Speeding Up Solutions of Interactive Dynamic Influence Diagrams Using Action Equivalence [*]

Yifeng Zeng
Dept. of Computer Science
Aalborg University
DK-9220 Aalborg, Denmark
yfzeng@cs.aau.dk

Prashant Doshi
Dept. of Computer Science
University of Georgia
Athens, GA 30602
pdoshi@cs.uga.edu

Abstract

Interactive dynamic influence diagrams (I-DIDs) are graphical models for sequential decision making in partially observable settings shared by other agents. Algorithms for solving I-DIDs face the challenge of an exponentially growing space of candidate models ascribed to other agents, over time. Previous approach for exactly solving I-DIDs groups together models having similar solutions into behaviorally equivalent classes and updates these classes. We present a new method that, in addition to aggregating behaviorally equivalent models, further groups models that prescribe identical actions at a single time step. We show how to update these augmented classes and prove that our method is exact for some cases. The new approach enables us to bound the aggregated model space by the cardinality of other agents' actions. We evaluate its performance and provide empirical results in support.

1 Introduction

Interactive dynamic influence diagrams (I-DIDs) [Doshi *et al.*, 2009] are graphical models for sequential decision making in uncertain settings shared by other agents. I-DIDs may be viewed as graphical counterparts of I-POMDPs [Gmytrasiewicz and Doshi, 2005]. They generalize DIDs [Tatman and Shachter, 1990], which may be viewed as graphical counterparts of POMDPs, to multiagent settings analogously to how I-POMDPs generalize POMDPs. Importantly, I-DIDs have the advantage of decomposing the state space into variables and relationships between them by exploiting the domain structure which allows computational benefits in comparison to I-POMDPs [Doshi *et al.*, 2009].

I-DIDs contribute to a growing line of work that includes multiagent influence diagrams (MAIDs) [Koller and Milch, 2001], and more recently, networks of influence diagrams (NIDs) [Gal and Pfeffer, 2003]. MAIDs objectively analyze the game, efficiently computing the Nash equilibrium profile by exploiting the independence structure. NIDs extend MAIDs to include agents' uncertainty over the game being played and over models of other agents. Both MAIDs and NIDs provide an analysis of the game from an external viewpoint, and adopt Nash equilibrium as the solution concept. Specifically, MAIDs do not allow us to define a distribution over non-equilibrium behaviors of other agents. Furthermore, their applicability is limited to static single play games. Interactions are more complex when they are extended over time, where predictions about others' future actions must be made using models that change as the agents act and observe. I-DIDs seek to address this gap by offering an intuitive way to extend sequential decision making to multiagent settings.

As we may expect, I-DIDs acutely suffer from both the curses of dimensionality and history [Pineau *et al.*, 2006]. This is because the state space in I-DIDs includes the models of other agents in addition to the traditional physical states. These models encompass the agents' beliefs, actions and sensory capabilities, and preferences, and may themselves be formalized as I-DIDs. The nesting is terminated at the 0^{th} level where the other agents are modeled using DIDs. As the agents act, observe, and update beliefs, I-DIDs must track the evolution of the models over time. Consequently, I-DIDs not only suffer from the curse of history that afflicts the modeling agent, but more so from that exhibited by the modeled agents. The exponential growth in the number of models over time also further contributes to the dimensionality of the state space. This is complicated by the nested nature of the space.

While we may solve I-DIDs exactly if the number of models of others is finite, we are unable to scale to large problems or even longer horizons for small problems. One way to mitigate the intractability is to group together behaviorally equivalent models [Rathnas *et al.*, 2006; Pynadath and Marsella, 2007] thereby reducing the cardinality of the model node. Doshi *et al.* [2009] proposed an approximation technique for solving I-DIDs based on clustering models that are likely to be behaviorally equivalent. Solutions of multiagent problems up to *six* horizons were shown using these approaches.

Although exact approaches face significant hurdles in scaling to realistic problems, nevertheless exact solutions play an important role: they serve as *optimal benchmarks* for solutions provided by approximation techniques. In this paper, we improve on the previous approach of exactly solving I-DIDs. We reduce the model space by grouping behaviorally equivalent models. A behavioral equivalence class contains models

[*] Prashant Doshi acknowledges partial support from grant #FA9550-08-1-0429 from US AFOSR. The authors thank Prof. Reid Simmons for discussions leading to changes in this paper.

that exhibit identical solutions for all time steps. We further compact the space of models in the model node by observing that behaviorally distinct models may prescribe identical actions at a single time step. We may then group together these models into a single equivalence class. In comparison to behavioral equivalence, the definition of our equivalence class is different: it includes those models whose prescribed action for the *particular* time step is the same, and we call it *action equivalence*. Since there are typically additional models than the behaviorally equivalent ones that prescribe identical actions at a time step, an action equivalence class often includes many more models. Consequently, the model space is partitioned into lesser number of classes than previously [Rathnas *et al.*, 2006] and is bounded by the number of actions of the other agent.

