Shortcut MixUp Policy: Toward Improving Robustness and Speed in Goal-Conditioned RL

Matt Hyatt*
mhyatt@luc.edu
Argonne National Laboratory
Lemont, Illinois, USA

Yassir Atlas
Argonne National Laboratory
Lemont, Illinois, USA

Hal Brynteson†
Argonne National Laboratory
Lemont, Illinois, USA

Diego A. Roa Perdomo‡
Argonne National Laboratory
Lemont, Illinois, USA

Athena Angara
Argonne National Laboratory
Lemont, Illinois, USA

Mengjiao Han
Argonne National Laboratory
Lemont, Illinois, USA

Joseph Insley
Argonne National Laboratory
Lemont, Illinois, USA

Janet Knowles
Argonne National Laboratory
Lemont, Illinois, USA

Yongho Kim
Argonne National Laboratory
Lemont, Illinois, USA

Victor Mateevitsi†
Argonne National Laboratory
Lemont, Illinois, USA

Michael E. Papka†
Argonne National Laboratory
Lemont, Illinois, USA

Silvio Rizzi
Argonne National Laboratory
Lemont, Illinois, USA

George K. Thiruvathukal§
gthiruvathukal@luc.edu
Loyola University Chicago
Chicago, Illinois
USA

Nicola Ferrier¶
nferrier@anl.gov
Argonne National Laboratory
Lemont, Illinois, USA

ABSTRACT

Neural networks trained on large datasets can be effective policies for the control of robotic manipulators. Using self supervised learning, these networks can achieve near-perfect success rates on complex pick and place style tasks. However, the speed of task completion is often a barrier to making learned policies practical for deployment. For instance, tasks that require 500 distinct token predictions will require many forward passes through the network, in real time. Moreover, to learn optimal task behavior - as in reinforcement learning - would require state value assignment across a long time horizon. This is often an impediment to learning.

To address these challenges, we present Shortcut Mixup Policy, a method to artificially reduce the task horizon length. Our method consists of training a model on next-token prediction task optionally conditioned on a target state-shortcut size. We present initial results using Shortcut Mixup Policy and propose future directions for improvement.

*Also with Loyola University Chicago.
†Also with University of Illinois Chicago.
‡Also with University of Delaware.
§Also with Argonne National Laboratory.
¶Also with University of Chicago.

SC '25, November 16–21, 2025, St. Louis, MO

ACM ISBN XXX-X-XXXX-XXXX-X/2025/11
https://doi.org/XXXXXXX.XXXXXXX

ACM Reference Format:
Matt Hyatt, Yassir Atlas, Hal Brynteson, Diego A. Roa Perdomo, Athena Angara, Mengjiao Han, Joseph Insley, Janet Knowles, Yongho Kim, Victor Mateevitsi, Michael E. Papka, Silvio Rizzi, George K. Thiruvathukal, and Nicola Ferrier. 2025. Shortcut MixUp Policy: Toward Improving Robustness and Speed in Goal-Conditioned RL. In *Proceedings of The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC '25).* ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 MOTIVATION

In addition to supervised token prediction, robotic manipulation tasks often use sparse rewards as an unbiased signal to guide learning optimal actions. It is common to bootstrap the network's state value estimate using the previous state value estimate, causing the learning signal to degrade over long time horizons. Consequently, it is difficult to learn which states are necessary for task completion. In [4] the authors show that the length of a task is associated with increasing errors in the network estimate of ground truth q-value of environment state.

Since long horizons can make learning difficult we propose a mechanism that will complete tasks quickly, even when learning from long sequences.

2 EXPERIMENT SETUP

In these experiments, we use the OGBench [3] environment suite based on the mujoco physics simulator. Each environment is designed as a goal-conditioned task, where the agent achieves success if the environment reaches a certain parameterized or randomized state. We focus this study on the manipulator task suite where the final state is a configuration of 2, 4, or 8, blocks.

The state s_t is defined as the 3D robot end effector position, rotation along z-axis, gripper closedness, joint positions and velocities, and the position and orientation of the blocks. The actions a_t are relative displacements in or absolute targets for end effector position, orientation, and gripper closedness.

Our training data consists of sequences collected from a scripted controller that approaches the blocks and moves them to new locations. It is likely that the sequences encountered during training are unlike the sequences/goal pairs at test time in that the environment may require that the policy seek a goal not reached in data from its current state. In [3, 4] this is called "goal-stitching".

3 POLICY SHORTCUT GUIDANCE

We first consider the naive sequence prediction (imitation learning) scenario. We adapt [2], which uses classifier free guidance to increase the probability of sampling actions from the predicted distribution which will lead to a given goal g. In this case the model predicts the next 1 action $a_t \propto s_{t+1}$, but optionally, we can condition on the step size gdt to instead predict the action s_{gdt}. We are inspired by [1] which uses a similar shortcut principle to perform one step diffusion denoising.

