A Particle Filtering Based Approach to Approximating Interactive POMDPs

Prashant Doshi and **Piotr J. Gmytrasiewicz**
Dept. of Computer Science,
University of Illinois at Chicago, IL 60607
{pdoshi,piotr}@cs.uic.edu

Abstract

POMDPs provide a principled framework for sequential planning in single agent settings. An extension of POMDPs to multiagent settings, called interactive POMDPs (I-POMDPs), replaces POMDP belief spaces with interactive hierarchical belief systems which represent an agent's belief about the physical world, about beliefs of the other agent(s), about their beliefs about others' beliefs, and so on. This modification makes the difficulties of obtaining solutions due to complexity of the belief and policy spaces even more acute. We describe a method for obtaining approximate solutions to I-POMDPs based on particle filtering (PF). We utilize the *interactive PF* which descends the levels of interactive belief hierarchies and samples and propagates beliefs at each level. The interactive PF is able to deal with the belief space complexity, but it does not address the policy space complexity. We provide experimental results and chart future work.

Introduction

Partially observable Markov decision processes (POMDPs) offer a principled framework for sequential decision-making. Their solutions map an agent's states of belief about the environment to policies, but optimal solutions are difficult to compute due to two sources of intractability: The complexity of the belief representation sometimes called the curse of dimensionality, and the complexity of the space of the policies, also called the curse of history. In this paper we focus on an extension of POMDPs to multiagent settings, called interactive POMDPs (I-POMDPs) (Gmytrasiewicz & Doshi 2005). The solutions to I-POMDPs are defined analogously to solutions of POMDPs, but complexity of the belief space is even greater; they include beliefs about the physical environment, and possibly the agent's beliefs about other agents' beliefs, their beliefs about others, and so on. This added complexity of interactive beliefs exasperates difficulties brought about by the curses of dimensionality and history. To address the problem of belief dimensionality we propose using an interactive version of the particle filtering (PF) approach.

Though POMDPs can be used in multiagent settings, it is so only under the strong assumption that the other agent's behavior be adequately represented implicitly within the state transition function. The approach taken in I-POMDPs is to include sophisticated models of other agents in the state space. These models called *intentional* models, ascribe beliefs, preferences, and rationality to others and are analogous to the notion of agent types in Bayesian games. An agent's beliefs are then called interactive beliefs, and they are nested analogously to the hierarchical belief systems considered in game theory and in theoretical computer science (Mertens & Zamir 1985; Brandenburger & Dekel 1993; Fagin *et al.* 1995; Heifetz & Samet 1998).

Since an agent's belief is defined over other agents' models, which may be a complex continuous space, sampling methods, which are immune to the high dimensionality of the underlying space are a promising approach. In (Doshi & Gmytrasiewicz 2005), we adapted PF (Doucet, Freitas, & Gordon 2001; Gordon, Salmond, & Smith 1993), and more specifically the bootstrap filter, resulting in the *interactive PF*, to approximate the state estimation in multiagent settings. In this paper, we combine the interactive PF with value iteration to present the first method for computing *approximately* optimal policies in the I-POMDP framework. We derive error bounds of our approach, and empirically demonstrate its performance on simple test problems.

Particle filters have previously been successfully applied to approximate the belief update in continuous state single agent POMDPs. While Thrun(2000) integrates PF with Q-learning to learn the policy, Poupart et al.(2001) assume the existence of an exact value function and present an error bound analysis of using particle filters. Loosely related to our work are the sampling algorithms that appear in (Ortiz & Kaelbling 2000) for selecting actions in influence diagrams, but this work does not focus on sequential decision making. In the multiagent setting, PFs have been employed for collaborative multi-robot localization (Fox *et al.* 2000). There, the emphasis was on predicting the position of the robot, and not the decisions and actions of the other robots.

Overview of Finitely Nested I-POMDPs

I-POMDPs (Gmytrasiewicz & Doshi 2005) generalize POMDPs to handle multiple agents. They do this by including models of other agents in the state space. We will limit the discussion to intentional models, analogous to *types* in Bayesian games, which include all private information influencing the other agents' behavior. For simplicity of pre-

sentation let us consider an agent, i, that is interacting with one other agent, j.

I-POMDP A *finitely nested interactive POMDP* of agent i, *I-POMDP*$_{i,l}$, is: *I-POMDP*$_{i,l} = \langle IS_{i,l}, A, T_i, \Omega_i, O_i, R_i \rangle$ where:

• $IS_{i,l}$ denotes a set of interactive states defined as, $IS_{i,l} = S \times \Theta_{j,l-1}$, for $l \geq 1$, and $IS_{i,0} = S$, where S is the set of states of the physical environment, and $\Theta_{j,l-1}$ is the set of $(l-1)^{th}$ level *intentional models* of agent j: $\theta_{j,l-1} = \langle b_{j,l-1}, A, \Omega_j, T_j, O_j, R_j, OC_j \rangle$. $b_{j,l-1}$ is the agent j's belief nested to the level $(l-1)$ and OC_j is j's optimality criterion. Rest of the notation is standard. Let us rewrite $\theta_{j,l-1}$ as, $\theta_{j,l-1} = \langle b_{j,l-1}, \widehat{\theta}_j \rangle$, where $\widehat{\theta}_j \in \widehat{\Theta}_j$ includes all elements of the intentional model other than the belief and is called the agent j's *frame*. We refer the reader to (Gmytrasiewicz & Doshi 2005) for a detailed inductive definition of the state space.

