Interaction Structure and Dimensionality Reduction in Decentralized MDPs

Martin Allen, Marek Petrik, and Shlomo Zilberstein
Computer Science Department
University of Massachusetts
Amherst, MA 01003
{mwallen, marek, shlomo}@cs.umass.edu

Abstract

Decentralized Markov Decision Processes are a powerful general model of decentralized, cooperative multi-agent problem solving. The high complexity of the general problem leads to a focus on restricted models. While worst-case hardness of such reduced problems is often better, less is known about the actual difficulty of given instances. We show tight connections between the structure of agent interactions and the essential dimensionality of various problems. Bounds are placed on problem difficulty, given restrictions on the type and number of interactions between agents. These bounds arise from a bilinear programming formulation of the problem; from such a formulation, a more compact reduced form can be automatically generated, and the original problem rewritten to take advantage of the reduction.

Introduction

Decentralized Markov decision processes (Dec-MDPs) extend MDPs to distributed, cooperative problems, in which each agent possesses only some local, unshared information, and act without full knowledge of what others observe, or plan to do. Finding a globally optimal policy for general Dec-MDPs is NEXP-complete (Bernstein *et al.* 2002). Optimal algorithms face doubly-exponential growth in space and time, rendering even simple problems intractable. An overview is found in Seuken and Zilberstein (2005).

Further, such problems are hard to solve approximately. Rabinovich *et al.* (2003) show that ϵ-approximate solutions are NEXP-hard. Locally optimal methods exist, but give no sharp quality guarantees. Other approaches isolate simpler special classes. Dec-MDPs with independent transition-functions are only NP-complete, and specialized algorithms solve reasonably-sized problems (Becker *et al.* 2004).

We show how to reduce the complexity of certain such problems, which limit agent interactions based on a framework of *events* and *constraints on joint reward*. We isolate the essential dimensionality of such Dec-MDPs via formulation as separable bilinear programs. This leads to some new results. We show how the bilinear program can be converted back into an event-based structure, and that doing so often reduces (and provably never increases) a key factor governing solution algorithm performance. A full version of the results we report here can be found in our technical report (Allen, Petrik, & Zilberstein 2008).

Decentralized MDPs

We focus on a class of problems first introduced by Becker, Lesser, and Zilberstein (2004). In such domains, agents operate on Markov processes that are independent, but for shared influence on the joint reward. These problems are defined based on single-agent MDPs.

Definition 1. *A* Markov decision process *is a tuple:*

$$\mathcal{M} = \langle S,\, A,\, P,\, R,\, \Delta_S,\, T \rangle$$

with individual components:

- S *is a finite set of world states.*
- A *is a finite set of available actions.*
- $P(s, a, s')$ *is a state-transition function.*
- $R : (S \times A) \rightarrow \Re$ *is the reward function.*
- Δ_S *is the initial state-distribution.*
- T *is the finite time-horizon of the problem.*

To define the shared reward structure of the multiagent Dec-MDP version, we require the following further notions.

Definition 2. *For any MDP* $\mathcal{M}$*, an* event from $\mathcal{M}$ *is some set of state-action pairs,*

$$\mathcal{E} = \{\langle s, a\rangle_1,\, \langle s, a\rangle_2, \ldots,\, \langle s, a\rangle_m\} \subseteq (S \times A).$$

Definition 3. *For a pair of MDPs* $\mathcal{M}^1$*,* $\mathcal{M}^2$*, a* reward-constraint on $\mathcal{M}^1$, $\mathcal{M}^2$ *is a triple* $c = \langle \mathcal{E}^1,\, \mathcal{E}^2,\, r_c \rangle$*, where each* $\mathcal{E}^i$ *is an event from* $\mathcal{M}^i$*, and* $r_c \in \Re$.

A reward-constraint defines a shared dependency between two processes, $\mathcal{M}^1$ and $\mathcal{M}^2$. Such structures define shared reward intuitively, and naturally describe many domains (e.g., those where agents have complementary or redundant subtasks). A *feasible set* of constraints assigns at most one reward to any pair of primitive events $(\langle s, a\rangle^1,\, \langle s, a\rangle^2)$.

Definition 4. *For two agents* x *and* y*, a two-agent* decentralized Markov decision process (Dec-MDP) *is a triple*

$$\mathcal{D} = \langle \mathcal{M}^x,\, \mathcal{M}^y,\, \rho \rangle$$

where $\mathcal{M}^x$ *and* $\mathcal{M}^y$ *are MDPs and* ρ*, the* shared-reward structure for $\mathcal{D}$*, is a feasible set of reward-constraints.*

This defines a proper subclass of the general Dec-MDP, properly called *transition and observation-independent, locally and jointly fully observable Dec-MDPs*; for convenience, we simply refer to them as Dec-MDPs.

