Extended Abstract

Motivation Long-horizon robotic manipulation tasks pose significant challenges because they require agents to retain historical context and procedural memory to make sequential decisions. Standard reactive policies, which operate strictly under the Markov assumption, often fall into behavioral loops and fail to recover from intermediate errors. To systematically address this, we target the RoboMME benchmark, which explicitly evaluates procedural memory in temporally extended robotics tasks. This project explores whether curiosity-driven reinforcement learning, memory-augmented imitation architectures, or continuous-time generative modeling can enable robots to successfully complete complex, multi-stage manipulation tasks, specifically focusing on the `BinFill` task.

Method We evaluate three distinct learning paradigms to tackle the sparse-reward, long-horizon nature of the benchmark. First, we implemented Proximal Policy Optimization (PPO) augmented with an Intrinsic Curiosity Module (ICM) to encourage exploration via forward-dynamics prediction errors. Second, we developed a memory-augmented Vision-Language-Action (VLA) policy utilizing a slot-based memory module and an auxiliary Past-Token Prediction (PTP) objective to reinforce temporal context retention. Finally, we implemented a Behavior Cloning (BC) model using Flow Matching, which frames action generation as an ordinary differential equation (ODE) solving process. To prevent compounding execution errors, we explicitly conditioned the Flow Matching policy on automatically segmented task subgoals.

Implementation The input to our algorithms consists of high-dimensional multi-modal data: 128×128 RGB camera images from front and wrist views, and a 15-dimensional proprioceptive state vector (joint angles, end-effector pose, and gripper status). The output is an 8-dimensional continuous action vector in joint space. For the offline imitation learning models (VLA and BC Flow Matching), we utilized a dataset of 100 human-teleoperated demonstration episodes, allocating 80 for training and 20 for validation. Online evaluation was conducted across 20 independent rollouts in the live simulation environment.

Results The three approaches yielded starkly contrasting results. Vanilla PPO and PPO+ICM achieved a 0% success rate, demonstrating severe susceptibility to reward hacking (e.g., hovering to exploit proximity rewards without grasping) and rapid intrinsic curiosity exhaustion. The VLA imitation learning approach also achieved a 0% task success rate; however, it successfully learned semantic object-directed reaching—reducing the TCP-to-object distance by an average of 0.410m and initiating gripper closure—but ultimately failed at contact-rich grasp acquisition. In contrast, the BC approach utilizing Flow Matching with subgoal segmentation successfully captured the multimodal expert distributions, achieving a 100% success rate across both the training and validation offline splits and executing smooth transitions between task phases.

Discussion Our comparative analysis highlights the fundamental bottlenecks of each paradigm. Standard RL struggles with exploration collapse and the immense difficulty of dense reward engineering. The regression-based VLA demonstrates that semantic understanding can be learned directly from limited demonstrations (<100 episodes) without handcrafted rewards, yet lacks the geometric precision required for contact-rich manipulation. Conversely, Behavior Cloning with generative Flow Matching effectively mitigates compounding regression errors and elegantly handles multi-modal trajectories. However, its purely supervised nature limits generalizability, leaving it highly sensitive to out-of-distribution physics and minor misalignments between offline datasets and live collision dynamics.

Conclusion We demonstrate that while standard RL and regression-based VLA architectures struggle with the precision and exploration demands of long-horizon robotic manipulation, modeling continuous actions via Flow Matching combined with structured subgoal segmentation provides a highly robust solution for offline imitation learning. Our findings suggest that effective long-horizon agents cannot rely on a single paradigm. Future work will focus on using pretrained memory-augmented VLAs to initialize RL fine-tuning, incorporating online DAgger (Dataset Aggregation) to address simulation-to-reality misalignments, and scaling representation models to seamlessly integrate semantic reasoning with precise, generative motor control.

