Generalized Point Based Value Iteration for Interactive POMDPs

Prashant Doshi
Dept. of Computer Science and AI Center
University of Georgia
Athens, GA 30602

Dennis Perez
AI Center
University of Georgia
Athens, GA 30602

Abstract

We develop a point based method for solving finitely nested *interactive* POMDPs approximately. Analogously to point based value iteration (PBVI) in POMDPs, we maintain a set of belief points and form value functions composed of those value vectors that are optimal at these points. However, as we focus on multiagent settings, the beliefs are nested and computation of the value vectors relies on predicted actions of others. Consequently, we develop a novel *interactive* generalization of PBVI applicable to multiagent settings.

Introduction

Interactive partially observable Markov decision processes (I-POMDPs; Gmytrasiewicz & Doshi 2005) are a framework for sequential decision-making in uncertain, multiagent environments. I-POMDPs facilitate planning and problem-solving at an agent's own individual level, and in the absence of any centralized controllers (cf. Nair *et al.* 2003) and assumptions about beliefs of other agents (cf. Nair *et al.* 2003; Szer and Charpillet 2006). Analogous to POMDPs (Kaelbling, Littman, & Cassandra 1998), I-POMDPs are disproportionately affected by growing dimensionalities of the state space (curse of dimensionality), and by a large policy space that grows exponentially with the number of actions and observations (curse of history).

Because I-POMDPs include models of other agents in the state space as well, the curses of dimensionality and history are particularly potent. First, if models of others encompass their beliefs, the state space is nested representing the beliefs over others' beliefs and their beliefs over others. Second, as the agents act and observe, their beliefs evolve over time. Thus, solutions of I-POMDPs are affected by not only the curse of history afflicting the modeling agent but also that exhibited by the modeled agents.

Previously, approximations of finitely nested I-POMDPs focused on mitigating the curse of dimensionality. One approach is to form a sampled representation of the agent's prior nested belief. Samples are then propagated over time and recursively in nesting, using a process called the interactive particle filter (Doshi & Gmytrasiewicz 2005), that generalizes the particle filter to multiagent settings. As the approach maintains a fixed set of N samples of the interactive state space, it saves on computations. However, the approach does not address the curse of history and is suited to solving I-POMDPs when an agent's prior belief is known.

In this paper, we focus on general solutions of finitely nested I-POMDPs; we do not assume a particular initial belief of the agent. For POMDPs, point based solutions (e.g. PBVI: Pineau *et al.* 2006, Perseus: Spaan & Vlassis 2005) provide effective approximations that reduce the impact of the curse of history and scale well to relatively large single-agent problems. This motivates their use in approximating multiagent decision making. Szer and Charpillet (2006) use point based iterative dominance to approximate DEC-POMDPs. Seuken and Zilberstein (2007) adopt a memory bounded and point based approach to compute approximately optimal joint policy trees for DEC-POMDPs. While its application in DEC-POMDPs is somewhat straightforward due to the assumption of common knowledge of initial beliefs of agents and a focus on team setting, we confront multiple challenges: (i) As point based techniques utilize a set of initial beliefs, we need computational representations of the nested beliefs to select them. (ii) Because there could be infinitely many computable models of other agents, the state space is prohibitively large. Finally, (iii) actions of agents in a multiagent setting depend on others' actions as well. Thus, solutions of others' models are required which suggests a recursive implementation of the technique.

We provide ways to address these challenges. We show that computational representations of multiply nested beliefs are non-trivial and restrictive assumptions are necessary to facilitate their representations. In this context, we limit the interactive state space by including a finite set of initial models of other agents and those models that are reachable from the initial set over time. Here, we make the assumption that the initial beliefs of the agent are *absolutely continuous* with the true models of all agents, as defined in (Doshi & Gmytrasiewicz 2006). Finally, we present *generalized* point based value iteration for finitely nested I-POMDPs that recurses down the nesting, approximately solving the models at each level.

Finitely Nested I-POMDPs

Interactive POMDPs generalize POMDPs to multiagent settings by including other agents' models as part of the state

space (Gmytrasiewicz & Doshi 2005). A finitely nested I-POMDP of agent i with a strategy level l interacting with one other agent, j, is defined as the tuple:

$$\text{I-POMDP}_{i,l} = \langle IS_{i,l}, A, T_i, \Omega_i, O_i, R_i \rangle$$

where: • $IS_{i,l}$ denotes a set of interactive states defined as, $IS_{i,l} = S \times M_{j,l-1}$, where $M_{j,l-1} = \{\Theta_{j,l-1} \cup SM_j\}$, for $l \geq 1$, and $IS_{i,0} = S$, where S is the set of states of the physical environment. $\Theta_{j,l-1}$ is the set of computable *intentional models* of agent j: $\theta_{j,l-1} = \langle b_{j,l-1}, \hat{\theta}_j \rangle$ where the *frame*, $\hat{\theta}_j = \langle A, \Omega_j, T_j, O_j, R_j, OC_j \rangle$. Here, j is Bayes rational and OC_j is j's optimality criterion. SM_j is the set of subintentional models of j. In this paper, we focus on intentional models only. We give a recursive bottom-up construction of the interactive state space:

$$\begin{array}{ll} IS_{i,0} = S, & \Theta_{j,0} = \{\langle b_{j,0}, \hat{\theta}_j \rangle \mid b_{j,0} \in \Delta(IS_{j,0})\} \\ IS_{i,1} = S \times \Theta_{j,0}, & \Theta_{j,1} = \{\langle b_{j,1}, \hat{\theta}_j \rangle \mid b_{j,1} \in \Delta(IS_{j,1})\} \\ \vdots & \vdots \\ IS_{i,l} = S \times \Theta_{j,l-1}, & \Theta_{j,l} = \{\langle b_{j,l}, \hat{\theta}_j \rangle \mid b_{j,l} \in \Delta(IS_{j,l})\} \end{array}$$

• $A = A_i \times A_j$ is the set of joint actions of all agents. The remaining parameters have their usual meaning.

