Interactive Dynamic Influence Diagrams

Kyle Polich and Piotr Gmytrasiewicz
Department of Computer Science, University of Illinois at Chicago
Chicago, IL, 60607-7053, USA
E-mail: kpolich@cs.uic.edu, piotr@cs.uic.edu

1. INTRODUCTION

Partially Observable Markov Decision Processes (POMDPs) emerged as the primary framework for decision-theoretic planning in single agent settings. Solutions to POMDPs are optimal plans which are conditional on future observations. Dynamic Influence Diagrams (DIDs) are computational representations of POMDPs which compute solutions for finite time horizons in an on-line fashion. Interactive POMDPs (I-POMDPs) [5] generalize POMDPs to multi-agent settings by including models of other agents in the state space. Interactive DIDs (I-DIDs), presented in this paper, are computational representations of I-POMDPs, and thus generalizations of DIDs. DIDs are themselves temporal generalizations of influence diagrams [6].

Solutions of I-POMDPs, computed by I-DIDs, are conditional plans which take into account the continually revised probabilistic prediction of other agents' behavior. The predictions are formed based on the models of the other agents. The models can themselves be I-DIDs, or, in simpler cases, DIDs or probability distributions over others' actions [5]. The possible nesting of agents' beliefs about each other's models has been studied in recent advances in game theory [1, 2, 3]. While the nesting of beliefs could be infinite, we assume finite nesting to ensure computability of the belief updates.

The solutions maximize the agent's expected utility, and thus do not rely on the notion of equilibrium. This approach has been called the decision-theoretic approach to game theory by some authors [7, 10].

2. SINGLE AGENT DECISION MAKING

A partially observable Markov decision processes (POMDP) [4, 8, 9] of an agent i is defined as

$$POMDP_i = \langle S, A_i, T_i, \Omega_i, O_i, R_i \rangle \quad (1)$$

where: S is a set of possible states of the environment, A_i is a set of actions agent i can execute, T_i is a transition function – $T_i : S \times A_i \times S \rightarrow [0, 1]$ which describes results of agent i's actions, Ω_i is the set of observations that the agent i can make, O_i is the agent's observation function – $O_i : S \times A_i \times \Omega_i \rightarrow [0, 1]$ which specifies probabilities of observations if agent executes various actions that result in different states, R_i is the reward function representing the agent i's preferences $R_i : S \times A_i \rightarrow \mathbf{R}$.

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

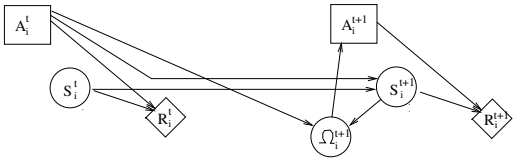

Figure 1: A Two Time Horizon Dynamic Influence Diagram

The nodes in a DID, like the one in Figure 1, are named for the elements of the POMDP which they represent. DIDs perform planning using a forward exploration technique known as reachability analysis. This technique explores the possible states of belief an agent may be in in the future, the likelihood of reaching each state of belief, and the expected utility of each belief state. The agent then adopts the plan which maximizes the expected utility.

3. FINITELY NESTED I-POMDPS

I-POMDPs generalize the POMDP framework to multi-agent environments [5]. Agent i considers the finitely nested *interactive* state space $IS_{i,l} = S \times M_j$ where S is the physical state and M_j is the set of models of agent j (for simplicity, we assume only two agents.) l is the level of nesting of i's I-POMDP. These models may include models j could have of i, nested to the level not greater than $l - 1$. Particular models may be referred to as m_j or, if the models are intentional (see [5] for details), θ_j.

(I-POMDP) An *I-POMDP* of agent i, is:

$$I\text{-}POMDP_{i,l} = \langle IS_{i,l}, A, T_i, \Omega_i, O_i, R_i \rangle \quad (2)$$

The belief update (derived in [5]) for I-POMDPs is:.

$$\begin{aligned} b_i^t(is^t) &= Pr(is^t | o_i^t, a_i^{t-1}) \\ &= \beta \textstyle\sum_{IS^{t-1}} \sum_{a_j^{t-1}} Pr(a_j^{t-1} | \theta_j^{t-1}) O_i(is^t, a^{t-1}, o_i^t) \\ &\times \textstyle\sum_{o_j^t} \tau_{\theta_j^t}(b_j^{t-1}, a_j^{t-1}, o_j^t, b_j^t) O_j(is_j^t, a^{t-1}, o_j^t) \\ &\times T_i(is^{t-1}, a^{t-1}, is^t) \end{aligned} \quad (3)$$

978-81-904262-7-5 (RPS)