• $A = A_i \times A_j$ is the set of joint moves of all agents

• T_i is a transition function, $T_i : S \times A \times S \rightarrow [0,1]$ which describes results of agents' actions

• Ω_i is the set of agent i's observations

• O_i is an observation function, $O_i : S \times A \times \Omega_i \rightarrow [0,1]$

• R_i is defined as, $R_i : IS_i \times A \rightarrow \mathbf{R}$. While an agent is allowed to have preferences over physical states and models of other agents, usually only the physical state will matter.

Belief Update

There are two differences that complicate a belief update in multiagent settings, when compared to single-agent ones. First, since the state of the physical environment depends on the actions performed by both agents, the prediction of how the physical state changes has to be made based on the predicted actions of the other agent. The probabilities of other's actions are obtained based on its models. Second, changes in the models of the other agent – update of the other agent's beliefs due to its new observation – has to be included. For better understanding, we decompose the I-POMDP belief update into two steps:

• *Prediction:* When an agent, say i, performs an action a_i^{t-1}, and agent j performs a_j^{t-1}, the predicted belief state is,

$$Pr(is^t | a_i^{t-1}, a_j^{t-1}, b_{i,l}^{t-1}) = \int_{IS^{t-1}: \widehat{\theta}_j^{t-1} = \widehat{\theta}_j^t} b_{i,l}^{t-1}(is^{t-1}) \times Pr(a_j^{t-1} | \theta_{j,l-1}^{t-1}) T_i(s^{t-1}, a_i^{t-1}, a_j^{t-1}, s^t) \times \sum_{o_j^t} O_j(s^t, a_i^{t-1}, a_j^{t-1}, o_j^t) \times \delta(SE_{\widehat{\theta}_j^t}(b_{j,l-1}^{t-1}, a_j^{t-1}, o_j^t) - b_{j,l-1}^t) d(is^{t-1})$$

where δ is the Dirac-delta function, $SE(\cdot)$ is an abbreviation for the belief update, $Pr(a_j^{t-1} | \theta_{j,l-1}^{t-1})$ is the probability that a_j^{t-1} is Bayes rational for the agent described by $\theta_{j,l-1}^{t-1}$.

• *Correction:* When agent i perceives an observation, o_i^t, the corrected belief state is a weighted sum of the predicted belief states for each possible action of j,

$$Pr(is^t | o_i^t, a_i^{t-1}, b_{i,l}^{t-1}) = \alpha \sum_{a_j^{t-1}} O_i(s^t, a_i^{t-1}, a_j^{t-1}, o_i^t) \times Pr(is^t | a_i^{t-1}, a_j^{t-1}, b_{i,l}^{t-1})$$

where α is the normalizing constant. Proofs are in (Gmytrasiewicz & Doshi 2005).

If j is also modeled as an I-POMDP, then i's belief update invokes j's belief update (via the term $SE_{\widehat{\theta}_j^t}(b_{j,l-1}^{t-1}, a_j^{t-1}, o_j^t)$), which in turn invokes i's belief update and so on. This recursion in belief nesting bottoms out at the 0^th level. At this level, belief update of the agent reduces to a POMDP belief update. [1] For additional details on I-POMDPs, and how they compare with other multiagent planning frameworks, see (Gmytrasiewicz & Doshi 2005).

Value Iteration

Each level l belief state in I-POMDP has an associated value reflecting the maximum payoff the agent can expect in this belief state:

$$V^t(\langle b_{i,l}, \widehat{\theta}_i \rangle) = \max_{a_i \in A_i} \left\{ \int_{is} ER_i(is, a_i) b_{i,l}(is) d(is) + \gamma \sum_{o_i \in \Omega_i} Pr(o_i | a_i, b_{i,l}) V^{t-1}(\langle SE_{\widehat{\theta}_i}(b_{i,l}, a_i, o_i), \widehat{\theta}_i \rangle) \right\} \quad (1)$$

where, $ER_i(is, a_i) = \sum_{a_j} R_i(is, a_i, a_j) Pr(a_j | \theta_{j,l-1})$ (since $is = (s, \theta_{j,l-1})$). Eq. 1 is a basis for value iteration in I-POMDPs, and can be succinctly rewritten as $V^t = HV^{t-1}$, where H is commonly known as the backup operator. Analogous to POMDPs, H is both isotonic and contracting, thereby making the value iteration convergent (Gmytrasiewicz & Doshi 2005).

Agent i's optimal action, a_i^*, is an element of the set of optimal actions for the belief state, $OPT(\theta_i)$, defined as:

$$OPT(\langle b_{i,l}, \widehat{\theta}_i \rangle) = \underset{a_i \in A_i}{argmax} \left\{ \int_{is} ER_i(is, a_i) b_{i,l}(is) d(is) + \gamma \sum_{o_i \in \Omega_i} Pr(o_i | a_i, b_{i,l}) V^{t-1}(\langle SE_{\widehat{\theta}_i}(b_{i,l}, a_i, o_i), \widehat{\theta}_i \rangle) \right\}$$

Approximation Technique

As we mentioned, there is a continuum of intentional models of an agent. Since an agent is unaware of the true models of interacting agents *ex ante*, it must maintain a belief over all possible candidate models. The complexity of this space precludes practical implementations of I-POMDPs for all but the simplest settings. Approximations based on sampling use a finite set of sample points to represent a complete belief state.