For a description of the belief update, additional details on I-POMDPs and how they compare with other multiagent frameworks, see (Gmytrasiewicz & Doshi 2005).

Exact Solution

Analogous to POMDPs, the value function for I-POMDPs, U^t, maps $\Theta_{i,l} \rightarrow \mathbb{R}$, and is PWLC. Because $\Theta_{i,l}$ is a continuous space (countably infinite if we limit to computable models), we cannot iterate over all the models of i to compute their values. Instead, analogous to POMDPs, we may decompose the value function into its components:

$$U^t(\langle b_{i,l}, \widehat{\theta_i} \rangle) = \sum_{is \in IS_{i,l}} \alpha^t(is) \times b_{i,l}(is) \tag{1}$$

where,

$$\alpha^t(is) = \max_{a_i \in A_i} \Big\{ ER_i(is, a_i) + \gamma \sum_{o_i} \sum_{is' \in IS_{i,l}} \Big\{ \sum_{a_j} Pr(a_j|\theta_{j,l-1}) \Big[T_i(s, a_i, a_j, s') O_i(s', a_i, a_j, o_i) \textstyle\sum_{o_j} O_j(s', a_i, a_j, o_j) \\ \delta_D(SE_{\widehat{\theta}_j}(b_{j,l-1}, a_j, o_j) - b'_{j,l-1}) \Big] \Big\} \alpha^{t+1}(is') \Big\}$$

δ_D is the Dirac-delta function and $SE(\cdot)$ denotes the belief update. Proof for Eq. 1 is given in the Appendix of (Gmytrasiewicz & Doshi 2005). For small problems and finite sets of models, we may compute α^t (called *alpha vector*) exactly.

Let $\mathcal{V}^{t+1}$ be the time $t+1$ alpha vectors, $\forall a_i \in A_i$, $o_i \in \Omega_i$:

$$\Gamma^{a_i,*} \leftarrow \alpha^{a_i,*}(is) = \sum_{a_j \in A_j} R_i(s, a_i, a_j) Pr(a_j|\theta_{j,l-1}) \tag{2}$$

$$\Gamma^{a_i,o_i} \overset{\cup}{\leftarrow} \alpha^{a_i,o_i}(is) = \gamma \sum_{is'} \sum_{a_j} Pr(a_j|\theta_{j,l-1}) T_i(s, a_i, a_j, s') \\ O_i(s', a_i, a_j, o_i) \sum_{o_j} O_j(s', a_i, a_j, o_j) \delta_D(SE_{\widehat{\theta}_j}(b_{j,l-1}, a_j, o_j) \\ - b'_{j,l-1})\, \alpha^{t+1}(is') \qquad \forall \alpha^{t+1} \in \mathcal{V}^{t+1} \tag{3}$$

Thus we generate $\mathcal{O}(|A_i||\Omega_i|)$ sets of $|\mathcal{V}^{t+1}|$ vectors each. Each vector is of length $|IS_{i,l}|$. Next, we obtain Γ^{a_i} by a cross-sum of previously computed sets of alpha vectors:

$$\Gamma^{a_i} \leftarrow \Gamma^{a_i,*} \oplus \Gamma^{a_i,o_i^1} \oplus \Gamma^{a_i,o_i^2} \oplus \cdots \oplus \Gamma^{a_i,o_i^{|\Omega_i|}} \tag{4}$$

We may generate $\mathcal{O}(|A_i||\mathcal{V}^{t+1}|^{|\Omega_i|})$ many distinct intermediate alpha vectors, and we utilize a linear program (LP) to pick those that are optimal for at least one belief point.

$$\mathcal{V}^t = \underset{\alpha^t}{\text{prune}} \ (\textstyle\bigcup_{a_i} \Gamma^{a_i})$$

Note that Eqs. 2 and 3 require $Pr(a_j|\theta_{j,l-1})$, which involves solving the level $l-1$ intentional models of j.

Computational Representation of Nested Beliefs

While we presented a mathematical definition of nested belief structures, their computational representations are also needed to facilitate implementations utilizing nested beliefs. However, as we show next, developing these representations is not a trivial task.

Complexity of Representation

To promote understanding, we assume that j's frame is known and agent i is uncertain about the physical states and j's beliefs only. We explore the representations bottom-up.

Agent i's level 0 belief, $b_{i,0} \in \Delta(S)$, is a vector of probabilities over each physical state: $b_{i,0} \overset{def}{=} \langle\ p_{i,0}(s_1), p_{i,0}(s_2), \ldots, p_{i,0}(s_{|S|})\ \rangle$. Since belief is a probability distribution, elements of the vector must sum to 1. We refer to this constraint as the simplex constraint. As we may write, $p_{i,0}(s_{|S|}) = 1 - \sum_{q=1}^{|S|-1} p_{i,0}(s_q)$, subsequently, only $|S|-1$ probabilities are needed to specify a level 0 belief.

