Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence

Influence-Based Abstraction for Multiagent Systems

Frans A. Oliehoek
MIT / Maastricht University
P.O. Box 616
6200 MD Maastricht, The Netherlands

Stefan J. Witwicki
INESC-ID / Instituto Superior Técnico, UTL
Av. Prof. Dr. Cavaco Silva
2744-016 Porto Salvo, Portugal

Leslie P. Kaelbling
MIT
32 Vassar Street
Cambridge, MA 02139-4307, USA

Abstract

This paper presents a theoretical advance by which factored POSGs can be decomposed into local models. We formalize the interface between such local models as the *influence* agents can exert on one another; and we prove that this interface is sufficient for decoupling them. The resulting influence-based abstraction substantially generalizes previous work on exploiting weakly-coupled agent interaction structures. Therein lie several important contributions. First, our general formulation sheds new light on the theoretical relationships among previous approaches, and promotes future empirical comparisons that could come by extending them beyond the more specific problem contexts for which they were developed. More importantly, the influence-based approaches that we generalize have shown promising improvements in the scalability of planning for more restrictive models. Thus, our theoretical result here serves as the foundation for practical algorithms that we anticipate will bring similar improvements to more general planning contexts, and also into other domains such as approximate planning, decision-making in adversarial domains, and online learning.

1 Introduction

Multiagent planning and learning are very active and challenging topics of study, particularly when problems involve uncertainties in the effect of actions and when agents only make individual partial observations of their environment. In the single-agent case, much research has focused on the *Markov decision process (MDP)* framework and the *partially observable MDP (POMDP)* (Kaelbling, Littman, and Cassandra 1998). The natural generalization of these frameworks to multiagent systems (MASs) are the *decentralized MDP (Dec-MDP)* and *Dec-POMDP* (Bernstein et al. 2002) and, in the case of self interested agents, the *partially observable stochastic game (POSG)*. However, due to the additional complexity of decentralization—planning for a Dec-MDP is NEXP complete—research on MASs has lead to a plethora of models, each of them geared at exploiting specific types of structure in the agents interactions (Guestrin, Koller, and Parr 2002; Becker, Zilberstein, and Lesser 2004; Becker et al. 2003; Nair et al. 2005; Spaan and Melo 2008; Mostafa and Lesser 2009; Varakantham et al. 2009; Witwicki and Durfee 2010; Melo and Veloso 2011).

In this paper we offer a unified perspective on a large number of these approaches by presenting *influence-based abstraction for factored POSGs (IBA)* and showing how it generalizes the abstractions performed by those approaches. The general idea of IBA is as follows. A standard way for agent i to compute a best response to the policy π_j of another agent j, is to maintain a belief over the full state s and observation history (or other form of internal state) of the other agent. Such a belief is a sufficient statistic, but is also difficult to maintain, since the total state space may be large and the number of histories grows exponentially with time. However, in many cases, the agent may not need to reason about the entire state space; its reward function may depend on a small number of state variables, called *state factors*, and so may his observations. Therefore, in many cases it is quite natural to specify a subset of state factors that will constitute the *local state* of an agent. Consequently, it becomes unnecessary to reason about the entire state space and every detail of the other agent's actions; it only cares about how the 'external' portion of the problem (i.e., state factors not modeled in agent i's state space and actions and observations of other agents) affects its local state. IBA is a two-stage process. First the agent performs inference to compute a (conditional) probability distribution over the relevant external variables, called the (incoming) influence point. Next, the agent uses the influence point to perform a marginalization of the conditional probability tables, thereby isolating its local problem from superfluous external variables.

When the local state is well-chosen, IBA may lead to improved efficiency of best-response computation and opens the door to intuitive approximations via approximate inference. More importantly, however, there are strong indications that for cooperative MASs where the agents are weakly coupled, it may be more efficient to search in the space of joint influences, rather than the space of joint policies. For instance, Witwicki and Durfee (2010) define an explicit form of influence and show how this can substantially scale up optimal solutions for a restricted sub-class of Dec-POMDPs. In this paper, we introduce a different, novel definition of influence and show how it can be used to decompose factored partially observable stochastic games (fPOSGs), one of the most general frameworks for multiagent planning under uncertainty. That is, we show how it is possible to perform IBA in virtually any multiagent system, providing a better un-

derstanding of weakly-coupled interactions and laying the foundation for influence-space search for all[1] MASs.

2 Background

Definition 1. A *partially observable stochastic game (POSG)* (Hansen, Bernstein, and Zilberstein 2004) is a tuple $\mathcal{M} = \langle \mathcal{D}, \mathcal{S}, \{\mathcal{A}_i\}, \{\mathcal{O}_i\}, T, O, \{R_i\}, b^0, h \rangle$, where $\mathcal{D} = \{1, \ldots, n\}$ is the set of n agents, $\mathcal{S}$ is a finite set of states s, $\mathcal{A}_i$ is the set of actions available to agent i, $\boldsymbol{\mathcal{A}} = \times_{i \in \mathcal{D}} \mathcal{A}_i$ is the set of *joint actions* $\boldsymbol{a} = \langle a_1, ..., a_n \rangle$, $\mathcal{O}_i$ is a set of observations available to agent i, $\boldsymbol{\mathcal{O}} = \times_{i \in \mathcal{D}} \mathcal{O}_i$ is the set of joint observations $\boldsymbol{o} = \langle o_1, ..., o_n \rangle$, T is the transition function which specifies $\Pr(s'|s, \boldsymbol{a})$, O is the observation function which specifies the probabilities $\Pr(\boldsymbol{o}|\boldsymbol{a}, s')$ of joint observations, $R_i(s, \boldsymbol{a}, s')$ is agent i's immediate reward (or payoff) function, $b^0 \in \Delta(\mathcal{S})$ is the initial state distribution at stage $t = 0$, h is the horizon of the problem, which we will assume to be finite.

