Shaping Mario with Human Advice (Demonstration)

Anna Harutyunyan
Vrije Universiteit Brussel
aharutyu@vub.ac.be

Tim Brys
Vrije Universiteit Brussel
timbrys@vub.ac.be

Peter Vrancx
Vrije Universiteit Brussel
pvrancx@vub.ac.be

Ann Nowé
Vrije Universiteit Brussel
anowe@vub.ac.be

ABSTRACT

In this demonstration, we allow humans to interactively advise a Mario agent during learning, and observe the resulting changes in performance, as compared to its unadvised counterpart. We do this via a novel potential-based reward shaping framework, capable for the first time of handling the scenario of online feedback.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning

Keywords

potential-based reward shaping; human advice

1. INTRODUCTION

Advice is an integral part of learning, both for humans and machines. While priceless in some situations, it is heuristic in nature, and may be extremely suboptimal. In reinforcement learning (RL) [10], where *learning* implies optimizing a given reward function that specifies the task, care must be taken so as to maintain focus on solving that task, and use advice only as guidance. The alternative is to learn to optimize the *advice* itself, which may solve the problem if the advice comes from a perfect oracle, but is likelier to result in suboptimal behaviors, and even prevent solving the problem altogether [9]. We wish to ensure that regardless of the quality of advice, the agent does not suffer negative consequences for heeding it.

For this, we place ourselves into the *potential-based* reward shaping (PBRS) framework [8], which gives the necessary form of modifying the reward function of an MDP without altering its (near-)optimal policies. Namely, PBRS constrains the additional *shaping* reward function F to:

$$F = \gamma\Phi' - \Phi \tag{1}$$

where Φ is a potential function over the state(-action) space.

We now make explicit the implicit assumption above of advice being expressed as a reward function R^A. While there is evidence that reward shaping offers more advantages than pure exploration guidance [2, 6], previous attempts of integrating human feedback into the reward scheme have not shown much promise [5, 1]. This is unsurprising, as these attempts are either not potential-based, or do not capture the advice properly, since previously there has been no clear way to translate the advice function R^A into Φ.[1] The recently proposed technique by Harutyunyan et al. [3] allows to express *any* arbitrary reward function in the potential-based form of Eq. (1). The authors do this by simultaneously with the main learning process, learning a second value function Φ^A w.r.t. a version of the input (advice) reward function R^A, and using the successive estimates of Φ^A as the potentials in Eq. (1). This process is shown to produce shaping rewards that are equivalent in expectation to R^A, while preserving all guarantees of PBRS, allowing to leverage R^A that is being provided sporadically *online* (e.g. by a human), in a true PBRS fashion.

2. SETTING AND METHOD

We assume a *reward-centric* feedback strategy [7], i.e. all feedback is positive, and punishment is implicit in the absence of feedback. The advice function is then an indicator A defined over the state-action space. We render A as a numerical reward function R^A in a natural way:

$$r^A_{t+1} = c \times A(s_t, a_t) \tag{2}$$

where r^A_t is the component of R^A at time t and c is a scaling constant.We then follow the framework of Harutyunyan et al. [3] and learn Φ^A to express R^A, with the following update rules at each step:

$$\begin{array}{rcl} \Phi^A_{t+1}(s_t,a_t) & \leftarrow & -r^A_{t+1} + \gamma\Phi_t(s_{t+1},a_{t+1}) - \Phi_t(s_t,a_t) \\ f^A_{t+1} & \leftarrow & \gamma\Phi^A_{t+1}(s_{t+1},a_{t+1}) - \Phi^A_t(s_t,a_t) \\ Q(s_t,a_t) & \leftarrow & r_{t+1} + f^A_{t+1} + \gamma Q(s_{t+1},a_{t+1}) - Q(s_t,a_t) \end{array} \tag{3}$$

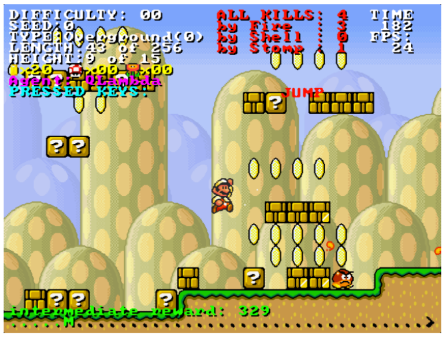
Note that we do not attempt to solve the advice *delay* problem. In our framework, the advice is implicitly propagated down the trajectory via the eligibility trace, with the remaining effect of delay being treated as noise.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

[1] Notice how naïve translation of $\Phi = R^A$ does not work. Say, $R^A(s,a) = 1$ (and 0 elsewhere), then $F(s,a,s') = \Phi(s',a') - \Phi(s,a) = -1$. The desired behavior got a *negative* shaping reward.