We begin by solving the individual models in the initial model node to obtain the policy trees. These trees are merged bottom-up to obtain a policy graph. As a result, behaviorally equivalent models, which have identical policy trees, are merged. We further group models at each time step whose prescribed actions at that step are identical. We show how we may compute the probability with which an equivalence class is updated to another class in the next time step. These probabilities constitute the new conditional probability distribution of the model node. We discuss computational savings and theoretically show that the approach preserves optimality for some cases. For other situations, we still can obtain a good approximation using our approach. We demonstrate the performance of our approach on two problem domains and show significant time savings in comparison to previous approaches.

2 Background: Interactive DID

We briefly describe interactive influence diagrams (I-IDs) for two-agent interactions followed by their extensions to dynamic settings, I-DIDs, and refer the reader to [Doshi *et al.*, 2009] for more details.

2.1 Syntax

In addition to the usual chance, decision, and utility nodes, I-IDs include a new type of node called the *model node* (hexagonal node, $M_{j,l-1}$, in Fig. $1(a)$). The probability distribution over the chance node, S, and the model node together represents agent i's belief over its *interactive state space*. In addition to the model node, I-IDs differ from IDs by having a chance node, A_j, that represents the distribution over other agent's actions, and a dashed link, called a *policy link*.

The model node contains as its values the alternative computational models ascribed by i to the other agent. We denote the set of these models by $\mathcal{M}_{j,l-1}$. A model in the model node may itself be an I-ID or ID, and the recursion terminates when a model is an ID or a simple probability distribution over the actions. Formally, we denote a model of j as, $m_{j,l-1} = \langle b_{j,l-1}, \hat{\theta}_j \rangle$, where $b_{j,l-1}$ is the level $l-1$ belief, and $\hat{\theta}_j$ is the agent's *frame* encompassing the action, observation, and utility nodes. We observe that the model node and the dashed policy link that connects it to the chance node, A_j, could be represented as shown in Fig. $1(b)$. The decision node of each level $l-1$ I-ID is transformed into a chance node. Specifically, if OPT is the set of optimal actions obtained by solving the I-ID (or ID), then $Pr(a_j \in A_j^1) = \frac{1}{|OPT|}$ if $a_j \in OPT$, 0 otherwise. The conditional probability table (CPT) of the chance node, A_j, is a *multiplexer*, that assumes the distribution of each of the action nodes (A_j^1, A_j^2) depending on the value of $Mod[M_j]$. The distribution over $Mod[M_j]$, is i's belief over j's models given the state.

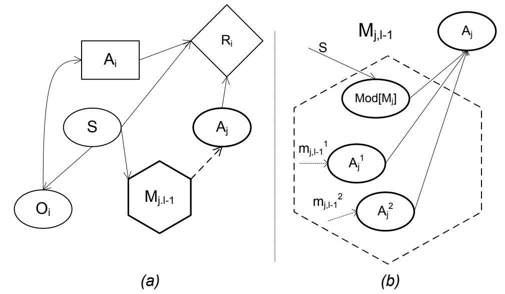


Figure 1: (a) A generic level $l > 0$ I-ID for agent i situated with one other agent j. The hexagon is the model node ($M_{j,l-1}$) and the dashed arrow is the policy link. (b) Representing the model node and policy link using chance nodes and dependencies between them. The decision nodes of the lower-level I-IDs or IDs ($m_{j,l-1}^1$, $m_{j,l-1}^2$) are mapped to the corresponding chance nodes (A_j^1, A_j^2), which is indicated by the dotted arrows.

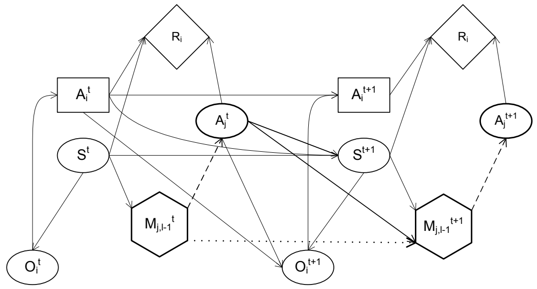


Figure 2: A generic two time-slice level l I-DID for agent i. Notice the dotted model update link that denotes the update of the models of j and of the distribution over the models, over time.

I-DIDs extend I-IDs to allow sequential decision making over several time steps. We depict a general two time-slice I-DID in Fig. 2. In addition to the model nodes and the dashed policy link, what differentiates an I-DID from a DID is the *model update link* shown as a dotted arrow in Fig. 2. We briefly explain the semantics of the model update next.