In a POSG, the agents act over h stages, selecting their actions based only on the private information they posses; the *policies* are specified as mappings from individual action-observation histories (AOHs) $\vec{\theta}_i^t = \left(a_i^0, o_i^1, \ldots, a_i^{t-1}, o_i^t\right)$ to probability distributions over actions: $\pi_i(\vec{\theta}_i^t) \in \Delta(\mathcal{A}_i)$.

In this paper, we are interested in the generalization where there are multiple state variables, or *factors*, that make up the state. A *factored POSG (fPOSG)* is a POSG where the states are induced by a set of state factors $\mathcal{SF} = \{F_1, \ldots, F_{|\mathcal{SF}|}\}$. A state is an assignment of a value to each state factor. In an fPOSG, the transition and observation model can be represented compactly using a two-stage dynamic Bayesian network (2DBN), which specifies a conditional probability table (CPT) for each variable in the problem (Boutilier, Dean, and Hanks 1999). Notably, the special case with identical payoffs is known as a (factored) Dec-POMDP, or (f)Dec-POMDP (Oliehoek et al. 2008). The 2DBN representation exploits conditional independence for more compact modeling. An example of a real-world problem that has a lot of such conditional independence is urban traffic light control (Wiering 2000), where each traffic light's actions will only (directly) influence the amount of traffic on adjacent road segments.

An important concept in MASs is that of *best response*: the best policy for an agent i given the policies of the other agents. There are large sub-classes of games, e.g., congestion games and potential games (Rosenthal 1973; Monderer and Shapley 1996) including Dec-POMDPs, where repeated application of best-response computation leads to an equilibrium. Best-response computation is also important from the perspective of an individual self-interested agent that finds itself surrounded by other agents that are not necessarily rational. In this setting the agent may want to estimate the policies of the other agents and use a best-response policy to try and maximize its expected payoff.

Nair et al. (2003) demonstrated how it is possible to use dynamic programming to compute a best response in a Dec-POMDP, but their result can be trivially extended to POSGs. In essence, given the fixed policy of the other agents π_{-i}, from the perspective of agent i the problem reduces to a special type of single-agent POMDP whose states are pairs $\langle s^t, \vec{\theta}_{-i}^t \rangle$. A belief $b_i^t(s^t, \vec{\theta}_{-i}^t)$ can be maintained by application of Bayes rule' and the solution of the POMDP will provide a value function mapping these beliefs to values. The best-response value for agent i is defined as the value of the initial state distribution: $V_i^*(\pi_{-i}) \triangleq V_i(b_i^0)$.

We will refer to the above belief as the *global* individual belief of agent i, since it is defined over the global state and history of all other agents.[2] Although the global belief is a sufficient statistic for predicting the value, it has the drawback that the agent needs to reason about the entire global state of the system and even the internal state (i.e., the action-observation histories) of the other agents. In the next section we will introduce Local-Form Models that will form the basis of a local, and potentially more compact, belief that is also a sufficient statistic.

3 Local-Form Models

Here we introduce the local-form fPOSG model, wherein we augment the vanilla fPOSG with a description of each agent's *local state*. Like in earlier models for influence abstraction in more restrictive settings (Witwicki and Durfee 2010), local state descriptions comprise potentially overlapping subsets of state factors that will allow us to decompose an agent's best-response computation from the global state.

Definition 2. The *local state function* $S : \mathcal{D} \rightarrow 2^{\mathcal{SF}}$ maps from agents to subsets of state factors $S(i)$ that are in the agents local state.

We say that a state factor F is *observed* by an agent i if it influences the probability of the agent's observation. That is, when in the 2DBN there is a path from F^t to o_i^t (i.e., F is an intra-stage ancestor of o_i^t). Similarly, a state factor F is *reward-relevant* for an agent i if it influences the agent's rewards, i.e., if F^t or F^{t+1} is an ancestor of R_i^t. We say that a state factor F is *modeled* by an agent i if it is part of its local state space: $F \in S(i)$. From the perspective of agent i, S partitions $S(i)$ in two sets: a set of *private* factors that it models but other agents do not, and a set of *mutually-modeled factors* (MMFs) that are modeled by agent i as well as some other agent j. We will write $\boldsymbol{x}_i$ and $\boldsymbol{m}_i$ for an instantiation of repectively the private factors and MMFs of agent i. A *local state* s_i of agent i is an assignment of values to all state factors modeled by that agent $s_i \triangleq \langle \boldsymbol{x}_i, \boldsymbol{m}_i \rangle$. We can now define the local-form model.

Definition 3. A *local-form POSG,* also referred to as *local-form model (LFM),* is a pair $\langle \mathcal{M}, S \rangle$, where $\mathcal{M}$ is an fPOSG and S is a local state function such that, for all agents:

- All observed factors are in the local state.
- All reward-relevant factors are in the local state.
- Private factors F can only be influenced by the own action, other private factors and MMFs.

