DynBridge: Bridging Imagination and Control through Interaction Dynamics for Robot Manipulation

Supplementary Material

Alex Wang[1*], Zhiwei Dong[2*], Qicheng Bai[1], Chenshi Zhang[3], Yujie Yi[4],
Guang Dai[1], Yong Liu[5], Mengmeng Wang[6,1†]
[1]SGIT AI Lab, State Grid Corporation of China, [2]Independent Researcher, [3]Central South University,
[4]Beijing University Of Technology, [5]Zhejiang University, [6]Zhejiang University of Technology
myx_dota@163.com, kivee@foxmail.com, guang.gdai@gmail.com, mengmewang@gmail.com

A. Details of the Simulation Environments

LIBERO-X. LIBERO is a tabletop manipulation benchmark built on the Franka robotic arm, designed to cover a broad range of complex and dexterous manipulation tasks. Each task is defined by a natural language instruction that specifies both the target object and its desired final state. The action space is 7-dimensional, consisting of the end-effector's positional and orientational deltas, along with the gripper control force.

To evaluate the capability of DynBridge on fine-grained manipulation scenarios, we conduct experiments on LIBERO-X, which consists of three subsets, LIBERO-Spatial, LIBERO-Object, and LIBERO-Goal, each containing ten tasks. These subsets isolate three fundamental aspects of manipulation: spatial configuration, object identity, and goal specification. ***LIBERO-Spatial*** includes tasks where the robot must place a bowl onto a plate, with two visually identical bowls differing only in their locations or spatial relations to other objects. The challenge lies in reasoning over varying spatial arrangements. ***LIBERO-Object*** consists of tasks centered on picking and placing different objects, requiring the robot to manage diverse object-level manipulation demands. ***LIBERO-Goal*** keeps the visual scene fixed while altering task objectives, resulting in different motion patterns and interaction strategies. The main difficulty is to learn distinct goal-conditioned behaviors and modulate them without behavioral interference.

LIBERO-Long. To assess long-horizon and multi-stage manipulation capabilities, we evaluate methods on LIBERO-Long, which contains ten extended-horizon tasks composed of sequences of interdependent subgoals. The feasibility of each subgoal depends on the correctness of preceding actions, creating strong temporal dependencies and requiring the model to reliably maintain intermediate states. These long sequences are highly sensitive to accumulated control errors, drift in dynamics representations, and degradation in action stability, thereby emphasizing the need for consistent and robust interaction dynamics.

LIBERO-90. We use LIBERO-90 to evaluate model performance under large-scale and highly diverse task conditions. LIBERO-90 contains ninety manipulation tasks that differ in scene layouts, object categories, language instructions, and task-phase structures. These variations create a wide range of perceptual and procedural conditions, allowing examination of a model's adaptability across diverse manipulation scenarios. The broad differences in manipulation primitives also reveal potential failure modes, such as representation collapse, overfitting to specific action patterns, and insufficient dynamics abstraction.

Meta-World. Meta-World is a widely used simulated manipulation benchmark built on a Sawyer robotic arm, covering diverse tabletop tasks involving object interaction and tool usage. The environment adopts a 4-dimensional action space controlling end-effector position deltas and gripper commands. Compared to LIBERO, Meta-World features clearer goal specifications and simpler visual scenes while emphasizing fine-grained low-level control. Meta-World varies target-object positions across episodes, creating a natural range of distinct spatial configurations. This inherent variability enables the evaluation of how well a model maintains stable manipulation behavior under changes in the spatial arrangement.

B. Details of Implementations

The complete list of hyperparameters used in our experiments is provided in Table 1.

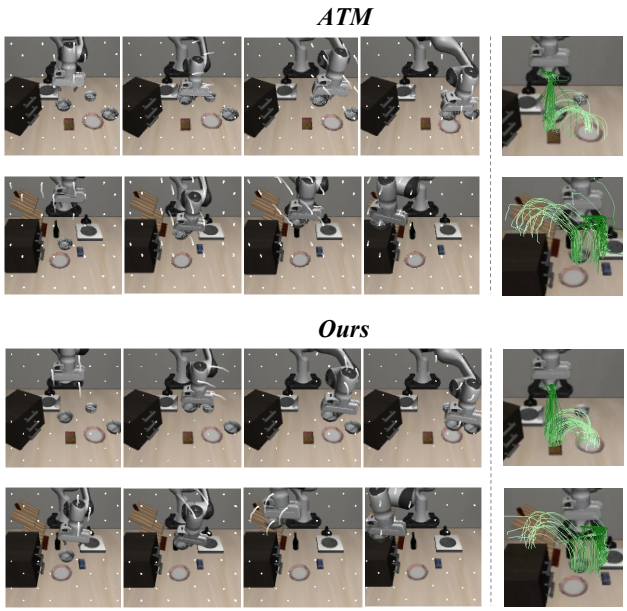

Figure 1. **DynBridge effectively bridges the gap between imagination and control.** *left:* the white curves show the model's imagined future trajectories. *right:* the green curves depict the end-effector trajectories during actual execution. Although both methods generate reasonable imagined trajectories, ATM's executed paths are dispersed with large variance, revealing a clear mismatch between imagination and control. In contrast, DynBridge's imagined trajectories align much more closely with the executed ones, effectively narrowing the imagination–control gap.