The update of the model node over time involves two steps: First, given the models at time t, we identify the updated set of models that reside in the model node at time $t+1$. Because the agents act and receive observations, their models are updated to reflect their changed beliefs. Since the set of optimal actions for a model could include all the actions, and the agent may receive any one of $|\Omega_j|$ possible observations, the updated set at time step $t+1$ will have up to $|\mathcal{M}_{j,l-1}^t||A_j||\Omega_j|$ models. Here, $|\mathcal{M}_{j,l-1}^t|$ is the num-

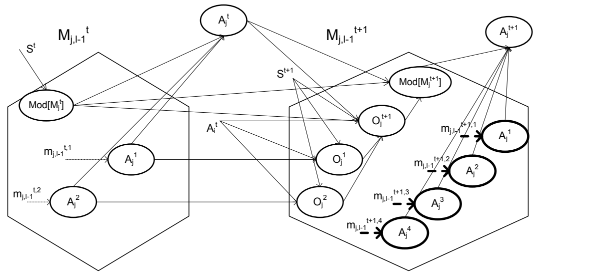


Figure 3: The semantics of the model update link. Notice the growth in the number of models in the model node at $t+1$ in bold.

ber of models at time step t, $|A_j|$ and $|\Omega_j|$ are the largest spaces of actions and observations respectively, among all the models. The CPT of $Mod[M^{t+1}_{j,l-1}]$ encodes the function, $\tau(b^t_{j,l-1}, a^t_j, o^{t+1}_j, b^{t+1}_{j,l-1})$ which is 1 if the belief $b^t_{j,l-1}$ in the model $m^t_{j,l-1}$ using the action a^t_j and observation o^{t+1}_j updates to $b^{t+1}_{j,l-1}$ in a model $m^{t+1}_{j,l-1}$; otherwise it is 0. Second, we compute the new distribution over the updated models, given the original distribution and the probability of the agent performing the action and receiving the observation that led to the updated model. The dotted model update link in the I-DID may be implemented using standard dependency links and chance nodes, as in Fig. 3, transforming it into a flat DID.

2.2 Solution

Solution of an I-DID (and I-ID) proceeds in a bottom-up manner, and is implemented recursively. We start by solving the level 0 models, which may be traditional IDs. Their solutions provide probability distributions which are entered in the corresponding action nodes found in the model node of the level 1 I-DID. The solution method uses the standard look-ahead technique, projecting the agent's action and observation sequences forward from the current belief state, and finding the possible beliefs that i could have in the next time step. Because agent i has a belief over j's models as well, the look-ahead includes finding out the possible models that j could have in the future. Each of j's level 0 models represented using a standard DID in the first time step must be solved to obtain its optimal set of actions. These actions are combined with the set of possible observations that j could make in that model, resulting in an updated set of candidate models (that include the updated beliefs) that could describe the behavior of j. Beliefs over these updated set of candidate models are calculated using the standard inference methods through the dependency links between the model nodes.

3 Aggregating Models Using Action Equivalence

As mentioned before, we seek to group at each step those models that prescribe identical actions at that time step. We describe our approach below.

3.1 Policy Graph and Behavioral Equivalence

Solutions of individual I-DIDs and DIDs that represent the models of other agents may be merged to obtain a *policy graph*. First, note that solutions of the models could be represented as policy trees. Each node in the policy tree represents an action to be performed by the agent and edges represent the agent's observations. The policy trees may be merged bottom-up to obtain a policy graph, as we demonstrate in Fig. 4 using the well-known tiger problem [Kaelbling *et al.*, 1998]. Analogous to a policy graph in POMDPs, each node in the graph is associated with a set of models for which the corresponding action is optimal.

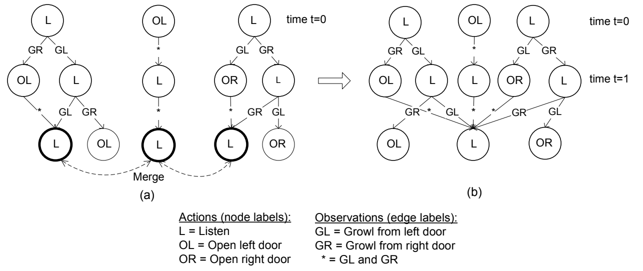


Figure 4: (a) Example policy trees obtained by solving three models of j for the tiger problem. We may merge the three L nodes to obtain the policy graph in (b). Because no policy trees of two steps are identical, no more merging is possible.

Implicit in the procedure of merging policy trees is the fact that if pairs of policy trees are identical – resulting from behaviorally equivalent models – they are merged into a single representative tree. The following proposition gives the complexity of merging the policy trees to obtain the policy graph.

Proposition 1 (Complexity of tree merge)**.** *Worst-case complexity of the procedure for merging policy trees to form a policy graph is* $\mathcal{O}((|\Omega_j|^{T-1})^{|\mathcal{M}^0_{j,l-1}|})$ *where* T *is the horizon.*

Proof. Complexity of the policy tree merge procedure is proportional to the number of comparisons that are made between parts of policy trees to ascertain their similarity. As the procedure follows a bottom-up approach, the maximum number of comparisons are made between leaf nodes and the worst case occurs when none of the leaf nodes of the different policy trees can be merged. Note that this precludes the merger of upper parts of the policy trees as well. Each policy tree may contain up to $|\Omega_j|^{T-1}$ leaf nodes, where T is the horizon. The case when none of the leaf nodes merge must occur when the models are behaviorally distinct. Hence, at most $\mathcal{O}((|\Omega_j|^{T-1})^{|\mathcal{M}^0_{j,l-1}|})$ comparisons are performed. ∎

Intuitively, merging policy trees is analogous to grouping behaviorally equivalent models, whose entire policy trees are similar. The utility of grouping behaviorally equivalent models toward reducing the model space is well known [Rathnas *et al.*, 2006; Pynadath and Marsella, 2007].