Agent i's level 1 belief, $b_{i,1} \in \Delta(S \times \Theta_{j,0})$, may be rewritten as, $b_{i,1}(s, \theta_{j,0}) = p_{i,1}(s) p_{i,1}(\theta_{j,0}|s)$. Therefore, i's level 1 belief is a vector: $b_{i,1} \overset{def}{=} \langle (p_{i,1}(s_1), p_{i,1}(\Theta_{j,0}|s_1)),\ (p_{i,1}(s_2), p_{i,1}(\Theta_{j,0}|s_2)),\ \ldots,\ (p_{i,1}(s_{|S|}), p_{i,1}(\Theta_{j,0}|s_{|S|})) \rangle$. Here, the discrete distribution, $\langle\ p_{i,1}(s_1), p_{i,1}(s_2), \ldots, p_{i,1}(s_{|S|})\ \rangle$ satisfies the simplex constraint, and each $p_{i,1}(\Theta_{j,0}|s_q)$ is a single density function over j's level 0 beliefs. We note that $p_{i,1}(\Theta_{j,0}|s_q)$ integrates to 1 over all level 0 models of j.

Agent i's level 2 belief, $b_{i,2} \in \Delta(S \times \Theta_{j,1})$, analogous to level 1 beliefs, is a vector: $b_{i,2} \overset{def}{=} \langle\ (p_{i,2}(s_1), p_{i,2}(\Theta_{j,1}|s_1)), (p_{i,2}(s_2), p_{i,2}(\Theta_{j,1}|s_2)), \ldots,\ (p_{i,2}(s_{|S|}), p_{i,2}(\Theta_{j,1}|s_{|S|}))\ \rangle$. In comparison to level 0 and level 1 beliefs, representing doubly-nested beliefs and beliefs with deeper nestings is difficult. This is because these are distributions over density functions whose representations need not be finite. For example, let j's singly-nested belief densities be represented using a mixture of Gaussians. Then, i's doubly nested belief over j's densities is a vector of normalized mathematical functions of variables where the variables are the parameters of lower-level densities. Because the lower-level densities are Gaussian mixtures which could have *any* number of components and therefore an arbitrary number of means and

covariances, functions that represent doubly nested beliefs may have an unbounded number of variables. Thus computational representations of i's level 2 beliefs are not trivial.

Restrictions on the complexity of nested beliefs are needed to allow for computability. One sufficient way is to focus our attention on a limited set of other's models.

Absolute Continuity Condition

Let $\tilde{\Theta}_{j,0}$ be a *finite* set of j's computable level 0 models. Then, define $\tilde{IS}_{i,1} = S \times \tilde{\Theta}_{j,0}$ and agent i's belief, $\tilde{b}_{i,1} \in \Delta(\tilde{IS}_{i,1})$. Here, each $p_{i,1}(\tilde{\Theta}_{j,0}|s_q)$ in the definition of $\tilde{b}_{i,1}$ is also a *discrete* distribution that satisfies the simplex constraint and facilitates higher order beliefs. We generalize to level l in a straightforward manner: Let $\tilde{\Theta}_{j,l-1}$ be a finite set of j's computable level $l-1$ models. Then, define $\tilde{IS}_{i,l} = S \times \tilde{\Theta}_{j,l-1}$ and agent i's belief, $\tilde{b}_{i,l} \in \Delta(\tilde{IS}_{i,l})$. Here, analogous to a level 1 belief, $b_{i,l} \stackrel{def}{=} \langle (p_{i,l}(s_1), p_{i,l}(\tilde{\Theta}_{j,l-1}|s_1)), (p_{i,l}(s_2), p_{i,l}(\tilde{\Theta}_{j,l-1}|s_2)),\ldots,(p_{i,l}(s_{|S|}), p_{i,l}(\tilde{\Theta}_{j,l-1}|s_{|S|})) \rangle$, where each distribution is discrete.

Agent i's belief over the physical states and j's models, together with its perfect information about its own model induces a predictive probability distribution over joint future observations in the interaction (Doshi & Gmytrasiewicz 2006). As we limit the support of i's belief to a finite set of models, an actual sequence of observations may not proceed along a path that is assigned a non-zero predictive probability by i's belief. In this case, i's observations may contradict its belief and the Bayesian belief update may not be possible.

Thus, we desire that i's belief, $\tilde{b}_{i,l}$, assign a non-zero probability to each potentially realizable observation path in the interaction – this condition has been called the truth compatibility condition (Doshi & Gmytrasiewicz 2006). We formalize it using the notion of *absolute continuity* of two probability measures: A probability measure p_1 is *absolutely continuous* with p_2, denoted as $p_1 \ll p_2$, if $p_2(E) = 0$ implies $p_1(E) = 0$, for any measurable set E. Let ρ_0 be the true distribution over possible observation paths induced by perfectly knowing the true models of i and j. Let $\rho_{b_{i,l}}$ be the distribution over observation paths induced by i's initial belief, $\tilde{b}_{i,l}$. Then,

Definition 1 (Absolute Continuity Condition (ACC)) *ACC holds for an agent, say i, if $\rho_0 \ll \rho_{b_{i,l}}$.*

Of course, a sufficient but not necessary way to satisfy the ACC is for agent i to include each possible model of j in the support of its belief. However, as our observation in the previous section precludes this, we select a finite set of j's candidate models with the partial (domain) knowledge that the true model of j is one of them.