[1] Influence search has only been demonstrated for cooperative MASs, but the core idea may transfer to search for Nash equilibria.

[2] Note that a global belief is entirely different from a so-called *joint belief* (a distribution over states induced by a joint AOH).

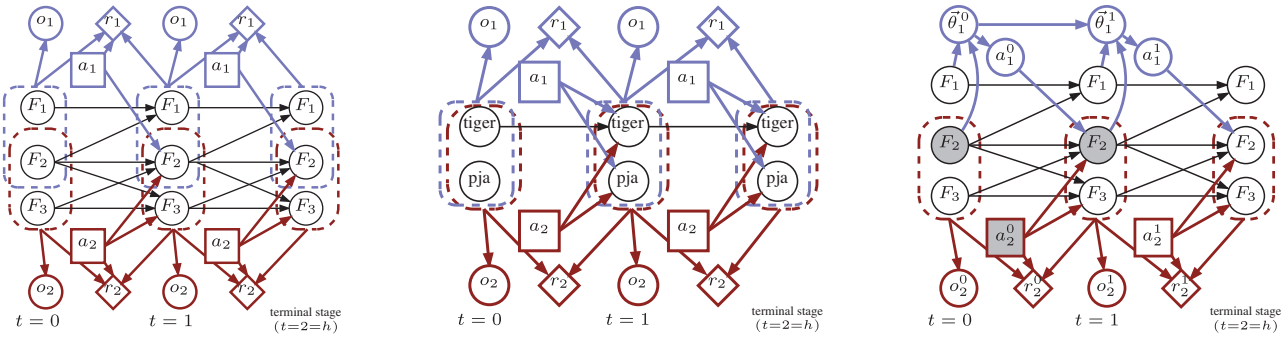

Figure 1: (Left) local form of a factored POSG. (Middle) DEC-TIGER in local form. (Right) agent 2's best response model.

Fig. 1(left) illustrates the LFM, showing that actions of multiple agents can affect the same state factor, that MMFs can affect private factors (illustrated for agent 1), and that private factors can influence MMFs (illustrated for agent 2). Intra-stage connections (not shown) are also allowed.

The LFM definition requires the additional specification of S that satisfies a number of properties (as per Def. 3), which might seem restrictive. On the contrary, here we show that any fPOSG can be converted to an LFM (although it may lead to the introduction of additional state factors).

Theorem 1. *Any fPOSG $\mathcal{M}$ can be converted to an equivalent problem in local form.*

Proof. Trivially, any fPOSG can be converted to a POSG by flattening the state representation. Here we will show how any POSG can be converted to local form. First, take the entire POSG state s^t and define it to be the first state factor of the LFM. Add a 'previous joint action' (pja) state factor, with incoming edges from all the individual actions. Add a 'joint observation' (jo) state factor with incoming edges from all the individual actions and an intra-stage edge from s^t, with CPT O (the observation function), and with an outgoing edge to every agent's individual observation (a deterministic function). Specify the local state mapping $\forall i\ S(i) = \mathcal{SF}$, which includes the entire set of state factors including {pja,jo}, all of which are consequently MMFs. At this point, the observations and rewards of agent only depend on their local state. Thus, we constructed an LFM. □

This shows that there is an LFM for every fPOSG, and therefore that the criteria from Def. 3 do not *what* can be modeled, but only on *how* it can be modeled; everything can be represented by promoting enough state factors to MMFs. However, it does not give us a recipe to find the LFM that will lead to the most compact local models. It may take some insight in order to determine which state factors should be modeled by an agent, but in practice it is usually possible find good descriptions of local state (Becker, Zilberstein, and Lesser 2004; Nair et al. 2005; Spaan and Melo 2008; Melo and Veloso 2011).

Example. The LFM for the DEC-TIGER benchmark (Nair et al. 2003) is shown in Fig. 1 (middle). Because the joint reward depends on the joint action, we introduce a state factor for the previous joint action (pja); the original state that encodes the tiger location is now called 'tiger'. Next we define the local reward functions as $R_i(s_i = \langle \text{tiger}, * \rangle, s_i' = \langle *, \text{pja} \rangle) \triangleq R(s = \text{tiger}, \boldsymbol{a} = \text{pja})/2$. We do not need to introduce a joint observation state factor, since the observation probabilities are specified as the product: $\Pr(\boldsymbol{o}|\boldsymbol{a}, s) = \Pr(o_1|\boldsymbol{a}, s) \Pr(o_2|\boldsymbol{a}, s)$.

4 Influence-Based Abstraction

Here we treat an LFM from the perspective of one agent and consider how that agent is affected by the other agents and can compute a best response against that 'incoming' influence.[3] Though the proposed framework can deal with intra-stage connections, in order to reduce the notational burden we will not consider them here.

4.1 Definition of Influence

As discussed in Section 2, when the other agents are following a fixed policy, they can be regarded as part of the environment. The resulting decision problem can be represented by the complete unrolled DBN, as illustrated in Fig. 1(right). In this figure, a node F^t is a different node than F^{t+1} and an edge at (emerging from) stage t is a different from the edge at $t+1$ that corresponds to the same edge in the 2DBN. Given this uniqueness of nodes and edges, we can define the interface between the agents as follows.

