Learning to Act Optimally in Partially Observable Multiagent Settings

(Doctoral Consortium)

Roi Ceren
Supervisor: Prashant Doshi
THINC Lab, Dept. of Computer Science
University of Georgia
Athens, Georgia, USA
ceren@cs.uga.edu

ABSTRACT

My research is focused on modeling optimal decision making in partially observable multiagent environments. I began with an investigation into the cognitive biases that induce subnormative behavior in humans playing games online in multiagent settings, leveraging well-known computational psychology approaches in modeling humans playing a strategic, sequential game. My subsequent work was in a scalable extension to Monte Carlo exploring starts for POMDPs (MCES-P), where I expanded the theory and algorithm to the multiagent setting. I first introduced a straightforward application with probably approximately correct guarantees (MCESP+PAC), and then introduced a more sample efficient partially model-based framework (MCESIP+PAC) that explicitly modeled the opponent.

Keywords

reinforcement learning; multiple agents; probably approximately correct; partial observability

1. INTRODUCTION

Arriving at optimal policies in single agent partially observable environments can be computationally taxing, but is significantly more so with the introduction of other agents who may have an impact on the environment as well. Exact solutions in all but the simplest domains may be far too intractable with conventional methodologies. I utilize online reinforcement learning (RL) as my departure point for partially observable multiagent settings.

My first investigation into RL was identifying whether humans utilized a reinforcement learning approach when evaluating their prospects of success in strategic, sequential games [1]. As a primary effort, I parameterized Q-learning utilizing known behavioral cognitive biases, including forgetfulness and erroneously ascribing reward to neighboring states. In addition, I implemented subproportional weighting *a la* prospect theory to represent humans misrepresenting probability judgments.

My recent efforts involve generalized frameworks for arriving at optimal policies in partially observable multiagent settings, while additionally providing statistical guarantees via probably approximately correct learning. I extend Monte Carlo Exploring Starts for POMDPs [2] with PAC bounds to the multiagent setting. I then add a partially model-based component explicitly modeling opponent behavior and show it dramatically reduces sample requirements. In addition, I customize ϵ-pruning to improve runtime.

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016),*
J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.),
May 9–13, 2016, Singapore.

2. MODELING HUMANS WITH BEHAVIORAL Q-LEARNING

My first work is concerned with the phenomenon of non-normative behavior in human subjects when engaging in strategic environments. I computationally model the effects of three cognitive biases on reinforcement learning: forgetfulness (erroneously discounting previous experience), reward spill over (where subjects attribute experiences in one state to a neighboring state as well), and subproportional weighting (from prospect theory, where probability assessments are categorically under- or over-weighted).

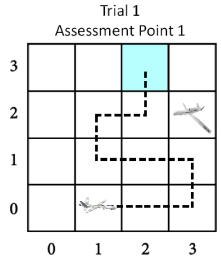

Figure 1: UAV problem domain, with the subject following a predefined path and the predator visible only initially.

In conjunction with the psychology department at my university, we conducted a series of experiments where subjects observed a UAV navigating in a 4x4 grid between a known start and goal sector. However, the environment included a predator UAV who, when arriving in the same sector as the subject UAV, would end the game in a loss. The predator's position is only known in the first step of the game.

I leveraged three parameters in our behavioral reinforcement learning model. α is the learning rate which normally appears in Q-learning. ϕ is an additional depreciation parameter affecting only the previous Q-value. ϵ represents the proportion of the reward that should be equally attributed to neighboring policies, where Eq. 1 represents the proportion given to the sector where the reward was experienced and Eq. 2 are the spilled-over states. Equation 3 is Prelec's one-parameter subproportional weighting. The state in our setting is the joint physical location of the subject and the predator UAVs.

$$Q(s,\pi(s)) = \phi \cdot Q(s,\pi(s)) + \alpha((1-\epsilon)r(s) + \gamma \cdot Q(s',\pi(s')) - \phi \cdot Q(s,\pi(s))) \quad (1)$$

$$Q(s_n,\pi(s)) = \phi \cdot Q(s_n,\pi(s)) + \alpha(\epsilon \cdot r(s) - \phi \cdot Q(s_n,\pi(s))) \quad (2)$$

$$w(p) = e^{-(-ln(p))^{\beta}} \quad (3)$$

I tested the predictive capabilities of the normative and behavioral models using Nelder-Mead downhill simplex with 5-fold cross-validation using 43 participants playing 20 trials each.

Model	total SSD
Behavioral Q-learning + Weighting	401.36
Default Q-learning + Weighting	406.26
Behavioral Q-learning	409.25
Default Q-learning	416.41

Table 1: Behavioral Q-learning with probability weighting shows the best fit and the differences are significant.

3. RL IN PARTIALLY OBSERVABLE MULTIAGENT ENVIRONMENTS

I extend Perkins' Monte Carlo exploring starts for POMDPs [2] (MCES-P) to multiagent domains with public observations of the environment and private observations of other agents' actions. The setting is non-cooperative, akin to the Interactive POMDP [3]. This extension, unlike the majority of the state of the art, does not require communication or collaboration between agents.