Interactive PBVI

Because I-POMDPs include possible models of other agents that are solved, their solution complexity additionally suffers from the curse of history that afflicts the modeled agents. This curse manifests itself in the $|A_j||\mathcal{V}_j^{t+1}|^{|\Omega_j|}$ alpha vectors that are generated at time t (Eq. 3) and the subsequent application of LPs to select the optimal vectors to solve the models of agent j. Point based approaches utilize a finite set of belief points to decide which alpha vectors to retain, and thereby do not utilize the LPs.

Bounded IS and Initial Beliefs

As we mentioned previously, we limit the space of j's candidate initial models to a finite set, $\tilde{\Theta}_{j,l-1}$. However, because the models of j may grow as it acts and observes, agent i must track these models over time in order to act rationally. Let $\mathsf{Reach}(\tilde{\Theta}_{j,l-1}, H)$ be the set of level $l-1$ models that j could have in the course of H steps. Note that $\mathsf{Reach}(\tilde{\Theta}_{j,l-1}, 0) = \tilde{\Theta}_{j,l-1}$. In computing $\mathsf{Reach}(\cdot)$, we repeatedly update j's beliefs in the models contained in $\tilde{\Theta}_{j,l-1}$. We define a bounded interactive state space below:

$$\tilde{IS}_{i,l} = S \times \mathsf{Reach}(\tilde{\Theta}_{j,l-1}, H),\ \tilde{\Theta}_{j,l} = \{\langle \tilde{b}_{j,l}, \hat{\theta}_j \rangle \mid \tilde{b}_{j,l} \in \Delta(\tilde{IS}_{j,l})\}$$

For each level of the nesting, we select an initial set of beliefs for the corresponding agent randomly. We may proceed recursively, selecting N initial beliefs randomly as we recurse down the nesting until we reach level 0, where each belief is simply a distribution over the physical states.

Point Based Back Projections

Given the bounded interactive state space defined previously, Eqs. 2 and 3 may be rewritten. $\forall a_i \in A_i$ and $o_i \in \Omega_i$:

$$\tilde{\Gamma}^{a_i,*} \leftarrow \alpha^{a_i,*}(\tilde{is}) = \sum_{a_j \in A_j} R_i(s, a_i, a_j) Pr(a_j|\tilde{\theta}_{j,l-1}) \quad (5)$$

$$\tilde{\Gamma}^{a_i,o_i} \stackrel{\cup}{\leftarrow} \alpha^{a_i,o_i}(\tilde{is}) = \gamma \sum_{\tilde{is}'} \sum_{a_j} Pr(a_j|\tilde{\theta}_{j,l-1}) T_i(s, a_i, a_j, s') O_i(s', a_i, a_j, o_i) \sum_{o_j} O_j(s', a_i, a_j, o_j) \delta_D(SE_{\hat{\theta}_j}(\tilde{b}_{j,l-1}, a_j, o_j) - \tilde{b}'_{j,l-1})\ \alpha^{t+1}(\tilde{is}') \qquad \forall \alpha^{t+1} \in \mathcal{V}^{t+1} \quad (6)$$

where $\tilde{is}, \tilde{is}' \in \tilde{IS}_{i,l}$ and $\tilde{is} = \langle s, \tilde{\theta}_{j,l-1} \rangle$.

Let $\tilde{B}_{i,l}$ be a finite set of level l belief points at some time t. As we seek alpha vectors of agent i that are optimal at the beliefs in $\tilde{B}_{i,l}$, we may simplify the cross-sum computations shown in Eq. 4. In particular, we need not consider all the vectors in a set, say $\tilde{\Gamma}^{a_i,o_i^1}$, but only those that are optimal at some belief point, $\tilde{b}_{i,l} \in \tilde{B}_{i,l}$. Formally,

$$\tilde{\Gamma}^{a_i} \leftarrow \tilde{\Gamma}^{a_i,*} \underset{o_i \in \Omega_i}{\oplus} \underset{\tilde{\Gamma}^{a_i,o_i}}{argmax} (\alpha^{a_i,o_i} \cdot \tilde{b}_{i,l}) \quad \forall \tilde{b}_{i,l} \in \tilde{B}_{i,l} \quad (7)$$

We again utilize $\tilde{B}_{i,l}$ to finally select the alpha vectors that form the set $\mathcal{V}^t$:

$$\mathcal{V}^t \leftarrow \underset{\alpha^t \in \bigcup_{a_i} \tilde{\Gamma}^{a_i}}{argmax} (\alpha^t \cdot \tilde{b}_{i,l}) \qquad \forall \tilde{b}_{i,l} \in \tilde{B}_{i,l}$$

Note that in Eq. 7, we generate at most $\mathcal{O}(|A_i||\mathcal{V}^{t+1}|^{|\Omega_i|})$ alpha vectors, typically less, and do not require a LP to select the optimal ones. The set $\mathcal{V}^t$ contains unique alpha vectors that are optimal for at least one belief point in $\tilde{B}_{i,l}$. Hence, $\mathcal{V}^t$ contains at most $|\tilde{B}_{i,l}|$ alpha vectors, typically less in practice. As the number of alpha vectors depends on the set of belief points, we may limit the latter to a constant size.

What remains is how we compute the term $Pr(a_j|\tilde{\theta}_{j,l-1})$ in Eqs. 5 and 6. We may solve agent j's I-POMDP of level $l-1$ or POMDP of level 0 in an analogous manner using a finite set of initial belief points of j. Consequently, *we recurse through the levels of nesting*, utilizing the pre-computed finite set of belief points at each level to generate the alpha vectors that are optimal at those points.

