiq Powerpick20 vacuum gripper. The task is to insert a window panel (0.5m × 0.5m) into a buck opening with a **strictly enforced 2 mm clearance**.

To reproduce the complexity of on-site assembly, the environment explicitly models:

- **Complex Contact Dynamics:** We use MuJoCo's convex decomposition to simulate multi-point collisions. Stick-slip friction parameters ($\mu \in [0.4, 0.8]$) are introduced to induce realistic binding behaviors.
- **Construction Variability:** At the beginning of each episode, we randomize the window's initial pose (±5mm, ±5°) and the buck friction coefficients. This randomization forces the agent to learn robust, reactive policies rather than memorizing a single trajectory.

The ±5 mm and ±5° initialization is a structured staging condition designed to isolate the tolerance-critical insertion–seating bottleneck from upstream coarse-positioning uncertainty. This reflects a deployment scenario in which the window unit is pre-staged with small residual perturbations, enabling a focused evaluation of the proposed intervention and temporal-abstraction mechanisms in the contact-rich phase most relevant to jamming recovery.

Table 1. MDP specification for the window-installation benchmark.

Component	Definition
Obs. s_t	$(\mathbf{I}_{\text{wrist}}, \mathbf{I}_{\text{side}}, \mathbf{x}_t)$: 2×128^2 RGB; $\mathbf{x}_t \in \mathbb{R}^{18}$ $(\mathbf{q}, \dot{\mathbf{q}} \in \mathbb{R}^7, \mathbf{p}_{\text{ee}} \in \mathbb{R}^3, g)$
Action $\mathbf{a}_t$	$[\Delta\mathbf{p}, \Delta\mathbf{r}, g]$; $\Delta\mathbf{p}$: 0.025 m; $\Delta\mathbf{r}$: 5°; chunk h=10
Reward	Sparse binary: r_t=1 at seating, 0 otherwise
Success	pos.<2 mm; cos. sim.>0.95; vel.<1 cm/s
Horizon	600 steps (60 s @ 10 Hz)

3.2 Installer-Guided Training Protocol

"Tabula rasa" reinforcement learning is prohibitively dangerous for tolerance-critical assembly. We therefore propose a three-stage protocol that safely integrates human expertise:

3.2.1 Stage I: Offline Initialization (Behavior Cloning)

A skilled operator collects N_{demo} successful insertion trajectories through teleoperation using a **3Dconnexion SpaceMouse**. This device enables intuitive 6-DoF control, allowing the operator to demonstrate the subtle wiggling motions required for insertion. The policy is pre-trained on this static dataset $\mathcal{D}_{demo}$, initializing the agent within a safe behavioral subspace.

3.2.2 Stage II: Contact-Stabilized Warm-Up

Directly deploying an offline policy in an online environment often causes performance collapse due to distribution shift, or the "Sim-to-Sim" gap. We introduce a **Non-Updating Warm-Up** phase [29]. The agent interacts with the environment for 2 000 steps to fill the replay buffer, while gradient updates are suspended. This procedure allows the value function, or Critic, to calibrate its estimates against the true contact dynamics before the policy, or Actor, begins optimization, thereby preventing catastrophic forgetting.

3.2.3 Stage III: Safety-Aware Online Intervention

During active learning, we employ a **Control Sharing** mechanism. The agent operates autonomously by default. The human installer monitors the same RGB and proprioceptive observations available to the policy and overrides the robot **only** when observable cues indicate hazardous or non-progressing contact, such as clearance loss, stalled insertion, asymmetric wedging, unstable suction, or repeated failed re-insertion attempts. No independent hazard-detection module is used, preserving deployment realism. These intervention segments are treated as expert corrective samples and appended to the online replay buffer $\mathcal{D}_{online}$. During training, we use a mixed sampling strategy, with 50% offline and 50% online samples, to balance exploration and expert guidance. This strategy performs hard-negative mining on near-failure states.

3.3 Generative Action-Sequence Policy

Standard RL policies, such as Gaussian policies in SAC, are unimodal and step-wise. They struggle with window jamming because recovery requires coherent multi-step motion, such as "wiggling", and because valid solutions are frequently multimodal, such as pivoting left *or* pivoting right. We address these challenges using the QC-FQL framework [17].