Curiosity-Driven Memory-Augmented Reinforcement Learning for Adaptive Robot Tasks

Yuening Huang
Department of Computer Science
Stanford University
yueningh@stanford.edu

Yifan Geng
ICME
Stanford University
yifangen@stanford.edu

Jevon Mao
Department of Computer Science
Stanford University
jevon@stanford.edu

Abstract

This paper investigates the efficacy of memory-augmented architectures and generative imitation learning for solving long-horizon robotic manipulation tasks. We evaluate three primary approaches: Proximal Policy Optimization (PPO) with an Intrinsic Curiosity Module (ICM), a Vision-Language-Action (VLA) model with a slot-based memory module and auxiliary Past-Token Prediction (PTP), and Behavior Cloning (BC) utilizing Flow Matching with subgoal segmentation. We demonstrate that while PPO falls victim to reward hacking and the VLA struggles with contact-rich execution, the Flow Matching BC approach achieves a 100% success rate on the RoboMME `BinFill` task. We discuss the implications of reward engineering, the challenges of continuous control, and the generalization limits of behavior cloning when deployed in live simulation environments.

1 Introduction

Robots operating in complex, unstructured environments frequently encounter tasks that require them to retain and utilize information over extended time horizons. Standard robotic control policies typically operate under the Markov assumption, relying solely on immediate sensory observations to dictate actions. However, in tasks such as sequential object sorting, assembly, or multi-step navigation, this lack of historical context causes agents to fall into behavioral loops and fail to recover from intermediate errors. Developing long-horizon autonomy requires mechanisms for procedural memory—allowing the agent to track task progress, remember sequences, and dynamically adjust to evolving state conditions.

Our motivation is driven by the recently introduced RoboMME benchmark, which explicitly tests an agent's ability to reason over historical context across long horizons. The core problem we address is how to effectively train policies that can solve these temporally extended manipulation tasks. We evaluate a spectrum of approaches: curiosity-driven reinforcement learning designed to overcome sparse rewards, memory-augmented imitation learning designed to compress historical context, and continuous-time generative modeling (Flow Matching) designed to handle multi-modal expert demonstrations.

While prior works have evaluated these methods in isolation on short-horizon tasks, our novelty stems from stress-testing them against the rigorous procedural memory demands of the RoboMME benchmark. Crucially, we introduce a segmented Flow Matching approach that uniquely succeeds in this high-complexity, multi-modal continuous control setting.

Stanford CS224R 2025 Final Report

2 Related Work

2.1 Memory Evaluation in Robotics

Recent advances in robot learning have enabled strong performance on short-horizon manipulation tasks, yet many real-world scenarios require agents to retain and utilize information over extended time horizons. To systematically evaluate such capabilities, the **RoboMME** Dai et al. (2026) benchmark was introduced as a large-scale evaluation suite for memory-augmented robotic manipulation. RoboMME comprises four categories of memory—temporal, spatial, object, and procedural memory. Unlike conventional manipulation benchmarks that focus primarily on immediate perception and control, RoboMME explicitly evaluates an agent's ability to reason over historical context.

2.2 Curiosity-Driven Exploration

Curiosity-driven reinforcement learning seeks to address sparse-reward challenges by augmenting the environment reward with intrinsic motivation signals. A prominent approach is *Curiosity-driven Exploration by Self-supervised Prediction* Pathak et al. (2017), which defines intrinsic reward as the prediction error of a learned forward dynamics model. The method jointly trains an inverse dynamics model to learn a feature representation that captures controllable aspects of the environment. While curiosity-based methods have demonstrated strong exploration capabilities, they generally rely on local novelty signals and do not explicitly maintain structured memory of previously visited states.

2.3 Memory-Augmented Policies

Several recent works have explored memory-enhanced architectures for long-horizon decision making. Among recent approaches, *Learning Long-Context Diffusion Policies via Past-Token Prediction (PTP)* Torne et al. (2025) introduces an auxiliary objective designed to improve long-context reasoning in diffusion-based policies. PTP jointly trains the policy to predict both future actions and historical action tokens, encouraging internal representations to preserve information from earlier timesteps.

