# Increasing Scalability in Algorithms for Centralized and Decentralized Partially Observable Markov Decision Processes

# (Extended Abstract)

Christopher Amato
Department of Computer Science
University of Massachusetts
Amherst, MA 01003 USA
camato@cs.umass.edu.com

## ABSTRACT

Real-world problems contain many forms of uncertainty, but current algorithms for solving sequential decision making problems under uncertainty are limited to small problems due to large resource usage. In my thesis, I study methods to increase the scalability of these approaches such as using memory bounded solutions, sampling or taking advantage of domain structure. I also plan to explore other methods to improve scalability and generate more practical real-world domains on which to test these algorithms.

## 1. INTRODUCTION

Sequential decision making under uncertainty is a thriving research area. In these problems, agents must choose a sequence of actions to maximize a given objective function. The actions must be chosen based on imperfect information about the system state due to stochastic action results and noisy sensors. When multiple cooperative agents are present, each agent must also reason about the action choices of the others in order to maximize joint value while making decisions based solely on local information. Using single and multi-agent sequential decision making under uncertainty a wide range of single and multi-agent problems can be represented, but the computational complexity of solving these models presents an important research challenge.

As a way to address this high complexity, some topics that I study in my thesis include: optimizing agent performance with limited resources, achieving coordination without communication, exploiting goals in multi-agent coordination and using sampling to reason about the future. The models used to represent single and multi-agent problems are the partially observable Markov decision process (POMDPs) and decentralized POMDP (DEC-POMDP). POMDPs represent stochastic actions and uncertainty about the current system state. DEC-POMDPs extend the POMDP model to multiple cooperative agents.

I first discuss the work that I have completed towards studying these problems. I then describe the additional research that I expect to complete for my thesis. Note that because no communication is assumed in my work with the DEC-POMDP model, agents must plan without explicitly sharing information.

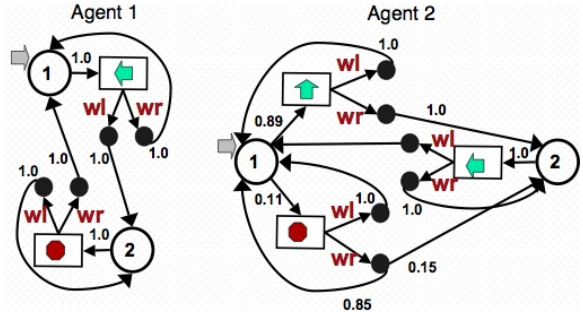


**Figure 1: A set of two node stochastic controllers for a two agent DEC-POMDP.**

## 2. OPTIMIZING CONTROLLERS FOR POMDPS AND DEC-POMDPS

Finite state controllers (depicted in Figure 1) have been shown to effectively model solutions for both infinite-horizon POMDPs [5] and DEC-POMDPs [4, 6]. This approach facilitates scalability as it offers a tradeoff between solution quality and the usage of available resources. That is, a controller may be optimized for a given amount of memory.

Unlike other controller based approaches for POMDPs and DEC-POMDPs, our formulation defines an optimal solution for a given size. This is accomplished by formulating the problem as a nonlinear program (NLP), and exploiting existing nonlinear optimization techniques to solve it. In the POMDP case, parameters are optimized for a fixed-size controller which produces the policy [2]. In the DEC-POMDP case, a set of fixed-size independent controllers is optimized, which when combined, produce the policy [1]. While an overview of how to solve these problems optimally is presented in the thesis, this would often be intractable in practice. As a result, we also evaluate an effective approximation technique using standard NLP solvers.

One premise of our work is that an optimal formulation of the problem facilitates the design of solution techniques that can overcome the limitations of previous controller-based algorithms and produce better stochastic controllers. The general nature of our formulation allows a wide range of solution methods to be used. This results in a search that is more sophisticated than those previously used in controller-based methods. Our approach also provides a framework for which future algorithms can be developed.

Two Agent Tiger Problem $|S| = 2$, $|A| = 3$, $|\Omega| = 2$

| BFS | DEC-BPI | NLP | Goal-directed |
|---|---|---|---|
| -14.1, 12007s | -52.6, 102s | -1.1, 6173s | 5.0, 75s |

Meeting in a Grid Problem $|S| = 16$, $|A| = 5$, $|\Omega| = 2$

| BFS | DEC-BPI | NLP | Goal-directed |
|---|---|---|---|
| 4.2, 17s | 3.6, 2227s | 5.7, 117s | 5.6, 4s |

Box Pushing Problem $|S| = 100$, $|A| = 4$, $|\Omega| = 5$

| BFS | DEC-BPI | NLP | Goal-directed |
|---|---|---|---|
| -2, 1696s | 9.4, 4094s | 54.2, 1824s | 149.9, 199s |

