Graphical Models for Online Solutions to Interactive POMDPs

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

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

Qiongyu Chen
Dept. of Computer Science
National Univ. of Singapore
117543, Singapore
chenqy@comp.nus.edu.sg

ABSTRACT

We develop a new graphical representation for interactive partially observable Markov decision processes (I-POMDPs) that is significantly more transparent and semantically clear than the previous representation. These graphical models called interactive dynamic influence diagrams (I-DIDs) seek to explicitly model the structure that is often present in real-world problems by decomposing the situation into chance and decision variables, and the dependencies between the variables. I-DIDs generalize DIDs, which may be viewed as graphical representations of POMDPs, to multiagent settings in the same way that I-POMDPs generalize POMDPs. I-DIDs may be used to compute the policy of an agent online as the agent acts and observes in a setting that is populated by other interacting agents. Using several examples, we show how I-DIDs may be applied and demonstrate their usefulness.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Theory

Keywords

Dynamic influence diagrams, decision-making, agent modeling

1. INTRODUCTION

Interactive partially observable Markov decision processes (I-POMDPs) [9] provide a framework for sequential decision-making in partially observable multiagent environments. They generalize POMDPs [13] to multiagent settings by including the other agents' computable models in the state space along with the states of the physical environment. The models encompass all information influencing the agents' behaviors, including their preferences, capabilities, and beliefs, and are thus analogous to *types* in Bayesian games [11]. I-POMDPs adopt a subjective approach to understanding strategic behavior, rooted in a decision-theoretic framework that takes a decision-maker's perspective in the interaction.

In [15], Polich and Gmytrasiewicz introduced **interactive dynamic influence diagrams** (I-DIDs) as the computational representations of I-POMDPs. I-DIDs generalize DIDs [12], which may be viewed as computational counterparts of POMDPs, to multiagents settings in the same way that I-POMDPs generalize POMDPs. I-DIDs contribute to a growing line of work [19] that includes multi-agent influence diagrams (MAIDs) [14], and more recently, networks of influence diagrams (NIDs) [8]. These formalisms seek to explicitly model the structure that is often present in real-world problems by decomposing the situation into chance and decision variables, and the dependencies between the variables. MAIDs provide an alternative to normal and extensive game forms using a graphical formalism to represent games of imperfect information with a decision node for each agent's actions and chance nodes capturing the agent's private information. 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 the other agents. Each model is a MAID and the network of MAIDs is collapsed, bottom up, into a single MAID for computing the equilibrium of the game keeping in mind the different models of each agent. Graphical formalisms such as MAIDs and NIDs open up a promising area of research that aims to represent multiagent interactions more transparently. However, MAIDs provide an analysis of the game from an external viewpoint and the applicability of both is limited to static single play games. Matters are more complex when we consider interactions that 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 address this gap by allowing the representation of other agents' models as the values of a special *model node*. Both, other agents' models and the original agent's beliefs over these models are updated over time using special-purpose implementations.

In this paper, we improve on the previous preliminary representation of the I-DID shown in [15] by using the insight that the static I-ID is a type of NID. Thus, we may utilize NID-specific language constructs such as *multiplexers* to represent the model node, and subsequently the I-ID, more transparently. Furthermore, we clarify the semantics of the special purpose "policy link" introduced in the representation of I-DID by [15], and show that it could be replaced by traditional dependency links. In the previous representation of the I-DID, the update of the agent's belief over the models of others as the agents act and receive observations was denoted using a special link called the "model update link" that connected the model nodes over time. We explicate the semantics of this link by showing how it can be implemented using the traditional dependency links between the chance nodes that constitute the model nodes. The net result is a representation of I-DID that is significantly more

AAMAS'07 May 14–18 2007, Honolulu, Hawaii, USA.

transparent, semantically clear, and capable of being implemented using the standard algorithms for solving DIDs. We show how I-DIDs may be used to model an agent's uncertainty over others' models, that may themselves be I-DIDs. Solution to the I-DID is a policy that prescribes what the agent should do over time, given its beliefs over the physical state and others' models. Analogous to DIDs, I-DIDs may be used to compute the policy of an agent online as the agent acts and observes in a setting that is populated by other interacting agents.

2. BACKGROUND: FINITELY NESTED I-POMDPS

Interactive POMDPs generalize POMDPs to multiagent settings by including other agents' models as part of the state space [9]. Since other agents may also reason about others, the interactive state space is strategically nested; it contains beliefs about other agents' models and their beliefs about others. For simplicity of presentation we consider an agent, i, that is interacting with one other agent, j.

A finitely nested I-POMDP of agent i with a strategy level l 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. Simple examples of subintentional models include a no-information model [10] and a fictitious play model [6], both of which are history independent. We give a recursive bottom-up construction of the interactive state space below.

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

Similar formulations of nested spaces have appeared in [1, 3].
• $A = A_i \times A_j$ is the set of joint actions of all agents in the environment; • $T_i : S \times A \times S \rightarrow [0, 1]$, describes the effect of the joint actions on the physical states of the environment; • Ω_i is the set of observations of agent i; • $O_i : S \times A \times \Omega_i \rightarrow [0, 1]$ gives the likelihood of the observations given the physical state and joint action; • $R_i : IS_i \times A \rightarrow \mathbb{R}$ describes agent i's preferences over its interactive states. Usually only the physical states will matter.

Agent i's policy is the mapping, $\Omega_i^* \rightarrow \Delta(A_i)$, where Ω_i^* is the set of all observation histories of agent i. Since belief over the interactive states forms a sufficient statistic [9], the policy can also be represented as a mapping from the set of all beliefs of agent i to a distribution over its actions, $\Delta(IS_i) \rightarrow \Delta(A_i)$.

