Concurrent Reinforcement Learning as a Rehearsal for Decentralized Planning Under Uncertainty

(Extended Abstract)

Landon Kraemer
Landon.Kraemer@eagles.usm.edu

Bikramjit Banerjee
Bikramjit.Banerjee@usm.edu

School of Computing
The University of Southern Mississippi
Hattiesburg, MS 39406-001

ABSTRACT

Decentralized partially-observable Markov decision processes (Dec-POMDPs) are a powerful tool for modeling multi-agent planning and decision-making under uncertainty. Prevalent Dec-POMDP solution techniques require centralized computation given full knowledge of the underlying model. Reinforcement learning (RL) based approaches have been recently proposed for distributed solution of Dec-POMDPs without full prior knowledge of the model, but these methods assume that conditions during learning and policy execution are identical. This assumption may not always be necessary and may make learning difficult. We propose a novel RL approach in which agents *rehearse* with information that will not be available during policy execution, yet learn policies that do not explicitly rely on this information. We show experimentally that incorporating such information can ease the difficulties faced by non-rehearsal-based learners, and demonstrate fast, (near) optimal performance on many existing benchmark Dec-POMDP problems.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*

General Terms

Algorithms, Experimentation, Performance

Keywords

Multi-agent reinforcement learning, Decentralized partially observable Markov decision processes

1. INTRODUCTION

Decentralized partially observable Markov decision processes (Dec-POMDPs) offer a powerful model of decentralized decision making under uncertainty and incomplete knowledge. Many exact and approximate solution techniques have been developed for finite-horizon Dec-POMDPs, e.g. [3], but these approaches are not scalable because the underlying problem is provably NEXP-complete [2].

Traditional Dec-POMDP solvers are limited in that they compute the set of prescribed behaviors for agents centrally, and assume that a comprehensive set of model parameters are available a priori; however, multi-agent reinforcement learning (MARL) techniques, have recently been applied [4, 1] to overcome these limitations. These RL-based approaches distribute the policy computation problem among the agents themselves but essentially solve a more difficult problem, because they do not assume the model parameters are known a priori. The hardness of the underlying problem translates to significant sample complexity for RL solvers.

Existing RL-based approaches subject learning agents to the same constraints that they would encounter when executing the learned Dec-POMDP policies, i.e. environment states are hidden and the other agents' actions and observations are invisible. In practice, however, it may actually be easy to allow learning agents to observe some otherwise hidden information *while they are learning*. We view such learning as a *rehearsal* – a phase where agents are allowed to access information that will not be available when executing their learned policies. Since agents must still learn policies that can be executed without hidden information, there is a principled incentive for agents to explore actions that will help them reduce reliance on these *rehearsal features*. Based on these ideas, we present a new approach to RL for Dec-POMDPs – *R*einforcement *L*earning *a*s a *R*ehearsal or RLaR, including a new exploration strategy based on information-gain. Our experiments show that this new approach can nearly optimally solve most existing benchmark Dec-POMDP problems with a low sample complexity.

2. PROBLEM

A Dec-POMDP is a tuple $\langle n, S, A, P, R, \Omega, O\rangle$ where n is a finite number of agents, S is finite set of (unobservable) states, $A = \times_i A_i$ is a set of joint actions(A_i being agent i's individual actions), $P(s'|s, \vec{a})$ gives the probability of transitioning from state s to s' when joint action $\vec{a}$ is executed, $R(s, \vec{a})$ gives the reward agents receive upon executing $\vec{a}$ in s, $\Omega = \times_i \Omega_i$ is a set of joint observations, and $O(\vec{w}|s', \vec{a})$ is the probability that agents jointly observe $\vec{\omega} \in \Omega$ in s' if $\vec{a}$ was executed. Typically, agents do not observe other agents' actions and observations, and thus each agent requires a policy π_i that prescribes an action for each individual action-observation history (h_t) it may encounter. For finite-horizon

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.

Method	Dectiger			Recycling			GridSmall			BoxPush			Alignment		
	T=3	T=4	T=5	T=3	T=4	T=5	T=3	T=4	T=5	T=2	T=3	T=4	T=3	T=4	T=5
RLaR	0.000	0.071	0.053	0.004	0.0028	0.024	0.000	0.000	0.002	0.000	0.000	0.000	0.328	0.089	0.141
Q-Conc	1.711	0.713	1.191	0.358	0.361	0.397	0.643	0.679	0.650	0.376	0.814	0.874	1.246	1.159	0.979

Table 1: Average relative error (compared to known optimal values) for RLaR and Q-Conc for Dectiger, Recycling, GridSmall, BoxPush, and Alignment after 60000 episodes.

Dec-POMDPs, the goal is to find a joint policy $\pi^* = \times_i \pi_i^*$ that maximizes expected reward over T steps of interaction.

3. MOTIVATING DOMAIN

Figure 1 shows two robots, each with two infra-red (IR) emitters and one IR receiver on their front faces. The robots can locomote, rotate, and emit or receive IR signals. They can communicate via IR; however, due to destructive interference, agents must take turns emitting and receiving IR, and, importantly, agents *must be facing each other*. Thus, larger tasks which require communication may require robots to become *aligned*, i.e. face each other.