2.4 Generative Imitation Learning

In continuous control, human demonstrations are often multimodal, causing standard Mean Squared Error (MSE) behavioral cloning to fail. Recent paradigms like Diffusion Policy Chi et al. (2023) and Flow Matching Lipman et al. (2023) formulate action generation as a gradual denoising or ODE solving process. These generative models have become the state-of-the-art for visual imitation learning because they can elegantly represent complex, multi-modal action distributions, avoiding the pitfalls of compounding execution errors seen in traditional unimodal regression.

3 Method

We tackle the problem using three distinct learning paradigms to evaluate the tradeoffs between exploration, memory retention, and demonstration modeling.

3.1 PPO with Intrinsic Curiosity Module

The agent receives multimodal observations $s_t = (I_t^{\text{front}}, I_t^{\text{wrist}}, p_t)$ consisting of two RGB camera views and a proprioceptive state vector. Visual observations are encoded using a shared ImageNet-pretrained ResNet-18 backbone, while proprioceptive inputs are embedded separately and fused into a shared latent representation:

$$\phi(s_t) = \big[\, \pi^{\text{front}}(h_\phi(I_t^{\text{front}})) \,\|\, \pi^{\text{wrist}}(h_\phi(I_t^{\text{wrist}})) \,\|\, g_\phi(p_t) \,\big], \tag{1}$$

To improve exploration under sparse supervision, we incorporate an intrinsic curiosity module (ICM) Pathak et al. (2017). The training reward decomposes as

$$r_t = r_t^{\text{ext}} + \eta r_t^{\text{int}}.$$

The intrinsic reward is defined as the forward-dynamics prediction error $r_t^{\text{int}} = \frac{1}{2}\|f_\omega^{\text{fwd}}(\phi(s_t), a_t) - \phi(s_{t+1})\|_2^2$, which encourages the agent to visit states whose dynamics it cannot yet predict.

3.2 VLA Imitation Learning with a Memory Module

We train a memory-augmented Vision-Language-Action (VLA) policy using imitation learning from RoboMME demonstrations. The model receives visual observations, proprioceptive robot states, natural-language task instructions, and a persistent memory state carried across timesteps within each episode. Images and task instructions are encoded using frozen CLIP vision and text encoders, respectively, and projected into a shared latent space. Proprioceptive observations, consisting of end-effector pose, joint positions, and gripper state, are encoded with a lightweight MLP.

The encoded visual, state, and language features are fused into a timestep representation x_t. To support long-horizon reasoning, we introduce a slot-based memory module,

$$M_t \in \mathbb{R}^{K \times D},$$

where $K = 8$ is the number of memory slots and $D = 512$ is the latent dimension. At each timestep, the model writes the current representation into memory using an attention-weighted residual update:

$$M_{t+1} = \mathrm{LayerNorm}(M_t + \alpha x_t).$$

A separate memory read operation retrieves task-relevant historical context, which is fused with the current observation representation. The policy head then predicts an 8-dimensional continuous joint-space action.

In addition to the standard behavior cloning objective, we introduce an auxiliary Past-Token Prediction (PTP) loss. Given the past observation sequence, the PTP head predicts the corresponding historical actions, encouraging the latent representation to capture temporal structure. The final training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{BC}} + \lambda \mathcal{L}_{\mathrm{PTP}}.$$

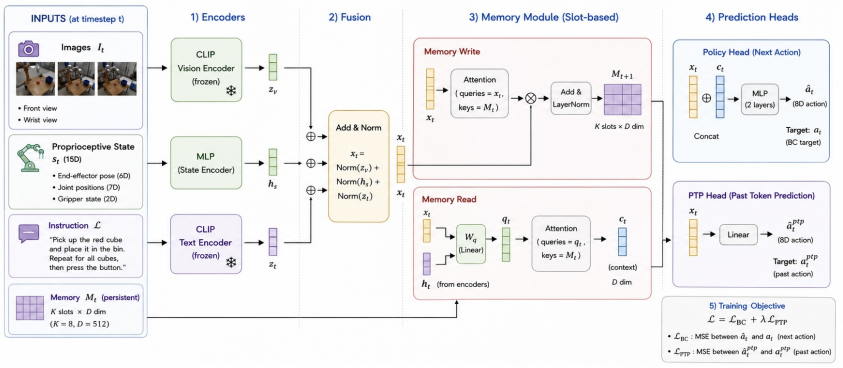

Figure 1: Offline Training Memory-Augmented VLA Model.