Intuitively, the influence of other agents is the effect of those edges leading into the agent's local state space. We say that every directed edge from the external portion of the problem to an MMF $m^t \in S(i)$ is an *influence link* $\langle u^t, m^t \rangle$, where u^t is called the *influence source* and m^t is the *influence destination*. Variable u^t can be either an action a_j^{t-1} or non-modeled state factor x_{-i}^{t-1}. We write $\boldsymbol{u}_{\rightarrow i}^t = \langle \boldsymbol{x}_u^{t-1}, \boldsymbol{a}_u^{t-1} \rangle$ for an instantiation of the all influence sources exerting influence on agent i at stage t. We write $\vec{\boldsymbol{\theta}}_u^t$ for the AOHs of those other agents whose action is an influence source.

We will define the influence as a conditional probability distribution over $\boldsymbol{u}_{\rightarrow i}^t$, making use of the concept of d-separation in graphical models (Koller and Friedman 2009). Let $d(I_{\rightarrow i}^{t+1})$ be a *d-separating set* for agent i's influence at stage $t+1$: a subset of variables

[3] An agent also exerts 'outgoing' influence on other agents, but this is irrelevant for best response computation.

at stages $0, \ldots, t$ that d-separates $\boldsymbol{x}_u^t, \vec{\boldsymbol{\theta}}_u^t$ from $\boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\theta}_i^t$ such that $\Pr(\boldsymbol{x}_u^t, \vec{\theta}_u^t | \boldsymbol{x}_i^t, \boldsymbol{m}_i^t, d(I_{\to i}^{t+1}), \vec{\theta}_i^t, b^0, \boldsymbol{\pi}_{-i}) = \Pr(\boldsymbol{x}_u^t, \vec{\theta}_u^t | d(I_{\to i}^{t+1}), b^0, \boldsymbol{\pi}_{-i})$.

Definition 4. The *incoming influence* at stage $t+1$, $I_{\to i}^{t+1}(\boldsymbol{\pi}_{-i})$, is a conditional probability distribution over values of the influence sources:

$$I(\boldsymbol{u}_{\to i}^{t+1} | d(I_{\to i}^{t+1})) \triangleq \sum_{\vec{\boldsymbol{\theta}}_u^t} \Pr(\boldsymbol{a}_u^t | \vec{\boldsymbol{\theta}}_u^t) \Pr(\boldsymbol{x}_u^t, \vec{\boldsymbol{\theta}}_u^t | d(I_{\to i}^{t+1}), b^0, \boldsymbol{\pi}_{-i}).$$

Here, I is shorthand for $I_{\to i}^{t+1}(\boldsymbol{\pi}_{-i})$.

The shaded nodes in Fig. 1(right) illustrate a d-separating set for $I_{\to 2}^2$. We will assume that the d-separating sets chosen can be expressed as the history of some subset $\boldsymbol{D}_i \subseteq S(i)$ of features. That is, we will assume that $d(I_{\to i}^{t+1})$ is given by $\vec{\boldsymbol{D}}_i^t$. In the figure, for instance, $\boldsymbol{D}_i = \{F_2, a_2\}$.

Definition 5. An *incoming influence point* $I_{\to i}(\boldsymbol{\pi}_{-i})$ for agent i, specifies the incoming influences for all stages $I_{\to i}(\boldsymbol{\pi}_{-i}) = \big(I_{\to i}^0(\boldsymbol{\pi}_{-i}), \ldots, I_{\to i}^{h-1}(\boldsymbol{\pi}_{-i})\big)$.

4.2 Influence-Augmented Local Models

Given the incoming influence $I_{\to i}(\boldsymbol{\pi}_{-i})$, agent i has an *influence-augmented local model (IALM)*, which we now define. The basic idea is that at each stage t, given $I_{\to i}^t$, we can, for all factors that are influence destinations, compute *induced CPTs* that only depend on the local state and $\vec{\boldsymbol{D}}_i^{t-1}$.

An IALM is a factored POMDP defined as follows. States are tuples $\langle \boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\boldsymbol{D}}_i^{t-1} \rangle$. Actions and observations are the unmodified sets $\mathcal{A}_i$, $\mathcal{O}_i$. Observation probabilities and rewards are directly given by the local form fPOSG; by construction the observation probabilities will only depend on $a_i^t, \boldsymbol{x}_i^{t+1}, \boldsymbol{m}_i^{t+1}$. The local model remains factored, which means that transition probabilities $\Pr(\boldsymbol{x}_i^{t+1}, \boldsymbol{m}_i^{t+1}, \vec{\boldsymbol{D}}_i^t | \boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\boldsymbol{D}}_i^{t-1}, a_i)$ of the local model are given by the product of the CPTs for all state factors in $S(i)$. These CPTs are defined as follows.

By construction, the parents of a private factor $x \in S(i)$ are contained within the IALM and no adaptations are necessary. The same holds for an MMF $m \in S(i)$ that is not an influence destination. Every $m \in S(i)$ that is the destination of one or more influence links has an *induced CPT*, that is specified as follows.

$$\Pr(m^{t+1} | \boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\boldsymbol{D}}_i^t, a_i, I_{\to i}^{t+1}) \triangleq \sum_{\boldsymbol{u}_{\to i}^{t+1} = \langle a_u^t, \boldsymbol{x}_u^t, \rangle} \Pr(m^{t+1} | \boldsymbol{x}_i^t, \boldsymbol{m}_i^t, a_i, \boldsymbol{x}_u^t, a_u) I(\boldsymbol{u}_{\to i}^{t+1} | \vec{\boldsymbol{D}}_i^t).$$

Since an IALM is just a special case of POMDP, regular POMDP solution methods can be used to compute $V_i^*(I_{\to i}(\pi_{-i}))$, the value of the IALM.

