Reinforcement Learning for Decentralized Planning Under Uncertainty

(Doctoral Consortium)

Landon Kraemer
Landon.Kraemer@eagles.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. But in real world scenarios, model parameters may not be known a priori, or may be difficult to specify. We propose to address these limitations with distributed reinforcement learning (RL).

Categories and Subject Descriptors

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

Keywords

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

1. INTRODUCTION

Decentralized partially-observable Markov decision processes (Dec-POMDPs) offer a formal model for planning in cooperative multiagent systems where agents operate with noisy sensors and actuators, and local information. These models are powerful representations of real-world environments, and allow us to pose multi-agent planning problems in realistic settings. For comparison, Dec-POMDPs allow world states to be known incompletely (and noisily), leading to state-aliasing, whereas decentralized Markov decision processes (Dec-MDPs) make the unrealistic assumption that agents always know their current states completely and accurately. The key problem of *coordination* in a cooperative multi-agent system becomes more realistic, but also concomitantly more challenging, in Dec-POMDPs.

In addition to the state being partially-observable, Dec-POMDPs generally assume that agents cannot communicate their observations and actions to each other. Thus, at each step, an agent must decide which action to execute based only upon the actions it previously executed and the observations it received (i.e. its individual action-observation history); however the state transition, reward, and observation functions depend on *joint* actions, and thus the quality of each agent's policy is dependent on the policies played by all agents or the *joint policy*. The goal of the Dec-POMDP problem, then, is to find the joint policy that maximizes the expected reward received by the agents. The problem of finding this optimal joint policy is NEXP-complete[2].

While many techniques, e.g. [8, 6, 1], have been developed for solving Dec-POMDPs exactly and approximately, they have been primarily centralized and reliant on full knowledge of the model parameters. Furthermore, although centralized solvers yield policies/plans that can be executed in a decentralized fashion, they fail to decentralize the process of policy computation itself.

We propose to address these limitations with distributed reinforcement learning (RL). In the RL approach to Dec-POMDP solution, agents *learn* their own policies, effectively distributing the policy computation problem. Furthermore, the learners do not need to know the model a priori, learning policies by repeated interaction with the environment and with each other instead. Hence, RL holds the promise to be a more realistic solution approach.

Recently, Zhang and Lesser[9] have applied reinforcement learning to a variant of the finite horizon Dec-POMDP problem (*viz.* ND-POMDPs), where agents are organized in a network, and agents' influences on one another are limited to cliques. While our goal is similar, we focus on less structured Dec-POMDPs that are inherently less scalable.

2. CONTRIBUTIONS

Informed Initial Policies: In [4], we investigated two approaches for applying *independent Q-learning* [3] to solve finite-horizon Dec-POMDP problems: one in which agents learn concurrently (Q-Conc), and one in which agents take turns to learn the best responses to each other's policies (Q-Alt). In these independent learning approaches, each agent essentially treats the other agents as part of the environment and is responsible for learning individual action quality values for each individual action-observation history h_t and each individual action a, $Q(h_t, a)$.

Since each agent treats other agents as part of the environment, in Q-Conc, agents must essentially learn against a dynamic environment. This can make learning difficult and can lead to oscillation. Our results showed that Q-Conc per-

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.

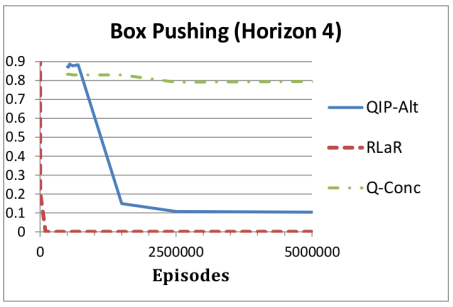

Figure 1: Average relative error (compared to known optimal value) of the solutions produced by QIP-Alt, RLaR, and Q-Conc for horizon-4 Box Pushing.

formed poorly for most Dec-POMDP benchmark problems.

In Q-Alt, agents learn against a static environment; however, agents that have not yet learned a policy must have some initial policy to follow. We proposed a simple yet effective technique for policy initialization that allowed agents to coordinate more effectively, and we found that by using these *informed initial policies* agents could learn (near) optimal values for most benchmark problems using Q-Alt (we call this approach QIP-Alt), and in many cases, the informed initial policies themselves were near-optimal.