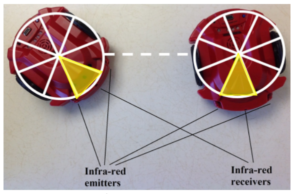

Figure 1:

During execution of this alignment subtask, robots will certainly be unable to observe their orientation (the state) and the actions of their counterparts, but learning under these conditions (i.e. only observing IR signals) is particularly challenging because agents must first become aligned and then coordinate IR signals to even receive meaningful observations. We argue, however, that robots could instead learn in a laboratory setting, where a computer connected to an overhead camera could relay hidden information (such as the state and the actions of their counterparts) to them. This apparatus would be unavailable in most scenarios that require alignment, however, so agents should learn policies that do not rely on this hidden information.

4. REINFORCEMENT LEARNING AS A REHEARSAL (RLAR)

We treat the MARL problem as a *rehearsal* before a final stage performance. During this rehearsal, agents learn under the supervision of a third-party observer that can convey $(s \in S, a_- \in A_-)$ - i.e. the hidden state and the others' actions - to them. The key challenge is that agents still must learn policies that will not rely on these *rehearsal features* because they will not be available during policy execution.

Providing agents with the rehearsal features allows them to maintain their own estimates of the transition function $\hat{P}(s'|s,\vec{a})$, the reward function $\hat{R}(s,\vec{a})$, the initial distribution over states $b_0 \in \Delta S$, and an *individual* observation probability function $\hat{O}_i(\omega_i|s',\vec{a})$. Importantly, agents can use this internal model to construct $P(s, a_-|h_t)$ for *prediction* of rehearsal features under execution conditions (i.e. given only an individual action-observation history h_t).

Our algorithm consists of two stages:

1. Agents treat the problem as the fully-observable MDP $\langle S, A, \hat{P}, \hat{R}\rangle$ and learn an MDP policy via action-quality values $Q(s,t,\vec{a}) = \hat{R}(s,\vec{a}) + \sum_{s' \in S} \hat{P}(s'|s,\vec{a}) \max_{\vec{a}' \in A} \cdot Q(s', t+1, \vec{a'})$.

2. Each agent i learns action-quality values $Q_i(h_t, a)$ which induce a valid policy (i.e. $\pi_i(h_t) = \arg\max_{a \in A_i} Q(h_t, a)$) that can be executed without rehearsal features. Agent i still uses rehearsal features to learn $Q_i(h_t, a)$; however, they are ultimately marginalized out using the agent's predictive model $P(s, a_-|h_t)$, *viz.*

$$Q_i(h_t, a) = \sum_{s \in S, a_- \in A_-} P(s, a_-|h_t) Q_i(s, h_t, \vec{a}). \quad (1)$$

$Q_i(s, h_t, \vec{a})$ represents the immediate reward of executing joint action $\vec{a}$ in state s (i.e. $\hat{R}(s,a)$) plus the future reward when rehearsal features are *not observable* (i.e. $\sum_{\omega \in \Omega_i} P(\omega|s,\vec{a}) \max_{a' in A_i} Q_i((h_t, a, \omega), a')$). In order to transfer the knowledge gained in stage 1, agents initialize $Q_i(s, h_t, \vec{a}) \leftarrow Q_i(s, t, \vec{a})$.

In our example, robots would first learn how to face each other in stage 1, and then they would learn how to coordinate their IR signals to detect alignment in stage 2.

Since agents must ultimately learn policies that are independent of the rehearsal features, there is a principled incentive to explore actions during stage 2 that help predict the rehearsal features. So, in addition to traditional exploration methods, we propose that, with some probability, agents explore actions which are expected to best reduce uncertainty about the hidden features.

Table 1 gives the average relative error (when compared to known optimal policies) of the expected values of policies produced by RLaR (with our new information-based exploration scheme) and Q-Conc - a concurrent learner which does not use rehearsal features - for various benchmark problems, including our new robot alignment[1] problem.

5. ACKNOWLEDGMENTS

We thank Matthjis Spaan for computing the optimal values for the alignment problem. This work was supported in part by the U.S. Army under grant #W911NF-11-1-0124.

6. REFERENCES

[1] B. Banerjee, J. Lyle, L. Kraemer, and R. Yellamraju. Sample bounded distributed reinforcement learning for decentralized pomdps. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12)*, pages 1256–1262, Toronto, Canada, July 2012.

[2] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of markov decision processes. *Mathematics of Operations Research*, 27:819–840, 2002.

[3] M. T. J. Spaan, F. A. Oliehoek, and C. Amato. Scaling up optimal heuristic search in Dec-POMDPs via incremental expansion. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI-11)*, pages 2027–2032, Barcelona, Spain, 2011.

[4] C. Zhang and V. Lesser. Coordinated multi-agent reinforcement learning in networked distributed POMDPs. In *Proc. AAAI-11*, San Francisco, CA, 2011.

[1] See `http://www.cs.usm.edu/~banerjee/alignment` for more details