Policy Regularization for Legible Behavior

Michele Persiani
Department of Computing Science
Umeå University, Umeå, Sweden
michelep@cs.umu.se

***Abstract*—In Reinforcement Learning, legible behavior requires to maintain a policy that is easily discernable from a set of other policies. While legibility has been thoroughly addressed in Explainable Planning, little work exists in the Reinforcement Learning literature. As we propose in this paper, injecting legible behavior inside an agent's policy doesn't require to modify components of its learning algorithm. Rather, the agent's optimal policy can be regularized for legibility, by evaluating how the policy may produce observations that would make an observer to infer an incorrect policy. In our formulation, the decision boundary introduced by legibility impacts the states in which the agent's policy returns an action that has high likelihood also in other policies. In these cases, a trade-off between such action, and legible/sub-optimal action occurs.**

***Index Terms*—Legibility, Reinforcement Learning, Q-Learning, Bayesian Networks**

I. Introduction

In Explainable AI, the interpretability of agent behavior has been addressed in several variants such as *explicability* [1], *predictability* [2] and *legibility* [3]. In general, interpretable behavior requires to capture the expectations of the user about the agent in the form of an expectations model. This model, often called a second-order theory of mind, describes the model of the agent that has been internalized by its user, therefore informing the agent on how it is being perceived, understood or explained ie. how the user performs inferences on the agent. Using a second-order theory of mind, explainability charges the agent with the additional task of keeping its true model (here P_R) and the user's expectations model (P_R^H) as similar a possible. In this way, the inferences produced by P_R^H lead to the correct agent model P_R.

In this paper we attempt to implement the legibility criteria from the Explanable Planning literature using the reinforcement learning framework. As we propose, injecting legible behavior inside an agent's policy (π_R) doesn't require to modify components of the learning algorithm (here we use tabular Q-learning with full observability). Rather, we propose to evaluate how π_R may produce state-action pairs that would make the observer infer a different policy, to later find a trade-off that minimizes those while remaining consistent to the original policy.

Little work exists in modeling legible behavior in reinforcement learning. In [4] a method relying on the original formulation of legibility is proposed. However, this method is applicable only for goal-driven policies, thus excluding all other types of policies. In addition, it requires to specify a distance measure between states that can be a difficult task for arbitrary state-spaces.

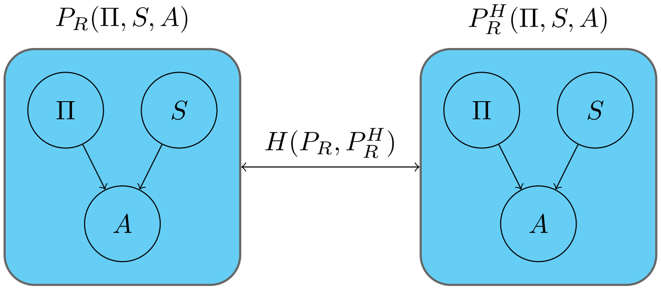

Fig. 1. Agent model and second-order theory of mind as equivalent Bayesian Networks. The networks model how agent and observer respectively select and infer actions using the current state and a set of predefined policies, while the function H measures the distance between these two processes.

Rather than relying of goal locations, we define a legibility criteria that is directly applicable on policies. A regularization method similar to ours is proposed in [5]. In their approach, during training the agent's policy is regularized towards an arbitrary behavior through a divergence function between the respective policies. We can see our method as a specific application of this method, where the policy is regularized towards the legible policy.

II. Method

We define a legible policy as:

An agent's policy is legible if it is discernable from a set of other policies.

We hypothesise an observer watching the agent and attempting to understand which is its policy among a set of candidates. The agent can be modeled to know it is being observed by implementing a theory of mind. The theory of mind can have many forms, for example, in [2] it is a label predicting whether the (human) observer is understanding the agent, while in [6] is a complete planning model. In general, simple observer models are easier to maintain, while those that are more complex allow to simulate with greater detail the inferences of the observer.

In this paper we utilize a middle way where agent and observer models are two equivalent Bayesian Networks (Figure 1). The networks are structurally the same, however, the random variables Π, S and A can be differently distributed in P_R and P_R^H. This setting avoids costly model alignments while implementing uncertainty in the observer beliefs.

The agent model (left part of Figure 1) selects actions based on the current state and policy. The right network instead simulates an observer, and tells the agent how the observer thinks it is selecting actions. When using Q-Learning, two corresponding Q-value tables $Q_R(a, \pi, s)$ and $Q^H_R(a, \pi, s)$ respectively define the probability distribution for selecting actions, with $P_R(a|\pi, s) \propto \exp\{Q_R(\pi, s, a)\}$, and for inferring which action the agent will take with $P^H_R(a|\pi, s) = \alpha \exp\{Q^H_R(\pi, s, a)\}$.

In this setting, the agent has a fixed set of pre-trained policies identified by the random variable $\Pi = \{\pi_0, ..., \pi_n\}$. Notably, among these there is the policy it is currently pursuing π_R with $P_R(\Pi = \pi_R) = 1$. We model the observer to not know which policy the agent is pursuing, thus having a uniform prior of the policies: $\forall i\ P^H_R(\pi_i) = k,\ k = \frac{1}{|\Pi|}$.