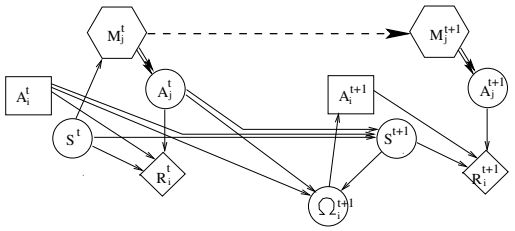

Figure 2: An Interactive Dynamic Influence Diagram

where $\theta_i = \langle b_i, A_i, \Omega_i, T_i, O_i, R_i, OC_i \rangle$, $is = (s, \theta_j)$, $is_j = (s, \theta_i)$, b_j^{t-1} and b_j^t are the belief elements of θ_j^{t-1} and θ_j^t, respectively, β is a normalizing constant, O_j is the observation function in θ_j^t, and $Pr(a_j^{t-1}|\theta_j^{t-1})$ is the probability that a_j^{t-1} is Bayesian rational for agent described by type θ_j^{t-1}. $\tau_{\theta_i}(b_i^{t-1}, a_i^{t-1}, o_i^t, b_i^t)$ stands for $Pr(b_i^t|b_i^{t-1}, a_i^{t-1}, o_i^t)$. OC_i is the optimality criterion used by agent i.

The τ_{θ_j} function performs agent j's belief update. The update equation makes explicit the fact that agent i must perform agent j's belief update as a part of it's own.

Analogously to POMDPs, each belief state in I-POMDP has an associated value reflecting the maximum payoff the agent can expect in this belief state:

$$U(\theta_i) = \max_{a_i \in A_i} \Big\{ \sum_{is} ER_i(is, a_i) b_i(is) + \gamma \sum_{o_i \in \Omega_i} Pr(o_i|a_i, b_i) U(\langle SE_{\theta_i}(b_i, a_i, o_i), \widehat{\theta}_i \rangle) \Big\} \quad (4)$$

where, $ER_i(is, a_i) = \sum_{a_j} R_i(is, a_i, a_j) Pr(a_j|m_j)$ (since $is = (s, \theta_j)$) [5].

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

$$OPT(\theta_i) = \underset{a_i \in A_i}{argmax} \Big\{ \sum_{is} ER_i(is, a_i) b_i(is) + \gamma \sum_{o_i \in \Omega_i} Pr(o_i|a_i, b_i) U(\langle SE_{\theta_i}(b_i, a_i, o_i), \widehat{\theta}_i \rangle) \Big\} \quad (5)$$

4. INTERACTIVE DYNAMIC INFLUENCE DIAGRAMS

I-DIDs contain some elements not present in classical dynamic influence diagrams (see Figure 2): hexagonal nodes, M_j, are called modeling nodes, dotted links are called belief update links, and double links are called policy links. Additionally, there is a chance node, A_j, representing the actions of the other agent.

The chance node A_j contains i's belief about j's action. The values of the node M_j are the possible models of j, which themselves are I-DIDs. Node S, representing the set of physical states, and M_j together represent the interactive state space, IS_i. The link between S_i and M_j captures the possible dependence between physical states and j's models.

The policy link between M_j and A_j represents the dependence between j's models and j's behavior, i.e., the $Pr(a_j^{t-1}|\theta_j^{t-1})$ in Eq 3. This is not a classical link, however, because the probability distribution over actions are obtained based on solutions (i.e., optimal actions) for each model in M_j.

The belief update link in I-DIDs connects the M_j nodes over time. It implements the τ_{θ_j} function in Eq. 3, representing the change in j's belief after its own observation and action. It clearly is not a classical link since it involves performing j's belief update.

5. SOLVING AN I-DID

Solution of an I-DID proceeds analogously to solution of a DID. This involves building a reachability tree containing the agent's beliefs for various sequences of its actions and observations. In case of I-DIDs the beliefs are over the interactive state space, and thus include beliefs over the models of the other agent. Using the example in Figure 2, the models of j contained in node M_j^t are solved first and impart the probability distribution to node A_j^t. This corresponds to computing $Pr(a_j^t|\theta_j^t)$ (this involves computing $OPT(\theta_j)$.) The policy link then becomes a conventional link between chance nodes (the node M_j^t values are now policies which are optimal for agent j.)

Now, given i's observation at time $t+1$, the probability distributions residing in nodes A_j^t and M_j^t are corrected. The corrected distribution over A_j^t can now be used to compute the corrected distribution over physical state, S^{t+1}. Also, given the distribution over A_j^t and i's action at time t, the probabilities of various observations available to j in each of its alternative models at time t can be computed.

Next, i simulates j's belief update for each of its models, and each possible action and observation pair. This implements the τ_{θ_j} function in Eq. 3 and updates the distribution in the M_j^{t+1} node. This completes the update of i's beliefs over one time step (i.e., implementing the τ_{θ_j} function).

Given the above, the recursive character of the solution process becomes transparent: Updating i's own beliefs involves computing solutions to i's models of j, and performing j's belief update, τ_{θ_j}. The recursion terminates at level not deeper than l, given the finitely nested *I-POMDP*$_{i,l}$ of agent i.

6. I-DID SOLUTION OF THE MULTI-AGENT PERSISTENT TIGER GAME

The single agent tiger game was first introduced in [8]. In this game, an agent is confronted with two doors. Behind one is a pot of gold (10 point reward for finding it) and behind the other is a tiger that attacks the agent (100 point penalty). During each time step, the agent may open either door or listen for the tiger's growl (listening incurs a 1 point cost). When listening, the agent hears growls which indicate the location of the tiger with some reliability.

