Review: Generic MODRL

Norio Kosaka

December 18, 2018

1 Paper Profile

- Title: A Multi-Objective Deep Reinforcement Learning Framework
- Authors: Thanh Thi Nguyen
- Organisation: Deakin University
- Publish Year: March 2018
- URL: https://arxiv.org/ftp/arxiv/papers/1803/1803.02965.pdf

2 Contents

1. Introduction
2. MORL Methods and Deep Learning Extensions
   (a) Overview of MORL methods
   (b) MODRL Framework Development
      i. Single-Policy DQN
      ii. Multi-Policy DQN
3. Experiment Settings and Evaluations
4. The Deep Sea Treasure(DST) problem
5. The MO-mountain car problem
6. Conclusions and Further work

3 Proposal

The present paper proposed a new MODRL framework on DQN coded in Python. It includes the use of linear and non-linear methods to develop the framework which is able to accommodate both single-policy and multi-policy strategies. And the resulting performance on the two previously proposed benchmarks, which are the Deep Sea Treasure and Mountain-Car problems, indicate the convergence to the optimal Pareto solutions effectively.

4 Introduction

The author has summarised the past researches by quoting some good researches as below.

- Q-Learning[13]: Tabular method requiring huge memory so that impractical in practice.
- DRL(e.g., Deep Q network by Mnih et al., 2015[2]) to overcome the problem using experience replay and function approximation techniques
- Mossalam et al. (2016) [3] extended deep Q-network to handle single-policy linear MORL
- Tajmajer(2017) [4] has extended DQN with a non-linear action selection approach based on a subsumption architecture
- Vamplew et al.(2017) [9] developed an MORL framework named MORL_Glue which is based on RL_Glue(Tanner and White, 2009) [5]. Unfortunately, this framework was not compatible with Deep neural networks.

5 MORL Methods and Deep Learning Extensions

5.1 Overview of MORL Methods

MORL extends the conventional single-objective RL methods to accommodate two or more objectives simultaneously; The reward signal of MORL is vectorised corresponding to each objective. Hence, by solving the multi-objectives problems leads us to the optimal Pareto Front which represent compromised the solutions among the objectives with the consideration regarding the preferences among the objectives. Indeed, current MORL methods can be classified into two categories

- Single-Policy(e.g., Chebyshev scalarization method by Van Moffaert et al,. 2013 [10])
- Multi-Policy(Van Moffaert et al,. 2014 [11])

In general, Single-policy methods require less computational resource though, they need appropriate prior information which bothering the practitioners. Whereas, in multi-policy methods, they can approximate the true Pareto front using multiple solutions so that users can select a suitable solution satisfying their requirements, yet predictably we have to compromise with regards to the computational cost.

Scalarisation method to transform the multi-objective problem into a single objective one(Vamplew et al,. 2008[8]).

- Nonlinear approach(Tesauro et al,. 2008[6])
- Linear approach(Castelletti et al,. 2013[1])
- Two-phase Local Search(Van Moddaert et al,. 2014[12])

- Analytic Hierarchy Process
- Geometric
- Ranking
- Convex Hull
- Varying Parameter approaches

5.2 MODRL Framework Development

5.2.1 Single-Policy DQN

This is to learn the optimal policy for a single-objective Markov Decision Process for which the objectives have been pre-scalarised into a single reward. In the present paper the agent, however, receives a vector of rewards on each time step. In addition, the agent is provided with a fixed weight vector w indicating the relative preference among the objectives.

$$w = \{w_1, w_2, \ldots, w_n\} \tag{1}$$

$$r = \{r_1, r_2, \ldots, r_n\} \tag{2}$$

$$L(\theta) = \sum_{i=1}^{n} L_i(\theta) \tag{3}$$

$$L_i(\theta) = E[\big(\gamma \max_{a'} Q_i(s', a'; \theta') - Q_i(s, a; \theta)\big)^2] \tag{4}$$

where γ is the discount factor, $L(\theta)$ is the loss function, $s, s', a, a', \theta, \theta'$ denote current state, next state, current action, next action, estimation network's weights and target network's weights respectively.

Fig.1 intuitively describes the network configuration used in our framework, which includes the layers as below.

Layer Index	Layer Description	Activation	# filters	Size	Stride
1	Convolutional Layer	ReLU	32	8x8	4
2	Fully Connected Layer	ReLU	N/A	N/A	N/A
3	Convolutional Layer	ReLU	64	4x4	2
4	Fully Connected Layer	ReLU	N/A	N/A	N/A
5	Convolutional Layer	ReLU	64	3x3	1
6	Fully Connected Layer	ReLU	N/A	N/A	N/A
7	Output Layer(the number of objectives)	N/A	N/A	N/A	N/A

5.2.2 Multi-policy DQN

I just copied this section from the original paper... sorry The choice of weights in a linear scalarisation is intended to represent the desirable trade-off between different objectives. In many problems, the user's preferences over objectives may change over time. In their framework, we implement multiple threads to allow the agents to learn in parallel multiple policies, such that it has an optimal