Top Down Expansion of Belief Points

For POMDPs, Spaan and Vlassis (2005) utilize a fixed set of beliefs obtained by randomly exploring the environment. During back projections, they progressively filter out belief points considering only those for which the previously back projected alpha vectors are not optimal. James *et al.* (2007) incrementally introduce belief points that have the potential of providing the largest gain, where gain is the difference between the current value of the policy at that point as obtained from previously selected alpha vectors and a minimal upper bound. As the authors conclude, finding the minimal upper bound is computationally expensive and for large belief spaces (as is the case in multiagent settings) may offset the runtime savings provided by point based approaches.

We utilize two alternatives to expand the sets of belief points over time that are used to select the alpha vectors:

• **Stochastic trajectory simulation (**Stoch**)** For each belief in a set, $\tilde{B}_{i,l}$, we sample a physical state and the other agent's model. We then uniformly sample i's action, a_i, and in combination with the sampled physical state and j's action obtained from solving j's model, we sample the next physical state using the transition function. Given the updated physical state and joint actions, we sample an observation of i, o_i, from the observation function. Agent i's belief is then updated given its action, a_i, and observation, o_i.

• **Error minimization (**Min**)** Approximation error in point based approaches, in part, depends on the density of the set of belief points. We prefer to generate a new belief point, $\tilde{b}_{i,l}^{t+1}$, such that the optimal alpha vector at that point is furthest in value from the alpha vector at an existing belief that is closest to the generated belief. This is because in the absence of such a point, a large error would be incurred at that point. As the optimal alpha vector at $\tilde{b}_{i,l}^{t+1}$ is not known, we may utilize the maximum (or minimum) value, $\frac{R_{max}}{1-\gamma}$ for each is, in its place. Consequently, we select a belief point, $b_{i,l}^t$ from the set $\tilde{B}_{i,l}$, which when updated will result in $\tilde{b}_{i,l}^{t+1}$.

Similar approaches used in (Pineau, Gordon, & Thrun 2006) for expanding beliefs in point based methods in context of single agent POMDPs, demonstrated good results. For each expansion technique, beliefs at all strategy levels are recursively generated in an analogous manner.

Algorithm

We show the main procedure for performing the interactive PBVI (I-PBVI) in Fig. 1. We generate the N initial belief points at each level, $\langle \tilde{B}_{k,l}^N, \tilde{B}_{-k,l-1}^N, \ldots, \tilde{B}_{k,0}^N \rangle$, randomly as mentioned before though other ways, for example, utilizing prior knowledge about probable beliefs, may be used. If the I-POMDP is not strategically nested, we back project the time $t+1$ vectors using a standard backup technique for single agent POMDPs, as given in, say (Pineau, Gordon, & Thrun 2006). Otherwise, a more sophisticated approach is needed (lines 2-7). The alpha vectors at time H (horizon 1) are initialized to their lower bounds, $\frac{R_{min}}{1-\gamma}$ (line 1). This is sufficient to ensure that the repeated back projections will gradually improve the value function. Though in Fig. 1, we recursively expand the set of beliefs, $\langle \tilde{B}_{k,l}^N, \tilde{B}_{-k,l-1}^N, \ldots, \tilde{B}_{k,0}^N \rangle$, after each backup, we may reduce our computational overhead by performing the expansions more sparsely. Here, we utilize the techniques in the previous section for carrying out the expansions (lines 7-8). We show the

I-PBVI (Initial beliefs: $\langle \tilde{B}_{k,l}^N, \tilde{B}_{-k,l-1}^N, \ldots, \tilde{B}_{k,0}^N \rangle$, Horizons:$H > 0$, Strategy level: $l \geq 0$)

1: $\tilde{\Gamma}^{H-1} \leftarrow$ INITIAL-ALPHAVECTORS ()
2: **for** $t \leftarrow H-2$ to 0 **do**
3: **if** $l = 0$ **then**
4: $\tilde{\Gamma}^t \leftarrow$ PBVI BACKUP($\tilde{B}_{k,0}^N$, $\tilde{\Gamma}^{t+1}$, $H-t$)
5: **else**
6: $\tilde{\Gamma}^t \leftarrow$ I-PBVI BACKUP($\tilde{B}_{k,l}^N, \ldots, \tilde{B}_{k,0}^N$, $\tilde{\Gamma}^{t+1}$, $H-t$, l)
7: Expand the previous set of beliefs at all levels recursively
8: Add the new beliefs to the existing sets
9: **return** $\tilde{\Gamma}^0$

Figure 1: Interactive PBVI for generating the alpha vectors at horizon, H. When $l = 0$, the vector projection is analogous to that for POMDPs. Here, k ($-k$) assumes agent i (j) or j (i).

procedure for back projecting the vectors for the case where the I-POMDP is nested to a level $l > 0$, in Fig. 2. In a nutshell, we utilize the steps outlined in Eqs. 5– 7 to identify the projected alpha vectors that are optimal at the belief points in the set, $\tilde{B}_{k,l}$ (lines 2-12). However, in doing so we need to predict the other agent's actions as well which is obtained by solving its models. Thus, in performing the backup, we descend through the nesting solving models at each level by recursively performing I-PBVI.