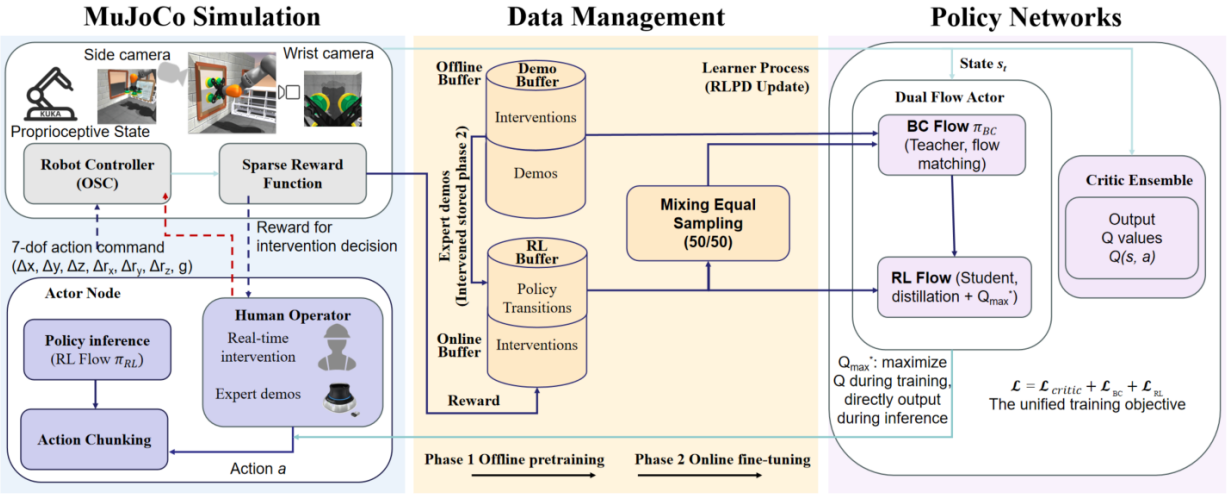

Figure 2. System architecture. The framework combines MuJoCo contact simulation, mixed replay of offline demonstrations and online interventions, and a generative policy that distills a Flow-Matching Teacher into a real-time Student policy.

3.3.1 Temporal Abstraction via Action Chunking

To eliminate high-frequency control jitter, the policy predicts a sequence of future actions $\mathbf{a}_{t:t+h}$, or a "chunk", with horizon $h = 10$, rather than a single-step action. This sequence is executed open-loop and functions as a learned temporal primitive. This design facilitates multi-step value propagation over action chunks: by evaluating the Q-value over the entire chunk $Q(s, \mathbf{a}_{t:t+h})$, the sparse success signal can be propagated across longer temporal spans, improving sample efficiency in long-horizon tasks.

3.3.2 Multimodal Modeling via Flow Matching

We employ a Teacher-Student distillation architecture to capture multimodal recovery behaviors.

The Teacher (f_ξ): A **Flow Matching** policy [27] learns a time-dependent vector field $v_t(x)$ that transports a Gaussian noise distribution p_0 to the complex target action distribution p_1. Flow Matching is adopted because it offers a practical tradeoff between expressive multimodal action modeling and efficient inference. In our setting, the teacher–student design also helps separate distribution learning from real-time control, which is important for sparse-reward contact tasks.

The Student (μ_ψ): To ensure real-time inference at 10 Hz or higher, we distill the Teacher into a one-step student network. The Student is trained to maximize the Q-value while minimizing the Wasserstein distance to the Teacher's distribution:

$$\mathcal{L}(\psi) = \underbrace{-Q_\theta(s, \mu_\psi(s, z))}_{\text{Maximize Task Reward}} + \alpha \underbrace{||\mu_\psi(s, z) - \mathbf{a}^\xi||_2^2}_{\text{Stay within Human Prior}} \quad (1)$$

This objective enables the agent to explore high-reward behaviors while remaining grounded in the safe, human-like action manifold.

3.4 Implementation Details

Perception: Visual inputs from wrist-mounted and side cameras are processed using a shared **ResNet-10** backbone. Spatial Learned Embeddings aggregate image features into a compact vector, which is then fused with proprioceptive information, including end-effector pose and velocity, through a Multi-Layer Perceptron (MLP).

Control Architecture: The high-level RL policy operates at 10 Hz and generates action chunks with horizon $h = 10$, corresponding to 1.0 second. These setpoints are tracked by a low-level Operational Space Controller (OSC) running at **500 Hz**. This hierarchical design supports compliant interaction with a stiff environment while maintaining strategic foresight. All experiments are conducted on a single NVIDIA RTX 4070 GPU.