2.1 Belief Update

Analogous to POMDPs, an agent within the I-POMDP framework updates its belief as it acts and observes. However, there are two differences that complicate the belief update in multiagent settings when compared to single agent ones. First, since the state of the physical environment depends on the actions of both agents, i's prediction of how the physical state changes has to be made based on its prediction of j's actions. Second, changes in j's models have to be included in i's belief update. Specifically, if j is intentional then an update of j's beliefs due to its action and observation has to be included. In other words, i has to update its belief based on its prediction of what j would observe and how j would update its belief. If j's model is subintentional, then j's probable observations are appended to the observation history contained in the model. Formally, we have:

$$\begin{array}{l}
Pr(is^t|a_i^{t-1}, b_{i,l}^{t-1}) = \beta \sum_{IS^{t-1}:\widehat{m}_j^{t-1}=\widehat{\theta}_j^t} b_{i,l}^{t-1}(is^{t-1}) \\
\times \sum_{a_j^{t-1}} Pr(a_j^{t-1}|\theta_{j,l-1}^{t-1}) O_i(s^t, a_i^{t-1}, a_j^{t-1}, o_i^t) \\
\times T_i(s^{t-1}, a_i^{t-1}, a_j^{t-1}, s^t) \sum_{o_j^t} O_j(s^t, a_i^{t-1}, a_j^{t-1}, o_j^t) \\
\times \tau(SE_{\widehat{\theta}_j^t}(b_{j,l-1}^{t-1}, a_j^{t-1}, o_j^t) - b_{j,l-1}^t)
\end{array} \qquad (1)$$

where β is the normalizing constant, τ is 1 if its argument is 0 otherwise it is 0, $Pr(a_j^{t-1}|\theta_{j,l-1}^{t-1})$ is the probability that a_j^{t-1} is Bayes rational for the agent described by model $\theta_{j,l-1}^{t-1}$, and $SE(\cdot)$ is an abbreviation for the belief update. For a version of the belief update when j's model is subintentional, see [9].

If agent 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 could invoke i's belief update and so on. This recursion in belief nesting bottoms out at the 0^{th} level. At this level, the belief update of the agent reduces to a POMDP belief update. [1] For illustrations of the belief update, additional details on I-POMDPs, and how they compare with other multiagent frameworks, see [9].

2.2 Value Iteration

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

$$\begin{array}{l}
U^n(\langle b_{i,l}, \widehat{\theta}_i \rangle) = \underset{a_i \in A_i}{max} \left\{ \sum\limits_{is \in IS_{i,l}} ER_i(is, a_i) b_{i,l}(is) + \right. \\
\left. \gamma \sum\limits_{o_i \in \Omega_i} Pr(o_i|a_i, b_{i,l}) U^{n-1}(\langle SE_{\widehat{\theta}_i}(b_{i,l}, a_i, o_i), \widehat{\theta}_i \rangle) \right\}
\end{array} \qquad (2)$$

where, $ER_i(is, a_i) = \sum_{a_j} R_i(is, a_i, a_j) Pr(a_j|m_{j,l-1})$ (since $is = (s, m_{j,l-1})$). Eq. 2 is a basis for value iteration in I-POMDPs.

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

$$\begin{array}{l}
OPT(\langle b_{i,l}, \widehat{\theta}_i \rangle) = \underset{a_i \in A_i}{argmax} \left\{ \sum\limits_{is \in IS_{i,l}} ER_i(is, a_i) b_{i,l}(is) \right. \\
\left. + \gamma \sum\limits_{o_i \in \Omega_i} Pr(o_i|a_i, b_{i,l}) U^n(\langle SE_{\widehat{\theta}_i}(b_{i,l}, a_i, o_i), \widehat{\theta}_i \rangle) \right\}
\end{array} \qquad (3)$$

3. INTERACTIVE INFLUENCE DIAGRAMS

A naive extension of influence diagrams (IDs) to settings populated by multiple agents is possible by treating other agents as automatons, represented using chance nodes. However, this approach assumes that the agents' actions are controlled using a probability distribution that does not change over time. Interactive influence diagrams (I-IDs) adopt a more sophisticated approach by generalizing IDs to make them applicable to settings shared with other agents who may act and observe, and update their beliefs.

3.1 Syntax

In addition to the usual chance, decision, and utility nodes, I-IDs include a new type of node called the *model* node. We show a general level l I-ID in Fig. $1(a)$, where the model node ($M_{j,l-1}$) is denoted using a hexagon. We note that the probability distribution over the chance node, S, and the model node together represents

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

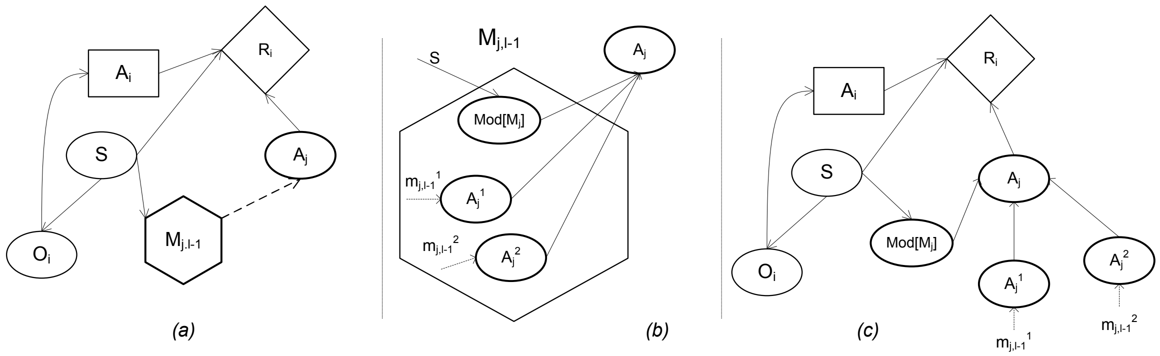

Figure 1: (a) **A generic level l I-ID for agent i situated with one other agent j. The hexagon is the model node ($M_{j,l-1}$) whose structure we show in (b). Members of the model node are I-IDs themselves ($m^1_{j,l-1}, m^2_{j,l-1}$; diagrams not shown here for simplicity) whose decision nodes are mapped to the corresponding chance nodes (A^1_j, A^2_j). Depending on the value of the node, Mod[M_j], the distribution of each of the chance nodes is assigned to the node A_j.** (c) **The transformed I-ID with the model node replaced by the chance nodes and the relationships between them.**

agent i's belief over its interactive states. In addition to the model node, I-IDs differ from IDs by having a dashed link (called the "policy link" in [15]) between the model node and a chance node, A_j, that represents the distribution over the other agent's actions given its model. In the absence of other agents, the model node and the chance node, A_j, vanish and I-IDs collapse into traditional IDs.

