Multi-Objective Deep Reinforcement Learning with Optimistic Linear Support[1]

Hossam Mossalam

University of Oxford, Department of Computer Science

Sequential decision-making problems with multiple objectives arise naturally in practice and pose unique challenges for research in reinforcement learning. While, reinforcement learning has largely focused on single-objective settings, most real-world problems have multiple objectives, and it is not always clear how to evaluate different available trade-offs between these objectives a priori. Therefore, it is often highly desirable to produce a so-called *coverage set*, i.e., a set containing at least one optimal policy (and policy value vector) for each possible utility function that a user might have. Recently, a lot of progress has been made in this field. However, until now, deep learning methods have not yet been developed for multi-objective reinforcement learning problems, while they have proven very effective for single-objective reinforcement learning problems. One main reason for this is that neural networks — which is the underlying data structure for deep learning — cannot account for unknown preferences and the resulting sets of value vectors. In this thesis, we aim to circumvent this issue by taking a so-called outer loop approach [3] to multi-objective reinforcement learning, i.e., we aim to learn an approximate coverage set of policies, represented by a neural network, by evaluating a sequence of scalarized (that is, single-objective) problems. In order to do so effectively, we expect that it is necessary to share parts of the neural networks from earlier scalarized problems, in later iterations of the algorithm, which poses interesting algorithmic challenges.

In this thesis, we restrict ourselves to the case in which a *convex coverage set (CCS) is the optimal solution*, which is the optimal solution when the utility of the user can be expressed with a linear scalarisation function, $f(\mathbf{V}^{\pi}, \mathbf{w}) = \mathbf{w} \cdot \mathbf{V}^{\pi}$, i.e., the scalarised value is a convex combination of the value, $\mathbf{V}^{\pi}$, of a policy, π, in each objective. To this end, we use the *Optimistic Linear Support (OLS)* [3, 4] framework for multi-objective planning, in combination with deep reinforcement learning to create a novel multi-objective reinforcement learning algorithm. We call the resulting algorithm *Deep OLS Learning (DOL)*. DOL learns a CCS by learning policies for different $\mathbf{w}$ sequentially. These $\mathbf{w}$ are selected in a smart way via the OLS framework. By selecting the appropriate *Deep Q-Network (DQN)* architecture for different MOMDP problems (e.g., we use convolutional neural networks for image-based problems), we can learn near-optimal approximate CCSs, for multi-objective RL problems of unprecedented size.

While DOL can already tackle very large MOMDPs, re-learning the parameters for the entire network when we move to the next $\mathbf{w}$ in the sequence is rather inefficient. Gladly, we can make use of the following observation: the optimal value vectors (and thus optimal policies) for a scalarised MOMDP with a $\mathbf{w}$ and a $\mathbf{w}'$ that are close together, are typically close as well [5]. Because deep Q-networks (auto-)encode those features of a problem that are relevant for the optimal value (and thereby policy) of an MOMDP, we could probably speed up computation by reusing the neural networks that we found earlier in the sequence. We therefore extend DOL with the ability to reuse already learnt neural networks to speed up the learning of new ones leading to our new algorithm *DOL with reuse (DOL-R)*. DOL-R has two variations: in *DOL Full Reuse DOL-FR*, we start learning for a new scalarisation weight $\mathbf{w}'$, using the complete DQN we optimised for the previous $\mathbf{w}$ that is closest to $\mathbf{w}'$ in the sequence of scalarisation weights that OLS generated so far; in *DOL Partial Reuse DOL-PR*, we take the same network as for full reuse, but we reinitialise the last layer of the network randomly, in order to escape local optima.

[1] Extended abstract of the MSc thesis "Multi-objective Deep Reinforcement Learning" from the University of Oxford [1]. A more extended version of this work is available in [2].