3.3 Behavior Cloning with Flow Matching and Subgoal Segmentation

We also implemented a continuous-time generative imitation learning framework based on Flow Matching Lipman et al. (2023). Instead of directly predicting an action vector, the neural network predicts a vector field $v_\theta(a_t|s_t)$ that dictates how to flow a sample from a base noise distribution to the target expert action distribution.

Because the `BinFill` task requires a long sequence of distinct phases (reach, grasp, lift, place), we augment the Flow Matching policy with **subgoal segmentation**. The full expert demonstration episode is automatically segmented into discrete subgoals using heuristic velocity and gripper state thresholds. To manage the complexity of these long-horizon sequential tasks, we explicitly condition the flow matching policy on these structured subgoals.

The robot's input representation consists of a 16-dimensional proprioceptive vector—comprising absolute joint angles, end-effector pose, and gripper status—temporally stacked over the last H frames to provide necessary historical context. Alongside this state history, the policy is conditioned on an active, dynamically parsed subgoal embedding that provides semantic and spatial priors, specifically the required action type (e.g., pick, put, press), target object color, and spatial pixel

location. This allows the Flow Matching model to learn high-fidelity, multimodal action distributions for each specific phase of the task, significantly reducing compounding errors over the long horizon.

During action generation, the conditional flow matching model utilizes Euler integration over K steps. It iteratively refines a base noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_8)$ guided by the learned velocity field v_θ, ultimately producing a normalized 8-dimensional joint-space action. This action is then de-normalized into 7 absolute joint targets and 1 gripper command to be sent to the robot's controller.

4 Experimental Setup

We mainly focus on one temporal memory task from the RoboMME benchmark that require accurate counting and timely termination: `BinFill`. This task requires the robot to place a specified number of colored cubes into the bin and press a button to finish.

For the offline imitation learning experiments (VLA and BC Flow Matching), we utilized a dataset of 100 human-teleoperated demonstration episodes per task. For our final Flow Matching model, we split the data into 80 episodes for training and 20 episodes for validation.

Our primary metrics are the **Success Rate** (percentage of episodes where the task is fully completed) and the **Average Evaluation Steps** to completion. Models were evaluated over 20 independent online rollout episodes in the live simulation environment.

4.1 PPO + ICM

We implement both vanilla PPO and PPO+ICM as online actor critic baselines for the sparse reward, long horizon RoboMME `BinFill` task. In the vanilla PPO setup, the agent is trained using only the extrinsic task reward, with parallel rollouts, generalized advantage estimation, entropy regularization, and the standard clipped PPO objective. The policy takes two RGB camera views and proprioceptive state as input, encodes the visual observations with a shared ImageNet pretrained ResNet-18 backbone, embeds the proprioceptive features separately, and fuses these representations before passing them into MLP policy and value networks. In the PPO+ICM setup, we add an intrinsic curiosity module to encourage exploration under sparse supervision. The ICM trains a forward dynamics model to predict the next latent state from the current latent state and action, as well as an inverse dynamics model to infer the action between consecutive latent states. This curiosity module shares the encoder with the policy, allowing the representation to be shaped jointly by the control objective and the auxiliary exploration objective.

4.2 VLA Imitation Learning

Our VLA policy receives an episode-level task instruction, RGB observations from two camera viewpoints, and a 15-dimensional proprioceptive state. The policy outputs an 8-dimensional continuous joint-space action at each timestep.