In order to sample from nested beliefs we need a language to represent them. We introduced a polynomial based representation for the nested beliefs in (Doshi & Gmytrasiewicz 2005), which we briefly review. i's level 0 belief is a probability distribution over S, i.e., a vector of length $|S|$. i's first level belief, which includes a distribution over j's level 0 beliefs is represented using a polynomial over j's level 0 beliefs, for each state and j's frames. Formally, $b_{i,1}$ is represented by $\{f_{i,1}^1, f_{i,1}^2, \ldots, f_{i,1}^{|S||\widehat{\Theta}_j|}\}$. Each polynomial $f_{i,1}^k$ can be written in a parametric form: $f_{i,1}^k = \langle d, c_1, c_2, \ldots, c_{(d+1)^{|S|-1}} \rangle$, where $d \in \mathbb{N}$ is the

[1] The 0^{th} level model is a POMDP: other agent's actions are treated as exogenous events and folded into T, O, and R.

degree, and $c \in \mathbb{R}$ is a coefficient of $f^k_{i,1}$. To be a legal probability distribution the areas under the polynomials must sum to 1. i's level 2 belief is represented using a tuple of polynomials over parameters of each of j's $|S||\widehat{\Theta}_i|$ level 1 polynomials ($f^k_{j,1}$), for each state and j's frame. Within each tuple, the first polynomial is defined over the degree d, of $f^k_{j,1}$. For each d, the remaining $(d+1)^{|S|-1}$ polynomials are defined over each coefficient of $f^k_{j,1}$. In other words, if j's level 1 belief is represented by $\{f^1_{j,1}, f^2_{j,1}, \ldots, f^{|S||\widehat{\Theta}_i|}_{j,1}\}$, then i's level 2 belief, $b_{i,2}$, is represented by $\{\{\langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^1, \ldots, \langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^{|S||\widehat{\Theta}_i|}\}_1, \ldots, \{\langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^1, \ldots, \langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^{|S||\widehat{\Theta}_i|}\}_{|S||\widehat{\Theta}_j|}\}$. Here the polynomials in each of the innermost tuples represent distributions over the parameters of the corresponding level 1 polynomial of j (that with the same superscript as the tuple). We specify higher levels of beliefs analogously.

Example 1: To illustrate our representation, we use the multiagent tiger game (Gmytrasiewicz & Doshi 2005) as an example. An example level 1 belief of i, $b_{i,1}$, in the tiger game is one according to which i is uninformed about j's beliefs and about the location of the tiger. Polynomial representation of this belief is $\{f^{TL}_{i,1}, f^{TR}_{i,1}\}$, where the polynomials $f^{TL}_{i,1} = f^{TR}_{i,1} = \langle 0, 0.5\rangle$. A level 2 belief of i is the one in which i considers increasingly complex level 1 beliefs of j (i.e. $f_{j,1}$ of higher degrees) as less likely (Occam's Razor), and is uninformed of the location of the tiger. We express this belief $b_{i,2}$ by $\{\langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^{TL}, \langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^{TR}\}_{TL}$, $\{\langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^{TL}, \langle f^1_{i,2}, f^2_{i,2}, \ldots\rangle^{TR}\}_{TR}\}$, $f^1_{i,2}$ in each tuple is the parametric form of the normalized Taylor series expansion of 2^{-d} defined over the degree d of j's level 1 polynomials: $\alpha \sum_{n=0}^{\infty} \frac{1}{k!}(-1)^n ln(2)^n (d - d_{max})^n$, where α is the normalizing constant and d_{max} is an upper bound on d.[2] The remaining upto $d_{max}+1$ polynomials in each tuple are p.d.f.s over the coefficients of j's level 1 polynomials and are of degree 0.

Interactive Particle Filter

The interactive PF (Doshi & Gmytrasiewicz 2005), similar to basic particle filtering, requires the key steps of *importance sampling* and *selection*. The resulting algorithm, described in Fig. 1, inherits the convergence properties of the original PF (Doucet, Freitas, & Gordon 2001). It requires an initial set of N particles, $\widetilde{b}^{t-1}_{k,l}$, that is approximately representative of the agent's belief, along with the action, a^{t-1}_k, the observation, o^t_k, and the level of belief nesting, $l > 0$. Each particle in the sample set represents the agent's possible interactive state. Here, k will stand for either agent i or j, and $-k$ for the other agent, j or i, as appropriate. We generate $\widetilde{b}^{t-1}_{k,l}$ by recursively sampling N particles from beliefs represented using polynomials at each level of nesting. The particle filtering proceeds by *propagating* each particle forward in time. However, as opposed to the basic particle filtering, this is not a one-step process. In order to perform the propagation, other agent's action must be known. This is obtained by solving the other agent's model (using the algorithm OPTIMALPOLICY described in the next subsection) to get its policy, and using its belief (contained in the particle) to find a distribution over its actions (line 3 in Fig. 1). Additionally, analogously to the exact belief update, for each of the other agent's possible observations, we must obtain its next belief state (line 6). If $l > 1$, updating the other agent's belief requires invoking the interactive PF for performing its belief update (lines 12–14). This recursion in depth of the belief nesting terminates when the level of nesting becomes one, and a POMDP belief update is performed (lines 8–10). Though the propagation step generates $|\Omega_{-k}|N$ appropriately weighted particles, we *resample* N particles out of these (line 19), using an unbiased resampling scheme. A visualization of the implementation is shown in Fig. 2.