4 HIERARCHICAL SHORTCUT PREDICTION

In long-horizon tasks, it is common to use a hierarchical neural network, where the high-level network predicts subgoals and the low-level network predicts actions that lead to subgoal achievement. We propose a hierarchical network that additionally learns the optimal shortcut size. We are motivated by the fact that not all states can be avoided via shortcut from all other states. For example, moving the manipulator directly from point A to B may cause a collision or failure if C is not visited first. The shortcut proposal network will learn when a shortcut would expidite safe task completion and when it would be detrimental.

5 PRELIMINARY RESULTS

We train the low-level network on a variety of shortcut sizes. In the most extreme case, the network is trained with a mix of no shortcut, 2 step, 4 step, and 8 step shortcuts. During task evaluation, we randomly sample from shortcut sizes with geometrically decreasing probability of large shortcuts. We find that the shortcut-guided policy completes tasks almost twice as fast as no-shortcut counterparts, but suffers from a lower task success rate. While using the hierarchical subgoal network helps, it does not completely compensate for the cost of taking shortcuts indiscriminately. We aim to address this by using the shortcut proposal network.

6 FUTURE DIRECTIONS

We hypothesize that taking policy shortcuts introduce additional complexities that reduce the success rate.

- **Shortcut Forecasting.** In some states, taking a shortcut or too large of shortcut would cause failure.
- **Out of Distribution States.** Excessive shortcuts cause the model to enter states not seen during training, which leads to undesirable actions on the subsequent timestep.

6.1 Shortcut Proposal Network

The shortcut proposal network remains an area for future study. We will use ablations to show that in some states the policy can safely take a shortcut while others cannot. This work will motivate the need for a shortcut size proposal.

6.2 State Mixup

Mixup [5–7] is a training strategy where data are sampled and combined to create new synthetic data to increase robustness during training. We plan to use mixup of goals and states during training to produce a low-level policy that is more robust to unseen scenarios after taking a shortcut through the environment.

6.3 Shortcut Filtering

We will compare the task success rates of shortcut-guided policy with a shortcut filtering policy to measure potential issues with training stability. The shortcut filtering policy will predict chunks of n actions and learn a filter network which executes actions in the chunk if they pass a shortcut confidence threshold.

7 CONCLUSION

We present Shortcut Mixup Policy, a method for expedited task execution from long-horizon training data. This method has the potential to increase the usefulness of existing robot policies by reconditioning on shortcut step size and to produce compressed datasets without including unnecessary state visits.

REFERENCES

[1] Kevin Frans, Danijar Hafner, Sergey Levine, and Pieter Abbeel. 2025. One Step Diffusion via Shortcut Models. arXiv:2410.12557 [cs.LG] https://arxiv.org/abs/2410.12557

[2] Kevin Frans, Seohong Park, Pieter Abbeel, and Sergey Levine. 2025. Diffusion Guidance Is a Controllable Policy Improvement Operator. arXiv:2505.23458 [cs.LG] https://arxiv.org/abs/2505.23458

[3] Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. 2025. OGBench: Benchmarking Offline Goal-Conditioned RL. In *International Conference on Learning Representations (ICLR)*.

[4] Seohong Park, Kevin Frans, Deepinder Mann, Benjamin Eysenbach, Aviral Kumar, and Sergey Levine. 2025. Horizon Reduction Makes RL Scalable. *arXiv preprint arXiv:2506.04168* (2025).

[5] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, Aaron Courville, David Lopez-Paz, and Yoshua Bengio. 2019. Manifold Mixup: Better Representations by Interpolating Hidden States. arXiv:1806.05236 [stat.ML] https://arxiv.org/abs/1806.05236

[6] Huaxiu Yao, Yiping Wang, Linjun Zhang, James Zou, and Chelsea Finn. 2022. C-Mixup: Improving Generalization in Regression. arXiv:2210.05775 [cs.LG] https://arxiv.org/abs/2210.05775

[7] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond Empirical Risk Minimization. arXiv:1710.09412 [cs.LG] https://arxiv.org/abs/1710.09412

ACKNOWLEDGEMENTS

This work was supported by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research (ASCR) under the EXPRESS initiative "Harnessing Technology Innovations to Accelerate Science through Visualization" (DOE-145-SE-DAIMSL, NF-24).

This research used resources of the Argonne Leadership Computing Facility, a U.S. Department of Energy (DOE) Office of Science user facility at Argonne National Laboratory and is based on research supported by the U.S. DOE Office of Science-Advanced Scientific Computing Research Program, under Contract No. DE-AC02-06CH11357.