The model node contains the alternative computational models ascribed by i to the other agent from the set, $\Theta_{j,l-1} \cup SM_j$, where $\Theta_{j,l-1}$ and SM_j were defined previously in Section 2. Thus, 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 subintentional. Because the model node contains the alternative models of the other agent as its values, its representation is not trivial. In particular, some of the models within the node are I-IDs that when solved generate the agent's optimal policy in their decision nodes. Each decision node is mapped to the corresponding chance node, say A^1_j, in the following way: if OPT is the set of optimal actions obtained by solving the I-ID (or ID), then $Pr(a_j \in A^1_j) = \frac{1}{|OPT|}$ if $a_j \in OPT$, 0 otherwise.

Borrowing insights from previous work [8], 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, as we mentioned previously, so that the actions with the largest value in the decision node are assigned uniform probabilities in the chance node while the rest are assigned zero probability. The different chance nodes (A^1_j, A^2_j), one for each model, and additionally, the chance node labeled $Mod[M_j]$ form the parents of the chance node, A_j. Thus, there are as many action nodes (A^1_j, A^2_j) in $M_{j,l-1}$ as the number of models in the support of agent i's beliefs. The conditional probability table of the chance node, A_j, is a *multiplexer* that assumes the distribution of each of the action nodes (A^1_j, A^2_j) depending on the value of $Mod[M_j]$. The values of $Mod[M_j]$ denote the different models of j. In other words, when $Mod[M_j]$ has the value $m^1_{j,l-1}$, the chance node A_j assumes the distribution of the node A^1_j, and A_j assumes the distribution of A^2_j when $Mod[M_j]$ has the value $m^2_{j,l-1}$. The distribution over the node, $Mod[M_j]$, is the agent i's belief over the models of j given a physical state. For more agents, we will have as many model nodes as there are agents. Notice that Fig. $1(b)$ clarifies the semantics of the "policy link", and shows how it can be represented using the traditional dependency links.

In Fig. $1(c)$, we show the transformed I-ID when the model node is replaced by the chance nodes and relationships between them. In contrast to the representation in [15], there are no special-purpose "policy links", rather the I-ID is composed of only those types of nodes that are found in traditional IDs and dependency relationships between the nodes. This allows I-IDs to be represented and implemented using conventional application tools that target IDs. Note that we may view the level l I-ID as a NID. Specifically, each of the level $l-1$ models within the model node are blocks in the NID (see Fig. 2). If the level $l = 1$, each block is a traditional ID, otherwise if $l > 1$, each block within the NID may itself be a NID. Note that within the I-IDs (or IDs) at each level, there is only a single decision node. Thus, our NID does not contain any MAIDs.

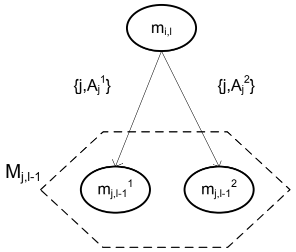

Figure 2: A level l I-ID represented as a NID. The probabilities assigned to the blocks of the NID are i's beliefs over j's models conditioned on a physical state.

3.2 Solution

The solution of an I-ID proceeds in a bottom-up manner, and is implemented recursively. We start by solving the level 0 models, which, if intentional, are traditional IDs. Their solutions provide probability distributions over the other agents' actions, which are entered in the corresponding chance nodes found in the model node of the level 1 I-ID. The mapping from the level 0 models' decision nodes to the chance nodes is carried out so that actions with the largest value in the decision node are assigned uniform probabilities in the chance node while the rest are assigned zero probability. Given the distributions over the actions within the different chance nodes (one for each model of the other agent), the level 1 I-ID is transformed as shown in Fig. $1(c)$. During the transformation, the conditional probability table (CPT) of the node, A_j, is populated such that the node assumes the distribution of each of the chance nodes depending on the value of the node, $Mod[M_j]$. As we mentioned previously, the values of the node $Mod[M_j]$ denote the different models of the other agent, and its distribution is the agent i's

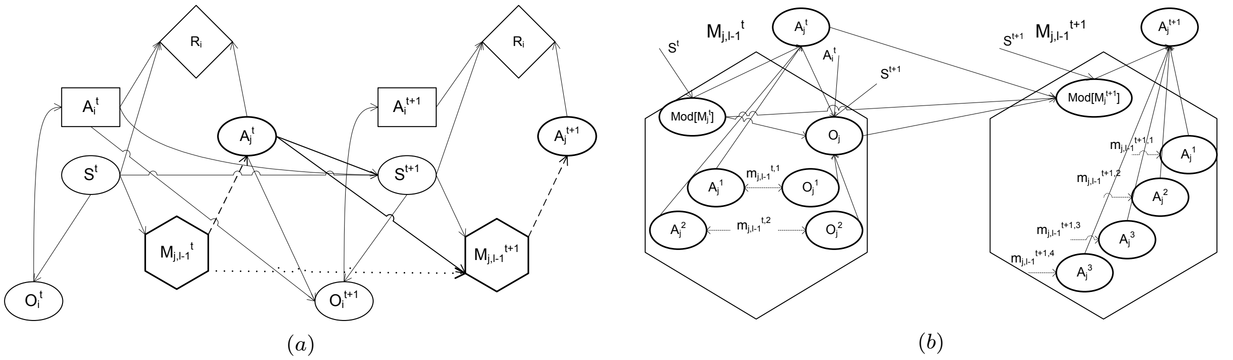

Figure 3: (a) **A generic two time-slice level l I-DID for agent i in a setting with one other agent j. Notice the dotted model update link that denotes the update of the models of j and the distribution over the models over time.** (b) **The semantics of the model update link.**

belief over the models of j conditioned on the physical state. The transformed level 1 I-ID is a traditional ID that may be solved using the standard expected utility maximization method [18]. This procedure is carried out up to the level l I-ID whose solution gives the non-empty set of optimal actions that the agent should perform given its belief. Notice that analogous to IDs, I-IDs are suitable for online decision-making when the agent's current belief is known.

