Decision Transformer with Flow Matching for Compounding Error Mitigation

Chang-Hun Ji
Future Convergence Engineering
Korea University of Technology and Education
Cheonan, South Korea
koir5660@koreatech.ac.kr

Asel Nurlanbek kyzy
Future Convergence Engineering
Korea University of Technology and Education
Cheonan, South Korea
aselbaekki@koreatech.ac.kr

Youn-Hee Han
Future Convergence Engineering
Korea University of Technology and Education
Cheonan, South Korea
yhhan@koreatech.ac.kr

***Abstract*—Offline reinforcement learning learns policies solely from pre-collected datasets without real-time environment interaction, addressing the cost and safety issues that online reinforcement learning faces in real-world applications. Among various offline reinforcement learning methods, the Decision Transformer (DT) is a representative algorithm that leverages the Transformer architecture to model long-term dependencies in sequential decision-making. However, autoregressive architecture of DT inherently suffers from the high compounding error problem, in which initial errors rapidly propagate and accumulate over time. To address this limitation, this paper proposes a novel architecture that integrates Flow Matching (FM) with DT to mitigate the high compounding error problem. Experimental results in a simple one-dimensional grid environment demonstrate that DT+FM not only delays the onset of compounding errors but also autonomously corrects them when they occur, proving substantially more robust than vanilla DT.**

***Index Terms*—offline reinforcement learning, decision transformer, flow matching**

I. Introduction

Reinforcement Learning (RL) enables agents to learn policies that maximize cumulative rewards through interactions with the environment. However, when data collection in real-world environments is costly or hazardous, Offline Reinforcement Learning, which learns solely from pre-collected datasets, becomes necessary. The Decision Transformer (DT) [1] is a representative offline reinforcement learning algorithm that leverages the self-attention mechanism of Transformers [2] to model long-term dependencies between past actions and rewards. However, due to its autoregressive architecture, DT is vulnerable to the compounding error problem, where initial errors rapidly propagate and accumulate over time. This paper proposes a novel architecture that integrates Flow Matching (FM) [3] with DT to mitigate the compounding error problem. The proposed DT+FM architecture learns vector fields that transform noise distributions into target data distributions along probability paths, thereby autonomously correcting errors when they occur by guiding samples toward the target distribution through learned vector fields. Experimental validation in a one-dimensional grid environment demonstrates that the proposed method effectively mitigates the compounding error problem compared to vanilla DT.

This research was supported by Two Basic Science Research Programs through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2023R1A2C1003143 & NRF-2018R1A6A1A03025526).

II. Preliminaries

A. Decision Transformer

DT learns a policy from a pre-collected dataset. The dataset consists of trajectory sequences of length k, where each timestep t contains a return-to-go (RTG) R_t, state s_t, and action a_t. DT is trained to predict actions conditioned on the given RTG through supervised learning, minimizing the difference between the predicted action $\hat{a}_t$ and the actual action a_t.

However, DT has an inherent limitation due to its autoregressive architecture. During inference, the model's predicted action is fed back as input for the next timestep. If the model encounters states or actions absent from the training data (out-of-distribution, OOD), it produces inaccurate predictions. The critical issue is that these initial errors are fed back as inputs, triggering a cascading chain of increasingly larger errors. Specifically, a small error at timestep t propagates to timesteps $t+1, t+2$, and beyond, exponentially accumulating and exacerbating the compounding error problem.

B. Flow Matching

FM models the transformation process from a noise distribution to a target data distribution as a probability path and directly learns the vector field that moves data along this path. Specifically, for flow matching time $T \in [0, 1]$, a probability path x_T is defined that continuously transforms from the noise

distribution x_0 to the target distribution x_1. The target vector field v_T is obtained by differentiating this probability path with respect to T ($v_T = dx_T/dT$), and the training process minimizes the difference between the target vector field v_T and the model's predicted vector field $\hat{v}_T$.