In order to prove that our definition of influence actually constitutes a sufficient statistic, and thus that resulting IALM actually achieves a best response against the policy π_{-i} that generated the influence $I_{\to i}(\pi_{-i})$, we show that the latter actually achieves the same optimal value.

Theorem 2. *For a finite-horizon fPOSG, we have that the solution of the IALM for the incoming influence point $I_{\to i}(\pi_{-i})$ associated with any π_{-i} achieves the same value as the best response computed against π_{-i} directly:*

$$\forall_{\pi_{-i}} \quad V_i^*(I_{\to i}(\pi_{-i})) = V_i^*(\pi_{-i}). \tag{1}$$

Sketch of Proof. We must show that the value based on the global belief $b_i^g(s^t, \vec{\theta}_{-i}^t)$ and local belief $b_i^l(\boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\boldsymbol{D}}_i^{t-1})$ are equal. The proof is by induction over the horizon, with the base case presented by the last stage.

For the induction step, we need to show that

$$Q_i^t(b_i^g, a_i) = R_i(b_i^g, a_i) + \sum_{o_i'} \Pr(o_i' | b_i^g, a_i) V_i^{t+1}(b_i^g) = R_i(b_i^l, a_i) + \sum_{o_i'} \Pr(o_i' | b_i^l, a_i) V_i^{t+1}(b_i^l) = Q_i^t(b_i^l, a_i)$$

This is proven if we show that the expected rewards and observation probabilities are equal (next-stage values V_i^{t+1} are equal per induction hypothesis). This also proves the base case (which requires only showing equality of expected rewards). An important step in proving these things is to show that b_i^l is sufficient to predict the next local state, i.e., $\Pr(\boldsymbol{x}_i^{t+1}, \boldsymbol{m}_i^{t+1} | b_i^g, a_i) = \Pr(\boldsymbol{x}_i^{t+1}, \boldsymbol{m}_i^{t+1} | b_i^l, a_i)$. This is achieved by decomposing the global belief in a local and external part via

$$b_i^\pi(s^t, \vec{\theta}_{-i}^t) = \sum_{\vec{\boldsymbol{D}}_i^{t-1}} b_i(\boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\boldsymbol{D}}_i^{t-1}) b_i(\boldsymbol{x}_{-i}^t, \vec{\theta}_{-i}^t | \boldsymbol{x}_i^t, \boldsymbol{m}_i^t, \vec{\boldsymbol{D}}_i^{t-1}),$$

and subsequently marginalizing out non-source $\boldsymbol{x}_{-i}^t, \vec{\theta}_{-i}^t$ variables. The remaining sources can be predicted from $\vec{\boldsymbol{D}}_i^{t-1}$, since it was chosen precisely to d-separate local state and the sources. □

4.3 Computing the Influence

The actual computation of $I_{\to i}^{t+1}(\boldsymbol{\pi}_{-i})$ is an inference problem (Koller and Friedman 2009). In general this may be a hard problem, but due to the d-separation it can be performed 'non-locally', i.e., it does not need to take into account agent i's private factors or actions. Also, in many cases it is possible to factor $I(\boldsymbol{u}_{\to i}^{t+1} | \vec{\boldsymbol{D}}_i^t)$ as the product of a number of non-correlated influences. For instance, when the set of MMFs is the same for all the agents and every $\vec{\boldsymbol{D}}_i^t$ contains this entire set, due to d-separation the distribution factors as a product of the distributions over sources corresponding to different agents.

Also, it is possible to use approximate inference for the influence computation. This leads to an intuitively sensible approximation: things that are difficult to observe and/or very complex such as private factors of other agents or their internal state are reasoned over only approximately, while variables that are more directly influential are treated exactly. For instance, a traffic light must predict the local rewards based on an accurate estimate of the traffic density of neighboring road segments, but in order to do that approximate inference about distant road segments may suffice.

In self-interested settings agents may not necessarily share their policies, which means that an agent should estimate the policy of other agents in order to compute a best response. In this case, instead of learning the policies of other agents and using those to compute the influences, it may be more useful to directly estimate the influence.

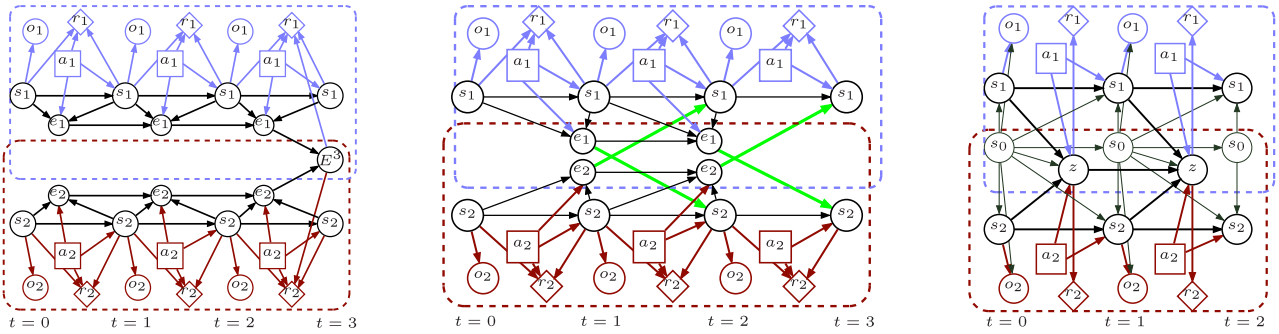

Figure 2: Local-form representations of TI-Dec-MDP (left), ED-Dec-MDP (middle) and ND-POMDP (right).