4. INTERACTIVE DYNAMIC INFLUENCE DIAGRAMS

Interactive dynamic influence diagrams (I-DIDs) extend I-IDs (and NIDs) to allow sequential decision-making over several time steps. Just as DIDs are structured graphical representations of POMDPs, I-DIDs are the graphical online analogs for finitely nested I-POMDPs. I-DIDs may be used to optimize over a finite look-ahead given initial beliefs while interacting with other, possibly similar, agents.

4.1 Syntax

We depict a general two time-slice I-DID in Fig. 3(a). 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. 3(a). We explained the semantics of the model node and the policy link in the previous section; we describe the model updates 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 $t+1$. Recall from Section 2 that an agent's intentional model includes its belief. 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 at most $|M^t_{j,l-1}||A_j||\Omega_j|$ models. Here, $|M^t_{j,l-1}|$ is the number of models at time step t, $|A_j|$ and $|\Omega_j|$ are the largest spaces of actions and observations respectively, among all the models. 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. These steps are a part of agent i's belief update formalized using Eq. 1.

In Fig. 3(b), we show how the dotted model update link is implemented in the I-DID. If each of the two level $l-1$ models ascribed to j at time step t results in one action, and j could make one of two possible observations, then the model node at time step $t+1$ contains four updated models ($m^{t+1,1}_{j,l-1}$,$m^{t+1,2}_{j,l-1}$, $m^{t+1,3}_{j,l-1}$, and $m^{t+1,4}_{j,l-1}$). These models differ in their initial beliefs, each of which is the result of j updating its beliefs due to its action and a possible observation. The decision nodes in each of the I-DIDs or DIDs that represent the lower level models are mapped to the corresponding chance nodes, as mentioned previously. Next, we describe how the distribution over the updated set of models (the distribution over the chance node $Mod[M^{t+1}_j]$ in $M^{t+1}_{j,l-1}$) is computed. The probability that j's updated model is, say $m^{t+1,1}_{j,l-1}$, depends on the probability of j performing the action and receiving the observation that led to this model, and the prior distribution over the models at time step t. Because the chance node A^t_j assumes the distribution of each of the action nodes based on the value of $Mod[M^t_j]$, the probability of the action is given by this chance node. In order to obtain the probability of j's possible observation, we introduce the chance node O_j, which depending on the value of $Mod[M^t_j]$ assumes the distribution of the observation node in the lower level model denoted by $Mod[M^t_j]$. Because the probability of j's observations depends on the physical state and the joint actions of both agents, the node O_j is linked with S^{t+1}, A^t_j, and A^t_i.[2] Analogous to A^t_j, the conditional probability table of O_j is also a multiplexer modulated by $Mod[M^t_j]$. Finally, the distribution over the prior models at time t is obtained from the chance node, $Mod[M^t_j]$ in $M^t_{j,l-1}$. Consequently, the chance nodes, $Mod[M^t_j]$, A^t_j, and O_j, form the parents of $Mod[M^{t+1}_j]$ in $M^{t+1}_{j,l-1}$. Notice that the model update link may be replaced by the dependency links between the chance nodes that constitute the model nodes in the two time slices. In Fig. 4 we show the two time-slice I-DID with the model nodes replaced by the chance nodes and the relationships between them. Chance nodes and dependency links that not in bold are standard, usually found in DIDs.

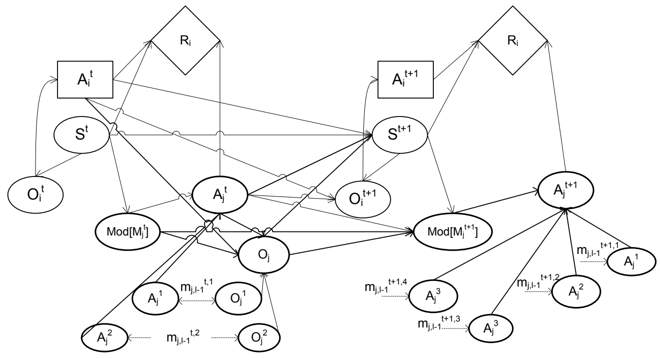

Figure 4: **Transformed I-DID with the model nodes and model update link replaced with the chance nodes and the relationships (in bold).**

[2] Note that O_j represents j's observation at time $t+1$.

Expansion of the I-DID over more time steps requires the repetition of the two steps of updating the set of models that form the values of the model node and adding the relationships between the chance nodes, as many times as there are model update links. We note that the possible set of models of the other agent j grows exponentially with the number of time steps. For example, after T steps, there may be at most $|M_{j,l-1}^{t=1}|(|A_j||\Omega_j|)^{T-1}$ candidate models residing in the model node.