Function I-PARTICLEFILTER($\widetilde{b}^{t-1}_{k,l}, a^{t-1}_k, o^t_k, l > 0$)
returns $\widetilde{b}^t_{k,l}$
1. $\widetilde{b}^{tmp}_{k,l} \leftarrow \phi$, $\widetilde{b}^t_{k,l} \leftarrow \phi$
   Importance Sampling
2. **for all** $is^{(n),t-1}_k = \langle s^{(n),t-1}, \theta^{(n),t-1}_{-k}\rangle \in \widetilde{b}^{t-1}_{k,l}$ **do**
3. $Pr(A_{-k}|\theta^{(n),t-1}_{-k}) \leftarrow$ OPTIMALPOLICY($\theta^{(n),t-1}_{-k}, l-1$)
4. Sample $a^{t-1}_{-k} \sim Pr(A_{-k}|\theta^{(n),t-1}_{-k})$
5. Sample $s^{(n),t} \sim T_k(S^t|a^{t-1}_k, a^{t-1}_{-k}, s^{(n),t-1})$
6. **for all** $o^t_{-k} \in \Omega_{-k}$ **do**
7. **if** (l = 1) **then**
8. $b^{(n),t}_{-k} \leftarrow$ POMDPBELIEFUPDATE($b^{(n),t-1}_{-k}, a^{t-1}_{-k}, o^t_{-k}$)
9. $\theta^{(n),t}_{-k} \leftarrow \langle b^{(n),t}_{-k}, \widehat{\theta}^{(n)}_{-k}\rangle$
10. $is^{(n),t}_k \leftarrow \langle s^{(n),t}, \theta^{(n),t}_{-k}\rangle$
11. **else**
12. $\widetilde{b}^{(n),t}_{-k} \leftarrow$ I-PARTICLEFILTER($\widetilde{b}^{(n),t-1}_{-k}, a^{t-1}_{-k}, o^t_{-k}, l-1$)
13. $\theta^{(n),t}_{-k} \leftarrow \langle \widetilde{b}^{(n),t}_{-k}, \widehat{\theta}^{(n)}_{-k}\rangle$
14. $is^{(n),t}_k \leftarrow \langle s^{(n),t}, \theta^{(n),t}_{-k}\rangle$
15. Weight $is^{(n),t}_k$: $w^{(n)}_t \leftarrow O_{-k}(o^t_{-k}|s^{(n),t}, a^{t-1}_k, a^{t-1}_{-k})$
16. Adjust weight: $w^{(n)}_t \overset{\times}{\leftarrow} O_k(o^t_k|s^{(n),t}, a^{t-1}_k, a^{t-1}_{-k})$
17. $\widetilde{b}^{tmp}_{k,l} \overset{\cup}{\leftarrow} (is^{(n),t}_k, w^{(n)}_t)$
18. Normalize all $w^{(n)}_t$ so that $\sum_{n=1}^{N} w^{(n)}_t = 1$
    Selection
19. Resample with replacement N particles $\{is^{(n),t}_k, n = 1...N\}$ from the set $\widetilde{b}^{tmp}_{k,l}$ according to the importance weights.
20. $\widetilde{b}^t_{k,l} \leftarrow \{is^{(n),t}_k, n = 1...N\}$
21. **return** $\widetilde{b}^t_{k,l}$
end function

Figure 1: Interactive PF for approximating the I-POMDP belief update. A nesting of PFs is used to update all levels of the belief.

Value Iteration

Because the interactive PF represents each belief of agent i, $b_{i,l}$, using a set of N particles, $\widetilde{b}_{i,l}$, a value backup operator which operates on samples is needed. Let $\widetilde{H}$ denote the required backup operator, and $\widetilde{V}$ the approximate value function, then the backup operation, $\widetilde{V}^t = \widetilde{H}\widetilde{V}^{t-1}$, is:

[2] We use $2^{-K(x)}$ where $K(\cdot)$ is the Kolmogorov complexity as a mathematical formalization of Occam's razor (Li & Vitanyi 1997).

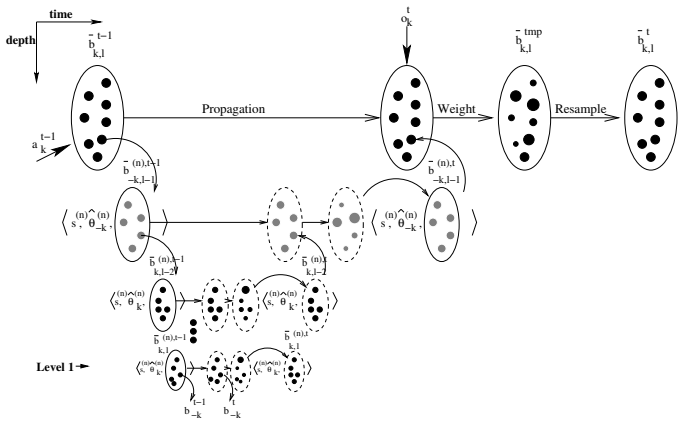

Figure 2: An illustration of the nesting in the interactive PF. Colors black and gray distinguish filtering for the two agents. Because the propagation step involves updating the other agent's beliefs, we perform particle filtering on its beliefs. The filtering terminates when it reaches the level 1 nesting, where an exact POMDP belief update is performed for the other agent.

$$\widetilde{V}^t(\langle \widetilde{b}_{i,l}, \widehat{\theta}_i \rangle) = \max_{a_i \in A_i} \left\{ \frac{1}{N} \sum_{is^{(n)} \in \widetilde{b}_{i,l}} ER_i(is^{(n)}, a_i) + \gamma \sum_{o_i \in \Omega_i} Pr(o_i | a_i, \widetilde{b}_{i,l}) \widetilde{V}^{t-1}(\langle \text{I-PF}(\widetilde{b}_{i,l}, a_i, o_i), \widehat{\theta}_i \rangle) \right\} \quad (2)$$

where ER_i is as defined previously, and I-PF denotes the belief update implemented using the interactive PF. The set of optimal actions at a given approximate belief, $\text{OPT}(\langle \widetilde{b}_{i,l}, \widehat{\theta}_i \rangle)$, is then calculated by returning the actions that have the maximum value.

