Message-Passing Algorithms for Large Structured Decentralized POMDPs

(Extended Abstract)

Akshat Kumar
Department of Computer Science
University of Massachusetts, Amherst
akshat@cs.umass.edu

Shlomo Zllberstein
Department of Computer Science
University of Massachusetts, Amherst
shlomo@cs.umass.edu

ABSTRACT

Decentralized POMDPs provide a rigorous framework for multi-agent decision-theoretic planning. However, their high complexity has limited scalability. In this work, we present a promising new class of algorithms based on probabilistic inference for infinite-horizon ND-POMDPs—a restricted Dec-POMDP model. We first transform the policy optimization problem to that of likelihood maximization in a mixture of dynamic Bayes nets (DBNs). We then develop the Expectation-Maximization (EM) algorithm for maximizing the likelihood in this representation. The EM algorithm for ND-POMDPs lends itself naturally to a simple message-passing paradigm guided by the agent interaction graph. It is thus highly scalable w.r.t. the number of agents, can be easily parallelized, and produces good quality solutions.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Multiagent systems

General Terms

Algorithms, Theory

Keywords

Agent Reasoning: Planning (single and multiagent)

1. INTRODUCTION

Decentralized partially observable MDPs (Dec-POMDPs) have emerged in recent years as an important framework for sequential multi-agent planning under uncertainty [2]. Their expressive power allows them to capture situations when agents must act based on different partial information about the environment and about each other to maximize a global objective function. Many problems such as multi-robot coordination [1], broadcast channel protocols [2] and target tracking by a team of sensor agents [7] can be modeled as a Dec-POMDP. However, their NEXP-Complexity even for two agents has limited their scalability.

To counter such scalability issues, an emerging paradigm is to consider restricted forms of interaction among agents that arise frequently in practice [1, 7]. In particular, we target the Network-Distributed POMDP (ND-POMDP) model that is inspired by the realistic problem of coordinating target tracking sensors [6, 7]. The key assumptions in this model are that of conditional transition independence and conditional observation independence along with factored immediate rewards. We aim to solve infinite-horizon ND-POMDPs using stochastic, finite-state controllers to represent policies. To the best of our knowledge, our work is the first approach to tackle infinite-horizon ND-POMDPs, and the first to solve such problems with 20 agents. We present a promising new class of algorithms, which combines decentralized planning with probabilistic inference. Our work is based on recently developed techniques for planning under uncertainty using probabilistic inference [8, 5].

The expectation-maximization algorithm we develop for ND-POMDPs lends itself naturally to a simple message passing implementation based on the agent interaction graph. In each iteration of EM, an agent only needs to exchange messages with its immediate neighbors. The complexity of computing and propagating such messages is *linear* in the number of links in the agent interaction graph. Thus EM is highly scalable w.r.t. the number of agents allowing us to solve a 20-agent problem. Furthermore, using the DBN representation, we efficiently exploit the highly factored state and action spaces of the ND-POMDP model, allowing us to solve large problems which are highly intractable when using a flat representation. To test the scalability of the EM, we also design new benchmarks that are much larger than the existing ND-POMDP instances. Empirically, EM provides good solution quality when compared against random controllers and a loose upper bound.

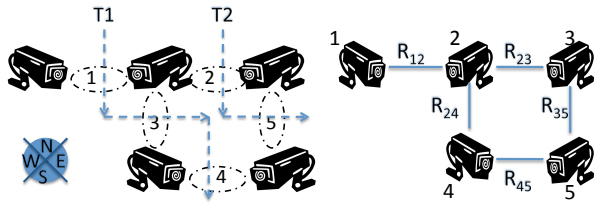
2. THE ND-POMDP MODEL

The ND-POMDP model is motivated by target tracking applications such as the one illustrated in Fig. 1. This example includes a sensor network with 5 camera sensors (or agents). For details, we refer to [3]. In our work, sensors also have an internal state, which indicates battery level. Each action consumed some power. Sensors could *recharge* at some cost and save battery power by being idle.

2.1 Policy evaluation in ND-POMDPs

We present two new results regarding policy evaluation in infinite-horizon ND-POMDPs. The stationary policy of each agent is represented using a fixed size, stochastic finite-state controller (FSC). An FSC for agent i is described by a

Cite as: Message-Passing Algorithms for Large Structured Decentralized POMDPs (Extended Abstract), Akshat Kumar and Shlomo Zilberstein, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp. XXX-XXX.



(a) Camera sensor network (b) Interaction graph

Figure 1: Targets T1 and T2 follow dotted trajectories.

tuple $\langle Q, \pi, \lambda, \nu \rangle$. Q denotes a set of controller nodes q. $\pi : Q \times S_i \rightarrow \Delta A_i$ denotes the stochastic action selection policy, i.e., $\pi_{a_i q s_i} = P(a_i|q, s_i)$. $\lambda : Q \times Y_i \rightarrow \Delta Q$ represents the stochastic node transition model, i.e., $\lambda_{q' q y_i} = P(q'|q, y_i)$. $\nu : Q \rightarrow \Delta Q$ denotes the initial distribution over the controller nodes, i.e., $\nu_q = P(q)$.