3.2 Action Equivalence

Definition

Notice from Fig. $4(b)$ that the policy graph contains multiple nodes labeled with the same action at time steps $t = 0$ and

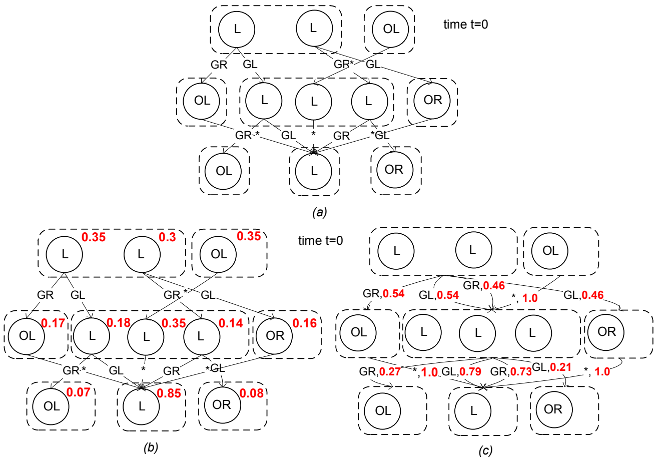


Figure 5: (a) We may group models that prescribe identical actions into classes as indicated by the dashed boxes. (b) Annotations are example probabilities for the models associated with the nodes. (c) Probabilities on the edges represent the probability of transition between classes given action and observation.

$t = 1$. The associated models while prescribing actions that are identical at a particular time step, differ in the entire behavior. We call these models *actionally equivalent*. Action equivalence further partitions the model space, $\mathcal{M}^t_{j,l-1}$, into classes, as we show in Fig. 5(a). If more than one action is optimal for a model, we may break ties randomly.

From Fig. 5(a), the partition of the model set, $\mathcal{M}^t_{j,l-1}$, induced by action equivalence at time step 0 is $\{\mathcal{M}^{t=0,1}_{j,l-1}, \mathcal{M}^{t=0,2}_{j,l-1}\}$, where $\mathcal{M}^{t=0,1}_{j,l-1}$ is the class of models in the model space whose prescribed action at $t = 0$ is L, and $\mathcal{M}^{t=0,2}_{j,l-1}$ is the class of models whose prescribed action at $t = 0$ is OL. Note that these classes include the behaviorally equivalent models as well. Thus, all models in a class prescribe an identical action at that time step. Furthermore at $t = 1$, the partition consists of 3 action equivalence classes and, at $t = 2$, the partition also consists of 3 classes.

Revised CPT of *Mod* Node

As we mentioned previously, the node $Mod[M^{t+1}_{j,l-1}]$ in the model node $M^{t+1}_{j,l-1}$, has as its values the different models ascribed to agent j at time $t+1$. The CPT of $Mod[M^{t+1}_{j,l-1}]$ implements the function $\tau(b^t_{j,l-1}, a^t_j, o^{t+1}_j, b^{t+1}_{j,l-1})$, which is 1 if $b^t_{j,l-1}$ in the model $m^t_{j,l-1}$ updates to $b^{t+1}_{j,l-1}$ in model $m^{t+1}_{j,l-1}$ using the action-observation combination, otherwise it is 0. However, now that the models have been aggregated into action equivalence classes, this CPT is no longer valid.

As we see from Fig. 5(a), updating an equivalence class given an action-observation combination may lead into multiple classes at the next time step. For example, updating $\mathcal{M}^{t=0,1}_{j,l-1}$ (left class at $t = 0$) with action L and observation GR leads into the left class with action OL and the middle class with action L at $t = 1$. Similarly, updating $\mathcal{M}^{t=0,1}_{j,l-1}$ with action L and observation GL leads into the middle class with action L and the singleton class with action OR at $t = 1$. Consequently, the update function, τ, and therefore the CPT of $Mod[M^{t+1}_{j,l-1}]$, is no longer deterministic (an indicator function) but is probabilistic.

The probability, $Pr(\mathcal{M}^{t+1,p}_{j,l-1} \mid \mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$, is zero trivially if a^t_j is not the optimal action for class $\mathcal{M}^{t,q}_{j,l-1}$. Otherwise, we show how we may derive the probability of class, $\mathcal{M}^{t+1,p}_{j,l-1}$, given $\mathcal{M}^{t,q}_{j,l-1}$ and an action-observation combination.