In the multi-agent persistent tiger game variation, first, there are two agents facing the doors, and second, the tiger's location is reset with the probability of 0.05 after any door is opened. Agents also receive richer observations. Their observations come from the set $\{GL, GR\}$ x $\{CL, CR, S\}$. The elements of $\{GL, GR\}$ represent tiger's growl from left and right door, respectively. The signals in the second set represent a creaking noise generated by the opponent's opening the left door (CL), the right door (CR), or silence (S) if the door has not been opened. Thus, the creaks are informative of the actions taken by the other agent and can be used to update their models.

Agents may have a variety of different reward functions. Let us consider two types of reward functions: Friend and

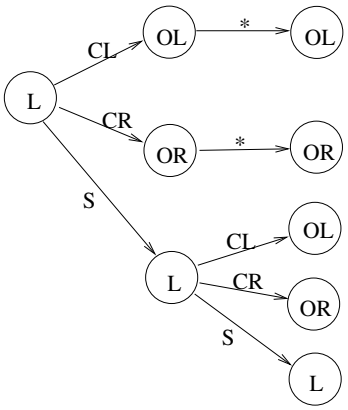

Figure 3: The policy of agent i's model of agent j.

Figure 4: The top-level policy of agent i.

Enemy. A Friend receives the sum of the single agent reward structure (e.g. 10 + -1 = 9 if the first agent opens the door with the gold and the second listens). An Enemy receives the difference of the single agent's rewards (e.g. -100 - 10 = -110 if the first agent opens the tiger door and the second agent opens the door with the gold). When either door is opened, the tiger switches location with 5% probability. If no door is opened, the tiger stays put.

In this simple game, I-DID solutions elucidate many interesting interactive phenomena. We will examine agent i which begins the game with 99% certainly that the tiger is behind the left door, a reward function of Enemy, and 85% reliability of both creaks and growls.

For simplicity, let us assume that agent i has only one model of agent j. According to this model, first, j has no information about the tiger's location. Second, j hears creaks with high reliability (95% accuracy), but its hearing of the tiger's growls is uninformative (uniform likelihood of growls). Third, j has the Friend reward function. Finally, j has two models of i (one for each possible location of the tiger) according to which i is almost certain (99%) of the Tiger's true location. In other words, i knows that j does not know where the tiger is, but also that j knows that i knows the tiger's location.

We will examine this situation for a game of three time horizons. The solutions proceeds bottom up, starting with j's two models of i. These models indicate that i will open the door i believes hides the gold for all three time horizons.

The solutions of the nesting level 0 models are now incorporated into the model of agent j. The policy of this agent is described in the Figure 3.

Since agent j's observation function prevent's it from learning anything from growls, it needs to rely entirely on the actions of agent i to infer anything about the location of the tiger. We can see this behavior, which can be called "Follow the Leader", in Figure 3.

Agent i's policy is shown in Figure 4. Note that the first action in this sequence opens the door which i believes is likely to hide the tiger! The reason is that, given j's "Follow the Leader" policy, and i's reward for when j opens the wrong door, it pays off for i to deceive agent j. The cost of this deception is rewarded in the subsequent two time steps.

Agent i does not open any door at time $t = 3$ due to mounting uncertainties about the tiger's location and j's actions.

7. CONCLUSION

I-DIDs are a powerful tool for deliberative agents acting in partially observable, non-deterministic, stochastic multi-agent environments. I-DIDs offer all the benefits traditionally associated with influence diagrams including the ability to make statistical queries, compact representation of probability distributions, and expressing the conditional elements of a network in a clear, intuitive, graphical manner. I-DIDs are also a finite look-ahead approximations of infinite horizon I-POMDPs.

8. REFERENCES

[1] R. J. Aumann and A. Heifetz. Incomplete information. In R. Aumann and S. Hart, editors, *Handbook of Game Theory with Economic Applications, Volume III, Chapter 43*. Elsevier, 2002.

[2] P. Battigalli and G. Bonano. Recent results on belief, knowledge and the epistemic foundations of game theory. Technical Report 98-14, University of California, Davis - Department of Economics, 1998.

[3] P. Battigalli and M. Siniscalchi. Hierarchies of conditional beliefs and interactive epistemology in dynamic games. *Journal of Economic Theory*, (88):188–230, 1999.

[4] C. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial intelligence Research*, 11:1–94, 1999.

[5] P. Gmytrasiewicz and P. Doshi. A framework for sequential planning in multiagent settings. *Journal of Artificial Intelligence Research*, 24:49–79, 2005. http://jair.org/contents/v24.html.

[6] R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *Readings on Principles and Applications of Decision Analysis*, pages 721–762. Strategic Decisions Group, Menlo Park, CA, 1984.

[7] J. B. Kadane and P. D. Larkey. Subjective probability and the theory of games. *Management Science*, 28(2):113–120, Feb. 1982.

[8] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(2):99–134, 1998.

[9] G. E. Monahan. A survey of partially observable markov decision processes: Theory, models, and algorithms. *Management Science*, pages 1–16, 1982.

[10] R. B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, 1991.