Enhancing autonomy and efficiency in goal-conditioned reinforcement learning

Yang, Z.

Chapter 7

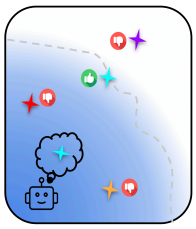
Conclusion

Goal-conditioned reinforcement learning is pivotal in training a generalist agent. Throughout this thesis, several new methods are proposed to enhance the goal-conditioned reinforcement learning framework. Whereas these methods show superior performance compared to the baselines, they are not without limitations. In this chapter, we first answer all research questions that are asked at the beginning, and then discuss the limitations of the methods we proposed. Then, we zoom out and provide a more general reflection on the current goal-conditioned reinforcement learning framework and how we can potentially improve it in the future. At the end, we summarize the main conclusion.

7.1 Answers to research questions

In Section 1.1, four research questions were formulated. Here, each research question is answered based on the results as described in Chapters 3-7, and the main conclusions for each question are given.

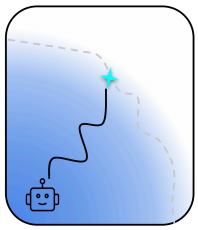
Q1 **Can RL agents learn without access to a 'reset'? [Chapter 3]**



(step 2) Having access to a 'reset' seems natural, but can be expensive in real-world tasks. For instance, training a robotic hand to spin a pen requires human intervention to reset when the pen falls from the hand (i.e. put the pen back in the robotic hand). Since such training schemes that require humans in the loop are infeasible to scale up, ideally, we want agents to operate with minimal human effort, i.e. without a reset (reset-free).

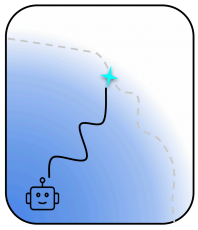
However, eliminating reset imposes challenges on exploration. We show model-based reinforcement learning (MBRL) methods can be applied 'out-of-the-box' to the reset-free setting and outperform state-of-the-art model-free methods due to their sophisticated exploration and high data efficiency, while requiring less human effort, such as environmental reward function or demonstrations. We then identify that applying MBRL methods directly causes over-exploration, i.e. the agent will squander a significant amount of time on 'task-irrelevant' states. To overcome over-exploration of MBRL methods, we propose a model-based reset-free agent (MoReFree), which biases exploration and policy training towards 'task-relevant' states to get better performance. More specifically, during the data collection in the real environment, MoReFree is commanded on three different state distributions, i.e. a goal state distribution and an initial state distribution for collecting more 'task-relevant' data, and an exploratory state distribution to encourage better exploration and collect novel data. During imagination training, the policy is explicitly trained to reach the initial and goal state distributions. In short, MBRL methods show promising results under the reset-free setting due to their sophisticated exploration and high data efficiency, and biasing towards task-relevant states during both data collection and policy training creates a synergistic cycle for MBRL methods, resulting in even stronger performance.

Q2 **Can non-parametric methods be applied to tasks with a continuous action space to achieve faster learning? [Chapter 4]**



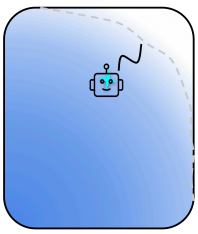
(step 3) Non-parametric methods, in our case, episodic control methods, store solutions in tabular forms and maintain values (episodic returns) for each possible action separately. Consequently, they can in principle only handle tasks with a discrete action space. We approach the proposed research question by maintaining both the action and the value for a state, replacing the existing action and its corresponding value with the newly encountered action which has a higher value. Since a continuous state will never be encountered twice, during action selection, we use the k-nearest neighbors algorithm to make an approximate estimation for the new coming state. We first find its k-nearest neighbor states, and then actions attached to neighbors with higher values are more likely to be selected. Our experimental results show that the proposed method achieves faster learning speed and better performance compared with deep RL baselines in various continuous control tasks.

