MoDem: Accelerating Visual Model-Based Reinforcement Learning with Demonstrations

Anonymous Author(s)
Affiliation
Address
email

Abstract: Poor sample efficiency continues to be the primary challenge for deployment of deep Reinforcement Learning (RL) algorithms for real-world applications, and in particular for visuo-motor control. Model-based RL has the potential to be highly sample efficient by concurrently learning a world model and using synthetic rollouts for planning and policy improvement. However, in practice, sample-efficient learning with model-based RL is bottlenecked by the exploration challenge. In this work, we find that leveraging just a handful of demonstrations can dramatically improve the sample-efficiency of model-based RL. Simply appending demonstrations to the interaction dataset, however, does not suffice. We identify key ingredients for leveraging demonstrations in model learning – policy pretraining, targeted exploration, and oversampling of demonstration data – which forms the three phases of our model-based RL framework. We empirically study three complex visuo-motor control domains and find that our method is $\mathbf{160\%} - \mathbf{250\%}$ more successful in completing sparse reward tasks compared to prior approaches in the low data regime ($\mathbf{100K}$ interaction steps, $\mathbf{5}$ demonstrations).

1 Introduction

Reinforcement Learning (RL) provides a principled and complete abstraction for training agents in unknown environments. However, poor sample efficiency of existing algorithms prevent their applicability for real-world tasks like object manipulation with robots. This is further exacerbated in visuo-motor control tasks which present both the challenges of visual representation learning as well as motor control. Model-based RL (MBRL) can in principle [1] improve the sample efficiency of RL by concurrently learning a world model and policy [2, 3, 4, 5]. The use of imaginary rollouts from the learned model can reduce the need for real environment interactions, and thus improve sample efficiency. However, a series of practical challenges like the difficulty of exploration, the need for shaped rewards, and the need for a high-quality visual representation, prevent MBRL from realizing its full potential. In this work, we seek to overcome these challenges from a practical standpoint, and we propose to do so by using expert demonstrations to accelerate MBRL.

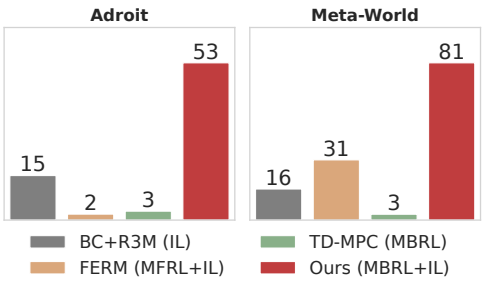

Figure 1. **Success rate (%) in sparse reward tasks.** Given only 5 demonstrations and limited online interaction, our method solves **21** hard robotics tasks from pixels, including dexterous manipulation, pick-and-place, and locomotion.

Expert demonstrations for visuo-motor control tasks can be collected using human teleoperation, kinesthetic teaching, or scripted policies. While these demonstrations provide direct supervision to learn complex behaviors, they are hard to collect in large quantities due to human costs and the degree of expertise needed [6]. However, even a small number of expert demonstrations can significantly accelerate RL by circumventing challenges related to exploration. Prior works have studied this in the context of *model-free* RL (MFRL) algorithms [7, 8, 9]. In this work, we propose a new framework to accelerate *model-based* RL algorithms with demonstrations. On a suite of challenging

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.