Reinforcement Learning as a Rehearsal: While QIP-Alt was able to produce high-quality policies, the sample complexity required was generally large for most problems. Compared to traditional Dec-POMDP solvers, the problem QIP-Alt and Q-Conc solve is more difficult because they do not assume the model is known a priori. The difficulty is exacerbated because the learners subject themselves to the same constraints they will encounter when executing the learned policies, i.e. environment states are hidden and other agents' actions and observations are invisible.

In some scenarios, however, it may actually be easy to allow learning agents to observe some otherwise hidden information *while they are learning*, even if the model is not completely known a priori. We can explicitly break up the problem into distinct "learning" and "execution" phases, and treat the learning problem as a *rehearsal* before a final stage performance. During this learning phase, agents rehearse under the supervision of a third party observer that can convey to them the hidden state and actions executed by other agents. While this additional information can facilitate the rehearsal, agents must learn policies that can indeed be executed without relying on these *rehearsal features*.

In [5] we presented a new RL approach based upon these principles: *R*einforcement *L*earning *a*s a *R*ehearsal or RLaR. We showed that RLaR was able to learn (near) optimal policies with low sample complexity. Figure 1 shows the average error (relative to the known optimal policy value) of the values of policies produced by QIP-Alt, RLaR, and Q-Conc for the Box Pushing problem with horizon 4 (the largest horizon for which any method has optimally solved this problem [7]). While QIP-Alt substantially outperforms Q-Conc for this problem, RLaR outperforms both approaches, achieving optimality in fewer than 20000 episodes.

Robot Alignment Domain: Many benchmark problems have been developed for evaluation of Dec-POMDP solvers in the past [7]. The majority of these problems were developed to illustrate concepts pertaining to methods which *compute* solutions given the full Dec-POMDP model. They also tend to be rather abstract. In [5], we introduced a more concrete Dec-POMDP benchmark, *robot alignment*, that is particularly difficult for *RL-based* Dec-POMDP solvers. We hope robot alignment will spur research to address the difficulties that it poses, leading to better RL-based Dec-POMDP solvers.

3. FUTURE WORK

My work has focused on learning policies for finite-horizon Dec-POMDPs by learning action-quality values $Q(h_t, a)$. Unfortunately, the number of action-observation histories grows exponentially with the horizon T, thus learning these Q-Values is intractable for infinite-horizon Dec-POMDPs. Moving forward, I intend to extend both QIP-Alt and RLaR to the infinite-horizon case.

4. ACKNOWLEDGMENTS

This work was supported in part by the U.S. Army under grant #W911NF-11-1-0124.

5. REFERENCES

[1] C. Amato and S. Zilberstein. Achieving goals in decentralized POMDPs. In *Proceedings of the Eighth International Conference on Autonomous Agents and Multiagent Systems*, pages 593–600, Budapest, Hungary, 2009.

[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] C. Claus and C. Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the 15th National Conference on Artificial Intelligence*, pages 746–752, Menlo Park, CA, 1998. AAAI Press/MIT Press.

[4] L. Kraemer and B. Banerjee. Informed initial policies for learning in dec-pomdps. In *Proceedings of the AAMAS-12 Workshop on Adaptive Learning Agents (ALA-12)*, pages 135–143, Valencia, Spain, June 2012.

[5] L. Kraemer and B. Banerjee. Concurrent reinforcement learning as a rehearsal for decentralized planning under uncertainty (extended abstract). In *Proceedings of the 12th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS-13)*, St. Paul, MN, May 2013. To appear.

[6] J. Pajarinen and J. Peltonen. Periodic Finite State Controllers for Efficient POMDP and DEC-POMDP Planning. In *Proceedings of the 25th Annual Conference on Neural Information Processing Systems (NIPS)*, pages 2636–2644, December 2011.

[7] M. Spaan. Dec-POMDP problem domains and format. `http://users.isr.ist.utl.pt/~mtjspaan/decpomdp/`.

[8] 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.

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