4 Experiments and Results

4.1 Experimental Setup

We evaluate the proposed method on a precision window-installation task in a physics-based simulation environment. The task requires inserting a 0.5m × 0.5m window panel into a frame opening with 2 mm tolerance. Success follows the acceptance-aligned criterion defined in Table 1. The environment models realistic contact dynamics, including static friction, jamming, and geometric constraints that challenge manipulation policies. We com-

pare the complete method against the following baselines and ablations:

- **BC:** Behavior Cloning trained exclusively on offline demonstrations, representing a pure imitation learning baseline.
- **HIL-SERL:** A standard human-in-the-loop baseline using a unimodal Gaussian policy, Soft Actor-Critic (SAC), and step-wise actions ($h = 1$), representing the original HIL-SERL framework [21].
- **Flow (1-step TD):** An ablation of the proposed method using action chunking with $h = 1$ to isolate the effect of temporal abstraction.
- **Ours (No Interv.):** An ablation that removes human intervention to evaluate the contribution of control sharing.
- **Ours (Full):** The complete method combining Flow Q-learning with action chunking and HIL-SERL human-in-the-loop training.

4.2 Performance and Sample Efficiency

Table 2 summarizes the quantitative results. *Ours (Full)* achieves the best performance, with a **93.3%** success rate, outperforming BC (29.0%) and HIL-SERL (73.3%). Removing human intervention reduces the success rate to 36.7%, while removing action chunking causes complete failure (0.0%). These results indicate that both guided recovery data and temporal abstraction are essential. All methods reach their reported final performance within approximately 2 hours of wall-clock time.

Table 2. Performance comparison across methods. Success rates are computed during training; training time represents wall-clock time to convergence.

Method	Success Rate	Training Time
BC	29.0%	~1.0h
HIL-SERL	73.3%	~2.0h
Flow (1-step TD)	0.0%	~1.5h
Ours (No Interv.)	36.7%	~1.8h
Ours (Full)	**93.3%**	**~2.0h**

4.3 Ablation Analysis

Effect of Action Chunking: The comparison between *Ours (Full)* and *Flow (1-step TD)* shows that temporal abstraction is critical for success. Without action chunking

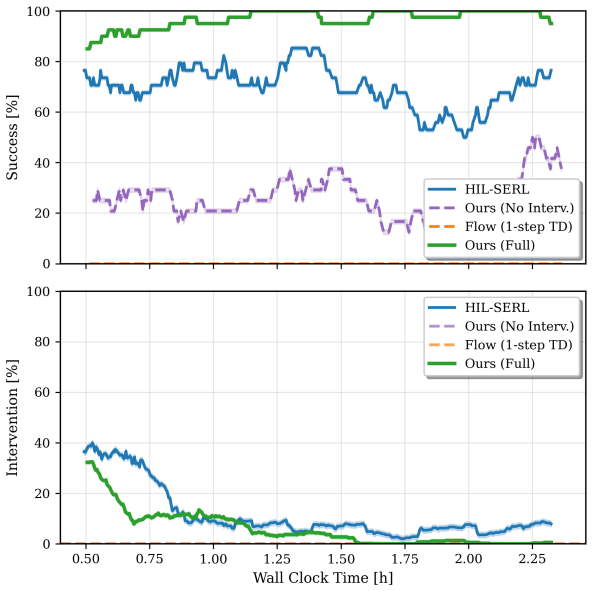

Figure 3. Training curves for success and intervention rates. Ours (Full) achieves the highest final success rate (93.3%) and the fastest decay in intervention frequency, indicating progressive autonomy.

($h = 1$), the agent fails completely, achieving a 0.0% success rate. This failure suggests that sparse-reward credit assignment, rather than control instability alone, is a major bottleneck. The single-step policy becomes "myopic", rapidly switching between conflicting corrective actions, such as pushing forward and retracting, based on instantaneous contact feedback. As a result, it fails to commit to the sustained and coordinated motions required to overcome static friction. Action chunking enforces temporal commitment, enabling the policy to execute coherent "push-and-align" primitives that are necessary for navigating geometric constraints.