I-PBVI BACKUP ($\langle \tilde{B}_{k,l}, \ldots, \tilde{B}_{k,0} \rangle$, $\tilde{\Gamma}_k^{t+1}$, h, l)

1: $\tilde{\Gamma}_{-k}^t \leftarrow$ I-PBVI ($\langle \tilde{B}_{-k,l-1}, \ldots, \tilde{B}_{k,0} \rangle$, h, $l-1$)
2: **for all** $a_k \in A_k$ **do**
3: Compute $\alpha_k^{a_i,*}$ (Eq. 5) where $Pr(a_{-k}|\tilde{\theta}_{-k,l-1}) \leftarrow$ GETACTION ($\tilde{\theta}_{-k,l-1}$, $\tilde{\Gamma}_{-k}^t$) and add $\alpha_k^{a_i,*}$ to $\tilde{\Gamma}^{a_i,*}$
4: **for all** $o_k \in \Omega_k$ **do**
5: Compute $\alpha_k^{a_i,o_i}$ (Eq. 6), where $Pr(a_{-k}|\tilde{\theta}_{-k,l-1}) \leftarrow$ GETACTION ($\tilde{\theta}_{-k,l-1}$, $\tilde{\Gamma}_{-k}^t$), add $\alpha_k^{a_i,o_i}$ to $\tilde{\Gamma}^{a_i,o_i}$
6: **for all** $\tilde{b}_{k,l} \in \tilde{B}_{k,l}$ **do**
7: Compute $\alpha_k^{a_i}$ (Eq. 7) and add $\alpha_k^{a_i}$ to $\tilde{\Gamma}^{a_i}$
8: $\tilde{\Gamma}^t \leftarrow \bigcup_{a_i} \tilde{\Gamma}^{a_i}$
9: **for all** $\tilde{b}_{k,l} \in \tilde{B}_{k,l}$ **do**
10: Select α_k^* in $\tilde{\Gamma}^t$ that maximizes $\alpha_k \cdot \tilde{b}_{k,l}$
11: Add α_k^* to $\tilde{\Gamma}_*^t$ if not already present
12: **return** $\tilde{\Gamma}_*^t$

Figure 2: Procedure for backing up alpha vectors when strategy level $l > 0$. We recursively call I-PBVI for solving other's models.

Computational Savings and Error Bounds

If the strategy level is 0, I-POMDP$_i$ collapses into a POMDP and we generate in the worst case $\mathcal{O}(|A_i||\mathcal{V}^{t+1}|^{|\Omega_i|})$ many alpha vectors at time t to solve the POMDP exactly. Let $M_{j,l-1} = \mathsf{Reach}(\widetilde{\Theta}_{j,l-1}, H)$. At level 1, because we include $|M_{j,0}|$ models of j in the state space, we need obtain at most $|M_{j,0}|$ alpha vectors assuming j's frame is known. These are used in solving the I-POMDP of i, which in the worst case generates $\mathcal{O}(|A_i||\mathcal{V}^{t+1}|^{|\Omega_i|})$ vectors of size $|\tilde{IS}_{i,l}|$. Thus, a total of $\mathcal{O}(|A_i||\mathcal{V}^{t+1}|^{|\Omega_i|} + |M_{j,0}|)$ alpha vectors are obtained at level 1. Generalizing to level l and assuming, for the sake of simplicity, that the same number of models of the other agent are included at any level, $|M|$, we need $\mathcal{O}(|A_i||\mathcal{V}^{t+1}|^{|\Omega_i|} + |M|l)$ alpha vectors to solve the I-POMDP$_{i,l}$ exactly. For I-PBVI, if at most N belief points are used at any level, approximate solution of a level 0 I-POMDP generates $\mathcal{O}(N)$ alpha vectors. For level 1, because solutions of $|M|$ models are obtained approximately using N belief points, we need obtain only $\mathcal{O}(N)$ vectors for j and another $\mathcal{O}(N)$ vectors to solve the I-POMDP of i at level 1 approximately. Generalizing to level l, we generate at most $\mathcal{O}(N(l+1))$ many alpha vectors. For the case where $N << |M|$, significant computational savings are obtained. Of course, for more than two agents, the number of alpha vectors are exponential in the number of agents.

The loss in optimality or error due to approximately solving the I-POMDP using I-PBVI is due to two reasons: (i) The alpha vectors that are optimal at selected belief points may be suboptimal at other points; and (ii) Models of the other agent are solved approximately as well. We begin a characterization of the error by noting that Eq. 1 may be rewritten as:

$$\alpha^t(is) = \sum_{a_j \in A_j} Pr(a_j|\theta_{j,l-1}) \times \max_{a_i \in A_i} \Big\{ R_i(s,a_i,a_j) + \gamma \sum_{o_i} \sum_{is' \in IS_{i,l}} \Big\{ \Big[T_i(s,a_i,a_j,s') O_i(s',a_i,a_j,o_i) \sum_{o_j} O_j(s',a_i, a_j,o_j) \delta_D(SE_{\widehat{\theta}_j}(b_{j,l-1},a_j,o_j) - b'_{j,l-1}) \Big] \Big\} \alpha^{t+1}(is') \Big\}$$
$$= Pr(a_j|\theta_{j,l-1}) \cdot \alpha^t_{a_j} \quad (8)$$