To be legible, the agent should select actions that communicate the observer its policy π_R, or that avoid communicating the others. This is obtained by selecting actions based on how they reduce the distance between the probability distribution over the agent policies, $P_R(\Pi)$, and the equivalent distribution that the observer infers, given an observation in term of state-action pair $P^H_R(\Pi|s, a)$. To implement this distance measure we utilize cross-entropy:

$$\begin{aligned} &H(P_R(\Pi), P^H_R(\Pi|s, a)) = \\ &- \log P^H_R(\pi_R|a, s) = \\ &- \log P^H_R(a|\pi_R, s) + \log \mathbb{E}[P^H_R(a|\pi, s)] - \log P^H_R(\pi_R) \end{aligned} \tag{1}$$

Assuming $Q_R = Q^H_R$ for simplicity, we can define the legible policy π_l as:

$$\pi_l(a|s) \propto \exp\{Q_R(\pi_R, s, a) - \alpha H(P_R(\Pi), P^H_R(\Pi|s, a))\} \tag{2}$$

where the right part of Eq. 2 regularizes the policy such that the selected actions also minimize the distance between the agent model and the model inferred by the observer.

Therefore, the decision boundary introduced by legibility impacts the states in which $\pi_R(s)$ returns an action that has high probability also in other policies. In these cases, a trade-off between such action, and sub-obtimal/legible action occurs.

III. Proof of concept

We tested the proposed method on a simple gridworld scenario. The grid is 7x7 and is without obstacles. There are 3 possible goals at the corners, for which we trained three corresponding policies with Q-learning. The value of α was set to 1. Figure 2 shows on the left column the learned optimal deterministic policy. On the right column the corresponding deterministic legible policies.

The learned policies have the behavior of going toward a wall adjacent the goal, to then approach the goal by walking along the wall. However, to be legible, it is important to approach the right wall that disambiguates the goal location. The legible policies systematically approach an unambiguous wall. Notice also how for g_1 the legible policy makes the agent walk in the middle to avoid approaching the other goals.

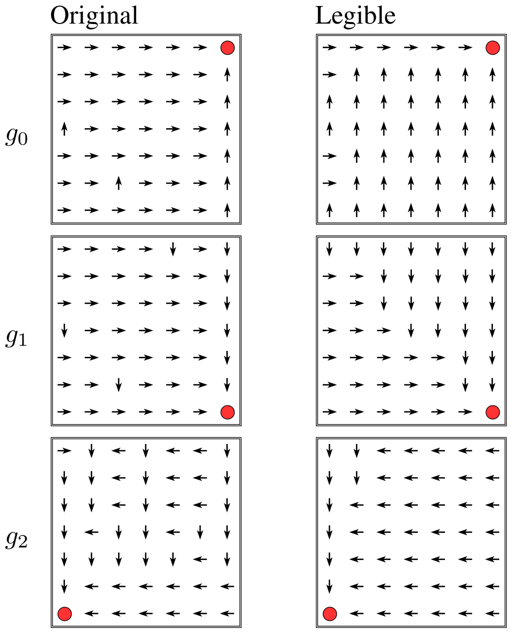

Fig. 2. Left: policies for the three goals (red dots) learned with Q-learning. Right: legible policies. The legible policies avoid ambiguity of goal location (of policy).

IV. Conclusion

In this abstract we introduce a framework that allows to incorporate legibility criterias into a reinforcement learning agent. We suggest that rather than modifying the learning procedure of the agent we can wrap a priorly learned set of policies by a pair of Bayesian Networks that model agent and observer respectively. The coupled networks forms a mirror setting of a second-order theory of mind, and have here the function of increasing the discrimination between the true agent policy and other candidates policies in the inferences of the observer. Future work includes further experimental validation of the proposed method.

References

[1] A. Kulkarni, Y. Zha, T. Chakraborti, S. G. Vadlamudi, Y. Zhang, and S. Kambhampati, “Explicable planning as minimizing distance from expected behavior,” in *AAMAS*, 2019, pp. 2075–2077.

[2] Y. Zhang, S. Sreedharan, A. Kulkarni, T. Chakraborti, H. H. Zhuo, and S. Kambhampati, “Plan explicability and predictability for robot task planning,” in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 1313–1320.

[3] A. D. Dragan, K. C. Lee, and S. S. Srinivasa, “Legibility and predictability of robot motion,” in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2013, pp. 301–308.

[4] M. Bied and M. Chetouani, “Integrating an observer in interactive reinforcement learning to learn legible trajectories,” in *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2020, pp. 760–767.

[5] Y. Wu, G. Tucker, and O. Nachum, “Behavior regularized offline reinforcement learning,” *arXiv preprint arXiv:1911.11361*, 2019.

[6] T. Chakraborti, S. Sreedharan, Y. Zhang, and S. Kambhampati, “Plan explanations as model reconciliation: Moving beyond explanation as soliloquy,” *arXiv preprint arXiv:1701.08317*, 2017.