Effect of Human Intervention: Removing human interventions, as in *Ours (No Interv.)*, reduces performance from 93.3% to 36.7%, corresponding to a reduction of 56.6 percentage points. This substantial gap indicates that human corrections at failure boundaries provide high-value training signals that are rarely discovered through autonomous exploration alone. The intervention mechanism enables the policy to learn from expert demonstrations of recovery behaviors, which are especially valuable in contact-rich scenarios where failure modes are diverse and difficult to explore autonomously.

Combined Effect: The complete method achieves the best performance, 93.3%, within approximately 2.0 hours, indicating that multimodal policy representation through Flow Matching, temporal abstraction through chunking, and human guidance are jointly necessary. The 20-point improvement over HIL-SERL demonstrates that Flow Matching is particularly effective for handling the non-

convex action space induced by contact constraints.

Failure Mode Analysis: Residual failures concentrate in three contact-dominated regimes: entry wedging caused by small yaw or roll errors, mid-depth frictional jamming requiring temporally extended recovery maneuvers, and unstable seating caused by residual terminal velocity. These modes indicate that temporal commitment, guided recovery data, and termination stability remain the principal bottlenecks.

4.4 Evolution of Human Guidance

A key objective of the proposed framework is to reduce human burden while maintaining learning efficiency. Figure 3 shows that the intervention frequency decreases from approximately 32% of steps to near 0% by the end of training, indicating progressive autonomy. This trend suggests that multimodal policy representation and action chunking enable efficient learning from sparse expert corrections. In absolute terms, the complete method requires approximately **12 minutes** of cumulative installer takeover time across the full training run, compared with approximately 20 minutes for the HIL-SERL baseline under the same budget. Takeovers are initiated from the same 10 Hz RGB and proprioceptive interface available to the policy, without additional sensing or automated hazard detection.

4.5 Qualitative Analysis: Jamming Recovery

We observed clear behavioral differences in jamming recovery. When the window becomes wedged diagonally, valid recovery strategies are often multimodal, such as pivoting left or pivoting right. The **HIL-SERL** baseline tends to average these modes, producing a "push straight" action that increases contact force without resolving the jam. In contrast, **Ours (Flow Matching)** exhibits mode-selecting behavior and commits to a specific recovery strategy, such as a coordinated retract-then-pivot maneuver. This capability is particularly important in tolerance-critical assembly, where the optimal recovery depends on subtle geometric variations.

5 Conclusion

This paper proposes an Installer-in-the-Loop reinforcement learning framework for tolerance-critical window installation. By combining physics-based simulation with a control-sharing protocol and a Generative Action-Sequence Policy, the framework addresses the safety and sample-efficiency challenges that limit RL deployment in construction assembly. The results demonstrate that expert interventions provide data-efficient learning signals for contact-rich recovery and that generative policies substantially outperform unimodal baselines in resolving friction-induced jamming. Although the current study is limited to simulation, contact-rich modeling and the strict safety protocol provide a necessary validation step before deployment on high-value architectural components. Physical deployment is nevertheless expected to introduce substantial sim-to-real mismatch, including unmodeled compliance, suction deformation, material-dependent friction and wear, sensing and communication delays, and site disturbances. Therefore, the effectiveness of the framework in real environments remains unvalidated. Future work will focus on bounded-budget on-site adaptation with domain randomization and hardware safety triggers.

References

[1] Filipe Barbosa, Jonathan Woetzel, and Jan Mischke. Reinventing construction: A route of higher productivity. Technical report, McKinsey Global Institute, 2017.

[2] Thomas Bock. The future of construction automation: Technological disruption and the upcoming ubiquity of robotics. *Automation in construction*, 59:113–121, 2015.

[3] Daniel E Whitney. Quasi-static assembly of compliantly supported rigid parts. *Journal of Dynamic Systems, Measurement, and Control*, 104(1):65–77, 1982.

[4] Mi Pan, Thomas Linner, Wei Pan, Huimin Cheng, and Thomas Bock. A framework of indicators for assessing construction automation and robotics in the sustainability context. *Journal of Cleaner Production*, 182:82–95, 2018.

[5] Christoph Heuer and Sigrid Brell-Cokcan. Adaptive robotic platform for the installation of refurbishment panels. *Construction Robotics*, 9(2):17, 2025.

[6] Decheng Wu, Xiaoyu Xu, Rui Li, Xuzhao Peng, Xinglong Gong, Chul-Hee Lee, Penggang Pan, and Shiyong Jiang. Curtain wall frame segmentation using a dual-flow aggregation network: Application to robot pose estimation. *Automation in Construction*, 168: 105816, 2024.