After training via imitation learning on demonstration trajectories, we evaluate the learned policy through online rollouts in the RoboMME environment. We report both task-level and subtask-level performance metrics, including episode success rate, subtask completion progress, TCP-to-object distance, object manipulation statistics, and gripper behavior. These additional diagnostics provide a more detailed understanding of policy behavior and failure modes beyond the binary success metric.

4.3 BC with Flow Matching and Subgoal Segmentation

The policy receives two primary inputs at each timestep. The first input is a stacked proprioceptive state $\hat{\mathbf{s}}_t \in \mathbb{R}^{16H}$, formed by concatenating the last H robot-state frames (default $H = 2$). Each frame $\mathbf{s}_t \in \mathbb{R}^{16}$ is composed of:

- Joint angles $\mathbf{q}_t \in \mathbb{R}^7$.
- End-effector pose $\mathbf{e}_t \in \mathbb{R}^6$ (Cartesian position plus continuous unwrapped roll-pitch-yaw).
- Gripper finger widths $\mathbf{g}_t \in \mathbb{R}^2$.
- A binary gripper-close indicator $⊮_{\text{gc}} \in \{0, 1\}$.

Missing history frames at the start of each subtask are filled by repeating the first available frame from the expert demonstration setups. All 16 dimensions are normalized using per-dimension means and standard deviations fitted on the training set.

The second input is a structured subgoal parsed from the environment's `grounded_subgoal_online` string. This consists of an action-type index $a \in \{0, 1, 2\}$ (pick/put/press), a color index $c \in \{0, 1, 2, 3\}$ (red/green/blue/none), and a pixel location $(p_y, p_x) \in [0, 1]^2$ normalized by the 256 px image resolution.

The output is an 8-dimensional de-normalized joint-space action $\mathbf{a}_t \in \mathbb{R}^8$, including 7 absolute joint-angle targets (in radians) and 1 gripper command (-1 close, $+1$ open). This action is generated by running $K = 20$ Euler steps of the flow-matching ODE from a standard-normal noise sample.

In addition to live simulation evaluations, we conduct offline rollouts to construct HDF5 trajectory files. These files are then replayed in the simulator to provide a more precise assessment of the policy.

5 Results

5.1 Quantitative Evaluation

Table 1 summarizes the performance of the three distinct algorithms on the `BinFill` task based on 20 evaluation rollouts. By shifting to Flow Matching with subgoal segmentation, we successfully captured the multimodal nature of the expert's approach trajectories, achieving a 100% success rate on both the training and validation splits.

Table 1: Performance Comparison on RoboMME `BinFill` (20 Evaluation Rollouts)

Method	**Final Shaped Reward**	**Eval Success Rate**
PPO + Extrinsic Rewards	10.3	0%
PPO + ICM	11.5	0%
VLA + Memory + PTP	N/A	0%
BC + Flow Matching (Subgoals)	N/A	**100%**

5.2 Qualitative Analysis

PPO and Reward Hacking: Despite utilizing a highly engineered dense reward signal, PPO variants failed to achieve meaningful task completion. Vanilla PPO exploited imperfections in the dense reward, learning shortcut behaviors such as hovering near an object to maximize proximity rewards while intentionally avoiding grasping to minimize the risk of dropping penalties. PPO+ICM improved early exploration, but the intrinsic curiosity signal rapidly decayed as the forward dynamics model became accurate, leading to curiosity exhaustion before any meaningful subgoal was solved (see Figure 3).

VLA Representation Learning: The memory-augmented VLA policy did not achieve successful task completion during online evaluation. However, it learned meaningful object-directed behavior, reducing the average TCP-to-object distance by $0.410\,\mathrm{m}$ and consistently closing the gripper near target objects. These results indicate successful semantic understanding and coarse reaching behavior. Nevertheless, the policy never achieved a successful grasp, with the minimum TCP-to-object distance remaining approximately $0.06\,\mathrm{m}$. This suggests that while the frozen CLIP-based visual representation provides sufficient information for object localization, it may lack the spatial precision required for contact-rich manipulation. Combined with compounding action prediction errors from behavior cloning, this likely prevents the precise alignment needed for successful grasp acquisition and task completion.