The key principle by which FM mitigates the compounding error problem is as follows. When errors occur due to OOD data during DT's autoregressive inference, these errors can be regarded as noise. The learned vector field is trained to always point toward the target distribution regardless of the current position (error-contaminated state). Therefore, even when errors occur, the vector field automatically guides the probability path toward the target distribution, thereby correcting the errors. This self-correction mechanism prevents initial errors from cascading and amplifying, consequently mitigating the compounding error problem.

III. Proposed Architecture

This paper proposes an architecture that integrates FM into DT's hidden states to mitigate the compounding error problem. The proposed architecture learns vector fields that converge toward the target action distribution by utilizing DT-generated features as conditioning information for FM.

1) Training Process: DT takes a window of k samples (RTG, states, actions) as input and outputs k hidden states through a Transformer with causal masking. After adding the embedded flow matching time T to each hidden state, they are passed through FiLM [4] to generate scale parameter β and shift parameter γ. Simultaneously, Gaussian noise, target actions, and flow matching time T are linearly interpolated to output probability path x_T and target vector field v_T. After linearly transforming x_T using β and γ, it is passed through a neural network to generate the predicted vector field $\hat{v}_T$. The entire model is trained end-to-end by minimizing the following loss function:

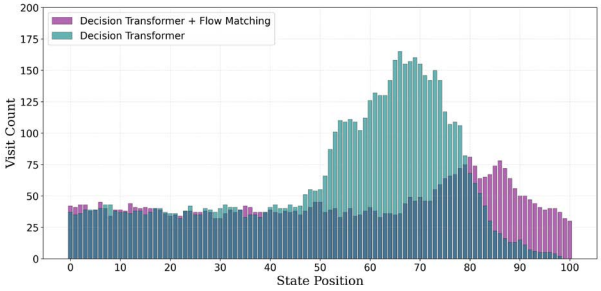
$$L_{DT+FM} = \left[||\hat{v}_T - v_T||_2^2\right] \tag{1}$$

2) Inference Process: Target RTG R_0 and initial state s_0 are input to DT to generate k hidden states. The final hidden state and Gaussian noise undergo N denoising steps to progressively refine and generate the final action.

IV. Experiments

1) Experimental Setup: In a one-dimensional grid environment (0$\tilde{1}$00), the agent can move left (-1) or right (+1) at each step. Reaching the goal position 100 results in success (reward 1), while exceeding 200 steps is treated as failure. The training dataset consists of trajectories where the agent always moves right.

During validation, to deliberately induce compounding errors, we designed an OOD environment where the agent is forced to move left (-1) with 10% probability regardless of the predicted action. A total of 30 episodes were executed, with results shown in Figure 1.



	Success Rate (%)	Most visited states (number of visits to that state)
DT	0.0	67 (201)
DT+FM	100.0	84 (84)

Fig. 1: Experimental results for vanilla Decision Transformer (DT) and proposed architenture (DT+FM).

2) Experimental Results: Figure 1 visualizes the visit count distribution across states. DT's visit counts surge from state 43, while DT+FM's counts increase from state 64, confirming the error accumulation delay effect.

The table in Figure 1 shows that DT+FM achieved 100% success rate, while vanilla DT recorded 0% success rate, demonstrating DT's compounding error problem and DT+FM's effective mitigation capability. Analysis of maximum visit counts reveals that DT visited state 72 a total of 176 times, whereas DT+FM visited state 81 only 75 times, showing that DT+FM delays error onset (64 vs 43) and mitigates error accumulation (75 visits vs 176 visits).

V. Conclusion

This study proposes an FM-integrated architecture to mitigate the compounding error problem in DT and demonstrates through grid environment experiments that the proposed architecture effectively alleviates compounding errors. Furthermore, this study presents the potential for improving existing algorithms suffering from compounding error problems and demonstrates the applicability of offline reinforcement learning to a broader range of real-world problem scenarios.

References

[1] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," *Advances in neural information processing systems*, vol. 34, pp. 15 084–15 097, 2021.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[3] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *The Eleventh International Conference on Learning Representations*, 2023.

[4] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.