Reward-Incremental Reinforcement Learning

Yannick Denker and Alexander Gepperth

University of Applied Sciences Fulda - Dept of Computer Science
Leipziger Strasse 123, Fulda - Germany

Abstract. We address the challenge of reward-incremental learning (RIL) within the context of continual reinforcement learning. RIL presents a novel continual learning (CL) scenario where the same data samples (observations for RL) are mapped to different classes (Q-values) at different times. This is in contrast to class-incremental CL where new sample classes may be added, but without the contradictions inherent in RIL. To tackle this issue, we propose the use of an innovative replay-based approach called adiabatic replay (AR) which is inherently suited for RL since it removes the need for large replay buffers. Based on a simple benchmark scenario for continual RL, we empirically demonstrate that RIL scenarios can be handled by our approach, in contrast to conventional DQN methods.

1 Introduction

This article is in the context of continual learning (CL), i.e, machine learning from non-stationary data distributions, applied to reinforcement learning (RL). Since learning affects an agents' actions, which in turn impact the environment, the distribution of observations in RL is generally non-stationary. As a remedy, replay buffers are traditionally used in, e.g., DQN to mitigate catastrophic forgetting which would be a natural consequence. We specifically address the common case of reward-incremental (reinforcement) learning (RIL) where environments may be non-stationary themselves, leading to situations where the exact same observations require different actions at different times. In this case, replay buffers are not helpful because old knowledge *should* actually be forgotten and replaced, and large buffer sizes will lead to delayed adaptation. We formalize this simple continual RL setting by positing the existence of two or more distinct *tasks* of stationary characteristics. At the (known) onset of each task, however, these characteristics may change abruptly.

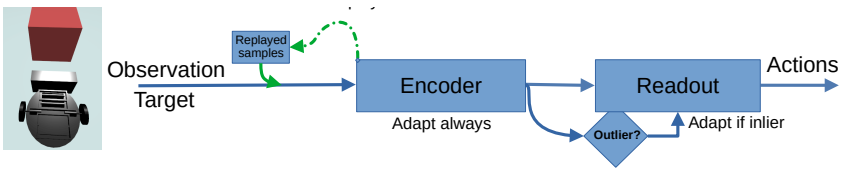

Fig. 1: Adiabatic Replay overview. In RL, selective replay is performed using a frozen AR model.

1.1 Proposed approach

We propose to use adiabatic replay (AR, see [7]) for addressing RIL scenarios. AR is centered around four concepts, see figure 1. model decomposition, inlier/outlier detection, selective updating and selective replay.

AR agents can be decomposed into an encoder implemented by a Gaussian Mixture Model (GMM) of K components, and a readout layer realized by a simple bias-free linear regression model. The GMM uses diagonal covariance matrices and is trained by SGD as outlined in [4], using the hyper-parameters recommended there. The readout(solver) is independently trained using SGD as well. The encoder provides a feature space for the readout layer and performs outlier detection as well as selective replay. For outlier detection, we consider a sample an outlier if the highest responsibility (posterior component probability) is superior to a threshold θ. New observations are used to query the encoder for similar samples, thus implementing selective replay. The generated samples are merged with newly arriving ones to form the training set for the current task. The readout layer is adapted only for inlier samples (since only inliers have a stable feature representation), whereas the encoder is adapted for all samples. Since the GMM update rule is local, the encoder naturally implements selective updating, i.e., only components close the the current input are adapted.

1.2 Related Work

Continual learning is a new and dynamic field (see, e.g., [9], [2] or [12]). In [3], CL approaches are grouped into three categories: parameter isolation, regularization and replay, all of which share the goal of preventing catastrophic forgetting (CF), the abrupt loss of knowledge after a distribution change. The application of CL methods to reinforcement learning has been discussed in, e.g., [6], although RL presents several challenges to the direct application of CL methods. Among the three methods, replay utilizes past experiences alongside current ones during the learning process. This effectively prevents the forgetting of previously acquired knowledge, ensuring a more robust and continuous learning experience. The previously acquired knowledge can be saved in a buffer or generated by another trained model, thus creating two distinct replay approaches [1]. Methods that use memory buffers to save prior knowledge in the reinforcement learning domain are ER [11], SER [5], MER [10] and a version of DQN [8].

1.3 Contribution

We propose the use of adiabatic replay (AR, [7]) for addressing RIL scenarios in continual RL, demonstrating the relevance and importance of Reward-Incremental Learning (RIL) as a significant problem in continual reinforcement learning. Additionally we contribute an empirical evaluation of a novel and a baseline RL method.