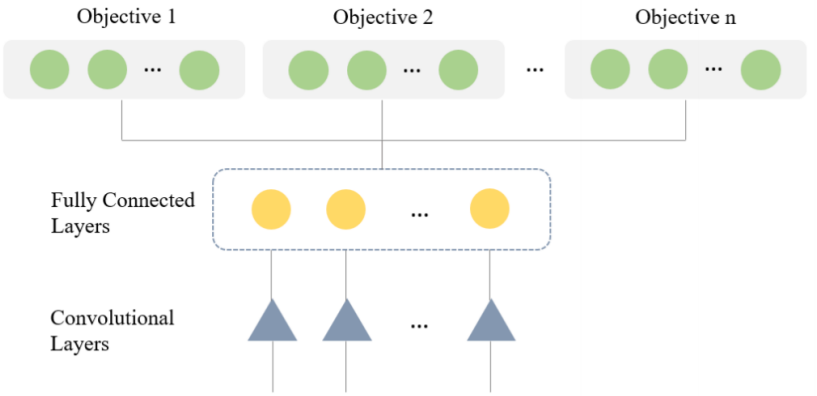

Fig. 1. Neural network structure used in our DQN-based MODRL framework.

policy in advance for any possible set of weights (linear weighted sum) or thresholds (nonlinear TLO) which it might encounter. In this way, it can immediately adapt its behaviour when informed of a change in weights or thresholds.

6 Proposal

The authors proposed two approaches

- Linear Weighted Sum
- Threshold Lexicographic Ordering(TLO)

Table 1. DQN settings for our experiments

Parameters	**Values**		
Initial epsilon	1.0		
Final epsilon	0		
Learning rate	0.0001		
Gamma (discounted rate)	0.9		
Target network update	1000 steps		
Root mean square (RMS) optimizer	decay = 0.99, epsilon = 1e-6		
Width of environment	DST $width = 3$	DST $width = 5$	Mountain-car
Action repeat	1	1	5
Epsilon annealing steps	46,000	190,000	200,000
Experience replay size	50,000	100,000	20,000
Warmup steps	5,000	10,000	2,000
Training steps	50,000	200,000	200,000

7 Experiment Settings and Evaluations

He has examined the proposition with two different benchmarkd games.

- MO-mountain-car: 2 objectives
- Deep sea treasure: 3 objectives

As for evaluation, he has adapted the *hypervolume approach* introduced by Vamplew et al,[7]. in 2011 as descibed in Fig.2

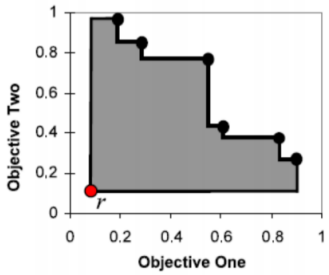

Fig. 2. The hypervolume is derived by the shaded region, bounded by the optimally approximated front and the reference point r (Vamplew et al., 2011).

8 Conclusions and Further Work

In this paper, the author proposed two generic novel approaches in MODRL. Most importantly, according to the author, it is so generic that they can accommodate different existing DRL algorithms, e.g., DQN, Dual DQN, A3C, UNREAL, Double DQN and so on, in various environments, e.g., gridworlds, Atari, and MuJoCo and so on.

Reference

[1] A Castelletti, F Pianosi, and M Restelli. "A multiobjective reinforcement learning approach to water resources systems operation: Pareto frontier approximation in a single run". In: *Water Resources Research* 49.6 (2013), pp. 3476–3486.

[2] Volodymyr Mnih et al. "Human-level control through deep reinforcement learning". In: *Nature* 518.7540 (2015), p. 529.

[3] Hossam Mossalam et al. "Multi-objective deep reinforcement learning". In: *arXiv preprint arXiv:1610.02707* (2016).

[4] Tomasz Tajmajer. "Multi-Objective Deep Q-Learning with Subsumption Architecture." In: *arXiv preprint arXiv:1704.06676* (2017).

[5] Brian Tanner and Adam White. "RL-Glue: Language-independent software for reinforcement-learning experiments". In: *Journal of Machine Learning Research* 10.Sep (2009), pp. 2133–2136.

[6] Gerald Tesauro et al. "Managing power consumption and performance of computing systems using reinforcement learning". In: *Advances in Neural Information Processing Systems*. 2008, pp. 1497–1504.

[7] Peter Vamplew et al. "Empirical evaluation methods for multiobjective reinforcement learning algorithms". In: *Machine learning* 84.1-2 (2011), pp. 51–80.

[8] Peter Vamplew et al. “On the limitations of scalarisation for multi-objective reinforcement learning of pareto fronts”. In: *Australasian Joint Conference on Artificial Intelligence*. Springer. 2008, pp. 372–378.

[9] Peter Vamplew et al. “Steering approaches to Pareto-optimal multiobjective reinforcement learning”. In: *Neurocomputing* 263 (2017), pp. 26–38.

[10] Kristof Van Moffaert, Madalina M Drugan, and Ann Nowé. “Scalarized multi-objective reinforcement learning: Novel design techniques.” In: *ADPRL*. 2013, pp. 191–199.

[11] Kristof Van Moffaert and Ann Nowé. “Multi-objective reinforcement learning using sets of pareto dominating policies”. In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 3483–3512.

[12] Kristof Van Moffaert et al. “A novel adaptive weight selection algorithm for multi-objective multi-agent reinforcement learning”. In: *Neural Networks (IJCNN), 2014 International Joint Conference on*. IEEE. 2014, pp. 2306–2314.

[13] Christopher J.C.H. Watkins and Peter Dayan. “Technical Note: Q-Learning”. In: *Machine Learning* 8.3 (May 1992), pp. 279–292. ISSN: 1573-0565. DOI: 10.1023/A:1022676722315. URL: https://doi.org/10.1023/A:1022676722315.