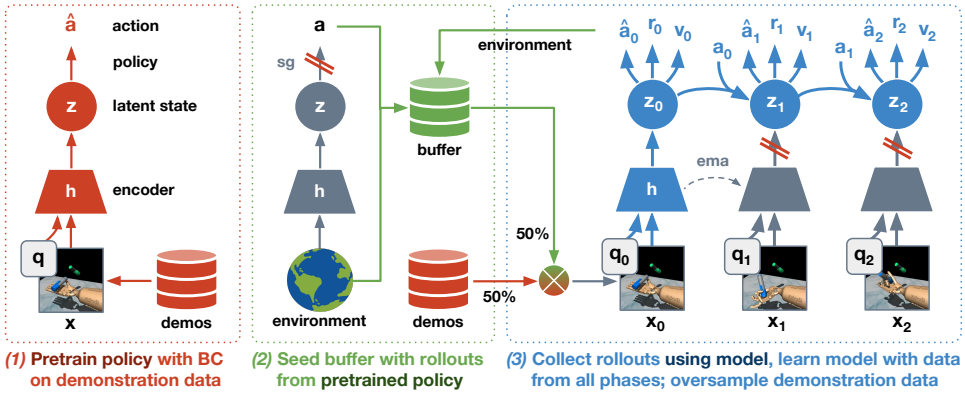

Figure 2. **Our framework (MoDem)** consists of three phases: *(1)* a ***policy pretraining*** phase where representation and policy is trained on a handful of demonstrations via BC, *(2)* a ***seeding*** phase where the pretrained policy is used to generate rollouts for targeted model learning, and *(3)* an ***interactive learning*** phase where the model iteratively collects new rollouts and is trained with data from all three phases. Crucially, we aggressively oversample demonstration data for model learning, regularize the model using data augmentation, and reuse weights across phases. sg: stop-gradient operator.

visuo-motor control tasks, we find that our method can train policies that are approx. $160\% - 250\%$ more successful than prior state-of-the-art (SOTA) baselines.

Off-policy RL algorithms [10] – both model-based and model-free – can in principle admit any dataset in the replay buffer. Consequently, it is tempting to naïvely append demonstrations to the replay buffer of an agent. However, we show that this is a poor choice (see Section 3), since the agent still starts with a random policy and must slowly incorporate the behavioral priors inherent in the demonstrations while learning in the environment. Simply initializing the policy by behavior cloning [11] the demonstrations is also insufficient. Any future learning of the policy is directly impacted by the quality of world model and/or critic – training of which requires sufficiently exploratory datasets. To circumvent these challenges and enable stable and monotonic, yet sample-efficient learning, we propose **Mo**del-based Reinforcement Learning with **Dem**onstrations (**MoDem**), a three-phase framework for visual model-based RL using only a handful of demonstrations. Our framework is summarized in Figure 2 and consists of:

- ***Phase 1: Policy pretraining***, where the visual representation and policy are pretrained on the demonstration dataset via behavior cloning (BC). While this pretraining by itself does not produce successful policies, it provides a strong prior through initialization.
- ***Phase 2: Seeding***, where the pretrained policy, with added exploration, is used to collect a small dataset from the environment. This dataset is used to pretrain the world model and critic. Empirically, data collected by the pretrained policy is far more useful for model and critic learning than random policies used in prior work, and *is key to the success of our work* as it ensures that the world model and critic benefit from the inductive biases provided by demonstrations. Without this phase, interactive learning can quickly cause policy collapse after the first few iterations of training, consequently erasing the benefits of policy pretraining.
- ***Phase 3: Finetuning with interactive learning***, where we interleave policy learning using synthetic rollouts and world model learning using data from all three phases including fresh environment interactions. Crucially, we aggressively oversample demonstration data during world model learning, and regularize with data augmentation in all phases.

Our Contributions. Our primary contribution in this work is the development of MoDem, which we evaluate on **18** challenging visual manipulation tasks from Adroit [7] and Meta-World [12] suites with only **sparse rewards**, as well as locomotion tasks from DMControl [13] that use dense rewards. Measured in terms of policy success after **100K** interaction steps (and using just **5** demonstrations), MoDem achieves $160\% - 250\%$ higher relative success and $38\% - 50\%$ higher absolute success compared to strong baselines. Through extensive empirical evaluations, we also elucidate the importance of each phase of MoDem.

2 Model-Based Reinforcement Learning with Demonstrations

In this work, our goal is to accelerate the sample efficiency of (visual) model-based RL with a handful of demonstrations. To this end, we propose **Mo**del-based Reinforcement Learning with **Dem**onstrations (**MoDem**), a simple and intuitive framework for visual RL under a strict environment-interaction budget. Figure 2 provides an overview of our method. MoDem consists of three phases: *(1)* a ***policy pretraining*** phase where the policy is trained on a handful of demonstrations via behavior cloning, *(2)* a ***seeding*** phase where the pretrained policy is used to collect a small dataset for targeted world-model learning, and *(3)* an ***interactive learning*** phase where the agent iteratively collects new data and improves using data from all the phases, with special emphasis on the demonstration data. We describe each phase in more detail below.