5 Influence Specifications for Sub-Classes

There are a number of important problem classes and associated models developed in previous work that emphasize weakly coupled problem structure in more restrictive settings. We now frame these classes in the context of IBA, thus taking a step towards demonstrating the practical usefulness of our theoretical formulation. All of the models that we review below are specialized instances of the factored Dec-POMDP (fDec-POMDP) model and since an fDec-POMDP is an fPOSG, our IBA is applicable to all of these models. However, as we illuminate below, some subclasses allow for particularly compact influence specifications.

TD-POMDP. Introduced by Witwicki and Durfee (2010), the Transition-Decoupled POMDP (TD-POMDP) is an fDec-POMDP with a particular form geared towards decoupling weakly-coupled agents. Like our local-form fPOSG, the TD-POMDP describes a local state for each agent and distinguishes mutually-modeled factors common to more than one agents' local state. However, there are two important differences that make the TD-POMDP more restrictive than our local-form model. First, the TD-POMDP does not allow intra-stage dependencies between private state variables and MMFs. Second, in a TD-POMDP each state factor can only be directly affected by (have an incoming edge from) the action (or private state variable) of just one agent. These constraints effectively limit the representational power of the TD-POMDP to *nonconcurrent* interactions.

Observation 1. *The TD-POMDP model is a special case of the LFM, imposing restrictions that limit its modeling capabilities to a subset of those interactions representable as local-form POSGs: TD-POMDP $\subset$ LFM.*

Witwicki and Durfee also propose an explicit form of influence. Their definition however, corresponds to our notion of 'induced CPT' and fixes the choice of the conditioning sets to the history of mutually-modeled factors. The authors also show that since there may be many more policies than possible influence points, searching in the space of joint influences can provide significant speed-ups over searching the space of joint policies. There is nothing apparent that precludes extending influence-space search techniques (Witwicki 2011; Witwicki, Oliehoek, and Kaelbling 2012) beyond TD-POMDPs, by using our more general definition of influence. That is, such algorithmic extensions might bring the computational leverage demonstrated for TD-POMDPs to a more general class of problems.

TI-Dec-MDP. Another model, the Transition-Independent Dec-MDP (Becker et al. 2003), imposes other more stringent restrictions on the dependencies between agents' local models. In particular, an agent fully observes its private factors and there are no paths of dependence in the DBN connecting one agent's private factors, actions, and observation to those of another. This implies that the agents are *transition and observation independent*. Agents' local models are instead coupled through their rewards, which can depend on the *events* $e_i = \langle s_i^t, a_i^t, s_i^{t+1} \rangle$ that occur (at most once) within another agents' state space. As shown in Figure 2(left), the combined occurrence of both agents' events (as represented by Boolean variable E) leads to a change in the reward (split between the agents as soon as the event occurs). When the discount factor is 1 (as the authors assume), the reward may as well be effected at the last time step as we have indicated. This leads to a very simple form of influence $I_{\rightarrow i}$: the probability of e_j^{h-1} being true. This corresponds exactly to the characterization of 'parameter space' presented by Becker et al. in their development of the *coverage set algorithm (CSA)*.

Our characterization of the TI-Dec-MDP immediately leads to some new insights.

Observation 2. *The properties that 1) events cannot occur more than once; and 2) events are unobserved, allow for history-independent influence encodings in TI-Dec-MDPs.*

Observation 3. *CSA and closely-related TI-Dec-MDP algorithms (Petrik and Zilberstein 2009) exploit structure that is also present in more general contexts, such as TI-Dec-POMDPs with partial observability of private factors.*

We expect that CSA and its successors can actually be extended to any problem whose joint value function is piecewise-linear and convex in the influence parameters.

Event-Driven Interactions. In a closely-related class, the Dec-MDP with Event-Driven Interaction (EDI-Dec-MDP), Becker, Zilberstein, and Lesser (2004) developed an explicit representation for structured transition dependencies. As shown in Figure 2(middle), agent 1's transition probabilities may be affected by the prior occurrence of agent 2's

event e_2 (and vice versa). In this case, the event features are mutually-observable, implying that their histories are necessary for d-separation, i.e., $\boldsymbol{D}_1 = \{e_1, e_2\}$. This leads us to develop a special influence specification.

Observation 4. *The influence on EDI-Dec-MDP agent i, $I^{t+1}_{\rightarrow i}(\pi_j)$ can be defined as $I(s^t_j, a^t_j, s^{t+1}_j | \vec{e}^t_i, \vec{e}^t_j)$. Moreover, the history $\vec{e}^t_i, \vec{e}^t_j$ can be represented compactly since events can only switch to true.*

The product of induced CPTs $Pr(e^{t+1}_1, e^{t+1}_2 | \vec{e}^t_1, \vec{e}^t_2)$ is similar to the parameters used by Becker *et al*. (in their application of CSA), but is slightly more compact, since it does not depend on private factors s^t_i. Having derived a more compact parameter form, we anticipate that this will translate directly into a more efficient application of CSA. We note that our reformulation of the TI-Dec-MDP and EDI-Dec-MDP also serve as influence specifications for the EDI-CR model (Mostafa and Lesser 2009), a model developed to include both event-driven interactions and reward dependencies (as in the TI-Dec-MDP).