Equation 2 is analogous to the equation 1 with exact integration replaced by Monte Carlo integration, and the exact belief update replaced with the interactive particle filter. Note that $\widetilde{H} \to H$ as $N \to \infty$. The algorithm for computing an approximately optimal finite horizon policy tree using value iteration when $l > 0$ is given in Fig. 3. When $l = 0$, the algorithm reduces to the POMDP policy tree computation which is carried out exactly.

Convergence and error bounds

The use of randomizing techniques such as particle filters means that value iteration does not necessarily converge. This is because, unlike the exact belief update, posteriors generated by the particle filter with finitely many particles are not guaranteed to be identical for identical input. The non-determinism of the approximate belief update rules out isotonicity and contraction for $\widetilde{H}$ as $N \to \infty$. [3]

Our inability to guarantee convergence implies that we must approximate an infinite horizon policy with the approximately optimal finite horizon policy tree. Let V^* be the value of the optimal infinite horizon policy, $\widetilde{V}^t$ be the value of the approximate and V^t be the value of the optimal

[3] One may turn particle filters into deterministic belief update operators (de-randomization) by generating several posteriors from the same input. A representative posterior is then formed by taking a convex combination of the different posteriors.

Function OPTIMALPOLICY($\theta_k, l > 0$) **returns** $\Delta(A_k)$
1. $\widetilde{b}^0_{k,l} \leftarrow \{is^{(n)}_k, n = 1...N | is^{(n)}_k \sim b_{k,l} \in \theta_k\}$
   Reachability Analysis
2. reach(0) $\leftarrow \widetilde{b}^0_{k,l}$
3. **for** $t \leftarrow 1$ **to** T **do**
4. reach(t) $\leftarrow \phi$
5. **for all** $\widetilde{b}^{t-1}_{k,l} \in$ reach($t-1$), $a_k \in A_k$, $o_k \in \Omega_k$ **do**
6. reach(t) $\overset{\cup}{\leftarrow}$ I-PARTICLEFILTER($\widetilde{b}^{t-1}_{k,l}, a_k, o_k, l$)
   Value Iteration
7. **for** $t \leftarrow T$ **downto** 0 **do**
8. **for all** $\widetilde{b}^t_{k,l} \in$ reach(t) **do**
9. $\widetilde{V}^{T-t,l}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) \leftarrow -\infty$, OPT($\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle$) $\leftarrow \phi$
10. **for all** $a_k \in A_k$ **do**
11. $\widetilde{V}^{T-t}_{a_k}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) \leftarrow 0$
12. **for all** $is^{(n),t}_k = \langle s^{(n),t}, \theta^{(n)}_{-k} \rangle \in \widetilde{b}^t_{k,l}$ **do**
13. $Pr(A_{-k} | \theta^{(n)}_{-k}) \leftarrow$ OPTIMALPOLICY($\theta^{(n)}_{-k}, l-1$)
14. **for all** $a_{-k} \in A_{-k}$ **do**
15. $\widetilde{V}^{T-t}_{a_k}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) \overset{+}{\leftarrow} \frac{1}{N} R(s^{(n),t}, a_k, a_{-k}) Pr(a_{-k} | \theta^{(n)}_{-k})$
16. **if** ($t < T$) **then**
17. **for all** $o_k \in \Omega_k$ **do**
18. sum $\leftarrow 0$, $\widetilde{b}^{t+1}_{k,l} \leftarrow$ reach($t+1$)[$|\Omega_k| a_k + o_k$]
19. **for all** $is^{(n),t+1}_k \in \widetilde{b}^{t+1}_{k,l}$, $is^{(n),t}_k \in \widetilde{b}^t_{k,l}$ **do**
20. $Pr(A_{-k} | \theta^{(n)}_{-k}) \leftarrow$ OPTIMALPOLICY($\theta^{(n)}_{-k}, l-1$)
21. **for all** $a_{-k} \in A_{-k}$ **do**
22. sum $\overset{+}{\leftarrow} O_k(o_k | s^{(n),t+1}, a_k, a_{-k}) \times Pr(is^{(n),t+1}_k | is^{(n),t}_k, a_k, a_{-k}) Pr(a_{-k} | \theta^{(n)}_{-k})$
23. $\widetilde{V}^{T-t,l}_{a_k}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) \overset{+}{\leftarrow} \frac{\gamma}{N} \times$ sum $\times \widetilde{V}^{T-t-1}(\langle \widetilde{b}^{t+1}_{k,l}, \widehat{\theta}_k \rangle)$
24. **if** $\widetilde{V}^{T-t}_{a_k}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) \geq \widetilde{V}^{T-t}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle)$) **then**
25. **if** $(\widetilde{V}^{T-t}_{a_k}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) > \widetilde{V}^{T-t}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle)$ **then**
26. $\widetilde{V}^{T-t}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle) \leftarrow \widetilde{V}^{T-t}_{a_k}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle)$
27. OPT($\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle$) $\leftarrow \phi$
28. OPT($\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle$) $\overset{\cup}{\leftarrow} a_k$
29. **for all** $a_k \in A_k$ **do**
30. **if** ($a_k \in$ OPT($\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle$)) **then**
31. $Pr(a_k | \theta_k) \leftarrow \frac{1}{|\text{OPT}(\langle \widetilde{b}^t_{k,l}, \widehat{\theta}_k \rangle)|}$
32. **else**
33. $Pr(a_k | \theta_k) \leftarrow 0$
34. **return** $Pr(A_k | \theta_k)$