I first introduce canonical MCES-P to the multiagent setting, adapting the algorithm to allow private observations, and show that the probably approximately correct bounds also hold, albeit with a larger policy space. MCESP+PAC obtains action-values for policies using sampled trajectories and compares neighboring policies (that is, policies that differ by an action in only one observation sequence) to determine dominating transformations in the local neighborhood. In order to guarantee this transformation dominates, MCESP+PAC requires exactly $k_m \leftarrow \left\lceil 2\left(\frac{\Lambda}{\epsilon}\right)^2 \ln \frac{2N}{\delta_m} \right\rceil$ samples.

samples for each policy, where m is the number of transformations already taken and $\delta_m = 6\delta/(m^2\pi^2)$, and the neighbor must dominate the original policy by at least epsilon. Λ is the range of possible rewards, defined in MCES-P as $2T(R_{max} - R_{min})$. The algorithm can determine a better neighbor in less than k_m samples if, after each policy has p samples after m transformations, the neighbor dominates by $\epsilon(m,p) \leftarrow \Lambda\sqrt{\frac{1}{2p} \ln \frac{2(k_m-1)N}{\delta_m}}$.

MCES for I-POMDPs with PAC bounds (MCESIP+PAC) adds a partially model-based component of explicit models of the opponent and requires dramatically fewer samples than MCES-P for the same configuration and arriving at the same locally-optimal policy. Since MCES-IP maintains a belief over opponent models, the range of rewards is smaller, since it is limited to the rewards achievable for a specific action the opponent takes, which we dub Λ^{α_j}.

Lastly, I provide a custom ϵ-pruning methodology for MCES-P and MCES-IP, referred to as observation sequence pruning, that dramatically reduces the runtime in lieu of introducing bounded regret on the reward lost from foregone transformations. Since sampling some of the observation sequences can be quite costly due to their relative rarity, and since the same observation sequences have limited impact on the reward due to their infrequency, pruning is a powerful technique for speeding up Monte Carlo exploring starts. Given a proportion of allowable regret ϕ (around 15-20% in our experiments), the maximum regret introduced by foregoing transformations cannot exceed ϕ, such that $\sum_{\vec{o}_i \in \mathcal{P}} \bar{regret}_{\vec{o}_i} \leq \phi$.

We ran over 200 trials for two problem domains: the multiagent Tiger problem and a 3x2 Autonomous UAV problem, where the subject is a predator seeking the opponent (prey) before they arrive at the goal sector. In almost every case, MCES-IP arrived at the same policy as MCES-P, but did so in significantly fewer samples.

Method	Policy	Mean # of samples per transform	Mean bound on k_m
MCESP+PAC	Single	102,775 ± 44,468	271,478 ± 28,310
	Mixed	142,137 ± 54,361	262,711 ± 58,315
MCESIP+PAC	Single	50,154 ± 16,979	119,859 ± 11,743
	Mixed	82,940 ± 39,384	118,888 ± 9,283

Method	Policy	Mean # of samples per transform	Mean bound on k_m
MCESP+PAC	Single	48,325 ± 21,694	96,877 ± 8,308
	Mixed	46,210 ± 31,505	108,449 ± 11,314
MCESIP+PAC	Single	14,111 ± 4,488	20,843 ± 1,948
	Mixed	14,191 ± 3,475	19,893 ± 2,007

Table 2: Mean effective sample size and theoretical bound across the stages for the multiagent Tiger problem (top) and the 3×2 AUAV (bottom) problem, stratified over method and opponent model type.

4. FUTURE WORK

My final effort involves first introducing the same PAC bounds and optimal policy search present in MCES-P/MCES-IP to the cooperative Multiagent POMDP (MPOMDP) [4] setting (MCES-IP). In this setting we consider the environment where n agents with private observations of the environment communicate with a centralized controller that iterates and tests a series of joint policies, arriving at a mutually-optimal set of policies for each individual agent. I will then follow up with a work on introducing new sampling techniques in order to further improve the runtime of each framework without impacting the overall optimality of the converged policies.

5. ACKNOWLEDGEMENTS

[1] was supported by a grant from Army RDECOM-#W911NF-09-1-0464 to Prashant Doshi, Adam Goodie, and Dan Hall. Section 3 is supported by grant NSF IIS-0845036 to Dr. Doshi. Thanks to Theodore Perkins for emails clarifying MCESP+PAC and Bikramjit Banerjee for his invaluable advice for MCESIP+PAC. Thanks to Yingke Chen and Junhuan Zhang for assistance in developing theory and problem domains.

REFERENCES

[1] Roi Ceren, Prashant Doshi, Matthew Meisel, Adam Goodie, and David Hall. On modeling human learning in sequential games with delayed reinforcements. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 3108–3113. IEEE, 2013.

[2] Theodore J Perkins. Reinforcement learning for pomdps based on action values and stochastic optimization. In *AAAI/IAAI*, pages 199–204, 2002.

[3] Piotr J Gmytrasiewicz and Prashant Doshi. A framework for sequential planning in multi-agent settings. *Journal of Artificial Intelligence Research (JAIR)*, pages 49–79, 2005.

[4] Joao V. Messias, Matthijs Spaan, and Pedro U. Lima. Efficient offline communication policies for factored multiagent pomdps. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1917–1925. 2011.