Bilinear Programs and Dec-MDPs

Petrik and Zilberstein (2007) have demonstrated how this class of Dec-MDPs can be represented and solved as separable bilinear programs. We have simplified that presentation somewhat here. For Dec-MDP $\mathcal{D} = \langle \mathcal{M}^x, \mathcal{M}^y, \rho \rangle$, we define the equivalent bilinear program:

$$\begin{aligned} \text{maximize} \quad & r_1^T x + x^T R y + r_2^T y \\ \text{subject to} \quad & A_x x = \Delta_{S^x} \qquad x \geq 0 \\ & A_y y = \Delta_{S^y} \qquad y \geq 0 \end{aligned} \tag{1}$$

The variable-vectors x and y correspond to the possible state-action pairs from the two MDPs. Each linear reward-vector r_i in the objective function is the individual agent reward, taken from $R^i \in \mathcal{M}^i$. The matrices A_i encode state-visitation information, so that the multiplication in the constraints generates the original state distribution Δ_{S^i}, preserving total flow in the system for each state. Note that all elements so far are linear. However, we get generally non-linear behavior in the objective function via the matrix R, encoding the shared-reward structure of the Dec-MDP.

The *dimensionality* of a bilinear program for a Dec-MDP is simply n, the size of the y-dimension of shared-reward matrix R. Dimensionality has been observed to dominate the complexity of solving such programs, and we prove that it is tightly bound to the shared reward constraint structure. A given Dec-MDP can easily and automatically be reduced to its essential dimensions, using elementary matrix operations to eliminate all constant dimensions of y (along which the best response for agent y is the same for anything x does). Furthermore, we can then go back to the original problem formulation, and replace the reward-constraint structure with a new one, often smaller. This means that we can preserve the often more intuitive structure, based on events, and use algorithms exploiting this sort of structure.

Constraints and Dimensionality

We show that reductions in bilinear dimensionality correspond to a reduced event-based formulation. We establish a theorem, and some related results:

Theorem 1: The essential dimensionality of the bilinear formulation can be bounded above in advance, based on the reward-structure of the original Dec-MDP.

Fact 1: Given a dimensionality-reduced bilinear formulation, we can convert it directly back into an event-based structure with a fixed number of constraints.

Fact 2: The number of such constraints can be bounded above based on the structure of the bilinear program.

These then allow us to establish our main result, namely that by putting a Dec-MDP, $\mathcal{D}$, in reduced bilinear form, and then rewriting it in terms of the induced reward structure, we can only reduce the number of constraints required.

Theorem 2. *Let $\mathcal{D}$ be a Dec-MDP with reward structure of size $|\rho| = n$; let $\mathcal{D}^-$ be the compactified bilinear form of the problem, and ρ^- be the corresponding event-based constraint structure. Then we have that:*

$$\left|\rho^-\right| \leq |\rho|,$$

and the number of necessary constraints never increases.

Applications and Conclusions

These techniques are of more than formal interest. Our ongoing research has applied the presented techniques to a number of standard benchmark domains. For instance, in a multi-access broadcast problem, dimensionality (and the number of necessary events) is reduced to 3 no matter what the original problem size, providing a potentially very large reduction for the event-based specification. When applied to instances of the standard version of the decentralized tiger problem, the number of necessary events is reduced by about a factor of 5, from 108 to 20, with a reward loss of at most 2%. Since even linear reductions in the number of events provides exponential potential speed-ups for specialized Dec-MDP methods like the Coverage Set Algorithm (Becker *et al.* 2004), this transforms such problem instances from ones that are simply infeasible to those that can be practically solved after all.

In analytical terms, this method allows us to reveal the essential structure of dependencies between agents in a Dec-MDP. By converting to the reduced form, find a more minimal set of events suitable for representing a domain. The event-based formulation is very convenient and intuitive, but can be highly inefficient. While simple techniques for merging events exist, they are limited. In fact, problems can be such that there is simply no way of reducing the size of the event formulation, so long as we use state- action pairs. Our reduction process handles this automatically, and has potential applications for multiagent planning systems, where it can be used to prune out unnecessary events, and simplify the structure of designed systems and control hierarchies.

Acknowledgments

This work was supported in part by the Air Force Office of Scientific Research under Grant No. FA9550-05-1-0254 and by the National Science Foundation under Grants No. IIS-0328601.

References

Allen, M.; Petrik, M.; and Zilberstein, S. 2008. Interaction structure and dimensionality in decentralized problem solving. Technical Report UM-CS-2008-11, University of Massachusetts Amherst, Department of Computer Science.

Becker, R.; Zilberstein, S.; Lesser, V.; and Goldman, C. V. 2004. Solving transition independent decentralized MDPs. *Journal of Artificial Intelligence Research* 22:423–455.

Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Math. Operations Research* 27(4):819–840.

Petrik, M., and Zilberstein, S. 2007. Anytime coordination using separable bilinear programs. In *Proceedings of the Twenty-Second Conference on Artificial Intelligence*, 750–755.

Rabinovich, Z.; Goldman, C. V.; and Rosenschein, J. S. 2003. The complexity of multiagent systems: The price of silence. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 1102–1103.

Seuken, S., and Zilberstein, S. 2005. Formal models and algorithms for decentralized control of multiple agents. Technical Report UM-CS-2005-68, University of Massachusetts, Amherst, Department of Computer Science.