THEOREM 1. *The value of starting the joint controller in the configuration* $\mathbf{q}$ *in the joint-state* $\mathbf{s}$ *is factored and additive along the links* l*, that is,* $V(\mathbf{q}, \mathbf{s}) = \sum_l \sum_{a_l} \pi_{a_l q_l s_l}$.

$$\left\{ R_l(s_u, s_l, a_l) + \gamma \sum_{s'_l, s'_u, y_l} p_u p_l \sum_{q'_l} \lambda_{q'_l q_l y_l} V_l(q'_l, s'_l, s'_u) \right\}$$

We can further use the fact that the external state-space S_u is factored; in the sensor network example each factor corresponds to a location of a target.

THEOREM 2. *Let the external state-space* S_u *be factored as* $S_{t_1} \times \ldots \times S_{t_m}$ *with each state-factor having its own independent transition function. Let the immediate reward* R_l *and the transition and observation probabilities of all the agents on a link* l *involve at most the state factors* $S_{t_l} \subseteq S_u$*, then the policy value along a link* l *satisfies:*

$$V_l(q_l, s_l, \mathbf{s_u}) = V_l(q_l, s_l, \mathbf{s_{t_l}}) \;\; s.t. \;\; s_u \in S_u \;\; , s_{t_l} \in S_{t_l}$$

3. EM ALGORITHM FOR ND-POMDPS

Algorithm 1 shows the message-passing implementation of EM. Messages are exchanged locally among immediate neighbors in the interaction graph. The function $f_{ij}([aqs]_j)$ is defined for each edge (i, j) of the interaction graph and both the agents i and j of this edge. The argument of this function, $[aqs]_j$, represents a specific action a, controller node q and internal state s of the agent j. The function is given by the following probabilistic inference in the DBN mixture corresponding to the edge (i, j):

$$f(a, q, s) = \sum_{T=0}^{\infty} P(T) \sum_{t=0}^{T} P_t(\hat{r} = 1, a, q, s|L, T; \theta). \qquad (1)$$

where $\hat{r}$ is the auxiliary reward variable as introduced in [5]. This inference can be implemented using a message-passing paradigm as in [8, 5], which makes EM highly scalable with the number of agents. EM also offers a great potential for parallelization. All the messages in EM for each link can be computed in parallel leading to a significant speedup when using massively parallel computing platforms, such as Google's MapReduce. This further highlights the scalability of EM for large multiagent planning benchmarks.

We experimented on several sensor network benchmarks from [7, 3]. In addition, we also used a 20-agent benchmark

Algorithm 1: *Message-Passing for ND-POMDPs*

```
Initialize parameters π_{[aqs]_i} randomly for each agent i
for iter = 1 until MaxIter do
    for Agent i = 1 until n do
        for each agent j ∈ Ne(i) do
            Compute f_ij([aqs]_j) for each [aqs]_j
            Send message μ_{i→j} = f_ij to agent j
        end
    end
    for Agent i = 1 until n do
        Receive all messages μ_{j→i} from j ∈ Ne(i)
        Set π*_{aqs} = (1/C_qs) Σ_{j∈Ne(i)} μ_{j→i}([aqs])
    end
    Set π_{[aqs]_i} ← π*_{[aqs]_i} for each agent i
end
```

from [4]. For all these problem, EM converged quickly, often within 200 iterations. When compared against random controllers, EM provided significantly better solution quality. Against a loosely computed upper bound, EM provide a solution within 45% of the bound.

4. CONCLUSION

We developed a new approach for solving infinite-horizon ND-POMDPs using probabilistic inference in a mixture of dynamic Bayes nets. We then derived the EM algorithm for iteratively improving the policy. The resulting algorithm can be easily implemented using local message passing among the agents. Each message can be computed efficiently and involves only the parameters of agents connected to a single interaction link, making this message passing scheme particularly scalable w.r.t. the number of agents and links in the interaction graph. Another practical advantage of EM is that it naturally lends itself to parallelization; our experiments on a multi-core machine showed linear speedup.

Acknowledgments

Support for this work was provided in part by the NSF Grant No. IIS-0812149 and AFOSR Grant No. FA9550-08-1-0181.

5. REFERENCES

[1] R. Becker, S. Zilberstein, V. Lesser, and C. V. Goldman. Solving transition independent decentralized markov decision processes. *JAIR*, 22:423–455, 2004.

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

[3] A. Kumar and S. Zilberstein. Constraint-based dynamic programming for decentralized pomdps with structured interactions. In *AAMAS*, pages 561–568, 2009.

[4] A. Kumar and S. Zilberstein. Event-detecting multi-agent mdps: complexity and constant-factor approximation. In *IJCAI*, pages 201–207, 2009.

[5] A. Kumar and S. Zilberstein. Anytime planning for decentralized POMDPs using expectation maximization. In *Uncertainty in Artificial Intelligence*, pages 294–301, 2010.

[6] V. Lesser, M. Tambe, and C. L. Ortiz, editors. *Distributed Sensor Networks: A Multiagent Perspective*. Kluwer Academic Publishers, Norwell, MA, USA, 2003.

[7] R. Nair, P. Varakantham, M. Tambe, and M. Yokoo. Networked distributed POMDPs: A synthesis of distributed constraint optimization and POMDPs. In *AAAI*, pages 133–139, 2005.

[8] M. Toussaint, S. Harmeling, and A. Storkey. Probabilistic inference for solving (PO)MDPs. Technical Report EDIINF-RR-0934, University of Edinburgh, 2006.