Phase 1: *policy pretraining*. We start by learning a policy from the demonstration dataset $\mathcal{D}^{\text{demo}} := \{D_1, D_2, \ldots, D_N\}$ where each demonstration D_i consists of $\{\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1, \ldots, \mathbf{s}_T, \mathbf{a}_T\}$. In general, the demonstrations may be noisy or sub-optimal – we do not explicitly make any optimality assumptions. Let $h_\theta : \mathcal{S} \mapsto \mathbb{R}^l$ denote the encoder and $\pi_\theta : \mathbb{R}^l \mapsto \mathcal{A}$ denote the policy that maps from the latent state representation to the action space. In Phase 1, we pretrain both the policy and encoder using a behavior-cloning objective, given by

$$\mathcal{L}_{\text{P1}}(\theta) = \mathbb{E}_{(\mathbf{s},\mathbf{a})\sim\mathcal{D}^{\text{demo}}} \left[-\log \pi_\theta\big(\mathbf{a}|h_\theta(\mathbf{s})\big) \right]. \tag{1}$$

When $\pi_\theta(\cdot|\mathbf{z})$ is parameterized by an isotropic Gaussian distribution, as commonly used in practice, Eq. 1 simplifies to the standard MSE loss. Behavior cloning with a small demonstration dataset is known to be difficult, especially from high-dimensional visual observations [14, 15, 16]. In Section 3, we indeed show that behavior cloning alone cannot train successful policies for the environments and datasets we study, even when using pre-trained visual representations [16, 17]. However, policy pretraining can provide strong inductive priors that facilitate sample-efficient adaptation in subsequent phases outlined below.

Phase 2: *seeding*. In the previous phase, we only pretrained the policy. In Phase 2, our goal is to also pretrain the critic and world-model, which requires a "seeding" dataset with sufficient exploration. A random policy is conventionally used to collect such a dataset in algorithms like TD-MPC. However, for visual RL tasks with sparse rewards, a random policy is unlikely to yield successful trajectories or visit task-relevant parts of the state space. Thus, we collect a small dataset with additive exploration using the policy from phase 1. Concretely, given π_θ^{P1} and h_θ^{P1} from the first phase, we collect a dataset $\mathcal{D}^{\text{seed}} = \{\tau_1, \tau_2, \ldots \tau_K\}$ by rolling out $\pi_\theta^{\text{P1}}(h_\theta^{\text{P1}}(\mathbf{s}))$. To ensure sufficient variability in trajectories, we add Gaussian noise to actions [5]. Let $\xi_t = (\mathbf{s}_i, \mathbf{a}_i, r_i, \mathbf{s}'_i)_{i=t}^{t+H}$ be a generic trajectory snippet of length H. In this phase, we learn π_θ, h_θ, d_θ, R_θ, Q_θ – the policy, representation, dynamics, reward, and value models – by minimizing the loss

$$\mathcal{L}_{\text{P2}}(\theta) := \kappa \cdot \mathbb{E}_{\xi_t\sim\mathcal{D}^{\text{demo}}} \left[\mathcal{L}_{\text{TD-MPC}}(\theta, \xi_t)\right] + (1-\kappa) \cdot \mathbb{E}_{\xi_t\sim\mathcal{D}^{\text{seed}}} \left[\mathcal{L}_{\text{TD-MPC}}(\theta, \xi_t)\right], \tag{2}$$

where κ is an "oversampling" rate that provides more weight to the demonstration dataset. In summary, the seeding phase plays the key role of initializing the world model, reward, and critic in the task-relevant parts of the environment, both through data collection and demonstration oversampling. We find in Section 3 that the seeding phase is crucial for sample-efficient learning, without which the learning agent is unable to make best use of the inductive priors in the demonstrations.