4.2 Solution

Analogous to I-IDs, the solution to a level l I-DID for agent i expanded over T time steps may be carried out recursively. For the purpose of illustration, let l=1 and T=2. The solution method uses the standard look-ahead technique, projecting the agent's action and observation sequences forward from the current belief state [17], 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. Consequently, each of j's subintentional or 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 this updated set of candidate models are calculated using the standard inference methods using the dependency relationships between the model nodes as shown in Fig. $3(b)$. We note the recursive nature of this solution: in solving agent i's level 1 I-DID, j's level 0 DIDs must be solved. If the nesting of models is deeper, all models at all levels starting from 0 are solved in a bottom-up manner.

```
Algorithm for solving I-DID
Input : level $l \geq 1$ I-ID or level 0 ID, $T$

Expansion Phase
 1. For $t$ from $1$ to $T-1$ do
 2.     If $l \geq 1$ then
            Populate $M_{j,l-1}^{t+1}$
 3.         For each $m_j^t$ in Range($M_{j,l-1}^t$) do
 4.             Recursively call algorithm with the $l-1$ I-ID (or ID)
                that represents $m_j^t$ and the horizon, $T-t+1$
 5.             Map the decision node of the solved I-ID (or ID),
                $OPT(m_j^t)$, to a chance node $A_j$
 6.             For each $a_j$ in $OPT(m_j^t)$ do
 7.                 For each $o_j$ in $O_j$ (part of $m_j^t$) do
 8.                     Update $j$'s belief, $b_j^{t+1} \leftarrow SE(b_j^t, a_j, o_j)$
 9.                     $m_j^{t+1} \leftarrow$ New I-ID (or ID) with $b_j^{t+1}$ as the
                        initial belief
10.                     Range($M_{j,l-1}^{t+1}$) $\overset{\cup}{\leftarrow} \{m_j^{t+1}\}$
11.         Add the model node, $M_{j,l-1}^{t+1}$, and the dependency links
            between $M_{j,l-1}^t$ and $M_{j,l-1}^{t+1}$ (shown in Fig. 3($b$))
12.     Add the chance, decision, and utility nodes for $t+1$ time
        slice and the dependency links between them
13.     Establish the CPTs for each chance node and utility node

Look-Ahead Phase
 14. Apply the standard look-ahead and backup method to solve
the expanded I-DID
```

Figure 5: Algorithm for solving a level $l \geq 0$ I-DID.

We briefly outline the recursive algorithm for solving agent i's level l I-DID expanded over T time steps with one other agent j in Fig. 5. We adopt a two-phase approach: Given an I-ID of level l (described previously in Section 3) with all lower level models also represented as I-IDs or IDs (if level 0), the first step is to expand the level l I-ID over T time steps adding the dependency links and the conditional probability tables for each node. We particularly focus on establishing and populating the model nodes (lines 3-11). Note that Range($\cdot$) returns the values (lower level models) of the random variable given as input (model node). In the second phase, we use a standard look-ahead technique projecting the action and observation sequences over T time steps in the future, and backing up the utility values of the reachable beliefs. Similar to I-IDs, the I-DIDs reduce to DIDs in the absence of other agents.

As we mentioned previously, the 0-th level models are the traditional DIDs. Their solutions provide probability distributions over actions of the agent modeled at that level to I-DIDs at level 1. Given probability distributions over other agent's actions the level 1 I-DIDs can themselves be solved as DIDs, and provide probability distributions to yet higher level models. Assume that the number of models considered at each level is bound by a number, M. Solving an I-DID of level l in then equivalent to solving $O(M^l)$ DIDs.

5. EXAMPLE APPLICATIONS

To illustrate the usefulness of I-DIDs, we apply them to three problem domains. We describe, in particular, the formulation of the I-DID and the optimal prescriptions obtained on solving it.

5.1 Followership-Leadership in the Multiagent Tiger Problem

We begin our illustrations of using I-IDs and I-DIDs with a slightly modified version of the multiagent tiger problem discussed in [9]. The problem has two agents, each of which can open the right door (OR), the left door (OL) or listen (L). In addition to hearing growls (from the left (GL) or from the right (GR)) when they listen, the agents also hear creaks (from the left (CL), from the right (CR), or no creaks (S)), which noisily indicate the other agent's opening one of the doors. When any door is opened, the tiger *persists* in its original location with a probability of 95%. Agent i hears growls with a reliability of 65% and creaks with a reliability of 95%. Agent j, on the other hand, hears growls with a reliability of 95%. Thus, the setting is such that agent i hears agent j opening doors more reliably than the tiger's growls. This suggests that i could use j's actions as an indication of the location of the tiger, as we discuss below. Each agent's preferences are as in the single agent game discussed in [13]. The transition, observation, and reward functions are shown in [16].

A good indicator of the usefulness of normative methods for decision-making like I-DIDs is the emergence of realistic social behaviors in their prescriptions. In settings of the persistent multiagent tiger problem that reflect real world situations, we demonstrate *followership* between the agents and, as shown in [15], *deception* among agents who believe that they are in a follower-leader type of relationship. In particular, we analyze the situational and epistemological conditions sufficient for their emergence. The followership behavior, for example, results from the agent knowing its own weaknesses, assessing the strengths, preferences, and possible behaviors of the other, and realizing that its best for it to follow the other's actions in order to maximize its payoffs.

Let us consider a particular setting of the tiger problem in which agent i believes that j's preferences are aligned with its own - both of them just want to get the gold - and j's hearing is more reliable in comparison to itself. As an example, suppose that j, on listening can discern the tiger's location 95% of the times compared to i's 65% accuracy. Additionally, agent i does not have any initial information about the tiger's location. In other words, i's single-level nested belief, $b_{i,1}$, assigns 0.5 to each of the two locations of the tiger. In addition, i considers two models of j, which differ in j's

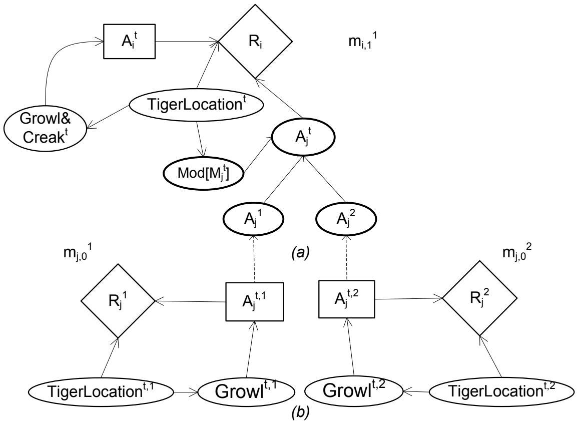

Figure 6: (a) **Level 1 I-ID of agent** i**,** (b) **two level 0 IDs of agent** j **whose decision nodes are mapped to the chance nodes,** A_j^1**,** A_j^2**, in** (a)**.**

flat level 0 initial beliefs. This is represented in the level 1 I-ID shown in Fig. $6(a)$. According to one model, j assigns a probability of 0.9 that the tiger is behind the left door, while the other model assigns 0.1 to that location (see Fig. $6(b)$). Agent i is undecided on these two models of j. If we vary i's hearing ability, and solve the corresponding level 1 I-ID expanded over three time steps, we obtain the normative behavioral policies shown in Fig 7 that exhibit followership behavior. If i's probability of correctly hearing the growls is 0.65, then as shown in the policy in Fig. $7(a)$, i begins to conditionally follow j's actions: i opens the same door that j opened previously iff i's own assessment of the tiger's location confirms j's pick. If i loses the ability to correctly interpret the growls completely, it blindly follows j and opens the same door that j opened previously (Fig. $7(b)$).