Let $\tilde{b}'_{i,l}$ be the belief point where the maximum error occurs, and α'' be the exact alpha vector that is optimal at this belief point. Let α be the approximate vector that is instead utilized at $\tilde{b}'_{i,l}$ for computing the policy. Note that in using α the solution suffers from both the sources of error mentioned previously, while α'' induces no error. Let α' be optimal at $\tilde{b}'_{i,l}$ while still exhibiting error due to the approximate solution of j's models. We may define the worst case error as:

$$\mathcal{E} = \alpha'' \cdot \tilde{b}'_{i,l} - \alpha \cdot \tilde{b}'_{i,l} = \alpha'' \cdot \tilde{b}'_{i,l} - \alpha \cdot \tilde{b}'_{i,l} + (\alpha' \cdot \tilde{b}'_{i,l} - \alpha' \cdot \tilde{b}'_{i,l})$$
$$= (\alpha'' \cdot \tilde{b}'_{i,l} - \alpha' \cdot \tilde{b}'_{i,l}) + (\alpha' \cdot \tilde{b}'_{i,l} - \alpha \cdot \tilde{b}'_{i,l}) \quad (9)$$

For the second term, $\alpha' \cdot \tilde{b}'_{i,l} - \alpha \cdot \tilde{b}'_{i,l}$, the difference is only due to the limited set of belief points, as both α' and α utilize the same approximate solution of j's models. Define $d_{\tilde{B}}$ as the largest of the distances between the pruned belief, $\tilde{b}'_{i,l}$, and the closest belief, $\tilde{b}_{i,l}$, among the selected points: $d_{\tilde{B}} = max_{\tilde{b}'_{i,l} \in \Delta_{i,l}} min_{\tilde{b}_{i,l} \in \tilde{B}_{i,l}} |\tilde{b}'_{i,l} - \tilde{b}_{i,l}|$. $d_{\tilde{B}}$ reflects the density of the selected belief points within the belief simplex. Derivation of this difference proceeds analogously to that in POMDPs (Pineau *et al.* 2006), and we obtain:

$$\alpha' \cdot \tilde{b}'_{i,l} - \alpha \cdot \tilde{b}'_{i,l} \leq \frac{R_i^{max} - R_i^{min}}{1-\gamma} d_{\tilde{B}}$$

Next, the term, $\alpha'' \cdot \tilde{b}'_{i,l} - \alpha' \cdot \tilde{b}'_{i,l}$, in Eq. 9 represents the error due to the approximate solution of the other agent's models obtained by using I-PBVI recursively:

$$\alpha'' \cdot \tilde{b}'_{i,l} - \alpha' \cdot \tilde{b}'_{i,l} = \tilde{b}'_{i,l} \cdot (\alpha'' - \alpha')$$
$$= \tilde{b}'_{i,l} \cdot (\alpha''_{a_j} \cdot Pr(a_j|\cdot) - \alpha''_{a_j} \cdot Pr'(a_j|\cdot)) \quad \text{(Using Eq. 8)}$$
$$= \tilde{b}'_{i,l} \cdot (\alpha''_{a_j} \cdot (Pr(a_j|\cdot) - Pr'(a_j|\cdot)))$$

The inner dot products are over j's actions. $Pr'(a_j|\cdot)$ represents the suboptimal probability due to the approximation. Consider the case where $Pr'(a_j|\cdot)$ prescribes an action, a'_j, different from that by $Pr(a_j|\cdot)$. Then the worst error is loosely bounded by, $\alpha''_{a_j} - \alpha''_{a'_j} \leq \frac{R_i^{max} - R_i^{min}}{1-\gamma}$. Therefore,

$$\alpha'' \cdot \tilde{b}'_{i,l} - \alpha' \cdot \tilde{b}'_{i,l} \leq \tilde{b}'_{i,l} \cdot \frac{R_i^{max} - R_i^{min}}{1-\gamma} = \frac{R_i^{max} - R_i^{min}}{1-\gamma}$$

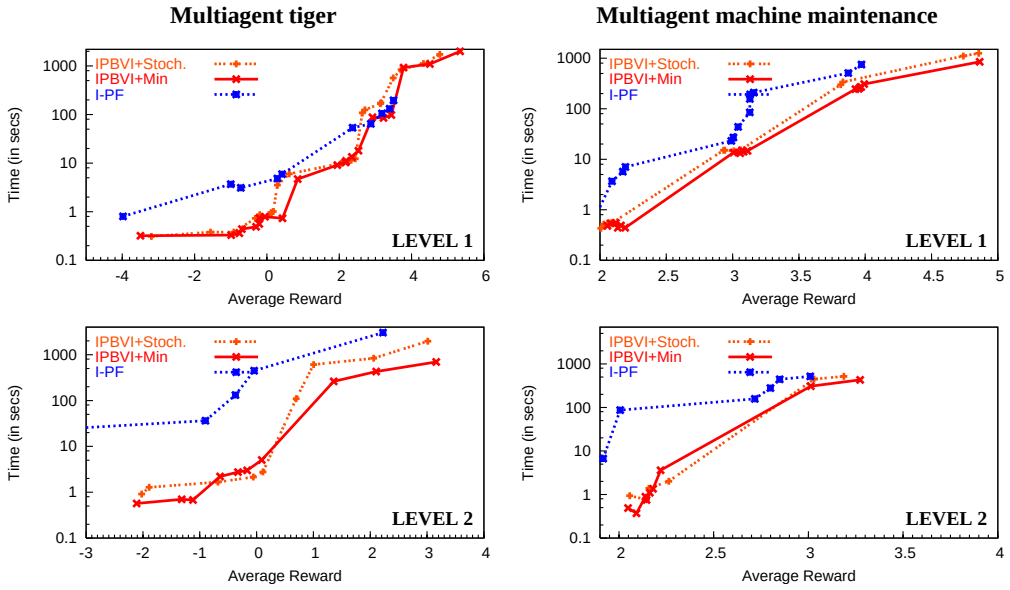
Thus, although the error due to pruning the belief points is bounded and depends on the density of the selected belief points, we are unable to usefully bound the error due to approximately solving other's models.