C. Additional Results

Visualization of Generated vs. Executed Trajectories. We visualize two types of trajectories for comparison. The first is the predicted future trajectories generated by the model, and the second is the end-effector trajectories executed by the robot, which are projected onto a 2D plane to reveal their execution behavior. As shown in Figure 1, the white curves depict the predicted future trajectories, and the green curves represent the actual end-effector trajectories recorded during execution. For simplicity and clearer interpretation, although our model generates 128 trajectories internally, only 32 uniformly sampled trajectories are visualized. ATM produces predicted trajectories on the left that appear visually coherent and directionally plausible, yet the corresponding execution trajectories on the right are highly dispersed with larger action variance. This reveals a pronounced imagination–control gap, referring to the mismatch between the internally imagined trajectories and the robot's actual executed behavior. In contrast, DynBridge generates predicted trajectories that align much more closely with their executed counterparts, exhibiting smaller spatial deviations and lower action variance. This alignment indicates DynBridge's ability to significantly reduce the imagination–control gap, a factor that directly supports its superior performance.

Table 1. List of hyperparameters.

Parameter	Value
Learning Rate	$1e^{-4}$
Image size	128 x 128 x 3 for LIBERO 84 x 84 x 3 for Meta-World
Batch size	64
Optimizer	Adam
Hidden dim	256
Interaction Attention	1 layers and 8 heads
Action Transfromer	8 layers and 4 heads
Action head	2-layer MLP
History length h	5
Number of dynamic token N_{tok}	8

Qualitative Analysis of Interaction Dynamics. To better illustrate the impact of interaction dynamics, we present two representative tasks as qualitative examples in Figure 2. In the top drawer-pulling task, our method guides the robot arm to accurately reach the handle and apply effective pulling force by modeling the interaction process, jointly capturing the interaction dynamics through trajectory generation for where to act and action prediction for how to act, ultimately opening the drawer successfully. In contrast, the baseline without interaction dynamics, *ours w/o InD*,

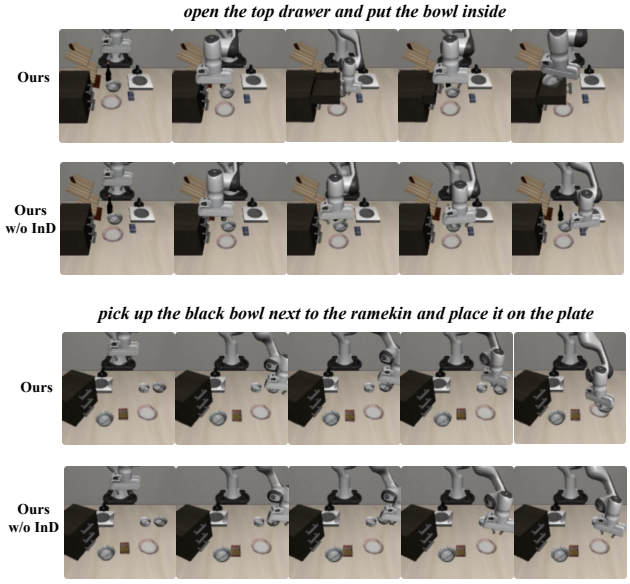

Figure 2. **Qualitative comparison on two LIBERO tasks.** Our method successfully pulls the drawer and recovers from a failed bowl grasp by leveraging interaction dynamics.

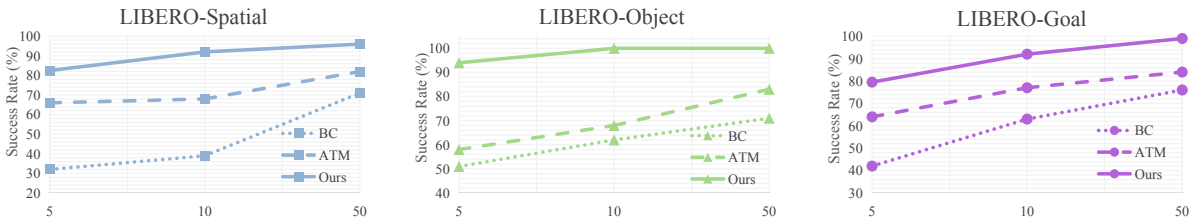

Figure 3. Effect of the number of training demonstrations.

also makes contact with the handle but fails to reason about meaningful interaction, leading to an unsuccessful pull. In the bottom bowl-grasping task, our method adapts the end-effector pose and retries after the initial grasp does not succeed, whereas the baseline proceeds to the next stage despite not retaining the object, revealing overfitting to the training trajectories. These qualitative examples highlight the importance of explicitly modeling interaction dynamics for robust manipulation.

