ICRA2026 Late Breaking Results Poster presented at
2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Exploring History-Aware Online Actor-Critic for Smart Manufacturing Tasks in the RICAIP Testbed

1st Tomas Horelican
CEITEC, Brno University of Technology, Brno, Czech Republic

***Abstract*—As manufacturing capabilities advance to greater autonomy, interest is increasingly directed toward versatile agents capable of performing complex tasks. Recently, learning-based approaches have shown more rapid progress compared to classical methods. While these advancements are enabled by the offline setting of Imitation Learning (IL), transfer to pure online exploration Reinforcement Learning (RL) remains less explored. This work experiments with a simple extension to the standard Markovian MLP policy by explicitly encoding a history of states using a tiny transformer model.**

***Index Terms*—reinforcement learning, transformer, locomotion, manipulation, control.**

I. Introduction

Recent breakthroughs in offline Reinforcement Learning (RL) are heavily supported by multimodal parametrization of diffusion policies together with the capacity of transformer networks to encode causal relationships in sequence generation [1]. These formulations provided better solutions for fitting multimodal datasets of collected observation/action trajectories using a simple regression loss in the Behavior Cloning (BC) objective of Imitation Learning (IL). Latest methods in the online regime also adapt the diffusion probability model but still focus predominantly on a simple single-step MLP architecture for the policy [2], which implicitly assumes the Markovian property. Additionally, as opposed to regressing a fixed pre-collected sequence, there is no clear method for continuous collection and batch retrieval of causally linked future horizon samples from an online replay buffer. This work will explore relaxing the Markov property and reparametrizing the policy with explicit history dependence. Evaluations were performed using three algorithms: SAC, TD3, and DIPO.

II. Policy Overview

The base MLP policy, parametrized as $a_t \sim \pi(\cdot \mid o_t)$, was adapted identically from [2]. With 3 hidden layers ($d : 512, 256, 128$) and *ELU* activations for SAC and TD3. For DIPO, the diffusion timestep is encoded with a sinusoidal positional embedding ($d : 256$) and passed through one hidden layer ($d : 1024$), which is fed into the conditional predictor with 3 hidden layers ($d : 1024, 512, 256$), using *Mish* activations in both stacks. Leveraging insights from [3],

The work was supported by the infrastructure of RICAIP that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857306 and from MinistryofEducation, Youth and Sports under OP RDE grant agreement No CZ.02.1.01/0.0/0.0/17_043/0010085.

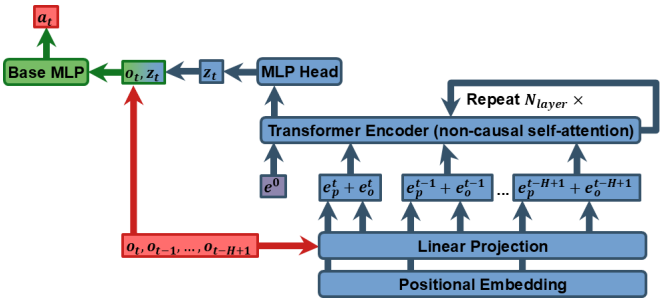

Fig. 1. Policy architecture.

a tiny non-causal self-attention transformer encoder ($N_{layer} : 2, N_{head} : 2, d_{model} : 128$, and $d_{FFW} : 512$ with *GELU* activation) built from the GPT implementation in [4] was adapted with the pre-pended extra token. Inputs are linearly embedded and learnable positional embeddings are added. Like ViT, the final head projects only the extra token embeddings, but the output is used as a latent feature vector ($d : 256$), proposing a Latent Temporal Transformer (LTT). This is concatenated with the base MLP single-state input, reparametrizing the policy as $a_t \sim \pi(\cdot \mid o_t, o_{t-1}, \ldots, o_{t-H+1})$ with history length H (see Fig. 1). Encoding is performed only once before diffusion iterations in DIPO. Special GPT-2/3 initialization and configuration is applied only to the LTT encoder network stack. Initializing linear layers and embeddings with $\mathcal{N}(0, \frac{1}{d_{model}})$, residual projections with $\mathcal{N}(0, \frac{0.02^2}{2 \cdot N_{layer}})$, setting weight decay only for 2D parameters, using the cosine-with-warm-up learning rate schedule, and optimizer betas: (0.9, 0.95). Additive biases in linear layers and layer normalizations are disabled.

III. Simulation Results

A single RTX 4090 was used for training in 6 diverse environments, covering a wide range of complex continuous control tasks, from whole-body control, maze navigation, to object manipulation, with dense and sparse rewards. To evaluate different parametrization schemes, three popular off-policy algorithms were assessed:

- stochastic squashed-Gaussian SAC with learnable coefficient α,
- deterministic TD3,
- multimodal diffusion DIPO using the target action modification from [2].