As we expect, the aggregation at time t may not impact the probability distribution of j's behaviors at other time steps. The revised probability $Pr(\mathcal{M}^{t+1,p}_{j,l-1} \mid \mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$ must satisfy two constraints given below (for all values of $m^{t-1}_{j,l-1}$,a^{t-1}_j, and o^t_j):

$$\begin{aligned}
Const.1: \quad & Pr(\mathcal{M}^{t+1,p}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j) \\
& = \textstyle\sum_{m^{t+1}_{j,l-1} \in \mathcal{M}^{t+1,p}_{j,l-1}} Pr(m^{t+1}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j) \\
Const.2: \quad & \textstyle\sum_{\mathcal{M}^{t,q}_{j,l-1}} Pr(m^{t+1}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j) \\
& Pr(\mathcal{M}^{t,q}_{j,l-1}|m^{t-1}_{j,l-1}, a^{t-1}_j, o^t_j) \\
& = \textstyle\sum_{m^t_{j,l-1}} Pr(m^{t+1}_{j,l-1}|m^t_{j,l-1}, a^t_j, o^{t+1}_j) \\
& Pr(m^t_{j,l-1}|m^{t-1}_{j,l-1}, a^{t-1}_j, o^t_j)
\end{aligned}$$

The first constraint ensures the correctness of j's behavior at $t+1$ while the second preserves the joint probability distribution of j's behaviors at adjacent time steps, $t-1$ and $t+1$.

By solving the first equation above, we get:

$$Pr(\mathcal{M}^{t+1,p}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j) = \frac{\sum_{\mathcal{M}^{t+1,p}_{j,l-1}, \mathcal{M}^{t,q}_{j,l-1}} Pr(m^{t+1}_{j,l-1}|m^t_{j,l-1}, a^t_j, o^{t+1}_j) Pr(m^t_{j,l-1}|m^{t-1}_{j,l-1}, a^{t-1}_j, o^t_j)}{\sum_{\mathcal{M}^{t,q}_{j,l-1}} Pr(m^t_{j,l-1}|m^{t-1}_{j,l-1}, a^{t-1}_j, o^t_j)} \tag{1}$$

$Pr(m^{t+1}_{j,l-1}|m^t_{j,l-1}, a^t_j, o^{t+1}_j)$ is equivalent to the τ function. As for the second constraint, we may simplify $Pr(m^t_{j,l-1}|m^{t-1}_{j,l-1}, a^{t-1}_j, o^t_j)$ to $Pr(m^t_{j,l-1})$, and further get:

$$Pr(\mathcal{M}^{t+1,p}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j) = \frac{\sum_{\mathcal{M}^{t+1,p}_{j,l-1}, \mathcal{M}^{t,q}_{j,l-1}} Pr(m^t_{j,l-1}) Pr(m^{t+1}_{j,l-1}|m^t_{j,l-1}, a^t_j, o^{t+1}_j)}{\sum_{\mathcal{M}^{t,q}_{j,l-1}} Pr(m^t_{j,l-1})} =$$

$$\frac{\sum_{m^{t+1}_{j,l-1} \in \mathcal{M}^{t+1,p}_{j,l-1}, m^t_{j,l-1} \in \mathcal{M}^{t,q}_{j,l-1}} b_i(m^t_{j,l-1}) \tau(b^t_{j,l-1}, a^t_j, o^{t+1}_j, b^{t+1}_{j,l-1})}{\sum_{m^t_{j,l-1} \in \mathcal{M}^{t,q}_{j,l-1}} b_i(m^t_{j,l-1})} \tag{2}$$

where $b_i(m^t_{j,l-1})$ is i's belief over the model of j at t , $b^t_{j,l-1}$ is the belief in the model $m^t_{j,l-1}$, and $b^{t+1}_{j,l-1}$ is in model $m^{t+1}_{j,l-1}$. In other words, if we could find a valid $Pr(\mathcal{M}^{t+1,p}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$ that meets for all $m^{t-1}_{j,l-1}$,a^{t-1}_j, and o^t_j in Eq. 1 by satisfying the second constraint, we could use Eq. 2 to compute the revised CPT of Mod node.

Intuitively, Eq. 2 gives the proportion of the total probability mass assigned to individual models in the class, $\mathcal{M}^{t,q}_{j,l-1}$, that update to models in the class, $\mathcal{M}^{t+1,p}_{j,l-1}$. We note that updating via Eq. 2 becomes approximate if there is no $Pr(\mathcal{M}^{t+1,p}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$ suiting all $m^{t-1}_{j,l-1}$,a^{t-1}_j, and o^t_j values in Eq. 1. In other words, our approach becomes approximate if Eq. 2 is used for updates in all cases.

What remains now is how we compute $b_i(m^t_{j,l-1})$ in Eq. 2. If $t = 0$, this is straightforward and may be obtained as: $b_i(m^0_{j,l-1}) = \sum_s Pr(m^0_{j,l-1}|s)Pr(s)$, where $Pr(m^0_{j,l-1}|s)$ is given in the CPT of the *Mod* node at time 0. For subsequent time steps, this is a challenge since we have aggregated the individual models into action equivalence classes at time t and have obtained the probability of each class. We may overcome this obstacle by computing the distribution over the individual updated models as in the original I-DID as well and caching it. This is done, of course, before we begin computing the probabilities of equivalence classes. A marginal of the previously cached values over all physical states for the particular model results in the required probability.