ND-POMDP. The Network Distributed POMDP (ND-POMDP) (Nair et al. 2005) is another transition and observation independent model whose structure can easily be represented in our framework. As we depict in Figure 2 (right), in addition to an unaffectable, mutually-modeled factor s_0, there are also reward dependencies involving joint actions, as captured with an unobservable variable z encoding the local state-action pair. In general, a ND-POMDP can consist of multiple local neighborhoods (which would translate to multiple variables z) that can connect different subsets of agents. Our reformulation presented here immediately leads to the first specification of influence that we are aware of for this problem class:

Observation 5. *The influence on ND-POMDP agent i, $I^{t+1}_{\rightarrow i}(\pi_{-i})$ can be defined as $I(s^t_{N(i)}, a^t_{N(i)} | \vec{s}^t_0)$, where $N(i)$ denotes the neighbors of agent i.*

Like the other mentioned subclasses, the ND-POMDP affords a very compact influence encoding, suggesting that influence-based planning methods could gain traction if applied here.

6 Other Related Work

Apart from the models and approaches reviewed in Section 5, there are important connections to be drawn with a large body of other work. Past studies of fDec-POMDPs with factored value functions (Nair et al. 2005; Kumar, Zilberstein, and Toussaint 2011; Witwicki and Durfee 2011; Varakantham et al. 2007) have shown that gains in computation efficiency are possible when the value function can be expressed as the sum of a number of local components, each of which is specified over subsets of agents and state factors (a property commonly referred to as 'locality of interaction'). However, for general fDec-POMDPs, these components involve all agents and factors. I.e., they are *not* local (Oliehoek et al. 2008; Oliehoek 2010). This paper shows that even in the most general case, provided there is enough conditional independence, we *can* find local (i.e., restricted scope) components, although this may be at the cost of introducing a dependence on the history. This means that it may be possible to extend the planning-as-inference method of Kumar et al. to exploit structure in general fDec-POMDPs. In general, IBA draws close connections to the paradigm of planning as inference (Toussaint 2009); it performs inference to compute a compact local model; subsequently, inference (among other choices of solution methods) could be used to solve the IALM.

A number of other approaches decompose multiagent decision-making problems by leveraging structured interactions. For instance, our approach resembles the distributed approximate planning method by Guestrin and Gordon (2002) in that both methods decompose an agent's decision model into internal and external parts. Our proposed abstraction, in addition to being sufficient for *optimal* decision-making, is more general in that it can deal with partial observability. Other models such as the IDMG (Spaan and Melo 2008) and the DPCL (Varakantham et al. 2009) have allowed for approximate decoupled local planning by leveraging a form of context-specific independence, where agents only influence each other in certain states. An important direction of research is to also exploit this type of independence in LFMs. I-POMDPs (Gmytrasiewicz and Doshi 2005) and I-DIDs (Doshi, Zeng, and Chen 2008) inherently provide a decomposition, modeling other agents recursively. Although quite different from the fPOSG, our definition of influence (and thus IBA) is readily applicable to factored-state versions of these models. Influence-based abstraction is conceptually similar to existing I-DID solution techniques that exploit *behavioral equivalence* (Pynadath and Marsella 2007; Rathnasabapathy, Doshi, and Gmytrasiewicz 2006; Zeng et al. 2011), but these approaches abstract classes of behaviors down to policies, whereas we abstract policies down to even more abstract influences.

7 Conclusions & Future Work

This paper introduced influence-based abstraction (IBA) for factored POSGs, a technique that allows us to decouple the model into a set of local models defined over subsets of state factors. Performing IBA for an agent consists of two steps: computing the incoming influence point using inference techniques and creating the agent's induced local model. As we described in Section 5, IBA for fPOSGs generalizes the abstractions made in a number of more specific problem contexts.

IBA is important for the following reasons: (1) It allows us to decompose the value function for *any* factored Dec-POMDP into the sum of a number of *local* value functions (that in general may depend on the local history), a property that is exploited in several solution methods (Nair et al. 2005; Oliehoek 2010; Kumar, Zilberstein, and Toussaint 2011), but that had been proven only for specific sub-classes of Dec-POMDPs. (2) It provides a better understanding of these sub-classes and how they relate to each other. The insightful connections that we have drawn promote extensions of specialized methods beyond their respective subclasses as well as comparisons with one another in more general contexts. (3) our general definition of influence paves the

way for influence-space search in general Dec-POMDPs. (4) IBA can enable more efficient best-response computation in many fPOSGs, as well as providing a very natural form of approximation via approximate inference. (5) Influences can provide a more compact, yet sufficient statistic for the behavior of other agents in a MAS. This is important in multiagent reinforcement learning, since it is often easier to learn a compact statistic from the same amount of data.