2 Methods

2.1 Simulation Setup

We design a benchmark scenario for continual RL, with conflicting rewards for the same observable task, as described in section 1. It contain a mobile robot navigating on a plane, equipped with a camera sensor and a differential drive. Cubes of different colors placed on the ground plane are either interactable or non-interactable, determined by their mass. One Cube exists twice with different interactability, to create the described conflict in expected behavior. The scenario is divided into a specified task structure, with a task for every colored cube. During training tasks are presented sequentially via a set amount of episodes, in which the robot is positioned in front of a cube. When approaching any cube and when colliding with an interactable a reward is given, while collision with non-interactable ones results in a punishment. The robot's action space comprises three actions: forward, left and right, each with three different speeds, resulting in a total of 9 discrete actions. The benchmark can be classified as a Reward-Incremental Learning scenario due to changes in the mass property of some objects during an experient which requires forgetting and retraining of learned knowledge. Performance is measured on a evaluation tasks taking place in between training tasks, where all previously learned tasks are evaluated.

2.2 Adiabatic Replay (AR)

Adiabatic replay as described in [7] is used as the learner in a Q-learning scenario mapping observations to Q-values for each possible action. At the beginnent of each task $t > 1$, we copy the current AR learner to a frozen model and re-initialize it. The frozen model is used for selective sampling, since it conserves the information learned at the end of the previous task.

2.3 Deep Q-Learning

We employ deep Q-Networks (DQNs) as a baseline for continual reinforcement learning (CRL), with and without prioritized experience replay (PER) as the sampling strategy for our replay buffer. An ϵ-greedy exploration strategy is utilized, where ϵ determines whether actions are selected randomly or by the neural network.

3 Experiments

3.1 Deep-Q Learning

We conduct a series of experiments utilizing our benchmark scenario from section 2.1, with object parameters displayed in table 1. Each individual training task consists of 5000 steps with episodes limited to 30 steps the robot can make. Each inbetween elvaluation task is 10 episodes long. The DQN controller uses epsilon-greedy exploration, starting with an initial epsilon value of 1.0. This

epsilon value decays linearly by 0.00015 per step until it reaches 0.2. This value resets for each new task to a value of 0.8 to increase the exploration again. We perform experiments with four different memory buffer sizes, utilizing simple and double Q-learning respectively based on the implementation from Mnih et al. [8]. Buffer capacities are 1000, 5000, 15000, 50000 and are chosen to represent samples from less than one task, exactly one task, three tasks and all tasks in memory respectively. All experiment combinations are repeated three times and with their results being averaged.

Task	T1	T2	T3	T4	T5
Cube color	**red**	blue	**red**	green	yellow
Interactable	**no**	yes	**yes**	no	yes

Table 1: All objects with their unique characteristics in the reinforcement learning experiments. The task number indicates in which order the tasks are presented during training. Each cube classified as interactable should be pushed and others approached but not touched.

The results of our experiments with Double DQN across the four buffer sizes reveal the well known distinct trade-off between buffer size and performance. Small buffers exhibit rapid learning of new tasks but suffer from catastrophic forgetting, erasing knowledge of previously learned tasks. In contrast, larger buffers retain knowledge of earlier tasks effectively, demonstrating minimal forgetting, but adapted more slowly to new tasks. This is evident in the table 2, where rows labeled *"T1 before T4"* show how small buffers completely forgot the behavior for task T1 while larger buffers retained all or some of it. Interestingly, in our benchmark scenario where a task repeats but requires different behaviors, small buffers successfully adapted by forgetting prior strategies, while larger buffers struggled due to their resistance to forgetting. This is seen in the table 2, where rows labeled *"T3 before T4"* and *"T3 before T5"* show that small buffers can learn the conflicting behavior immediately, while larger buffers are unable to or require much more time to. This highlights a critical limitation: while small buffers enables rapid adaptation in such conflicting task settings through catastrophic forgetting, it remains undesirable in broader continual learning contexts.

3.2 Adiabatic Replay

We conduct the same experiments dexscribed in section 3.1 using AR, meaning without the use of a memory buffer as seen in section 2.2. The outlier detection threshold is set to $\theta = 0.7$, the number of GMM components to $K = 100$. The learning rates for GMM and readout layer are set to 0.01 using a plain SGD optimizer. At each iteration, we draw mini-batches of size 32 from a memory buffer of size 64 which acts just as a tool to aggregate samples into mini-batches. All other settings are kept. For task T1, this mini-batch is directly used for training. For task T2 (and indeed all higher tasks), an additional mini-batch

↓ eval task - baseline →	DDQN 1K	DDQN 5K	DDQN 15K	DDQN 50K
Experience Replay				
T1 before T3	7.41	6.47	7.45	6.78
T1 before T4	−1.35	−1.20	2.99	8.93
T1 before T5	−1.27	−1.09	−0.64	7.44
T3 before T4	17.11	17.33	12.91	8.70
T3 before T5	17.22	17.37	15.38	10.95
Prioritized Experience Replay				
T1 before T3	7.91	8.55	6.94	6.78
T1 before T4	−1.16	−1.22	6.71	5.25
T1 before T5	−2.24	−1.39	−1.01	1.78
T3 before T4	17.26	17.28	8.71	10.45
T3 before T5	16.25	16.68	16.35	13.11

Table 2: Average episodic rewards for each contradictory task from inbetween evaluations, with double dqn, experience replay (ER) or prioritized experience replay (PER) and on all buffer sizes respectively. For T1 the optimal reward is around 8, while the optimal reward for T3 is around 18 (a reward of 10 is given when colliding with interactable cubes).