We use Deep Sea Treasure world problem which is a well known multi-objective benchmark [6] to test the potential of DOL and DOL-R. In Deep Sea Treasure world problem, the agent controls a submarine searching for 10 treasures (terminal states) which are at different distances from the starting state. The two objectives in this problem are fuel usage (-1 per step) and the value of the treasures. We adjusted the standard rewards of the problem so that the optimal policies for finding the different treasures constitute a CCS. The main reason for using this problem is that it has a well-known structure, and we have a ground-truth CCS that we can compare our results to. We test on both the standard raw version, in which the state is a tuple of grid coordinates, and a new version, in which we use an image of the deep sea treasure problem as input for the learner. While the input for the learner (i.e., an image) of the latter is incomparably higher than the raw version (i.e., two integers), the ground truth CCS stays the same. In order to solve the image version of the problem, deep learning methods are essential, as the input would be too large for tabular reinforcement learning methods. For the raw version of the problem we use a feed-forward neural network for DOL(-R), while for the image version we employ a convolutional neural network.

To measure the effectiveness of our algorithms, we employ the Max CCS Difference, ε, i.e., we compare the output set of our algorithms with the ground truth CCS, and measure the maximum difference in (linearly) scalarised value [5]. The max CCS error is defined as $\varepsilon_i = \max_{w \in W} |(\max_{c \in CCS} w.c) - (\max_{s \in S_i} w.s)|$, where i represents the i^{th} iteration.

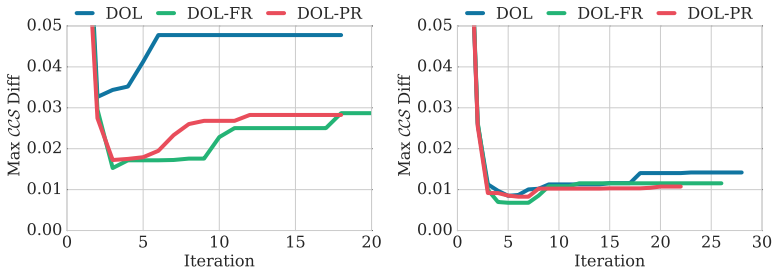

Figure 1: The Max CCS difference error (ε) after the i-th iteration of DOL and DOL-R: (left) Raw version; (right) Image version.

First, we evaluate our proposed methods, in a simple scenario, where the agent has direct access to s_t (Figure 1 (left)). Hence, we employ a simple feed forward neural network architecture, to measure the maximum error in scalarised value with respect to the true CCS. We trained the neural network for 4000 episodes and we observe that DOL suffers from the worst performance amongst the three proposed algorithms. Also, the difference in performance between DOL and DOL-R is considerably high indicating the benefit of reuse on the overall accuracy. For the image version, we employed a convolutional neural network which was trained for 6000 episodes. We observe that DOL and DOL-R achieve a very similar performance with with DOL-PR achieving the best results amongst all three algorithms. The results from both experiments indicate that DOL-R consistently achieves better results than DOL.

We therefore conclude that DOL can successfully learn a CCS, in problems (such as image deep sea treasure) that are intractable with traditional multi-objective RL algorithms. Furthermore, DOL-PR leads to a significant improvement over DOL without reuse. To our knowledge, DOL and DOL-R are the first multi-objective RL methods that can harness the power of deep reinforcement learning.

References

[1] Hossam Mossalam. Multi-objective deep reinforcement learning. Master's thesis, MSc in Computer Science, University of Oxford, 2016.

[2] Hossam Mossalam, Yannis M Assael, Diederik M Roijers, and Shimon Whiteson. Multi-objective deep reinforcement learning. *arXiv preprint arXiv:1610.02707*, 2016.

[3] Diederik M. Roijers. *Multi-Objective Decision-Theoretic Planning*. PhD thesis, University of Amsterdam, 2016.

[4] Diederik M. Roijers, Shimon Whiteson, and Frans A. Oliehoek. Computing convex coverage sets for faster multi-objective coordination. *Journal of Artificial Intelligence Research*, 52:399–443, 2015.

[5] Diederik M. Roijers, Shimon Whiteson, and Frans A. Oliehoek. Point-based planning for multi-objective POMDPs. In *IJCAI 2015: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 1666–1672, July 2015.

[6] P. Vamplew, R. Dazeley, A. Berry, E. Dekker, and R. Issabekov. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Machine Learning*, 84(1-2):51–80, 2011.