end function

Figure 3: Algorithm for computing an approximately optimal finite horizon policy tree given a model containing an initial sampled belief. When $l = 0$, the exact POMDP policy tree is computed.

t-horizon policy tree, then the error bound (using the supremum norm $||\cdot||$) is, $||V^* - \widetilde{V}^t|| = ||V^* - V^t + V^t - \widetilde{V}^t|| \leq ||V^* - V^t|| + ||V^t - \widetilde{V}^t||$. Note that the first term is bounded by $\gamma^t ||V^* - V^0||$. The bound for the second term is calculated below:

$$\begin{aligned}
\mathcal{E}^t &= ||\widetilde{V}^t - V^t|| \\
&= ||\widetilde{H}\widetilde{V}^{t-1} - HV^{t-1}|| \\
&= ||\widetilde{H}\widetilde{V}^{t-1} - H\widetilde{V}^{t-1} + H\widetilde{V}^{t-1} - HV^{t-1}|| && \text{(add zero)} \\
&\leq ||\widetilde{H}\widetilde{V}^{t-1} - H\widetilde{V}^{t-1}|| + ||H\widetilde{V}^{t-1} - HV^{t-1}|| && (\Delta \text{ inequality}) \\
&\leq ||\widetilde{H}\widetilde{V}^{t-1} - H\widetilde{V}^{t-1}|| + \gamma ||\widetilde{V}^{t-1} - V^{t-1}|| && \text{(contracting } H) \\
&\leq ||\widetilde{H}\widetilde{V}^{t-1} - H\widetilde{V}^{t-1}|| + \gamma \mathcal{E}^{t-1}
\end{aligned}$$

We will turn our attention to $||\widetilde{H}\widetilde{V}^{t-1} - H\widetilde{V}^{t-1}||$. In the analysis that follows we focus on level 1 beliefs. Let $\dot{V}^t =$

$H\widetilde{V}^{t-1}$, $\widetilde{V}^t = \widetilde{H}\widetilde{V}^{t-1}$, and $b_{i,1}$ be the singly-nested belief where the worst error is made: $b_{i,1} = \underset{b_{i,1} \in B_{i,1}}{\text{argmax}} |\dot{V}^t - \widetilde{V}^t|$.

Let $\widetilde{\alpha}$ be the policy tree (alpha vector) that is optimal at $\widetilde{b}_{i,1}$ (the sampled estimate of $b_{i,1}$), and $\dot{\alpha}$ be the policy tree that is optimal at $b_{i,1}$. We will use Chernoff-Hoeffding (C-H) upper bounds (Theorem A.1.4, pg 265 in (Alon & Spencer 2000)) [4], a well-known tool for analyzing randomized algorithms, to derive a confidence threshold $1-\delta$ at which the observed estimate, $\widetilde{V}^t_{\widetilde{\alpha}}$, is within 2ϵ of the true estimate $\dot{V}^t_{\dot{\alpha}}$ (= $E[\dot{\alpha}]$):

$$Pr(\widetilde{V}^t_{\widetilde{\alpha}} > \dot{V}^t_{\dot{\alpha}} + \epsilon) \leq e^{-2N\epsilon^2/(\widetilde{\alpha}_{max} - \widetilde{\alpha}_{min})^2}$$
$$Pr(\widetilde{V}^t_{\widetilde{\alpha}} < \dot{V}^t_{\dot{\alpha}} - \epsilon) \leq e^{-2N\epsilon^2/(\widetilde{\alpha}_{max} - \widetilde{\alpha}_{min})^2}$$

For a confidence probability of $1-\delta$, the error bound is:

$$\epsilon = \sqrt{\frac{(\widetilde{\alpha}_{max} - \widetilde{\alpha}_{min})^2 ln(2/\delta)}{2N}} \quad (3)$$

where $\widetilde{\alpha}_{max} - \widetilde{\alpha}_{min}$ may be loosely upper bounded as $\frac{R_{max}-R_{min}}{1-\gamma}$. Note that Eq. 3 can also be used to derive the number of particles, N, for some given δ and ϵ. To get the desired bound, we note that with probability $1-\delta$ our error bound is 2ϵ and with probability δ the worst possible sub-optimal behavior may result: $||\widetilde{H}\widetilde{V}^{t-1} - H\widetilde{V}^{t-1}|| \leq (1-\delta)2\epsilon + \delta\frac{R_{max}-R_{min}}{1-\gamma}$. The final error bound now obtains:

$$\begin{aligned}\mathcal{E}^t &\leq (1-\delta)2\epsilon + \delta\tfrac{R_{max}-R_{min}}{1-\gamma} + \gamma\mathcal{E}^{t-1} \quad \text{(geom. series)} \\ &= (1-\delta)\tfrac{2\epsilon(1-\gamma^t)}{1-\gamma} + \delta\tfrac{(R_{max}-R_{min})(1-\gamma^t)}{(1-\gamma)^2}\end{aligned}$$

where ϵ is as defined in Eq. 3.