Rover Problems $|S| = 256$, $|A| = 6$, $|\Omega| = 8$

| BFS | DEC-BPI | NLP | Goal-directed |
|---|---|---|---|
| x | -1.1, 11262s | 9.6, 379s | 26.9, 491s |
| x | -1.2, 14069s | 8.1, 438s | 21.5, 956s |

**Table 1: The values produced by each method along with controller size and time in seconds.**

Our results demonstrate that local optimization of the NLP formulation provides concise high quality solutions. In POMDP domains, our technique was competitive in general and outperformed a leading approximate algorithm on a set of problems. In DEC-POMDP domains, our approach significantly outperformed other approximate algorithms, often producing the highest value while using the least amount of time. Further improvement in solution quality is likely as more specialized solution methods are developed.

# 3. ACHIEVING GOALS IN DEC-POMDPS

Another method of improving scalability is to take advantage of structure inherent in domains. One such structure is the achievement of goals, after which the problem terminates. We have demonstrated that when certain goal conditions are present in DEC-POMDPs, this structure can be used to improve scalability and solution quality [3].

To demonstrate this, we have extended the indefinite-horizon framework to decentralized domains using common assumptions – that terminal actions exist for each agent and rewards for non-terminal actions are negative. Under these assumptions we showed that dynamic programming could be adapted to solve the indefinite-horizon problem. We also developed a sample-based algorithm which is able to solve problems with more relaxed goal conditions. For this algorithm, we provided a bound detailing the number of samples required to ensure that the optimal solution is approached. As shown in Table 1 this algorithm was often able to significantly outperform other DEC-POMDP approximate algorithms on a range of goal-directed problems. The approach also provides the framework for sample-based methods to be extended to other classes of decentralized problems.

# 4. FUTURE CONTRIBUTIONS

In addition to the work above, I also plan to work on the following projects for my thesis.

## Incremental policy generation

We are currently developing a method to improve optimal DEC-POMDP algorithms by reducing the necessary search space. This will allow larger problems to be solved optimally and better solutions to be found for other problems. This is accomplished by determining what states are reachable after different action choices are made and observations are seen by the agents. Because not all states will be reachable, not all states will need to be considered to determine a solution. An example of this is a robot observing a wall to its left. The exact system state may not be known, but it can be limited to those states in which the agent has a wall to its left. If the number of solutions can be sufficiently limited, we may be able to identify natural lower complexity subclasses. This approach can also be incorporated in a number of approximate methods, which will improve their performance as well.

## Attribute-based planning

We are also working on other ways to make use of domain information to simplify the planning process in DEC-POMDPs. This approach would utilize user generated or learned information in the form of attributes or landmarks that serve to summarize parts of agent histories. These attributes could include the last location of a wall seen or the number of steps since another agent was observed. Agents could remember only these attributes, allowing planning to be conducted over a smaller set of attributes rather than over all possible histories.

# 5. CONCLUSIONS

In conclusion, my thesis work improves scalability and solution quality for solving uncertain single and multi-agent domains. This is accomplished by such methods as determining the optimal use of a fixed solution space and utilizing domain structure to improve solution search. These approaches perform well in a wide range of problems. In the future, we plan to further improve scalability and solution quality while applying our methods to real-world domains such as e-commence, manufacturing or medical diagnosis.

# 6. REFERENCES


[1] C. Amato, D. S. Bernstein, and S. Zilberstein. Optimizing memory-bounded controllers for decentralized POMDPs. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*, Vancouver, Canada, 2007.

[2] C. Amato, D. S. Bernstein, and S. Zilberstein. Solving POMDPs using quadratically constrained linear programs. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence*, pages 2418–2424, Hyderabad, India, 2007.

[3] C. Amato and S. Zilberstein. Achieving goals in decentralized pomdps. In *Proceedings of the Eighth International Joint Conference on Autonomous Agents and Multiagent Systems*, Budapest, Hungary, 2009.

[4] D. S. Bernstein, E. Hansen, and S. Zilberstein. Bounded policy iteration for decentralized POMDPs. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 1287–1292, Edinburgh, Scotland, 2005.

[5] P. Poupart and C. Boutilier. Bounded finite state controllers. In *Advances in Neural Information Processing Systems*, 16, Vancouver, Canada, 2003.

[6] D. Szer and F. Charpillet. An optimal best-first search algorithm for solving infinite horizon DEC-POMDPs. In *Proceedings of the Sixteenth European Conference on Machine Learning*, Porto, Portugal, 2005.