We illustrate the application of Eq. 2 to the policy graph of Fig. $5(a)$ below:

Example For simplicity, let the left action equivalence class, $\mathcal{M}^{t=0,1}_{j,l-1}$, comprise of two models, $m^1_{j,l-1}$ and $m^2_{j,l-1}$, both of which prescribe action L. Let i's marginal belief over these two models be 0.35 and 0.3, respectively (see Fig. $5(b)$). Updating $\mathcal{M}^{t=0,1}_{j,l-1}$ using the action-observation combination (L, GR) leads into two classes, $\mathcal{M}^{t=1,1}_{j,l-1}$ and $\mathcal{M}^{t=1,2}_{j,l-1}$ with the probabilities 0.54 and 0.46, respectively (see Fig. $5(c)$). This is because the model $m^1_{j,l-1}$ which updates to the model in $\mathcal{M}^{t=1,1}_{j,l-1}$ using (L, GR) has the probability proportion $\frac{0.35}{0.35+0.3} = 0.54$. Model $m^2_{j,l-1}$ which updates to a model in $\mathcal{M}^{t=1,2}_{j,l-1}$ has the probability proportion $\frac{0.3}{0.35+0.3} = 0.46$. Similarly, updating $\mathcal{M}^{t=0,1}_{j,l-1}$ using the action-observation combination of (L, GL) leads into $\mathcal{M}^{t=1,2}_{j,l-1}$ and $\mathcal{M}^{t=1,3}_{j,l-1}$ with the probabilities 0.54 and 0.46 respectively. We shall note that updating $\mathcal{M}^{t=1}_{j,l-1}$ becomes approximate at $t = 1$ since no probability values, $Pr(\mathcal{M}^{1,p}_{j,l-1}|\mathcal{M}^{2,q}_{j,l-1}, a^t_j, o^{t+1}_j)$, could be found to satisfy Eq. 1.

In summary, we implement the proposed method by revising the model update phase in the procedure for solving I-DIDs [Doshi *et al.*, 2009]. We aggregate actionally equivalent models and represent their probabilistic update using the new CPT for the node $Mod[M^{t+1}_{j,l-1}]$.

4 Computational Savings and Optimality

The complexity of exactly solving a level l I-DID is, in part, due to solving the lower-level models of the other agent, and given the solutions, due to the exponentially growing space of models. In particular, at some time step t, there could be at most $|\mathcal{M}^0_{j,l-1}|(|A_j||\Omega_j|)^t$ many models, where $\mathcal{M}^0_{j,l-1}$ is the set of initial models of the other agent. Although $|\mathcal{M}^0_{j,l-1}|$ models are solved, considering action equivalence bounds the model space to at most $|A_j|$ distinct classes. Thus, the cardinality of the interactive state space in the I-DID is bounded by $|S||A_j|$ elements at any time step. This is a significant reduction in the size of the state space. In doing so, we additionally incur the computational cost of merging the policy trees, which is $\mathcal{O}((|\Omega_j|^{T-1})^{|\mathcal{M}^0_{j,l-1}|})$ (from Proposition 1). We point out that our approach is applied recursively to solve I-DIDs at all levels down to 1.

Analogous to [Rathnas *et al.*, 2006], which showed that considerations of behavioral equivalence do not upset the solution of I-DIDs, we show that aggregating actionally equivalent models preserves the optimality. Since the aggregation using Eq. 2 does not affect the probability distribution of j's behaviors we further prove that the predictive distribution over j's actions remains unchanged at any time step.

Proposition 2 (Optimality)**.** *The predictive distribution over j's actions on aggregating the model space due to action equivalence is preserved.*

Proof. We prove by showing that for some action, a^{t+1}_j, $Pr(a^{t+1}_j)$ remains unchanged when $\mathcal{M}^{t+1}_{j,l-1}$ is replaced by a partition. Let $\mathcal{M}^{t+1,p}_{j,l-1}$ be the set of models whose optimal action is a^{t+1}_j with probability 1:

$$Pr(a^{t+1}_j) = \textstyle\sum_{m^{t+1}_{j,l-1} \in \mathcal{M}^{t+1,p}_{j,l-1}} Pr(a^{t+1}_j|m^{t+1}_{j,l-1})Pr(m^{t+1}_{j,l-1})$$
$$= \textstyle\sum_{m^{t+1}_{j,l-1} \in \mathcal{M}^{t+1,p}_{j,l-1}} Pr(m^{t+1}_{j,l-1})$$
$$= \textstyle\sum_q \sum_{m^{t+1}_{j,l-1} \in \mathcal{M}^{t+1,p}_{j,l-1}, m^t_{j,l-1} \in \mathcal{M}^{t,q}_{j,l-1}} Pr(m^{t+1}_{j,l-1}|m^t_{j,l-1}, a^t_j, o^{t+1}_j) \times Pr(m^t_{j,l-1}, a^t_j, o^{t+1}_j)$$

Here, we do not show the sum over all a^t_j and o^{t+1}_j for clarity. Notice that $Pr(m^{t+1}_{j,l-1}|m^t_{j,l-1}, a^t_j, o^{t+1}_j)$ is equivalent to $\tau(\cdot)$.

$$Pr(a^{t+1}_j) = \textstyle\sum_q \frac{\sum_{\mathcal{M}^{t+1,p}_{j,l-1}, \mathcal{M}^{t,q}_{j,l-1}} \tau(b^t_{j,l-1}, a^t_j, o^{t+1}_j, b^{t+1}_{j,l-1})}{\sum_{\mathcal{M}^{t,q}_{j,l-1}} Pr(m^t_{j,l-1}, a^t_j, o^{t+1}_j)}$$
$$\times Pr(m^t_{j,l-1}, a^t_j, o^{t+1}_j)Pr(\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$$

$Pr(m^t_{j,l-1}, a^t_j, o^{t+1}_j)$ simplifies to $Pr(m^t_{j,l-1})$ and analogously for $Pr(\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$ if the second constraint is met (in Section 3.2).