In future work we hope to develop a general form of influence space search for factored Dec-POMDPs. We also plan to investigate how the two design choices in IBA—the choice of the local state definition and the choice of influence encoding (i.e., the choice of d-separating sets)—interact. A number of other directions of future work seem promising, such as the use of approximate inference for computing influence points, the clustering of d-set-histories (thereby imposing a clustering of the influences), and the clustering of influence points themselves.

Acknowledgments

This work was supported by AFOSR MURI project #FA9550-09-1-0538, by NWO CATCH project #640.005.003, and by the Fundação para a Ciência e a Tecnologia (FCT) and the CMU-Portugal Program under project CMU-PT/SIA/0023/2009.

References

Becker, R.; Zilberstein, S.; Lesser, V.; and Goldman, C. V. 2003. Transition-independent decentralized Markov decision processes. In *AAMAS*, 41–48.

Becker, R.; Zilberstein, S.; and Lesser, V. 2004. Decentralized Markov decision processes with event-driven interactions. In *AAMAS*, 302–309.

Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research* 27(4):819–840.

Boutilier, C.; Dean, T.; and Hanks, S. 1999. Decision-theoretic planning: Structural assumptions and computational leverage. *JAIR* 11:1–94.

Doshi, P.; Zeng, Y.; and Chen, Q. 2008. Graphical models for interactive POMDPs: representations and solutions. *Journal of Autonomous Agents and Multi-Agent Systems* 18(3):376–416.

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

Guestrin, C., and Gordon, G. 2002. Distributed planning in hierarchical factored MDPs. In *UAI*, 197–206.

Guestrin, C.; Koller, D.; and Parr, R. 2002. Multiagent planning with factored MDPs. In *Advances in Neural Information Processing Systems 14*, 1523–1530.

Hansen, E. A.; Bernstein, D. S.; and Zilberstein, S. 2004. Dynamic programming for partially observable stochastic games. In *AAAI*, 709–715.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *Artificial Intelligence* 101(1-2):99–134.

Koller, D., and Friedman, N. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Kumar, A.; Zilberstein, S.; and Toussaint, M. 2011. Scalable multiagent planning using probabilistic inference. In *IJCAI*, 2140–2146.

Melo, F. S., and Veloso, M. 2011. Decentralized MDPs with sparse interactions. *Artificial Intelligence* 175(11):1757–1789.

Monderer, D., and Shapley, L. S. 1996. Potential games. *Games and Economic Behavior* 14(1):124–143.

Mostafa, H., and Lesser, V. 2009. Offline planning for communication by exploiting structured interactions in decentralized MDPs. In *Proc. of Int. Conf. on Web Intelligence and Intelligent Agent Technology*, 193–200.

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

Nair, R.; Varakantham, P.; Tambe, M.; and Yokoo, M. 2005. Networked distributed POMDPs: A synthesis of distributed constraint optimization and POMDPs. In *AAAI*, 133–139.

Oliehoek, F. A.; Spaan, M. T. J.; Whiteson, S.; and Vlassis, N. 2008. Exploiting locality of interaction in factored Dec-POMDPs. In *AAMAS*, 517–524.

Oliehoek, F. A. 2010. *Value-Based Planning for Teams of Agents in Stochastic Partially Observable Environments*. Ph.D. Dissertation, University of Amsterdam.

Petrik, M., and Zilberstein, S. 2009. A bilinear programming approach for multiagent planning. *JAIR* 35(1):235–274.

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

Rathnasabapathy, B.; Doshi, P.; and Gmytrasiewicz, P. 2006. Exact solutions of interactive POMDPs using behavioral equivalence. In *AAMAS*, 1025–1032.

Rosenthal, R. W. 1973. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory* 2(1):65–67.

Spaan, M. T. J., and Melo, F. S. 2008. Interaction-driven Markov games for decentralized multiagent planning under uncertainty. In *AAMAS*, 525–532.

Toussaint, M. 2009. Probabilistic inference as a model of planned behavior. *Kunstliche Intelligenz* 3(9):23–29.

Varakantham, P.; Marecki, J.; Yabu, Y.; Tambe, M.; and Yokoo, M. 2007. Letting loose a SPIDER on a network of POMDPs: Generating quality guaranteed policies. In *AAMAS*.

Varakantham, P.; young Kwak, J.; Taylor, M.; Marecki, J.; Scerri, P.; and Tambe, M. 2009. Exploiting coordination locales in distributed POMDPs via social model shaping. In *ICAPS*, 313–320.

Wiering, M. 2000. Multi-agent reinforcement learning for traffic light control. In *Proc. of the International Conference on Machine Learning*, 1151–1158.

Witwicki, S., and Durfee, E. 2010. Influence-based policy abstraction for weakly-coupled Dec-POMDPs. In *ICAPS*, 185–192.

Witwicki, S., and Durfee, E. 2011. Towards a unifying characterization for quantifying weak coupling in Dec-POMDPs. In *AAMAS*, 29–36.

Witwicki, S.; Oliehoek, F. A.; and Kaelbling, L. P. 2012. Heuristic search of multiagent influence space. In *AAMAS*. (To appear).

Witwicki, S. J. 2011. *Abstracting Influences for Efficient Multiagent Coordination Under Uncertainty*. Ph.D. Dissertation, University of Michigan.

Zeng, Y.; Doshi, P.; Pan, Y.; Mao, H.; Chandrasekaran, M.; and Luo, J. 2011. Utilizing partial policies for identifying equivalence of behavioral models. In *AAAI*, 1083–1088.