Theorem 1 (Error Bound). *For a singly-nested t-horizon I-POMDP, the error introduced by our approximation technique is bounded and is given by:*

$$||\widetilde{V}^t - V^t|| \leq (1-\delta)\frac{2\epsilon(1-\gamma^t)}{1-\gamma} + \delta\frac{(R_{max}-R_{min})(1-\gamma^t)}{(1-\gamma)^2}$$

where ϵ is as defined in Eq. 3.

At levels of belief nesting greater than one, j's beliefs are also approximately represented. Hence the error in the value function is not only due to the sampling from i's beliefs, but also due to the possible incorrect prediction of j's actions based on its approximate beliefs. We are currently investigating if it is possible to derive bounds that are useful, that is, tighter than the usual difference between the best and worst possible behavior, for this case.

Computational savings

Since the complexity of solving I-POMDPs is dominated by the complexity of solving the models of other agents we look at the reduction of the number of agent models that must be solved. In an M+1-agent setting with the number of particles bounded by N, each particle in $\widetilde{b}^{t-1}_{k,l}$ of level l contains M models of level $l-1$. Solution of each of these level $l-1$ models requires solution of the lower level models recursively. The upper bound on the number of models that are solved is $O((MN)^{l-1})$. Given that there are M level $l-1$ models in a particle, and N such possibly distinct particles, we need to solve $O((MN)^l)$ models. Each of these (level 0) models is a POMDP with an initial belief, and is solved exactly. Our upper bound on the number of models is polynomial in M. This can be contrasted with $O((M|\Theta_*|^M)^l)$ models that need to be solved in the exact case, which is exponential in M. Amongst the spaces of models of all agents, Θ_* is the largest space. Typically, $N \ll |\Theta_*|^M$, resulting in a substantial reduction in computation.

Experiments

The goal of our experimental analysis is to demonstrate empirically, (a) the reduction in error with increasing sample complexity, and (b) savings in computation time and space when our approximation technique is used. We use the multiagent tiger game introduced previously, and a multiagent version of the machine maintenance (MM) problem (Smallwood & Sondik 1973) as test problems. Because the problems are rather simplistic (tiger: $|S|$=2, $|A_i|$=$|A_j|$=3, $|\Omega_i|$=$|\Omega_j|$=6; MM: $|S|$=3, $|A_i|$=$|A_j|$=4, $|\Omega_i|$=$|\Omega_j|$=2), our results should be considered preliminary.

To demonstrate the reduction in error, we construct performance profiles showing an increase in performance as more computational resources – in this case particles – are allocated to the approximation algorithm. In Figs. 4(a) and (c) we show the performance profile curves when agent i's prior belief is the level 1 belief described previously in example 1, and suitably modified for the MM problem. As expected the average rewards for both, horizon 2 and 3 approach the exact expected reward as the number of particles increases. We show the analogous plots for the level 2 belief in Figs. 4(b) and (d). In each of these cases the average of the rewards accumulated by i over a 2 and 3 horizon policy tree (computed using the algorithm in Fig. 3) while playing against agent j were plotted. To compensate for the randomness in sampling, we generated i's policy tree 10 times, and performed 100 runs each time. Within each run, the location of the tiger and j's prior beliefs were sampled according to i's prior belief.

In Table 1, we compare the worst observed error – difference between the exact expected reward and the observed expected reward – with the worst case theoretical error bound (δ=0.1,γ=0.9) from the previous section, for horizons 2 and 3. The difference between the best and the worst possible behavior for the tiger game for $t=2$ is 209.00, and for $t=3$ is 298.1. For the multiagent MM problem, the differences are 8.84 and 12.61, respectively. The theoretical error bounds appear loose due to the worst-case nature of our analysis but (expectedly) are tighter than the worst bounds, and reduce as the number of particles increases. Table 2 compares the average run times of our sample-based approach (SB) with the exact approach, for computing policy trees of

[4] At horizon t, samples in $\widetilde{b}_{i,1}$ are i.i.d. However, at horizons $< t$, the samples are generated by the interactive PF and exhibit limited statistical independence, but independent research (Schmidt, Spiegel, & Srinivasan 1995) reveals that C-H bounds still apply.

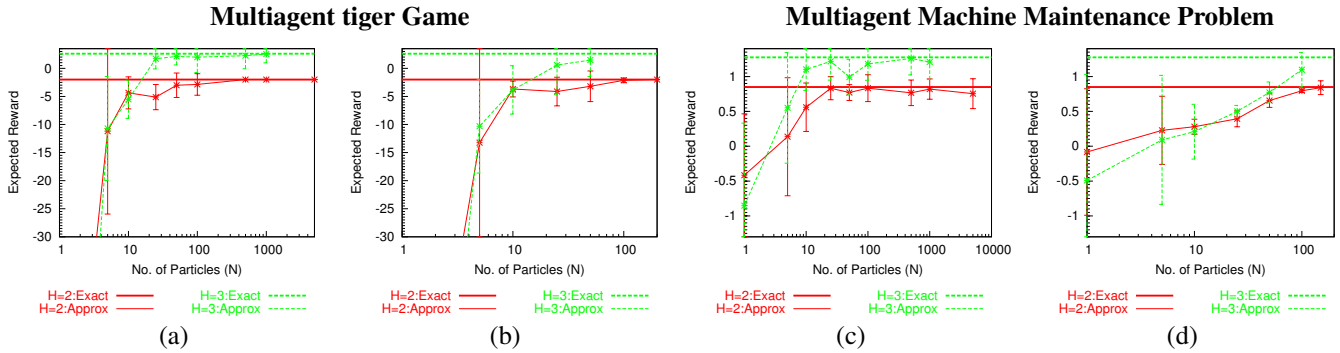

Figure 4: Performance profiles: The multiagent tiger game using the (a) level 1, and (b) level 2 belief as the prior for agent i. The multiagent MM using the (c) level 1, and (d) level 2 belief as i's prior.