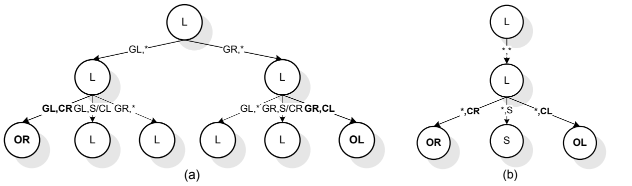

Figure 7: **Emergence of** (a) **conditional followership, and** (b) **blind followership in the tiger problem. Behaviors of interest are in bold. * is a wildcard, and denotes any one of the observations.**

We observed that a *single level* of belief nesting - beliefs about the other's models - was sufficient for followership to emerge in the tiger problem. However, the epistemological requirements for the emergence of leadership are more complex. For an agent, say j, to emerge as a leader, followership must first emerge in the other agent i. As we mentioned previously, if i is certain that its preferences are identical to those of j, and believes that j has a better sense of hearing, i will follow j's actions over time. Agent j emerges as a leader if it believes that i will follow it, which implies that j's belief must be nested *two levels* deep to enable it to recognize its leadership role. Realizing that i will follow presents j with an opportunity to influence i's actions in the benefit of the collective good or its self-interest alone. For example, in the tiger problem, let us consider a setting in which if both i and j open the correct door, then each gets a payoff of 20 that is double the original. If j alone selects the correct door, it gets the payoff of 10. On the other hand, if both agents pick the wrong door, their penalties are cut in half. In this setting, it is in both j's best interest as well as the collective betterment for j to use its expertise in selecting the correct door, and thus be a good leader. However, consider a slightly different problem in which j gains from i's loss and is penalized if i gains. Specifically, let i's payoff be subtracted from j's, indicating that j is antagonistic toward i - if j picks the correct door and i the wrong one, then i's loss of 100 becomes j's gain. Agent j believes that i incorrectly thinks that j's preferences are those that promote the collective good and that it starts off by believing with 99% confidence where the tiger is. Because i believes that its preferences are similar to those of j, and that j starts by believing almost surely that one of the two is the correct location (two level 0 models of j), i will start by following j's actions. We show i's normative policy on solving its singly-nested I-DID over three time steps in Fig. $8(a)$. The policy demonstrates that i will blindly follow j's actions. Since the tiger persists in its original location with a probability of 0.95, i will select the same door again. If j begins the game with a 99% probability that the tiger is on the right, solving j's I-DID nested *two levels* deep, results in the policy shown in Fig. $8(b)$. Even though j is almost certain that OL is the correct action, it will start by selecting OR, followed by OL. Agent j's intention is to deceive i who, it believes, will follow j's actions, so as to gain \$110 in the second time step, which is more than what j would gain if it were to be honest.

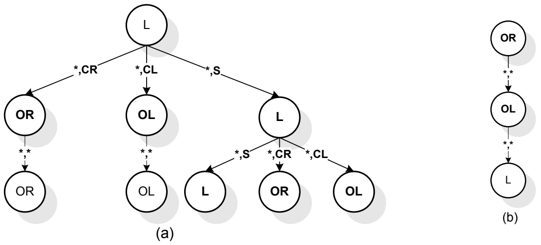

Figure 8: **Emergence of deception between agents in the tiger problem. Behaviors of interest are in bold. * denotes as before.** (a) **Agent** i**'s policy demonstrating that it will blindly follow** j**'s actions.** (b) **Even though** j **is almost certain that the tiger is on the right, it will start by selecting OR, followed by OL, in order to deceive** i**.**

5.2 Altruism and Reciprocity in the Public Good Problem

The public good (PG) problem [7], consists of a group of M agents, each of whom must either contribute some resource to a public pot or keep it for themselves. Since resources contributed to the public pot are shared among all the agents, they are less valuable to the agent when in the public pot. However, if all agents choose to contribute their resources, then the payoff to each agent is more than if no one contributes. Since an agent gets its share of the public pot irrespective of whether it has contributed or not, the dominating action is for each agent to not contribute, and instead "free ride" on others' contributions. However, behaviors of human players in empirical simulations of the PG problem differ from the normative predictions. The experiments reveal that many players initially contribute a large amount to the public pot, and continue to contribute when the PG problem is played repeatedly, though in decreasing amounts [4]. Many of these experiments [5] report that a small core group of players persistently contributes to the public pot even when all others are defecting. These experiments also reveal that players who persistently contribute have altruistic or reciprocal preferences matching expected cooperation of others.

For simplicity, we assume that the game is played between $M = 2$ agents, i and j. Let each agent be initially endowed with X_T amount of resources. While the classical PG game formulation permits each agent to contribute any quantity of resources ($\leq X_T$) to the public pot, we simplify the action space by allowing two possible actions. Each agent may choose to either *contribute* (C) a *fixed* amount of the resources, or not contribute. The latter action is denoted as *defect* (D). We assume that the actions are not observable to others. The value of resources in the public pot is discounted by c_i for each agent i, where c_i is the marginal private return. We assume that $c_i < 1$ so that the agent does not benefit enough that it contributes to the public pot for private gain. Simultaneously, $c_i M > 1$, making collective contribution pareto optimal.

i/j	**C**	**D**
C	$2c_i X_T, 2c_j X_T$	$c_i X_T - c_p, X_T + c_j X_T - P$
D	$X_T + c_i X_T - P, c_j X_T - c_p$	X_T, X_T

Table 1: The one-shot PG game with punishment.