BC with Flow Matching and Subgoal Execution: Unlike the regression-based VLA, the continuous-time generative formulation of Flow Matching successfully captured the multimodal nature of the expert's approach trajectories. By explicitly conditioning the vector field prediction on segmented subgoals, the model effectively localized its actions to the current phase of the task (e.g., reaching, grasping, or lifting). Qualitatively, this segmentation prevented the policy from becoming "lost" over the long temporal horizon and significantly curtailed the compounding execution errors that

Table 2: Online evaluation results of the memory-augmented VLA policy on `BinFill`

Metric	Value
Initial TCP-to-Object Distance (m)	0.544
Minimum TCP-to-Object Distance (m)	0.063
TCP-to-Object Improvement (m)	0.481
Initial Gripper Opening	0.040
Minimum Gripper Opening	0.008
Gripper Closure	0.033

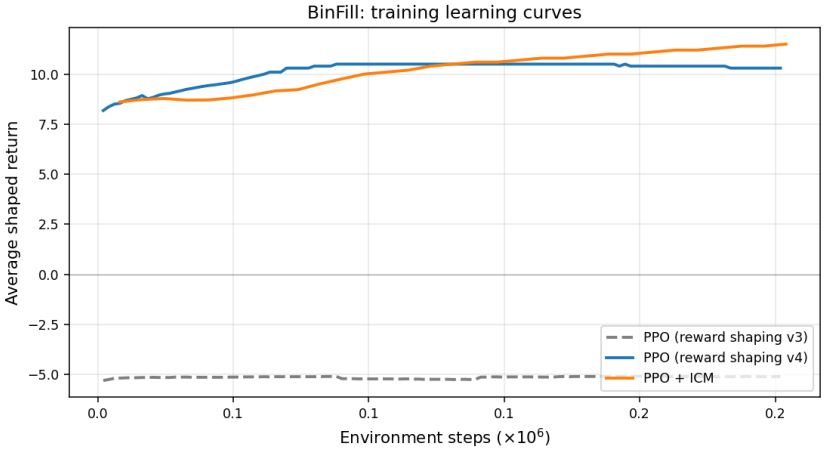

Figure 2: Learning curves comparing vanilla PPO variants and PPO augmented with ICM.

typically plague behavior cloning. The robot exhibited smooth, confident transitions between free-space movement and contact-rich manipulation, yielding a flawless success rate on validation states. However, qualitative observation during live deployment highlighted its sensitivity to out-of-distribution physics; the policy's precision is tightly coupled to the training data, making it vulnerable to the slight simulation misalignments between the offline dataset and live collision dynamics.

6 Discussion

While behavior cloning with Flow Matching achieved a 100% success rate on the training and validation splits, we observed a critical limitation during live evaluation: *simulation misalignment*. There is a slight physical misalignment between the offline training states and the live simulation environment's collision dynamics.

Furthermore, while the generative formulation successfully mitigates the compounding errors typically seen in long-horizon MSE regression, its purely supervised nature limits generalizability. The BC Flow Matching model works flawlessly on seen scenarios and validation start states, but it cannot guarantee success on radically unseen states or perturbations not covered in the 80 training episodes. Conversely, reinforcement learning approaches such as PPO theoretically offer better generalization through online interaction, but remain fundamentally bottlenecked by the difficulty of dense reward engineering in multi-stage manipulation tasks.