different horizons starting from the level 1 belief. The values of the policy trees generated by the two approaches were similar. The run times demonstrate the dominant impact of the curse of dimensionality on the exact method as shown by the higher run times for the MM in comparison to the tiger game. Our sample based implementation is immune to this curse, but is affected by the curse of history, as illustrated by the higher run times for the tiger game (branching factor = 18) compared to the MM problem (branching factor = 8).

Problem	**Error**	$t=2$		$t=3$	
		N=10^2	**N**=10^3	**N**=10^2	**N**=10^3
Multiagent tiger	Obs.	5.61	0	4.39	2.76
	$\mathcal{E}^t$	108.38	48.56	207.78	86.09
Multiagent MM	Obs.	0.28	0.23	0.46	0.40
	$\mathcal{E}^t$	4.58	2.05	8.79	3.64

Table 1: Comparison of the observed errors and the theoretical error bounds.

Problem	**Method**	**Run times**			
		$t=2$	$t=3$	$t=4$	$t=5$
Multiagent tiger	Exact	37.84s $\pm$ 0.6s	11m 22.25s $\pm$ 1.34s	*	*
	SB	1.44s $\pm$ 0.05s	1m 44.29s $\pm$ 0.6s	19m 16.88s $\pm$ 17.5s	146m 54.35s $\pm$ 39.0s
Multiagent MM	Exact	5m 26.57s $\pm$ 0.07s	20m 45.69s $\pm$ 0.29s	*	*
	SB	5.75s $\pm$ 0.01s	34.52.06s $\pm$ 0.01s	3m 24.9s $\pm$ 0.04s	17m 58.39s $\pm$ 0.57s

Table 2: Run times on a P-IV 2.0 GHz, 2.0GB RAM and Linux. * = program ran out of memory.

Conclusion

This paper described a randomized method for obtaining approximate solutions to I-POMDPs based on an extension of particle filtering to multiagent settings. The extension is not straightforward because we are confronted with an interactive belief hierarchy when dealing with multiagent settings. We used the interactive particle filter which descends the levels of interactive belief hierarchies and samples and propagates beliefs at each level. The interactive particle filter is able to deal with the belief space dimensionality, but it does not address the policy space complexity. We provided performance profiles for the multiagent tiger and the machine maintenance problems. They show that our approach saves on computation over the space of models but it does not scale (usefully) to large values of time horizons and needs to be combined with methods that deal with the curse of history.

References

Alon, N., and Spencer, J. 2000. *The Probabilistic Method*. John Wiley and Sons.

Brandenburger, A., and Dekel, E. 1993. Hierarchies of beliefs and common knowledge. *Journal of Economic Theory* 59:189–198.

Doshi, P., and Gmytrasiewicz, P. J. 2005. Approximating state estimation in multiagent settings using particle filters. In *AAMAS*.

Doucet, A.; Freitas, N. D.; and Gordon, N. 2001. *Sequential Monte Carlo Methods in Practice*. Springer Verlag.

Fagin, R.; Halpern, J.; Moses, Y.; and Vardi, M. 1995. *Reasoning about Knowledge*. MIT Press.

Fox, D.; Burgard, W.; Kruppa, H.; and Thrun, S. 2000. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots on Heterogenous Multi-Robot Systems* 8(3).

Gmytrasiewicz, P. J., and Doshi, P. 2005. A framework for sequential planning in multi-agent settings. *JAIR* 23.

Gordon, N.; Salmond, D.; and Smith, A. 1993. Novel approach to non-linear/non-gaussian bayesian state estimation. *IEEE Proceedings-F* 140(2):107–113.

Heifetz, A., and Samet, D. 1998. Topology-free typology of beliefs. *Journal of Economic Theory* 82:324–341.

Li, M., and Vitanyi, P. 1997. *An Introduction to Kolmogorov Complexity and its Applications*. Springer.

Mertens, J., and Zamir, S. 1985. Formulation of bayesian analysis for games with incomplete information. *International Journal of Game Theory* 14:1–29.

Ortiz, L., and Kaelbling, L. 2000. Sampling methods for action selection in influence diagrams. In *AAAI*, 378–385.

Poupart, P.; Ortiz, L.; and Boutilier, C. 2001. Value-directed sampling methods for monitoring pomdps. In *UAI*, 453–461.

Schmidt, J.; Spiegel, A.; and Srinivasan, A. 1995. Chernoff-hoeffding boundings for applications with limited independence. *SIAM Journal on Discrete Mathematics* 8:223–250.

Smallwood, R., and Sondik, E. 1973. The optimal control of pomdps over a finite horizon. *Op. Res.* 21:1071–1088.

Thrun, S. 2000. Monte carlo pomdps. In *NIPS 12*, 1064–1070.