Phase 3: *interactive learning*. After initial pretraining of model and policy, we continue to improve the agent using fresh interactions with the environment. To do so, we initialize the replay buffer from the second phase, i.e. $\mathcal{B} \leftarrow \mathcal{D}^{\text{seed}}$. A naïve approach to utilizing the demonstrations is to simply append them to the replay buffer. However, we find this to be ineffective in practice, since online interaction data quickly outnumbers demonstrations. In line with the seeding phase, we propose to aggressively oversample demonstration data throughout training, but progressively anneal away the oversampling through the course of training. Concretely, we minimize the loss

$$\mathcal{L}_{\text{P3}}(\theta) := \kappa \cdot \mathbb{E}_{\xi_t\sim\mathcal{D}^{\text{demo}}} \left[\mathcal{L}_{\text{TD-MPC}}(\theta, \xi_t)\right] + (1-\kappa) \cdot \mathbb{E}_{\xi_t\sim\mathcal{B}} \left[\mathcal{L}_{\text{TD-MPC}}(\theta, \xi_t)\right]. \tag{3}$$

Finally, we find it highly effective to regularize the visual representation using **data augmentation**, which we apply in all phases.

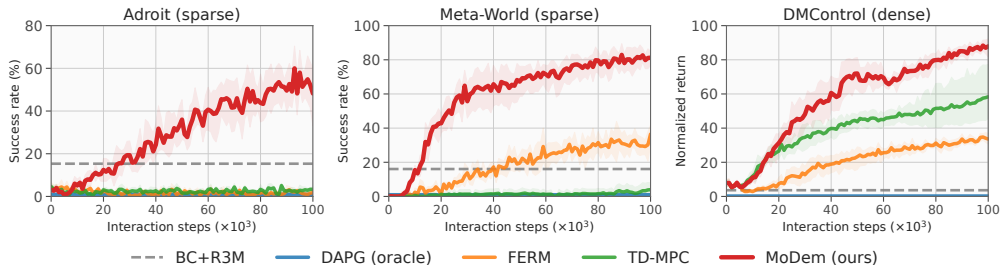

Figure 3. **Main result.** Success rate and episode return as a function of interaction steps for each of the three domains that we consider (Adroit, Meta-World, DMControl), aggregated across a total of **21** challenging, visual robotics tasks. Adroit and Meta-World use *sparse* rewards. Mean of 5 seeds; shaded area indicates 95% CIs. Our method is significantly more sample-efficient than prior methods.

3 Results & Discussion

Environments and Evaluation For our experimental evaluation, we consider **21** challenging visual control tasks. This includes **3** dexterous hand manipulation tasks from the *Adroit* suite [7], **15** manipulation tasks from *Meta-World*, as well as **3** locomotion tasks involving high-dimensional embodiments from *DMControl* [13]. For Adroit and Meta-World, we use sparse task completion rewards instead of human-shaped rewards. We use DM-Control to illustrate that MoDem provides significant sample-efficiency gains even for visual RL tasks with dense rewards. In the case of Meta-World, we study a diverse collection of *medium*, *hard*, and *very hard* tasks as categorized by Seo et al. [18]. We put strong emphasis on sample-efficiency and evaluate methods under an extremely constrained budget of only **5 demonstrations**[1] and **100K online interactions**.

Baselines for Comparison We consider a set of strong baselines from prior work on both visual IL, model-free RL (MFRL) with demonstrations, and visual model-based RL (MBRL): *(1)* **BC + R3M** that leverages the *pretrained* R3M visual representation [17] to train a policy by behavior cloning the demonstration dataset. *(2)* state-based (oracle) **DAPG** [7] that regularizes a policy gradient method with demonstrations. *(3)* **FERM** [9] combines model-free RL, contrastive representation learning, and imitation learning. Finally, we also compare with *(4)* **TD-MPC** [5] instantiated both *with* and *without* demonstrations. Our TD-MPC baseline appends demonstrations to the replay buffer at the start of training following Zhan et al. [9] and can thus be interpreted as a model-based analogue of FERM (but without contrastive pretraining). We evaluate all baselines under the same experimental setup as our method (*e.g.*, frame stacking, action repeat, data augmentation, and access to robot state) for a fair comparison.