[7] Saeed Talebi, Lauri Koskela, Patricia Tzortzopoulos, Michail Kagioglou, Chris Rausch, Faris Elghaish, and Mani Poshdar. Causes of defects associated with tolerances in construction: A case study. *Journal of Management in Engineering*, 37(4):05021005, 2021.

[8] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pages 651–673. PMLR, 2018.

[9] Brandon Johns, Mehrdad Arashpour, and Elahe

Abdi. Curtain wall installation for high-rise buildings: critical review of current automation solutions and opportunities. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 37, pages 393–400. IAARC Publications, 2020.

[10] Kepa Iturralde, Malte Feucht, Daniel Illner, Rongbo Hu, Wen Pan, Thomas Linner, Thomas Bock, Ibon Eskudero, Mariola Rodriguez, Jose Gorrotxategi, et al. Cable-driven parallel robot for curtain wall module installation. *Automation in Construction*, 138:104235, 2022.

[11] Lei Huang, Zihan Zhu, and Zhengbo Zou. To imitate or not to imitate: Boosting reinforcement learning-based construction robotic control for long-horizon tasks using virtual demonstrations. *Automation in construction*, 146:104691, 2023.

[12] Christopher Rausch, Mohammad Nahangi, Carl Haas, and Jeffrey West. Kinematics chain based dimensional variation analysis of construction assemblies using building information models and 3d point clouds. *Automation in Construction*, 75:33–44, 2017.

[13] Aleksandra Anna Apolinarska, Matteo Pacher, Hui Li, Nicholas Cote, Rafael Pastrana, Fabio Gramazio, and Matthias Kohler. Robotic assembly of timber joints using reinforcement learning. *Automation in Construction*, 125:103569, 2021.

[14] Kangkang Duan and Zhengbo Zou. Safety-constrained deep reinforcement learning control for human–robot collaboration in construction. *Automation in Construction*, 174:106130, 2025.

[15] Kangkang Duan and Zhengbo Zou. Enhancing construction robot collaboration via multiagent reinforcement learning. *Journal of Intelligent Construction*, 3(2):1–16, 2025.

[16] Tao Sun, Beining Han, Jimmy Wu, Szymon Rusinkiewicz, and Yi Shao. Mobile robotic rebar cage assembly via imitation learning. *Automation in Construction*, 181:106671, 2026.

[17] Qiyang Li, Zhiyuan Zhou, and Sergey Levine. Reinforcement learning with action chunking. *arXiv preprint arXiv:2507.07969*, 2025.

[18] Christian Arzate Cruz and Takeo Igarashi. A survey on interactive reinforcement learning: Design principles and open challenges. In *Proceedings of the 2020 ACM designing interactive systems conference*, pages 1195–1209, 2020.

[19] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[20] Michael Kelly, Chelsea Sidrane, Katherine Driggs-Campbell, and Mykel J Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8077–8083. IEEE, 2019.

[21] Jianlan Luo, Charles Xu, Jeffrey Wu, and Sergey Levine. Precise and dexterous robotic manipulation via human-in-the-loop reinforcement learning. *Science Robotics*, 10(105):eads5033, 2025.

[22] Jianlan Luo, Zheyuan Hu, Charles Xu, You Liang Tan, Jacob Berg, Archit Sharma, Stefan Schaal, Chelsea Finn, Abhishek Gupta, and Sergey Levine. Serl: A software suite for sample-efficient robotic reinforcement learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 16961–16969. IEEE, 2024.

[23] William Saunders, Girish Sastry, Andreas Stuhlmueller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. *arXiv preprint arXiv:1707.05173*, 2017.

[24] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[25] Canran Xiao, Liwei Hou, Ling Fu, and Wenrui Chen. Diffusion-based self-supervised imitation learning from imperfect visual servoing demonstrations for robotic glass installation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10401–10407. IEEE, 2025.

[26] Pete Florence, Corey Lynch, Andy Zeng, Oscar A Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. In *Conference on robot learning*, pages 158–168. PMLR, 2022.

[27] Seohong Park, Qiyang Li, and Sergey Levine. Flow q-learning. *arXiv preprint arXiv:2502.02538*, 2025.

[28] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

[29] Zhiyuan Zhou, Andy Peng, Qiyang Li, Sergey Levine, and Aviral Kumar. Efficient online reinforcement learning fine-tuning need not retain offline data. *arXiv preprint arXiv:2412.07762*, 2024.