Q3 **Can non-parametric and parametric methods be combined to achieve both fast learning and optimality? [Chapter 5]**



(step 3) Since episodic control methods replace existing values using newly encountered better ones, only the best solution the agent ever discovered is stored, even if such a solution only occurs with low probabilities. Consequently, although they learn faster, solutions stored by episodic control are non-optimal in stochastic tasks. RL, on the other hand, learns slowly but can handle stochasticity. We combine these two approaches together to form a single agent called the Two-Memory (2M) agent. For each episode during training, either episodic control or RL is selected to collect data, which is then used to train both of them. During evaluation, the one with better historical performance is used for action selection. Quantitative results show that 2M outperforms both sole RL and episodic control agents in five simplified Atari games, illustrating the success and effectiveness of such a combination. Interestingly, qualitative results show that initially, the 2M agent prefers to select the episodic control agent for data collection since it learns reasonably good behaviors quickly. But gradually, the RL agent catches up on the performance while maintaining better solutions, and thus the 2M agent switches to RL for data collection. Thereby, by switching between two methods and employing the better one for evaluation, 2M achieves both fast learning and optimality.

Q4 **What benefits can post-exploration introduce compared to not having it? [Chapter 6]**



(step 4) Post-exploration is defined as the exploration that occurs after the selected goal is reached. The intuition behind post-exploration is that the agent should explore when it is in states where exploration is beneficial. For instance, if the agent reaches frontier states, performing additional exploration will likely bring the agent into new regions, thus collecting more novel data. Our experimental results show that post-exploration indeed can lead the agent into new, unseen areas, where it can acquire more novel and diverse data. Therefore, the goal space extends to new areas, and the goal-conditioned policy is trained on new goals, resulting in a better performance compared with the agent without post-exploration (which resets directly after reaching the goal), using the same amount of environmental interactions. In conclusion, post-exploration can allocate the environmental interaction budget to more interesting areas, collecting more valuable data and leading to better performance.

7.2 Limitations

Although methods proposed in this thesis show promising results and outperform baselines, they are not without limitations. In each chapter, we already briefly discussed them, and here, we provide a more detailed and thorough discussion.

7.2.1 Limitations of model-based reset-free methods

In Chapter 3, MoReFree shows superior performance compared to baselines in various simulated robotic tasks. However, as a model-based unsupervised agent, it has all the disadvantages associated with model-based unsupervised approaches. Furthermore, evaluating reset-free methods using tasks designed for episodic settings limits their ability to fully showcase their potential. We discuss the limitations of MoReFree from these two perspectives.

Unsupervised MBRL backbone MoReFree is built on an unsupervised MBRL backbone, PEG (E. S. Hu et al., 2023), thus it inherits all drawbacks of both unsu-

pervised RL and MBRL methods. Unsupervised RL methods learn a self-supervised reward function, which is subsequently used to train the behavior policy. Although this eliminates the need for human-defined reward functions, in tasks requiring complex and precise behaviors, such self-supervised reward functions might be inadequate for providing effective learning signals. In these cases, incorporating a sparse reward function, which is generally easy to obtain, into the learned reward function can be beneficial. Similarly, MoReFree is constrained by its model-based backbone. MBRL methods first fit a model on collected data, and then use the model to either train the policy or perform online planning. Consequently, the upper bound of the performance of MBRL methods depends on the accuracy of the learned model. Namely, if the model is inaccurate, the trained policy or the online planning procedure will also be imprecise. Thereby, enhancing the underlying MBRL backbone, such as by employing more accurate world model architectures (Deng et al., 2023; Gu & Dao, 2023) or developing better self-supervised reward functions, will further improve the performance of MoReFree.