3. MARIO DOMAIN

The Mario benchmark problem [4] is based on Infinite Mario Bros, a public reimplementation of Super Mario Bros®. There are 12 discrete actions, corresponding to the buttons (with valid combinations) on a NES controller. Environment rewards correspond to the points collected in the game: the agent is rewarded for killing an enemy, collecting a coin, etc, and punished for getting hurt by a creature or dying. The state space includes information about Mario's state (can jump, can shoot, etc), as well as the coordinates of the closest enemy within a given range, totaling in 7072 features for each action, and 84864 features total. The state-action values are initialized to 0, resulting in near-random starting behavior.


DIFFICULTY: 00
SEED:0
TYPE:0(Overground(0)
LENGTH:43 of 256
HEIGHT:9 of 15
×20 ×00 ×00
Agent: QLambda
PRESSED KEYS:
ALL KILLS: 4
by Fire : 3
by Shell : 0
by Stomp : 1
TIME
182
FPS:
24

Intermediate reward: 329

Figure 1: A screenshot from Mario executing the level used in the demonstration

4. EXPERIMENTAL RESULTS

The participants were asked to advise Mario for the first 5 episodes by watching the agent play (at full speed of 25 decisions per second), and pressing a key, whenever in their opinion it had performed a good action. Recall that feedback is exclusively positive. After the advice had stopped, Mario continued learning on its own for another 95 episodes. We have considered two classes of advice:

Expert advice The advice is provided by a domain expert with knowledge of the state space

Non-expert advice Advisors are unaware of the state space (and, occasionally, of Mario)

Tab. 1 gives the comparison between the performance of Mario learning without any advice, with expert advice and with non-expert advice. Each variant is an average of 21 independent runs. The average advising rate was recorded to be 0.015, amounting to ≈ 45 advising steps per trial. Note that there is no significant difference between expert and non-expert advice, suggesting robustness to advice quality.

These results show that even with incredibly sparse advice rates, a large state space, noise incurred by the complexities of the domain and the delay in advice, our method is able to significantly improve the learning performance of Mario.

5. DEMONSTRATION

The demonstration[2] mirrors the experiment setup from Section 4. To highlight the immediate effects of the advice, a second *unadvised* Mario run is shown alongside. At the end of a trial the participant is presented with the learning curve of his Mario, and that of the average of all advisors, as compared to the unadvised autonomous learner.

Variant	Advice phase	Cumulative
Baseline	-376±51	470±83
Non-expert	**401±54**	**677±60**
Expert	**402±62**	**774±47**

Table 1: Points collected by Mario in the three considered scenarios (indicated with standard error of the mean). The best ($p < 0.05$) performance is given in bold.

6. CONCLUSION

With the inherent noise and lack of qualitative guarantees in human advice, it is imperative to be cautious when integrating it in the RL process. PBRS specifies the necessary form of modifying one's MDP without altering optimality w.r.t. the original task. We demonstrate the performance of a framework that for the first time effectively integrates human advice in the true PBRS fashion, showing promise for reward shaping methods in this avenue.

7. ACKNOWLEDGMENTS

Anna Harutyunyan is supported by the IWT-SBO project MIRAD (grant nr. 120057). Tim Brys is funded by a Ph.D grant of the Research Foundation-Flanders (FWO). The authors thank all volunteer Mario advisors.

REFERENCES

[1] S. Griffith, K. Subramanian, J. Scholz, C. Isbell, and A. L. Thomaz. Policy shaping: Integrating human feedback with reinforcement learning. In *Proc. of NIPS*, pages 2625–2633, 2013.

[2] A. Harutyunyan, T. Brys, P. Vrancx, and A. Nowé. Off-policy shaping ensembles in reinforcement learning. In *Proc. of ECAI*, pages 1021–1022, 2014.

[3] A. Harutyunyan, S. Devlin, P. Vrancx, and A. Nowé. Expressing arbitrary reward functions as potential-based advice. In *Proc. of AAAI*, 2015.

[4] S. Karakovskiy and J. Togelius. Mario AI benchmark and competitions. *IEEE CIG*, 4(1):55–67, 2012.

[5] W. B. Knox and P. Stone. Reinforcement learning from simultaneous human and MDP reward. In *Proc. of AAMAS*, pages 475–482, 2012.

[6] A. Laud and G. DeJong. The influence of reward on the speed of reinforcement learning: An analysis of shaping. In *Proc. of ICML*, 2003.

[7] R. Loftin, J. MacGlashan, and M. Taylor. A strategy-aware technique for learning behaviors from discrete human feedback. In *Proc. of AAAI*, 2014.

[8] A. Y. Ng, D. Harada, and S. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *In Proc. of ICML*, 1999.

[9] J. Randløv and P. Alstrøm. Learning to drive a bicycle using reinforcement learning and shaping. In *Proc. of ICML*, 1998.

[10] R. Sutton and A. Barto. *Reinforcement learning: An introduction*, volume 116. Cambridge Univ Press, 1998.

[2] Please see `https://vimeo.com/121085629` for an example video.