Robust Model Equivalence using Stochastic Bisimulation for N-Agent Interactive DIDs

Muthukumaran Chandrasekaran
University of Georgia
Athens, GA, USA 30602
mkran@uga.edu

Junhuan Zhang
Beihang University
Beijing, China 100191
junhuan.zhang@gmail.com

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

Yifeng Zeng
Teesside University
Middlesbrough, UK TS1 3BX
y.zeng@tees.ac.uk

Abstract

I-DIDs suffer disproportionately from the curse of dimensionality dominated by the exponential growth in the number of models over time. Previous methods for scaling I-DIDs identify notions of equivalence between models, such as behavioral equivalence (BE). But, this requires that the models be solved first. Also, model space compression across agents has not been previously investigated. We present a way to compress the space of models across agents, possibly with different frames, and do so without having to solve them first, using stochastic bisimulation. We test our approach on two non-cooperative partially observable domains with up to 20 agents.

1 INTRODUCTION

Autonomous agents must be capable of perceiving the environment, interact with other agents, and make rational decisions to achieve their goals under uncertainty. Interactive partially observable markov decision process (I-POMDP) [19] is a recognized framework that models the decision-making process of a self-interested agent in a partially observable multiagent setting. I-POMDPs cover an important portion of the multiagent planning problem space [10, 15]. Applications in diverse areas such as security [23, 21], robotics [26, 25], ad hoc teams [6, 7] and human behavior modeling [12, 27] testify to its wide appeal while motivating better scalability.

Interactive dynamic influence diagrams [14] provide a graphical and naturally factored representation for I-POMDPs. They compactly represent the problem of how an agent should act in an uncertain environment shared with others with unknown behaviors. I-DIDs typically handle the uncertainty over the other agents' behaviors by maintaining a belief over a large but finite set of models, and updating it over time [22]. However, I-DIDs suffer disproportionately from the curse of dimensionality [14]. The curse of dimensionality is dominated by the exponential growth in the number of models over time. Toward this, previous methods for scaling I-DIDs identify notions of equivalence between models, such as behavioral equivalence (BE) [28]. But, this requires that the models be solved first [28, 8]. All existing approaches group models that differ only in their beliefs while sharing a common frame (i.e., transition, observation, and reward functions). Hence, they have been evaluated on domains involving one other agent only.

Is there a way to compress the space of models across agents, possibly with different frames, and do so without having to solve them first? To answer this question, we draw upon the well-known concept of *stochastic bisimulation* [18, 4], which allows us to establish equivalence relationships (bisimilarity) between models under conditions of uncertainty. An exact bisimilar relation between two models (say, DIDs) implies that, for all the actions, the expected immediate rewards are equal and transitions occur to models that are themselves bisimilar. The base case requires we transition to beliefs that are the same.

However, this notion of exact bisimilarity is too stringent to use in practice because it requires that the frames of agents agree exactly. Obviously, this is not robust because even a small change in the rewards or the transition probabilities cause these models to appear dissimilar although their solutions may not be different. Therefore, we are motivated to measure the *degree* to which two models with differing frames may be bisimilar. Ferns *et.* al [16, 17] define a distance metric, called the *bisimulation metric*, that varies relative to the quantitative difference between two MDPs. We leverage the theoretical guarantees of this metric and generalize it to partially observable settings. Hence, for the first time in the context of I-DIDs, we can operate on models across one or more frames whose "similarity" can be measured by using our generalized metric. We are excited about the prospects of this metric: computational savings achievable by pruning *similar* models across agents, and the ability to do so without having to solve the models first.

Specifically, the contributions of this paper are three-fold: (i) We leverage the existing equivalence notion of stochastic bisimulation and extend Ferns et. al's bisimulation metrics for MDPs [16, 17] to partially observable settings represented by models such DIDs. We formally define and use this metric to quantitatively measure the *similarity* between any two models in the space of models that the subject agent ascribes to the other agents in its I-DID. (ii) Using a tolerance parameter ϵ, we present a way to partition the model space into ϵ-bisimiar regions using *barycentric subdivision* [20]. We also present a way to mitigate the combinatorial explosion due to barycentric subdividing by merging all those adjacent regions which, when merged, continue to satisfy the ϵ-bisimiarity constraints. (iii) Finally, we generalize I-DIDs to N-agents and compare the performance of our model space compression technique against a baseline I-DID solver that uses the current state-of-the-art BE-based technique – discriminative model updates (DMU) [28] – on two non-cooperative multiagent domains exhibiting partial observability with up to 20 agents.