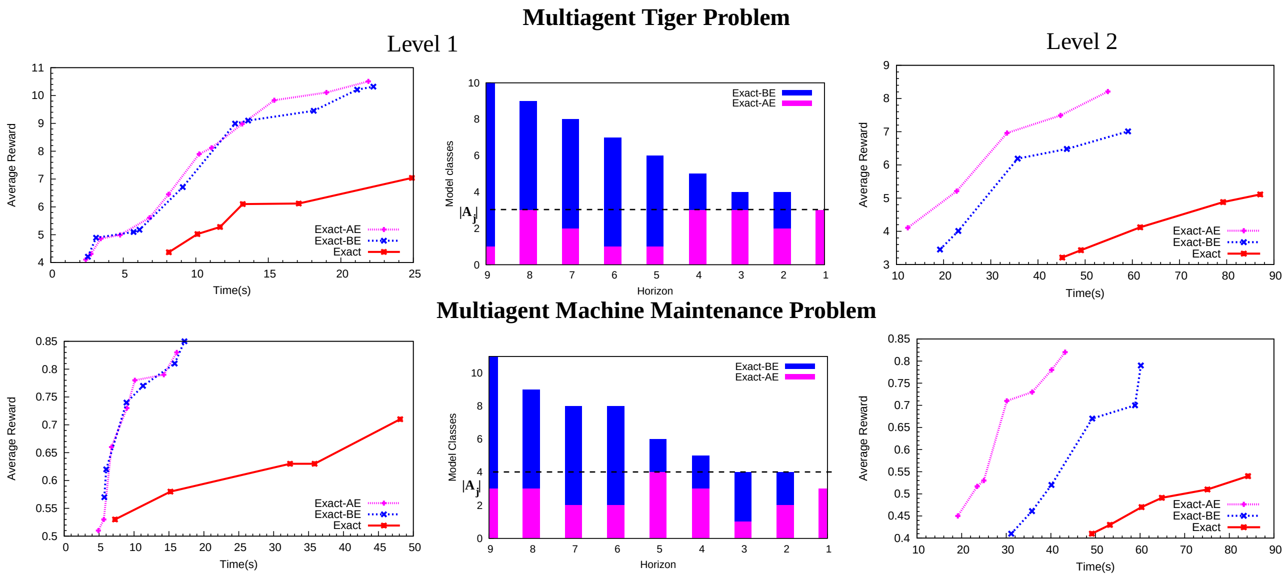
$$Pr(a^{t+1}_j) = \textstyle\sum_q \frac{\sum_{\mathcal{M}^{t+1,p}_{j,l-1}, \mathcal{M}^{t,q}_{j,l-1}} \tau(b^t_{j,l-1}, a^t_j, o^{t+1}_j, b^{t+1}_{j,l-1})}{\sum_{\mathcal{M}^{t,q}_{j,l-1}} b_i(m^t_{j,l-1})}$$
$$\times b_i(m^t_{j,l-1})Pr(\mathcal{M}^{t,q}_{j,l-1})$$

Using Eq. 2 we get:

$$Pr(a^{t+1}_j) = \sum_q Pr(\mathcal{M}^{t+1,p}_{j,l-1}|\mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)Pr(\mathcal{M}^{t,q}_{j,l-1})$$

Recall that $\mathcal{M}^{t+1,p}_{j,l-1}$ is the set whose optimal action is a^{t+1}_j. Thus, the last line (with summations not shown) is used to obtain $Pr(a^{t+1}_j)$ given the action equivalence classes. ∎

We notice that the solution optimality is ensured only if a valid value, $Pr(\mathcal{M}^{t+1,p}_{j,l-1} \mid \mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$ as computed in Eq. 1, could be found for all values of $m^{t-1}_{j,l-1}$,a^{t-1}_j, and o^t_j at time t. If such a value could not found in the updates we have to skip the aggregation in order to preserve the optimality. It is quite often that we can not find a suitable $Pr(\mathcal{M}^{t+1,p}_{j,l-1} \mid \mathcal{M}^{t,q}_{j,l-1}, a^t_j, o^{t+1}_j)$ due to constrains in Eq. 1. We may still perform the aggregation and get a good approximation by using Eq. 2. Consequently, our method becomes approximate since the updates are inexact in some time steps.