Effect of the Number of Demonstrations. We evaluate the performance of DynBridge on the LIBERO benchmark under different numbers of expert demonstrations. Specifically, we train the model with 5, 10, and 50 demonstrations per task. As shown in Figure 3, performance improves monotonically as the number of demonstrations increases, and DynBridge consistently achieves higher success rates across all data scales. It is also worth noting that even though ATM uses additional videos for pre-training, DynBridge still performs better under every data regime. This advantage mainly comes from DynBridge's ability to model interaction dynamics, which substantially reduces the gap between imagined trajectories and executed actions. DynBridge also exhibits strong sample efficiency: with only 5 demonstrations, it already surpasses all baselines by a clear margin. As the number of demonstrations increases to 10 and 50, its performance continues to improve and approaches near-saturation, with success rates close to or even reaching 100%. This suggests that DynBridge can make effective use of additional demonstrations and demonstrates the stable and scalable learning capability of our method.

Table 2. Training time and inference latency.

Method	Train↓ **(s/iter)**	Latency↓ **(s)**
ATM (2-Stage)	0.69	0.028
Ours (1-Stage)	**0.31**	**0.019**

Computation Cost. As demonstrated in Table 2, DynBridge achieves faster training speed and reduced inference latency relative to the two-stage baseline. This efficiency gain primarily stems from our latent interaction dynamics, which bypasses the computational burden of pixel-space reconstruction and the complexities of disjointed multi-stage optimizations.

D. Real-World Experiments

To validate the effectiveness of DynBridge in real-world scenarios, we train and evaluate the model in a real robotic setup equipped with a Franka Research 3 (FR3) arm. A statically mounted RGB camera provides observations from a third-person perspective. We design five real-world tasks: (1) pressing a button; (2) pulling out a tissue; (3) picking up a cup and pouring water; (4) placing cherries into a plate and then placing a small snack into the same plate; (5) folding a towel. These tasks involve diverse interactive objects and manipulation skills. For each task, we collect 10 demonstrations performed by human demonstrators, with trajectories recorded at 20 fps. We deliberately retain the raw expert demonstrations rather than cleaning them, thereby preserv-

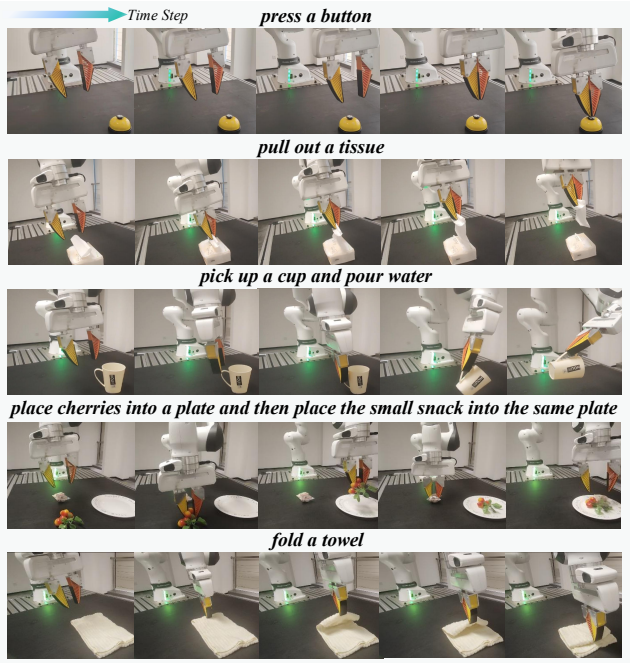

Figure 4. Qualitative results in real-world experiments.

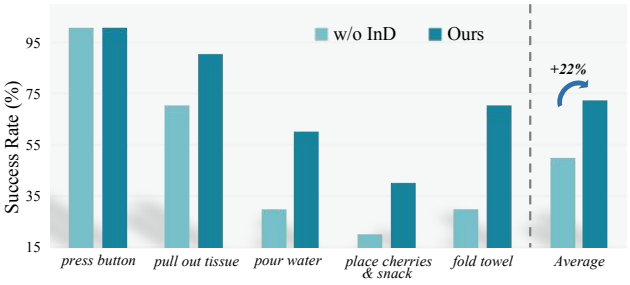

Figure 5. Quantitative comparison in real-world experiments.

ing inherent perception noise and naturally occurring imperfections in execution to better reflect realistic operating conditions and assess the model's robustness.

To evaluate the performance of our method, all experiments are conducted over 10 trials, and the average success rate is reported. Figure 4 shows qualitative results of DynBridge in real-world experiments, illustrating that it can make real-time predictions and execute tasks effectively. Figure 5 presents quantitative results, demonstrating that DynBridge significantly outperforms the variant without interaction-dynamics modeling (*w/o InD*), indicating that predicting future interactions enhances control stability and task success. Notably, the performance gains are most pronounced in long-horizon tasks and deformable-object tasks. For long-horizon tasks such as *placing cherries into a plate and then placing a small snack into the same plate*, where small deviations accumulate over multiple steps, our method reduces the imagination–control gap and effectively limits error accumulation. For deformable-object manipulation such as *folding a towel*, where future configurations are difficult to infer from vision alone, modeling future interaction dynamics allows DynBridge to maintain substantially more reliable control.