In order to encourage contributions, the contributing agents punish free riders but incur a small cost for administering the punishment. Let P be the punishment meted out to the defecting agent and c_p the non-zero cost of punishing for the contributing agent. For simplicity, we assume that the cost of punishing is same for both the agents. The one-shot PG game with punishment is shown in Table. 1. Let $c_i = c_j$, $c_p > 0$, and if $P > X_T - c_i X_T$, then defection is no longer a dominating action. If $P < X_T - c_i X_T$, then defection is the dominating action for both. If $P = X_T - c_i X_T$, then the game is not dominance-solvable.

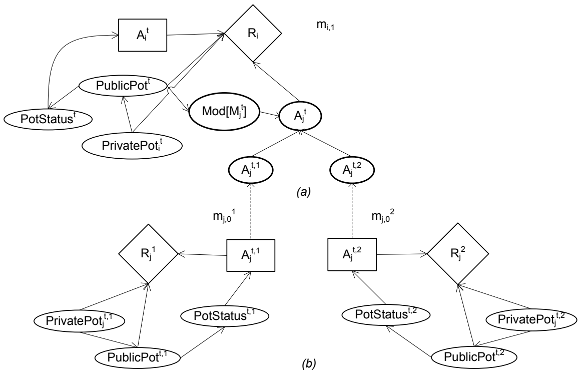

Figure 9: (a) **Level 1 I-ID of agent** i**,** (b) **level 0 IDs of agent** j **with decision nodes mapped to the chance nodes,** A_j^1 **and** A_j^2**, in** (a)**.**

We formulate a *sequential* version of the PG problem with punishment from the perspective of agent i. Though in the repeated PG game, the quantity in the public pot is revealed to all the agents after each round of actions, we assume in our formulation that it is hidden from the agents. Each agent may contribute a fixed amount[3], x_c, or defect. An agent on performing an action receives an observation of *plenty* (PY) or *meager* (MR) symbolizing the state of the public pot. Notice that the observations are also indirectly indicative of agent j's actions because the state of the public pot is influenced by them. The amount of resources in agent i's private pot, is perfectly observable to i. The payoffs are analogous to Table. 1. Borrowing from the empirical investigations of the PG problem [5], we construct level 0 IDs for j that model altruistic and non-altruistic types (Fig. 9(b)). Specifically, our altruistic agent has a high marginal private return (c_j is close to 1) and does not punish others who defect. Let x_c = 1 and the level 0 agent be punished half the times it defects. With one action remaining, both types of agents choose to contribute to avoid being punished. With two actions to go, the altruistic type chooses to contribute, while the other defects. This is because c_j for the altruistic type is close to 1, thus the expected punishment, $0.5P > (1 - c_j)$, which the altruistic type avoids. Because c_j for the non-altruistic type is less, it prefers not to contribute. With three steps to go, the altruistic agent contributes to avoid punishment ($0.5P > 2(1 - c_j)$), and the non-altruistic type defects. For greater than three steps, while the altruistic agent continues to contribute to the public pot depending on how close its marginal private return is to 1, the non-altruistic type prescribes defection.

We analyzed the decisions of an altruistic agent i modeled using a level 1 I-DID expanded over 3 time steps. i ascribes the two level 0 models, mentioned previously, to j (see Fig. 9). If i believes with a probability 1 that j is altruistic, i chooses to contribute for each of the three steps. This behavior persists when i is unaware of whether j is altruistic (Fig. 10(a)), and when i assigns a high probability to j being the non-altruistic type. However, when i believes with a probability 1 that j is non-altruistic and will thus surely defect, i chooses to defect to avoid being punished and because its marginal private return is less than 1. These results demonstrate that the behavior of our altruistic type resembles that found experimentally. The non-altruistic level 1 agent chooses to defect regardless of how likely it believes the other agent to be altruistic. We analyzed the behavior of a reciprocal agent type that matches expected cooperation or defection. The reciprocal type's marginal private return is similar to that of the non-altruistic type, however, it obtains a greater payoff when its action is similar to that of the other. We consider the case when the reciprocal agent i is unsure of whether j is altruistic and believes that the public pot is likely to be half full. For this prior belief, i chooses to defect. On receiving an observation of plenty, i decides to contribute, while an observation of meager makes it defect (Fig. 10(b)). This is because an observation of plenty signals that the pot is likely to be greater than half full, which results from j's action to contribute. Thus, among the two models ascribed to j, its type is likely to be altruistic making it likely that j will contribute again in the next time step. Agent i therefore chooses to contribute to reciprocate j's action. An analogous reasoning leads i to defect when it observes a meager pot. With one action to go, i believing that j contributes, will choose to contribute too to avoid punishment regardless of its observations.

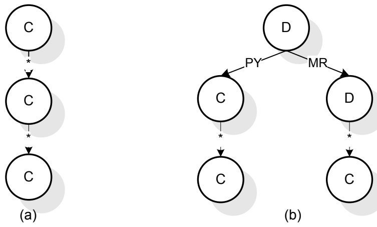

Figure 10: (a) **An altruistic level 1 agent always contributes.** (b) **A reciprocal agent** i **starts off by defecting followed by choosing to contribute or defect based on its observation of plenty (indicating that** j **is likely altruistic) or meager (**j **is non-altruistic).**

5.3 Strategies in Two-Player Poker

Poker is a popular zero sum card game that has received much attention among the AI research community as a testbed [2]. Poker is played among $M \geq 2$ players in which each player receives a *hand* of cards from a deck. While several flavors of Poker with varying complexity exist, we consider a simple version in which each player has three plys during which the player may either exchange a card (E), keep the existing hand (K), fold (F) and withdraw from the game, or call (C), requiring all players to show their hands. To keep matters simple, let $M = 2$, and each player receive a hand

[3] The amount is selected so that the resources last the entire game.

consisting of a single card drawn from the same suit. Thus, during a showdown, the player who has the numerically larger card (2 is the lowest, ace is the highest) wins the pot. During an exchange of cards, the discarded card is placed either in the L pile, indicating to the other agent that it was a low numbered card less than 8, or in the H pile, indicating that the card had a rank greater than or equal to 8. Notice that, for example, if a lower numbered card is discarded, the probability of receiving a low card in exchange is now reduced.