Performance Evaluation

We implemented the algorithms in Figs. 1 and 2 and evaluated its performance on the multiagent *tiger* problem (Gmytrasiewicz & Doshi 2005; our formulation is different from that in Seuken & Zilberstein 2007 having more observations) and a multiagent version of the machine maintenance problem (MM; Smallwood & Sondik 1973). Although the two problems have a small physical state space, the interactive state space, $IS_{i,l}$, tends to get large.

For both problems, we provide the least time taken in reaching a particular performance in terms of the rewards gathered by agent i. The time consumed is a function of the number of belief points used during I-PBVI, the horizons of the policy and the number of j's models. We gradually increased the number of belief points, horizons and models and simulated the performance of the resulting policies over 10 trials with 50 runs each. In each trial, we selected a different initial belief of agent i, and sampled the starting state and belief of j from this belief. We compare results across both the top down expansion strategies mentioned previously.

We show the level 1 and 2 plots for the two problems in Fig. 3, respectively. Lower values on y-axis indicate better performance. Notice that for level 1, Min (denoted as IP-BVI+Min) performs better than the approach of using Stoch (IPBVI+Stoch) to expand the belief points, in both domains. Specifically, IPBVI+Min takes less time in providing an identical performance as when the IPBVI+Stoch is used. However, the distinction is less evident at level 2 where the greater computations incurred in using the minimization approach assume significance. These observations are analogous to the mixed performance of the different expansion



Level 1	H	Time (s)	
		I-PBVI	I-PF
Tiger	7	322	496
	20	2290	*
	40	3043	*
MM	8	498	752
	20	2303	*
	40	3360	*

Figure 3: Level 1 (j's models are POMDPs) and level 2 (j's models are level 1 I-POMDPs) plots of time consumed in achieving a desired performance. Note that the y-axis is in log scale. The I-PBVI significantly improves on the I-PF, a previous approximation technique for I-POMDPs. All experiments are run on a Linux platform with dual processor Xeon 3.4GHz with 4GB memory.

techniques in POMDPs (Pineau, Gordon, & Thrun 2006). One way to assess the impact of deeper modeling is to measure the average rewards obtained by i across levels for the same number of belief points. Our experiments do not reveal a significant overall improvement when agent i's beliefs are doubly nested, although level 2 solutions are computationally more intensive as evident from Fig. 3. However, there is evidence in the tiger problem that modeling at level 1 results in better performance in comparison to naively treating the other agent as noise (Gmytrasiewicz & Doshi 2005).

We compare the performance of I-PBVI with the interactive particle filter (I-PF) based approximation (Doshi & Gmytrasiewicz 2005), which is the previous best approximation method for I-POMDPs. We generate policy trees for as many initial beliefs of i as the number of belief points used in I-PBVI. Although the I-PF is able to mitigate the curse of dimensionality, it must generate the full reachability tree to compute the policy and therefore it continues to suffer from the curse of history that affects I-POMDPs. Better performance of I-PBVI in comparison to I-PF demonstrates that the generalized PBVI is able to mitigate the impact of the curse of history that affects solutions of both the agents' decision processes. Furthermore, we were unable to run the I-PF beyond a few time horizons due to excessive memory consumption. As we show in the table in Fig. 3, we were able to solve for 40 horizons (using less belief expansions), significantly improving on the previous approach for I-POMDPs that could solve only up to 8 horizons.

Although I-PBVI mitigates the impact of having to maintain the history of interaction, we need to maintain the set of reachable models of the other agent (in Reach()) that quickly grows over time. Due to ACC, we cannot trivially limit this set. Hence, we are unable to solve beyond tens of horizons of the problems. Further improvement seems possible by carefully limiting the set of reachable models.

References

Doshi, P., and Gmytrasiewicz, P. J. 2005. A particle filtering based approach to approximating interactive pomdps. In *AAAI*, 969–974.

Doshi, P., and Gmytrasiewicz, P. J. 2006. On the difficulty of achieving equilibrium in interactive pomdps. In *AAAI*, 1131–1136.

Gmytrasiewicz, P., and Doshi, P. 2005. A framework for sequential planning in multiagent settings. *JAIR* 24:49–79.

James, M.; Samples, M.; and Dolgov, D. 2007. Improving anytime PBVI using principled point selections. In *IJCAI*, 865–870.

Kaelbling, L.; Littman, M.; and Cassandra, A. 1998. Planning and acting in partially observable stochastic domains. *AI Journal* 2.

Nair, R.; Tambe, M.; Yokoo, M.; Pynadath, D.; and Marsella, S. 2003. Taming decentralized pomdps : Towards efficient policy computation for multiagent settings. In *IJCAI*, 705–711.

Pineau, J.; Gordon, G.; and Thrun, S. 2006. Anytime point-based approximations for pomdps. *JAIR* 27:335–380.

Seuken, S., and Zilberstein, S. 2007. Improved memory bounded dynamic programming for decentralized pomdps. In *UAI*.

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

Spaan, M., and Vlassis, N. 2005. Perseus: Randomized point-based value iteration for pomdps. *JAIR* 24:195–220.

Szer, D., and Charpillet, F. 2006. Point based dynamic programming for dec-pomdps. In *AAAI*, 1233–1238.