↓ eval task - K →	AR 81	AR 100	AR 121
T1 before T3	7.41	6.47	7.45
T1 before T4	−1.35	−1.20	2.99
T1 before T5	−1.27	−1.09	−0.64
T3 before T4	17.11	17.33	12.91
T3 before T5	17.22	17.37	15.38

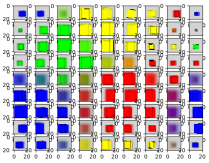

Table 3: Experimental results. Left: average reward for the contradictory evaluation tasks obtained using AR with a varying number of K of GMM components. Right: visualization of the GMM (encoder) centroids after processing task 1. We observe a sampling from all relevant sensory situations, and also a decoupling from the readout since the GMM centroids do not carry information about actions.

is obtained by variant generation from the frozen model as described in section 2.2. Thus, training for task T2-T5 is performed with a mini-batch size of 64.

A visualization of GMM centroids is given in figure 3, indicating that the distribution of sensory input has been acquired, and more specifically that GMM training has converged. Concerning the average reward obtained in the evaluation tasks, we refer to table 3. Generally, the obtained rewards are notably higher than those obtained on the evaluation task for DQN, indicating that catastrophic forgetting has occurred to a much lesser degree.

4 Discussion

In our experiments, DQNs seem incapable of addressing the Reward-Incremental Learning (RIL), suggesting that the problem has to be solved differently. When increasing the replay buffer size for DQN, reaction times to environment changes

increase strongly, rendering this strategy infeasible, whereas AR rewards indicate a rapid reaction is possible, because only conflicting knowledge is updated but non-conflicting knowledge is retained.

Of course, our continual RL benchmark is intentionally simplified, in particular since task onsets are known. Furthermore, the structure of objects is such that the underlying learning problem is rather simple. Relaxing this assumption for more difficult problems could lead to further interesting developments in continual RL.

5 Conclusion and Outlook

We present a unique scenario for Continual Learning (CL) influenced by reinforcement learning principles and introduce adiabatic replay (AR) as a potential solution, integrated into a suitable architecture. Our study demonstrates that AR can effectively address Reward-Incremental Learning (RIL) challenges in reinforcement learning settings, even without the use of a replay buffer. In future work, we aim to explore the feasibility of Reinforcement Learning (RL) without replay buffer in a broader context by leveraging Continual Learning (CL) methods such as adiabatic replay (AR).

References

[1] Benedikt Bagus and Alexander Gepperth. "An investigation of replay-based approaches for continual learning". In: *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2021, pp. 1–9.

[2] Yifan Chang et al. "Reviewing continual learning from the perspective of human-level intelligence". In: *arXiv preprint arXiv:2111.11964* (2021).

[3] Matthias De Lange et al. "A continual learning survey: Defying forgetting in classification tasks". In: *IEEE transactions on pattern analysis and machine intelligence* 44.7 (2021), pp. 3366–3385.

[4] Alexander Gepperth and Benedikt Pfülb. "Gradient-based training of Gaussian Mixture Models for High-Dimensional Streaming Data". In: *Neural Processing Letters* 53.6 (2021), pp. 4331–4348.

[5] David Isele and Akansel Cosgun. "Selective experience replay for lifelong learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 2018.

[6] Khimya Khetarpal et al. "Towards continual reinforcement learning: A review and perspectives". In: *Journal of Artificial Intelligence Research* 75 (2022), pp. 1401–1476.

[7] Alexander Krawczyk and Alexander Gepperth. "Adiabatic Replay for Continual Learning". In: *International Joint Conference on Neural Networks (IJCNN)*. 2024.

[8] Volodymyr Mnih et al. "Playing atari with deep reinforcement learning". In: *arXiv preprint arXiv:1312.5602* (2013).

[9] German I Parisi et al. "Continual lifelong learning with neural networks: A review". In: *Neural networks* 113 (2019), pp. 54–71.

[10] Matthew Riemer et al. "Learning to learn without forgetting by maximizing transfer and minimizing interference". In: *arXiv preprint arXiv:1810.11910* (2018).

[11] David Rolnick et al. "Experience replay for continual learning". In: *Advances in Neural Information Processing Systems* 32 (2019).

[12] Eli Verwimp et al. "Continual Learning: Applications and the Road Forward". In: *Transactions on Machine Learning Research (TMLR)* (2024). arXiv: `2311.11908 [cs.LG]`.