The memory-augmented VLA model occupies an interesting middle ground between reinforcement learning and imitation learning. Unlike PPO, it learned meaningful task semantics directly from demonstrations, consistently exhibiting object-directed behavior without requiring handcrafted rewards. However, despite successfully approaching target objects and initiating grasp attempts, the policy never completed a subtask. This suggests that semantic understanding alone is insufficient for contact-rich manipulation. Moreover, the relatively small demonstration dataset ((<100) training episodes) is likely insufficient for training a VLA policy from scratch, particularly for long-horizon

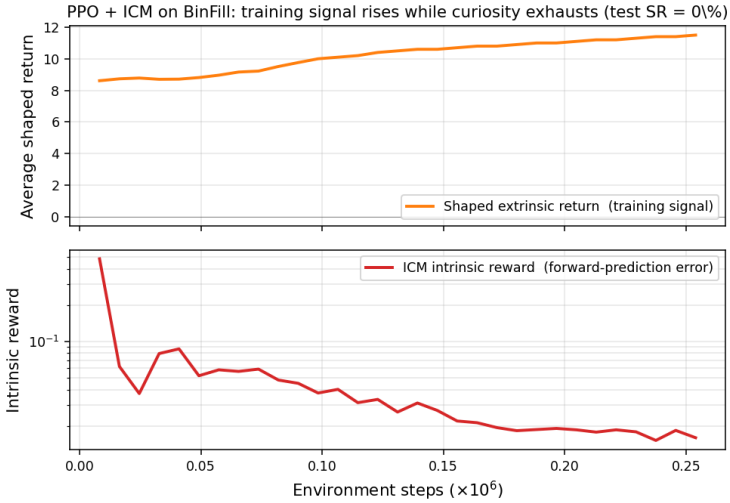

Figure 3: PPO + ICM training dynamics on BinFill. Shaped extrinsic return steadily increases while intrinsic curiosity reward rapidly decays during training.

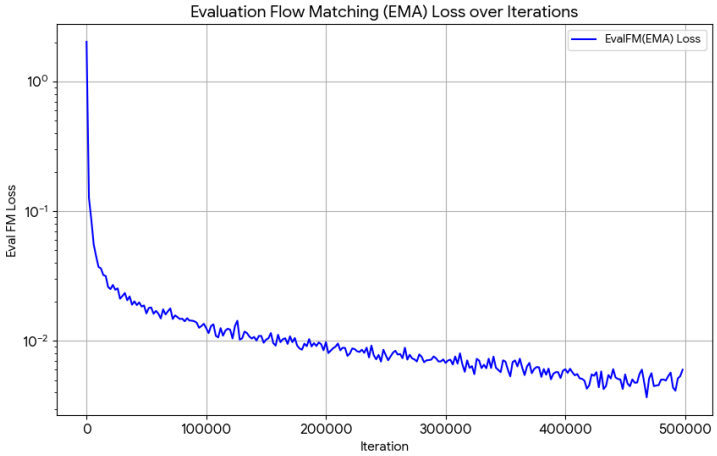

Figure 4: Evaluation Flow Matching (EMA) loss curve over 500,000 training iterations on the `BinFill` validation set. The logarithmic y-axis highlights the rapid initial convergence during the first 10,000 iterations, followed by a steady, stable refinement down to a final evaluation loss of approximately 0.005, indicating highly accurate modeling of the expert action distribution.

manipulation tasks that require both semantic reasoning and precise motor control. We hypothesize that limitations in geometric precision, limited demonstration coverage, and compounding errors from regression-based behavior cloning jointly prevent the fine-grained control required for successful grasp acquisition and long-horizon task completion.

Taken together, our results highlight a fundamental challenge in long-horizon robotic learning. Reinforcement learning struggles with exploration and reward specification, while supervised imitation learning struggles with distribution shift and precision-sensitive manipulation. Future work would benefit from combining the strengths of these distinct paradigms. For example, pretrained memory-augmented VLA policies could serve as a robust initialization for reinforcement learning fine-tuning. Additionally, incorporating stronger geometric representations or employing generative action models would better capture multimodal manipulation behaviors. More broadly, our findings suggest that

