Embodying and Representing States in Reflexive Reinforcement Learning

Billy I. Lyons J. Michael Herrmann

Institute for Perception, Action and Behaviour
School of Informatics, University of Edinburgh
10 Crichton St, Edinburgh, EH8 9AB, U.K.
{Billy.Lyons|Michael.Herrmann}@ed.ac.uk

1 Introduction

1.1 Inverse Reinforcement Learning

Inverse reinforcement learning (IRL) is an instance of the inverse problem relavant for reinforcement learning (RL). The aim is to reconstruct the reward function from the obeservation of an Markov devision process (MDP). The methods assume the the observed behaviour is optimal or near optimal with respect to the original reinforcement learning problem [3], which we will call the *primal* problem.

We ask here whether an agent that is optimally trained on the primal problem can modulate its behaviour so that any observations are suitable for the IRL task. This question can be asked for a specific or for numerous IRL algorithms, but it can also be formulated as the task for the primal agent to act in such a way as to convey maximal information about primal states and action sequences such that

1. the agent indicates the equality of the value of states
2. identifies suboptimal actions that would be avoided by a greedy agent
3. provides information about probabilistic and deterministic state transitions oif this is compatiple with the previous item.
4. it produces a maximum entropy distribution of paths under previously mentioned constraints.

by choosing a noise level that reveals the full

This paper concerns itself with the solution to item one, by our method of reflexive reinforcement learning, by developing an agent which can "reflect" on-line and act in such a way as to maximise information provided to the maximum entropy inverse reinforcement learning agent [5].

1.1.1 Maximum Entropy Inverse Reinforcement Learning

brief descriptor of the specific setting of maximum entropy inverse reinforcement learning. discussion of partition function and the need to integrate over all valid trajectories and the computational difficulties of such in continuous and sufficiently large discrete spaces. cite some approaches to using neural networks for such domains and different approaches and their successes in estimating the partition function

[1]

2 Reflexive Reinforcement Learning

very brief overview [2]

3 Methods

As we have shown in our previous work, what constitutes as a reflexive component in an RRL system is highly flexible and variable, and it is our opinion that in larger and more complex systems one may need several reflexive components to develop a complex system which may serve to form a robust behavioural hierarchy, and that one may wish to pre-train such components, as in the case of identifying spaces of empowerment; however, with this task the aim is to provide an agent with a simple reflexive reward working in tandem with the task related reward, only using the readily available information one might expect the agent to have.

3.1 Environment

We will start with an illustrative toy experiment in a 5×5 world, consisting of two terminal states in the two corners on sharing a side in the world space.

Upon initialisation the agent is uniformly placed in any of the grids in the state space, barring either terminal. The agent's state consists entirely of the single state it is currently residing in state $= [s]$ such that $s \in S$.

3.2 Policy

In this paper we use a traditional function approximation approach for state-action pairs

$$Q^{\pi}(x,a) = E_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}r_{t}|x_{0}=x,a_{0}=a\right] \tag{1}$$

$$V^{\pi}(x) = E_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}r_{t}|x_{0}=x\right] \tag{2}$$

where $Q^{\pi}(x,a)$ is updated by [4]

$$\Delta Q^{\pi}(x,a) = \alpha\left(r_{t}+\gamma V^{\pi}(x_{t+1})-Q^{\pi}(x,a)\right) \tag{3}$$

with learning rate α and the value is given by

$$V^{\pi}(x_{t+1}) = \max_{a} Q^{\pi}(x_{t+1}, a). \tag{4}$$

With softmax action selection:

$$\pi(a|s) = \frac{\exp(Q_t(a)/\tau)}{\sum_{b=1}^{n} \exp(Q_t(b)/\tau)} \tag{5}$$

4 New idea

During each episode, all non visited state-action pairs have a quantity ζ added to their value: Given some episode E of maximum duration T, we have an episode history $e_h = \{s_1, a_1, ..., s_t, a_t\}$ where $t \leq T$, such that s_i, a_i represents the state occupied and the action selected at time i

$$\forall s, a \notin e_h, \quad Q^{\pi}(s, a) = Q^{\pi}(s, a) + \zeta \tag{6}$$

The entropy of each state within the space can be calculated such that, for any state $s \in S$

$$H(s) = -\sum_{a \in A} p(a|s) \log p(a|s) \tag{7}$$

this will serve as our reflexive reward, r_r.

We adapt equation 3 such that, for any desired "future insight window" of length n time steps, with

$$\Delta Q^{\pi}(x, a) = \alpha \left(r_t + \gamma V^{\pi}(x_{t+1}) - Q^{\pi}(x, a)\right) + \alpha \left(r_r + \gamma V^{\pi}(x_{t+n}) - V^{\pi}(x_{t+1})\right) \tag{8}$$

NOTE: Not entirely sure on the last half of the equation. This could just be the future entropy of the next states in the $t+n$ window. Its likely the learning rate would have to be different and the same with step size to keep the impact very minor.

But essentially, the agent will be rewarding future uncertainty, and by adding back some small amount for unvisited states per episode, whenever the agent is in the symmetry state it will look ahead and see that the region, in our case LHS or RHS of the world space that has less recently been visited will be the most likely choice.

Perhaps instead the effect would be better if: At each time step, prior to action the agent reflects

$$\forall a \in A, \quad \Delta Q^{\pi}(x, a) = \alpha \left(r_r + \gamma V^{\pi}(x_{t+n}) - V^{\pi}(x_{t+1})\right) \tag{9}$$

Agent moves

$$\Delta Q^{\pi}(x, a) = \alpha \left(r_t + \gamma V^{\pi}(x_{t+1}) - Q^{\pi}(x, a)\right) \tag{10}$$

So the tuple for reflexive reinforcement learning would be, instead of SARSA (state, action, reward, state, action) it would be SRARS (state, reflection, action, reward, state ... and so on), you could even possibly say "state, reflection, action, reflection, state, action...

Do you think this will work? It has the prerequisite from our previous conversations that we are not augmenting the state in any way, this is simply using the additional information that is normally lost in the learning process to make such decisions.

References

[1] Chelsea Finn, Paul Christiano, Pieter Abbeel, and Sergey Levine. A connection between generative adversarial networks, inverse reinforcement learning, and energy-based models. *arXiv preprint arXiv:1611.03852*, 2016.

[2] Billy Lyons and J. Michael Herrmann. Reflexive reinforcement learning: Reflexive reinforcement learning: Methods for self-referential autonomous learning. In *NCTA*, 2020. (accepted).

[3] Andrew Y Ng, Stuart J Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, page 2, 2000.

[4] Richard S Sutton and Andrew G Barto. Reinforcement learning. *Journal of Cognitive Neuroscience*, 11(1):126–134, 1999.

[5] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.