2 RELATED WORK

Previous notions of equivalence like stochastic bisimulation and trajectory equivalence between states have been used in the context of model abstraction to provide a principled way to reduce a model into something more compact [18, 4]. The reduced model can then be solved using traditional solvers utilizing much lesser computational power than what would have been needed otherwise. However, exact equivalence is too stringent to use in practice. In their first piece of related work, Ferns *et.* al devised metrics which quantitatively measured the degree of similarity between states in an MDP [16]. More recently, they went on to extend their metrics to MDPs with continuous states [17]. In the context of I-DIDs, previous efforts focus predominantly on addressing their *curse of dimensionality* which is *in part* due to exponential growth in the number of models over time[1] [14, 13, 11, 5, 28, 8], and the *curse of history* due to exponential increase in the size of the model solutions – policy trees and actions – with the planning horizon [30, 29]. All these approaches identify some notion of *equivalence* between models – including behavioral equivalence (BE), action equivalence (AE), and value equivalence (VE) – and require solving of *all* the models and comparing their solutions. A few approaches exploit the spatial closeness of beliefs in order to identify equivalence between models [14] while others operate directly on candidate model solutions instead of the model specifications [28]. A general limitation of the former – utilizing the spatial proximity of beliefs – is that it is less likely that two such models will result in the same behavior if their frames (say, the transition or reward functions) were different. The latter, however, continue to apply, albeit with increased computational complexity, even if there was some uncertainty in the frames.

3 BACKGROUND

We briefly review I-DIDs next followed by the concept of stochastic bisimulation for MDPs.

3.1 INTERACTIVE DIDs (I-DIDs)

Representation We illustrate a generic two time-slice level $l > 0$ I-DID with $N = 2$ agents in Fig. 1. I-DIDs have a *model* node (denoted by the hexagonal node) in addition to the nodes already present in traditional DIDs. DIDs utilize chance nodes to model the uncertainty in the subject agent i's decision making problem through random variables such as those for modeling the physical state, S, and the agent's observations, O_i. They additionally use decision nodes and utility nodes to model the agent's actions, A_i, and reward function, R_i, respectively. In addition to the model node, I-DIDs also have a chance node, A_j, to represent the distribution over actions of the other agent j. The model node $M_{j,l-1}$ in the I-DID houses a candidate set of computable *intentional models* (and possibly *subintentional models*) ascribed by i to agent j. Subscript $l-1$ denotes the *strategy level* indicating the cognitive capability of the other agent j. A model in the model node may be a level $l-1$ I-DID or a DID. The recursion ends at level 0, when the models are DIDs. We note that the other agents' level is one less than that of i which follows from established previous hierarchical formulations in game theory [1, 3] and decision theory [19]. In order to operationalize this formulation of I-DIDs, the state space is augmented with the models of the other agents, referred to as the *interactive state space*, IS_i (shown in Fig. 1). A link from the chance node, S, to the model node, $M_{j,l-1}$, represents agent i's beliefs over j's models. Specifically, it is a probability distribution in the conditional probability table (CPT) of the chance node, $Mod[M_j]$ (in Fig. 2). An individual model of an agent j, denoted $m_{j,l-1}$, is a 2-tuple

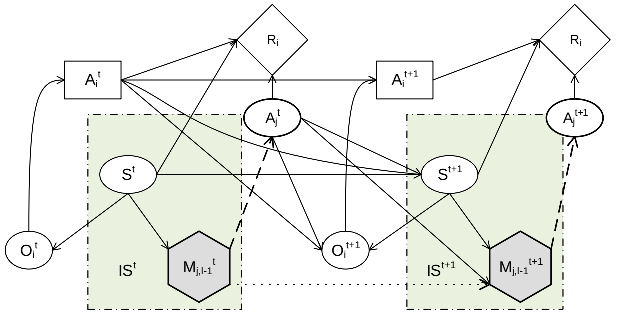

Figure 1: A generic two time-slice level l I-DID for agent i situated with one other agent j.

[1] The curse of dimensionality in I-DIDs can also be due to the exponential growth in the model space with the number of agents.

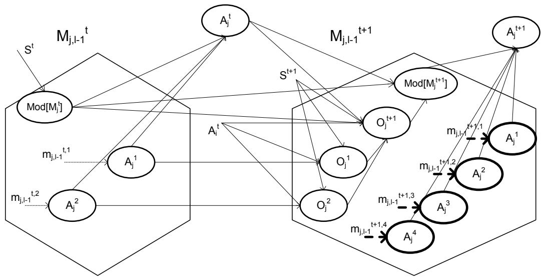

Figure 2: Implementation of the model update link using standard dependency links and chance nodes.