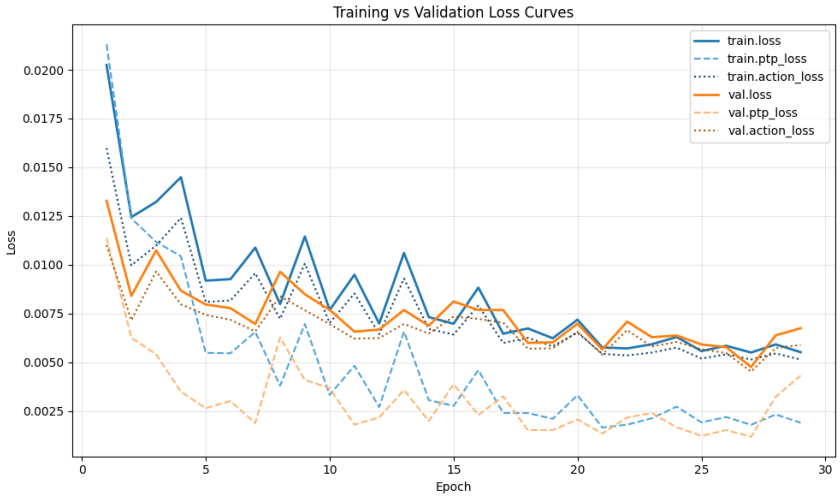

Figure 5: Training and validation losses for `BinFill`.

rather than relying on a single learning paradigm, effective long-horizon robotic agents will require the seamless integration of semantic reasoning, persistent memory, and interactive online adaptation.

7 Conclusion

In this project, we demonstrated that standard RL (PPO) is highly susceptible to reward hacking in long-horizon robotic tasks, and regression-based VLA models trained from scratch with frozen CLIP encoders struggle with the precision required for contact-rich manipulation. However, our implementation of Behavior Cloning using Flow Matching, augmented with subgoal segmentation, proved highly effective, achieving a 100% success rate on the `BinFill` task. Future work will focus on integrating online DAgger (Dataset Aggregation) to address the generalizability limitations of pure behavior cloning, and scaling our VLA architecture using pretrained foundational weights to improve sample efficiency.

8 Team Contributions

- **Yuening Huang:** Formulated the core problem, implemented the Behavior Cloning model with Flow Matching and subgoal segmentation, conducted the final 80/20 data split evaluations, and led the drafting of the final report.
- **Yifan Geng:** Developed the Memory-Augmented VLA architecture, implemented the auxiliary Past-Token Prediction (PTP) loss, conducted the comprehensive literature review, and prepared the project poster.
- **Jevon Mao:** Implemented the PPO and PPO+ICM baselines, designed the dense reward shaping functions, managed the parallelized environment infrastructure, instrumented the experiments, and generated the reinforcement-learning training curves

Changes from Proposal Originally, our proposal focused entirely on reinforcement learning (PPO and SAC). However, after observing severe reward hacking and exploration collapse, we pivoted our effort toward imitation learning, specifically VLA architectures and generative Flow Matching, to better leverage the available expert demonstrations.

Generative AI Usage Statement Generative AI tools(claude code, Gemini) were utilized during this project strictly for proofreading the final report for grammatical clarity, for assisting in setting up data pipeline, and for assisting in debugging PyTorch tensor dimension mismatch errors during the implementation of the Flow Matching ODE solver. No AI tools were used to generate the core conceptual algorithms, experimental results, or primary codebase.

References

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. 2023. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. In *Robotics: Science and Systems (RSS)*.

Yinpei Dai, Hongze Fu, Jayjun Lee, Yuejiang Liu, Haoran Zhang, Jianing Yang, Chelsea Finn, Nima Fazeli, and Joyce Chai. 2026. RoboMME: Benchmarking and Understanding Memory for Robotic Generalist Policies. arXiv:2603.04639 [cs.RO]

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2023. Flow Matching for Generative Modeling. In *International Conference on Learning Representations (ICLR)*.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. 2017. Curiosity-driven Exploration by Self-supervised Prediction. arXiv:1705.05363 [cs.LG] `https://arxiv.org/abs/1705.05363`

Marcel Torne, Andy Tang, Yuejiang Liu, and Chelsea Finn. 2025. Learning Long-Context Diffusion Policies via Past-Token Prediction. arXiv:2505.09561 [cs.LG]