Benchmark Results shown in Chapter 3 illustrate the strength of MoReFree on various simulated robotics tasks, which are all adapted from existing tasks designed for the conventional episodic setting. Although these tasks are already quite challenging for RL agents, they are constrained in ways that rely on resets, which are not considered in the reset-free setting. For instance, in the Fetch-Push and Pick&Place tasks, a fixed-location robot cannot reach an object that has fallen to the ground if the object is not reset back to the table. However, if the robot were equipped with motility, it could then navigate around the table, pick up the block from the ground, and successfully complete the task in a reset-free setting. This increased flexibility would bring the simulated tasks closer to real-world scenarios, where robots need to operate in environments with less predefined constraints. To achieve this, we should specifically design tasks for the reset-free setting, instead of directly adapting them from episodic benchmarks.

7.2.2 Limitations of episodic control methods

In Chapter 4 and Chapter 5, we show that non-parametric methods, i.e., episodic control, can latch onto discovered solutions quickly, resulting in better performance than deep RL methods in some scenarios. We now discuss the main limitations of episodic control from three perspectives: generalizability, optimality, and representation. In

the end, we also provide a reflection on why we think it is still worth investigating episodic control methods in the future.

Generalizability Episodic control methods store discovered solutions in a tabular format. Therefore, in continuous tasks where it is impossible to encounter exactly the same state twice, we need mechanisms to retrieve solutions of states that are similar in some way to the newly encountered state. The most commonly used mechanism is the K-Nearest Neighbors algorithm (KNN), which outputs the top-k states that are most similar to the queried state according to a given distance metric (Euclidean distance, cosine similarity, etc). Unfortunately, in high-dimensional spaces, KNN does not work well due to the curse of dimensionality, which might output unreasonable neighbors and limit the generalizability of episodic control methods.

Stochasticity Episodic control methods learn quickly due to their aggressive update rule: instead of taking the expectation over all possible trajectories, they overwrite the existing solution with a better encountered one. Consequently, the best solution ever encountered is stored, even though it might have an extremely low probability of occurring due to stochastic transition dynamics. This limitation is inherent and cannot be eliminated unless combined with other methods. For example, in Chapter 5, we combine episodic control methods with deep RL methods to form one single agent, which gains the ability of dealing stochasticity from the RL side, resulting in better performance.

Representation Unlike parametric methods (e.g. deep RL) where representations are learnable and trained to be informative for learning behavior policies, episodic control methods do not utilize any learnable parameters. Episodic control methods either store original states as representations or employ dimensionality reduction methods like random projection to preprocess the original states. However, these non-parametric dimensionality reduction methods do not scale up well and might filter out information that is important for learning good behaviors. Consequently, Pritzel et al. (2017a) integrates trainable features into episodic control methods, leading to improved performance and highlighting the advantages of using better representations. Furthermore, employing pre-trained features in episodic control methods could be beneficial, as the significant success of pre-trained representations in fields such as computer vision or natural language processing.

Reflection Episodic control methods draw inspiration from episodic memory in the human brain, which refers to the ability to recall specific events, experiences, and situations from one's past (Tulving, 1983). The primary reason we believe these methods are still worth investigating is that humans frequently rely on episodic memory in daily life. For example, when driving home after work, we constantly use episodic memory to recall the route and landmarks along the way to ensure a quick return. More importantly, even when unexpected events occur, such as encountering traffic we have never seen before or meeting unfamiliar people, we can still navigate home correctly and successfully. Apparently, humans demonstrate the ability to handle stochasticity when using episodic memory, a capability that current episodic control methods lack, as seen in Algorithm 1 and Chapter 5. This discrepancy raises questions: Is it because our brain has such good attention mechanisms that it selectively filters out all irrelevant information, such as the new traffic? Or is it because the generalization ability of our brain is so strong that it can adapt to these unseen situations seamlessly? Or is there something even more sophisticated at play? These questions should motivate us to continually explore the potential of better implementation of episodic memory in artificial systems.