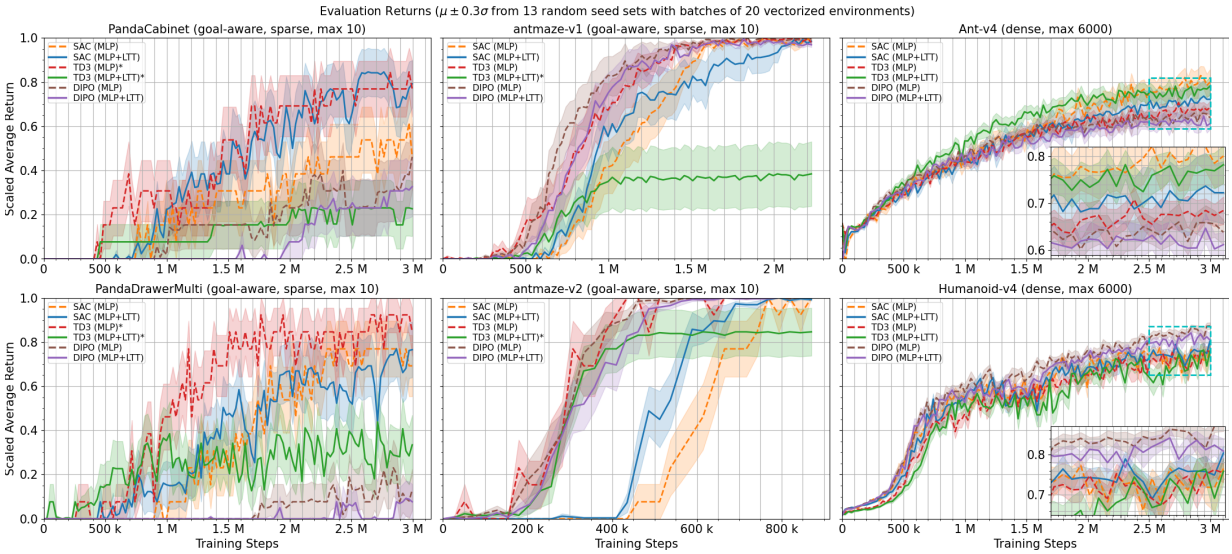

Fig. 2. All evaluation returns.

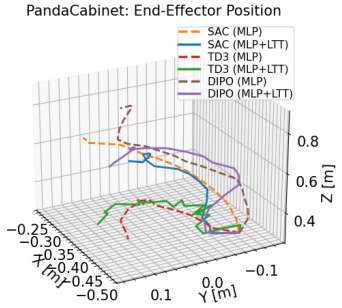

Fig. 3. Shortest successful trajectories.

An identical hyperparameter setup, scalar double-Q learning, and the single-state novelty intrinsic reward from [2] were used in all combinations. TD3 policy updates were not delayed for *AntMaze* tasks, as it resulted in generally worse performance. The first 32 exploration steps always used random warm-up sampling from the action space. Vectorized environments with identical settings were used for training (256) and evaluation (20) episodes. 13 starting random seeds were chosen such that there was no overlap across the vectorized environments and evaluation episodes between training sessions. A seed was always applied for the random warm-up action space sampling, the initial training reset, and each evaluation episode reset. Standard MuJoCo *Ant* and *Humanoid* control tasks serve to verify general applicability in dense reward settings. Environments from the *D4RL* and *panda-gym* suites also verify behavior in goal-aware sparse reward settings and were adapted directly from [2]. The *AntMaze-v2* task was adjusted with both goals being symmetrically spaced and giving identical rewards. All combinations were always trained for 3 million steps.

As seen in Fig. 2, even with the highly compact encoder module, potential for utilizing the explicit information is visible with the SAC baseline in sparse reward exploration. In contrast, TD3 had worse compatibility (several seeds maintained zero gradients with sparse rewards) and realized improvements only in dense settings. All DIPO implementations remained comparable. As a qualitative example (see Fig. 3), the explicit policies more naturally avoided a redundant upwards motion when retreating to open the cabinet door in *PandaCabinet*.

IV. Conclusion and Future Work

This work showed there are realizable improvements in formulating explicit history dependence for online RL policies. More complete future work will extend experiments with encoder ablations, network size and history length scaling, real-world partial observability, dimension reduction, and explore switching to a pure transformer policy to study the utility of causal horizon sequence predictions in online learning.

References

[1] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, vol. 44, no. 10-11, pp. 1684–1704, 2025. [Online]. Available: https://doi.org/10.1177/02783649241273668

[2] Z. Li, R. Krohn, T. Chen, A. Ajay, P. Agrawal, and G. Chalvatzaki, "Learning multimodal behaviors from scratch with diffusion policy gradient," in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 38 456–38 479.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[4] A. Karpathy, "mingpt," Available at https://github.com/karpathy/minGPT, 2026, accessed: 2026-04-08. [Online]. Available: https://github.com/karpathy/minGPT