(a) (b) (c)

Figure 6: Performance profiles for the multiagent tiger and MM problems generated by executing the solutions obtained using different exact approaches. (a,b) Profiles for level 1 I-DIDs. Notice that AE maintains less classes at larger horizons in comparison to BE and never more than $|A_j|$. (c) Solving level 2 I-DIDs reveals the efficiency facilitated by aggregation using action equivalence.

5 Experimental Results

We evaluate our improvement to the exact approach using action equivalence (AE) in the context of both the multiagent tiger [Nair *et al.*, 2003; Gmytrasiewicz and Doshi, 2005] and a multiagent version of the machine maintenance (MM) problem [Smallwood and Sondik, 1973]. We compare its performance with the previous exact method that uses behavioral equivalence (BE) [Rathnas *et al.*, 2006] and the exact approach with no refinement (Exact) [Doshi *et al.*, 2009]. We show that AE solves the I-DIDs more efficiently than its counterparts by comparing the time taken in achieving a level of expected reward. We experimented with both level 1 and level 2 I-DIDs. We shall note that AE method approaches the exact although the updates are inexact at some steps.

In Figs. 6(a) and (b), we show the reward gathered by executing the policy trees obtained from solving the I-DIDs for level 1. The time consumed is a function of the initial number of models and the horizon of the I-DID, both of which are varied beginning with $|\mathcal{M}^0| = 50$. We observe that the approaches which aggregate the model space perform significantly better than the traditional exact approach. In particular, these approaches obtain the same reward in much less time because they are able to solve the same I-DID more quickly. However, the time difference between AE and BE is not significant, although AE maintains significantly less number of model classes at each horizon as is evident from Figs. 6(b). This is because solution of level 0 models in our problem domains is fast and AE incurs the overhead of computing the update probabilities.

The reduced time needed to obtain a level of reward is more evident for level 2 I-DIDs, as we see in Figs. 6(c). Level 2 I-DIDs for both the problem domains show a significant speed up in solving them when models are aggregated using action equivalence in comparison to behavioral equivalence. Here, the approaches are recursively applied to the lower level I-DIDs that represent models of j, as well.

Level 2	T		Time (s)	
		AE	BE	Exact
Tiger	3	12.35	20.43	49.29
	6	37.56	89.14	*
	11	331.41	*	*
MM	3	17.29	33.15	64.13
	5	54.63	120.31	*
	10	423.12	*	*

Table 1: Aggregation using action equivalence scales significantly better to larger horizons. All experiments are run on a WinXP platform with a dual processor Xeon 2.0GHz and 2GB memory.

Finally, as we show in Table 1, we were able to solve level 2 I-DIDs over more than 10 horizons using AE ($|\mathcal{M}^0|$=25), improving significantly over the previous approach which could comparably solve only up to 6 horizons.

6 Discussion

I-DIDs provide a general formalism for sequential decision making in the presence of other agents. The increased complexity of I-DIDs is predominantly due to the exponential growth in the number of candidate models of others, over time. These models may themselves be represented as I-DIDs or DIDs. We introduced the concept of action equivalence which induces a partition of the model space. The resultant number of classes is often significantly less in comparison to those obtained by considerations of behavioral equivalence. The empirical performance demonstrates the computational savings provided by this approach and its significant improvement over the previous exact technique. We note that action equivalence could be seen as a less stringent criteria for model aggregation compared to behavioral equivalence, leading to more models in a class and less classes.

References

[Doshi *et al.*, 2009] P. Doshi, Y. Zeng, and Q. Chen. Graphical models for interactive pomdps: Representations and solutions. *JAAMAS*, 18(3):376–416, 2009.

[Gal and Pfeffer, 2003] Y. Gal and A. Pfeffer. A language for modeling agent's decision-making processes in games. In *AAMAS*, pages 265–272, 2003.

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

[Kaelbling *et al.*, 1998] L. Kaelbling, M. Littman, and A. Cassandra. Planning and acting in partially observable stochastic domains. *AIJ*, 2:99–134, 1998.

[Koller and Milch, 2001] D. Koller and B. Milch. Multi-agent influence diagrams for representing and solving games. In *IJCAI*, pages 1027–1034, 2001.

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

[Pineau *et al.*, 2006] J. Pineau, G. Gordon, and S. Thrun. Anytime point-based value iteration for large pomdps. *JAIR*, 27:335–380, 2006.

[Pynadath and Marsella, 2007] D. Pynadath and S. Marsella. Minimal mental models. In *AAAI*, pages 1038–1044, 2007.

[Rathnas *et al.*, 2006] B. Rathnas, P. Doshi, and P. J. Gmytrasiewicz. Exact solutions to interactive pomdps using behavioral equivalence. In *AAMAS*, pages 1025–1032, 2006.

[Smallwood and Sondik, 1973] R. Smallwood and E. Sondik. The optimal control of partially observable markov decision processes over a finite horizon. *OR*, 21:1071–1088, 1973.

[Tatman and Shachter, 1990] J. A. Tatman and R. D. Shachter. Dynamic programming and influence diagrams. *IEEE Trans. on Systems, Man, and Cybernetics*, 20(2):365–379, 1990.