7.2.3 Limitations of post-exploration

Post-exploration is exploration after the goal is achieved. In Chapter 6, we made an attempt to study post-exploration and key design choices are made as straightforward as possible:

- **When the agent should post-explore**: post-exploration is only performed when the goal is successfully achieved, otherwise the agent is reset back to the initial state, and the next goal is selected for the agent to start over again.
- **For how long the agent should post-explore**: post-exploration is always performed with a fixed number of steps, which is treated as a hyper-parameter.
- **How the agent should post-explore**: during post-exploration, the agent is always taking random actions.

Although these simple choices allow us to isolate irrelevant factors and better study the properties of post-exploration, they also limit the scope of our analysis. As tasks become more complex, the effectiveness and applicability of the aforementioned simple design choices may need to be reconsidered.

To improve the current post-exploration scheme, these processes should be dynamically adapted. For instance, dynamically determining the number of steps for post-exploration. Intuitively, the areas that take longer to reach deserve more exploration. Or, as in PEG (E. S. Hu et al., 2023), replacing random exploration with more sophisticated exploration strategies mentioned in Chapter 2.

7.3 Reflections on goal-conditioned reinforcement learning

The current goal-conditioned reinforcement learning (GCRL) framework consists of four fundamental steps: defining the goal space, selecting goals for the agent, training the agent to achieve these goals, and post-exploration. However, upon reflection, several considerations arise:

- *Autonomy*: Each step in the GCRL framework relies heavily on human intervention, which hinders autonomy and scalability. The design of crucial components such as the reward function, reset mechanisms, and strategies for goal selection often require human expertise. However, human involvement in these processes introduces bottlenecks, making it challenging to develop agents that can scale.

- *Goal space*: Exploration in the GCRL framework is largely influenced by the commanded goals selected from the predefined goal space. When humans perform goal-conditioned learning, we utilize a broad spectrum of goal types (Berkman, 2018), ranging from abstract to concrete, and from objective to subjective. Humans can also pursue goals that may not yet exist, driven by personal aspirations or creative visions. Importantly, different types of goals motivate individuals in distinct ways. In contrast, the goal space typically defined in the current GCRL framework is too simplistic and often aligns closely with the state space or a latent representation thereof, which may restrict the exploration of the agent, potentially limiting the performance.

- *State space*: A large portion of the current GCRL research focuses on tasks that utilize proprioceptive state spaces. These internal state representations are relatively easy to obtain in simulated environments but challenging to acquire in real-world settings. This limitation restricts the applicability of GCRL methods in real-world applications where only visual perception is available. Consequently, incorporating visual or even multi-modal inputs is essential to enable

GCRL agents to operate effectively in diverse and more realistic environments.

Excessive reliance on human expertise, unimaginative goal space definition, and unrealistic state space representation limit the potential for training GCRL agents at scale in the real world, highlighting the need for improvements in several key areas.

7.4 Future research

As future research of each work has already been discussed in the corresponding chapters separately, here we provide high-level future research directions on the GCRL field.

Recent developments in foundation models (Mu et al., 2024; Touvron et al., 2023) that are trained on internet-scale data have demonstrated a strong ability in generalization and human-level understanding across a wide range of tasks. All these foundation models came out during the course of my PhD study, and they are already being integrated into the RL loop, yielding promising results. For instance, they are demonstrated to be able to design better reward functions (Ma et al., 2023) for complex robotic tasks than human experts or provide bonuses for better exploration (Klissarov et al., 2023). Essentially, the findings of previous studies indicate that it is possible to replace human-designed components with those designed by foundation models, enhancing the overall performance and enabling the RL framework to scale up effectively.

The GCRL framework would greatly benefit from the integration of foundation models, especially given the increased necessity for human design in the training loop. Foundation models can play a crucial role in several aspects of GCRL:

- *Goal selection*: With their extensive understanding across various domains, foundation models are likely to comprehend the tasks at hand, thus they can be used to replace human experts in providing meaningful goals under different contexts for the RL agents to pursue (C. Lu et al., 2024). For instance, multi-modal foundation models like vision-language models (Radford et al., 2021) can be used to propose goals in text space while the robot has a vision input.