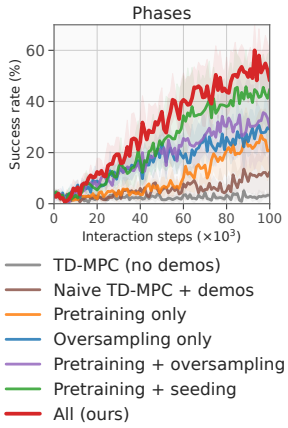

Figure 4. **Relative contribution** of each phase. Mean success across all Adroit tasks. 5 seeds, shaded areas are 95% CIs.

Benchmark Results Our main results are summarized in Figure 3. Our method achieves an average success rate of 53% at 100K steps across Adroit tasks, whereas all baselines fail to achieve any non-trivial results under this sample budget. FERM solves a small subset of the Meta-World tasks, whereas our method solves *all* 15 tasks. We find that our TD-MPC and FERM baselines fare relatively better in DM-Control, which uses dense rewards. Nevertheless, we still observe that MoDem outperforms all baselines. We also find that behavior cloning – even with pretrained visual representations – is ineffective with just 5 demonstrations. Finally, we study the relative importance of phases by considering all three Adroit tasks, and exhaustively evaluating all valid combinations of *policy pretraining* – as opposed to random initialization; BC *seeding* – as opposed to seeding with random interactions; and oversampling during *interactive learning* – as opposed to adding demonstrations to the interaction data buffer. Results are shown in Figure 4. We find that each aspect of our framework – policy pretraining, seeding, and oversampling – greatly improve sample-efficiency, both individually and in conjunction.

[1] Each demonstration corresponds to 50-500 online interaction steps, depending on the task.

References

[1] R. I. Brafman and M. Tennenholtz. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *J. Mach. Learn. Res.*, 3:213–231, 2002.

[2] D. Ha and J. Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems 31*, pages 2451–2463. Curran Associates, Inc., 2018.

[3] J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, T. P. Lillicrap, and D. Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588 7839:604–609, 2020.

[4] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

[5] N. Hansen, X. Wang, and H. Su. Temporal difference learning for model predictive control. In *ICML*, 2022.

[6] B. Baker, I. Akkaya, P. Zhokhov, J. Huizinga, J. Tang, A. Ecoffet, B. Houghton, R. Sampedro, and J. Clune. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *ArXiv*, abs/2206.11795, 2022.

[7] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine. Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

[8] R. Shah and V. Kumar. Rrl: Resnet as representation for reinforcement learning. *ArXiv*, abs/2107.03380, 2021.

[9] A. Zhan, P. Zhao, L. Pinto, P. Abbeel, and M. Laskin. A framework for efficient robotic manipulation. *ArXiv*, abs/2012.07975, 2020.

[10] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*, volume 1. 1998.

[11] D. A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 1. Morgan-Kaufmann, 1988.

[12] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2019.

[13] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. de Las Casas, D. Budden, A. Abdolmaleki, et al. Deepmind control suite. Technical report, DeepMind, 2018.

[14] Y. Duan, M. Andrychowicz, B. C. Stadie, J. Ho, J. Schneider, I. Sutskever, P. Abbeel, and W. Zaremba. One-shot imitation learning. *ArXiv*, abs/1703.07326, 2017.

[15] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *CoRL*, 2021.

[16] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. K. Gupta. The unsurprising effectiveness of pre-trained vision models for control. In *ICML*, 2022.

[17] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3m: A universal visual representation for robot manipulation. *ArXiv*, abs/2203.12601, 2022.

[18] Y. Seo, D. Hafner, H. Liu, F. Liu, S. James, K. Lee, and P. Abbeel. Masked world models for visual control. *ArXiv*, abs/2206.14244, 2022.