We show the level 1 I-ID for the simplified two-player Poker in Fig. 11. We considered two models (personality types) of agent j. The *conservative* type believes that it is likely that its opponent has a high numbered card in its hand. On the other hand, the *aggressive* agent j believes with a high probability that its opponent has a lower numbered card. Thus, the two types differ in their beliefs over their opponent's hand. In both these level 0 models, the opponent is assumed to perform its actions following a fixed, uniform distribution. With three actions to go, regardless of its hand (unless it is an ace), the aggressive agent chooses to exchange its card, with the intent of improving on its current hand. This is because it believes the other to have a low card, which improves its chances of getting a high card during the exchange. The conservative agent chooses to keep its card, no matter what its hand is because its chances of getting a high card are slim as it believes that its opponent has one.

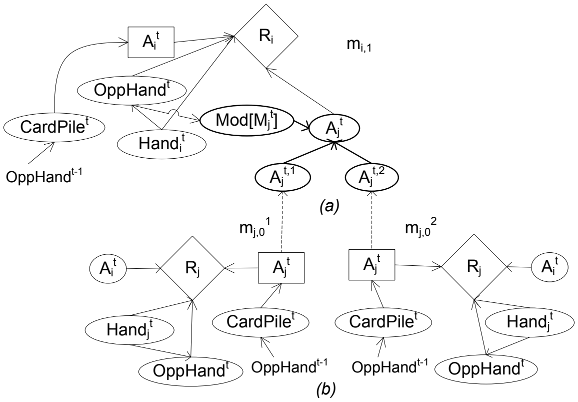

Figure 11: (a) **Level 1 I-ID of agent i. The observation reveals information about j's hand of the previous time step,** (b) **level 0 IDs of agent j whose decision nodes are mapped to the chance nodes, A_j^1, A_j^2, in** (a)**.**

The policy of a level 1 agent i who believes that each card except its own has an equal likelihood of being in j's hand (neutral personality type) and j could be either an aggressive or conservative type, is shown in Fig. 12. i's own hand contains the card numbered 8. The agent starts by keeping its card. On seeing that j did not exchange a card (N), i believes with probability 1 that j is conservative and hence will keep its cards. i responds by either keeping its card or exchanging it because j is equally likely to have a lower or higher card. If i observes that j discarded its card into the L or H pile, i believes that j is aggressive. On observing L, i realizes that j had a low card, and is likely to have a high card after its exchange. Because the probability of receiving a low card is high now, i chooses to keep its card. On observing H, believing that the probability of receiving a high numbered card is high, i chooses to exchange its card. In the final step, i chooses to call regardless of its observation history because its belief that j has a higher card is not sufficiently high to conclude that its better to fold and relinquish the payoff. This is partly due to the fact that an observation of, say, L resets the agent i's previous time step beliefs over j's hand to the low numbered cards only.

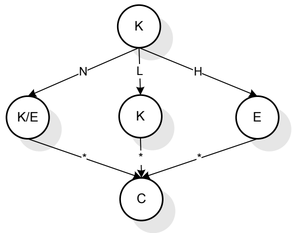

Figure 12: A level 1 agent i's three step policy in the Poker problem. i starts by believing that j is equally likely to be aggressive or conservative and could have any card in its hand with equal probability.

6. DISCUSSION

We showed how DIDs may be extended to I-DIDs that enable online sequential decision-making in uncertain multiagent settings. Our graphical representation of I-DIDs improves on the previous work significantly by being more transparent, semantically clear, and capable of being solved using standard algorithms that target DIDs. I-DIDs extend NIDs to allow sequential decision-making over multiple time steps in the presence of other interacting agents. I-DIDs may be seen as concise graphical representations for I-POMDPs providing a way to exploit problem structure and carry out online decision-making as the agent acts and observes given its prior beliefs. We are currently investigating ways to solve I-DIDs approximately with provable bounds on the solution quality.

7. REFERENCES

[1] R. J. Aumann. Interactive epistemology i: Knowledge. *International Journal of Game Theory*, 28:263–300, 1999.

[2] D. Billings, A. Davidson, J. Schaeffer, and D. Szafron. The challenge of poker. *AIJ*, 2001.

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

[4] C. Camerer. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press, 2003.

[5] E. Fehr and S. Gachter. Cooperation and punishment in public goods experiments. *American Economic Review*, 90(4):980–994, 2000.

[6] D. Fudenberg and D. K. Levine. *The Theory of Learning in Games*. MIT Press, 1998.

[7] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

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

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

[10] P. Gmytrasiewicz and E. Durfee. Rational coordination in multi-agent environments. *JAAMAS*, 3(4):319–350, 2000.

[11] J. C. Harsanyi. Games with incomplete information played by bayesian players. *Management Science*, 14(3):159–182, 1967.

[12] R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *The Principles and Applications of Decision Analysis*. Strategic Decisions Group, Menlo Park, CA 94025, 1984.

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

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

[15] K. Polich and P. Gmytrasiewicz. Interactive dynamic influence diagrams. In *GTDT Workshop, AAMAS*, 2006.

[16] B. Rathnas., P. Doshi, and P. J. Gmytrasiewicz. Exact solutions to interactive pomdps using behavioral equivalence. In *Autonomous Agents and Multi-Agent Systems Conference (AAMAS)*, 2006.

[17] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach (Second Edition)*. Prentice Hall, 2003.

[18] R. D. Shachter. Evaluating influence diagrams. *Operations Research*, 34(6):871–882, 1986.

[19] D. Suryadi and P. Gmytrasiewicz. Learning models of other agents using influence diagrams. In *UM*, 1999.