- *Reward signals*: Foundation models can also be employed to provide step-wise reward signals for learning. For example, given a goal, they can distinguish good states that are likely to lead to the goal from bad ones that are far off. This distinction can then be used as a reward to train the agent, guiding it to reach the given goal more effectively.

However, most powerful foundation models like ChatGPT (OpenAI, 2024), Claude (Anthropic, 2024) or Gemini (Google, 2024) are close-sourced. This means users can only query these models and cannot access their intermediate output, such as embedding, which limits their utility. In contrast, researchers worldwide are developing open-source domain-specific pre-trained models in areas like robotics (M. Kim et al., 2024; Padalkar et al., 2023), self-driving cars (J. Yang et al., 2024) and MineCraft (Fan et al., 2022). If these initiatives can replicate the success seen with pre-training in natural language processes and computer vision, RL could benefit immensely from such pre-trained models:

- *Pre-trained representations*: As seen in natural language processes and computer vision, pre-trained representations capture the compact and informative features of the original input space. In RL, where learning representations only from reward signals can be time-consuming, the use of such pre-trained representations could bypass the need for encoding steps and facilitate the policy learning. Meanwhile, effective representations often ensure that similar states are also close in the latent space. This property also enables leveraging similarities between representations as reward signals.
- *Dynamic models*: The pre-trained predictive models can also be used as world models (Escontrela et al., 2024). These models are trained to predict the subsequent states given the current state-action pairs and enable zero-shot model-based RL or planning, where agents can simulate future states and plan actions without interactions with the environment.

These applications illustrate just a part of the potential of using foundation models and domain-specific pre-trained models to enhance GCRL agents. As research progresses, we anticipate that even more groundbreaking possibilities will emerge, further improving the efficiency and performance of GCRL agents.

7.5 Conclusion

In the thesis, we proposed four research questions and focused on three parts of the goal-conditioned reinforcement learning framework. First, in Chapter 3, we studied a more autonomous goal-conditioned reinforcement learning setting where there is no access to reset, which poses challenges on exploration. By using world models and selecting various goals to command the agent for data collection, our proposed agent can autonomously operate in reset-free tasks and outperform state-of-the-art baselines.

Once a goal is selected, a goal-conditioned policy is typically trained to reach the goal. Previous research on a non-parametric method, called model-free episodic control (Blundell, Uria, Pritzel, Li, Ruderman, Leibo, Rae, et al., 2016), shows that episodic control can quickly latch onto previously discovered solutions and learn faster than deep RL methods which are known to be slow. Instead of training a goal-conditioned policy, episodic control can also be employed to train the agent to reach the selected goal. However, model-free episodic control was designed for tasks with a discrete action space. In Chapter 4, we extend the episodic control to continuous episodic control that can now tackle tasks with a continuous action space, and demonstrate its strong performance on various continuous robotic control tasks. Moreover, limitations of episodic control methods are identified in Chapter 5, namely, they will be non-optimal in stochastic tasks. Then we proposed to combine episodic control with deep reinforcement learning methods to gain from both approaches. Experimental results show that by combining these two methods, the unified agent achieves both the fast learning attributed to episodic control and the optimality attributed to reinforcement learning.

In Chapter 6, we demonstrated that post-exploration is an important mechanism to improve efficiency of GCRL agents. By performing post-exploration, agents successfully step into new, unseen areas and acquire more diverse data, resulting in better performance compared to agents without post-exploration.

The methods proposed in the thesis improved components of the goal-conditioned reinforcement learning framework, including goal selection, policy learning and exploration, consequently enhancing the performance and increasing the autonomy of the entire goal-conditioned reinforcement learning framework. In future work, we hope that the goal-conditioned reinforcement learning framework will serve as a recipe for training a generalist agent, and ultimately, an embodied artificial general intelligent agent.