$< b_{j,l-1}, \hat{\theta}_j >$ where $b_{j,l-1}$ is the level $l-1$ belief, which is the probability distribution over j's interactive state space, and $\hat{\theta}_j$ is agent j's *frame* that encompasses the decision, observation, transition, and utility nodes. Solutions to the model are the predicted behavior of j and are encoded into the chance node, A_j, through a dashed link, called a *policy link*. Connecting A_j with other nodes in the I-DID structures how agent j's actions influence i's decision-making process. As agent j acts and receives observations over time, its models should be updated. A dotted link, called the *model update link*, from $M_{j,l-1}^{t}$ to $M_{j,l-1}^{t+1}$ in Fig. 1 denotes the update of the model node over time. For example, two models, $m_{j,l-1}^{t,1}$ and $m_{j,l-1}^{t,2}$, are updated into four models at time $t+1$ (shown in Fig. 2). Models at $t+1$ reflect the updated belief of j, and their solutions provide the probability distributions for the corresponding action node. We may implement the model node, the policy link, and the model update link using chance nodes and standard dependency links, as shown in Fig. 2, and transform an I-DID into a traditional DID.

Solution We outline a generic procedure for solving I-DIDs below and refer the readers to [13, 28] for more information. The solution for a level l I-DID expanded over T time steps proceeds in a bottom-up manner. In order to solve the subject agent i's I-DID at level l, all models of the other agent j at level $l-1$ must first be solved. The solution to a level $l-1$ model $m_{j,l-1}$ is j's policy that prescribes an optimal decision in A_j initially given its belief $b_{j,l-1}$, and the actions thereafter conditional on its observations in O_j up to time T. We perform this process for each level $l-1$ model of j and obtain the fully expanded level l model.

3.2 EQUIVALENCE USING STOCHASTIC BISIMULATION

Stochastic bisimulation can be used to define equivalence relations between states in stochastic processes. Givan *et.* al extended the notion of stochastic bisimulation to probabilistic transition systems with rewards in the context of MDPs [18] providing a principled way for model abstraction in MDPs with pleasing theoretically guarantees.

The 4-tuple $\langle S, A, T, R\rangle$ defines an MDP with a finite set of physical states S and actions A, transition function $T : S \times A \rightarrow \Delta(S)$, and reward function $R : S \times A \rightarrow \mathbb{R}$. A bisimulation relation between states in an MDP is defined as follows:

Definition 1 (**Stochastic bisimulation relation**). *An equivalence relation $E \subseteq S \times S$ between two states $s, s' \in S$ is a* ***stochastic bisimulation relation*** *if whenever sEs', then the following holds $\forall a \in A$: (i) $R(s,a) = R(s',a)$, and (ii) $\forall C \in S/E,\ T(s,a)(C) = T(s',a)(C)$.*

where, S/E is the state partition induced by E and $T(s,a)(C) = \sum_{s'' \in C} T(s,a,s'')$. Stochastic bisimulation, $\approx$, is the largest stochastic bisimulation relation.

Givan et. al show an iterative procedure to compute the stochastic bisimulation (henceforth simply bisimulation) partition. Castro *et.* al extend bisimulation to partially observable settings [4] in the context of a *belief MDP*. A *belief MDP* equivalently converts a POMDP into a MDP with a continuous state space comprising of the entire belief simplex. We note that DIDs are just graphical representations of belief MDPs.

A belief MDP is a 4-tuple $M = \langle B, A, \mathcal{T}, \rho\rangle$, where B denotes the belief simplex; A is the set of actions; $\mathcal{T} : B \times A \rightarrow \Delta(B)$ is the belief-transition function; $\rho : B \times A \rightarrow \mathbb{R}$ is the reward function. $\rho(b,a)$ and $\mathcal{T}(b,a)(b')$ are defined below:

$$\rho(b,a) = \sum_{s \in S} R(s,a) b(s) \tag{1}$$

$$\mathcal{T}(b,a)(b') = \sum_{\omega \in \Omega} Pr(b'|b,a,\omega) Pr(\omega|a,b) \tag{2}$$

$$Pr(b'|a,b,\omega) = \begin{cases} 1 & \text{if } b' = \tau(b,a,\omega). \\ 0 & \text{otherwise.} \end{cases}$$

$$\tau(b,a,\omega) \triangleq b'(s') = \frac{O(s',a,\omega)\sum_{s \in S} T(s,a,s')b(s)}{Pr(\omega|a,b)}$$

$$Pr(\omega|a,b) = \sum_{s' \in S} O(s',a,\omega) \sum_{s \in S} T(s,a,s')b(s)$$

where, $O(s',a,\omega)$ denotes the probability of observing ω given the state s' and action a. Castro et. al defines *stochastic bisimulation* between 2 beliefs as follows:

Definition 2 (**Belief bisimulation relation**). *A relation $E \subseteq B \times B$ is a* ***belief bisimulation relation*** *if whenever bEc, then the following holds: (i) $\forall a \in A$ $\rho(b,a) = \rho(c,a)$, (ii) $\forall a \in A$, $\forall \omega \in \Omega$ $O(b,a,\omega) = O(c,a,\omega)$, (iii) $\forall a \in A$, $\forall \omega \in \Omega$ $\tau(b,a,\omega)$ and $\tau(c,a,\omega)$ are belief bisimilar. (Def. 3).*

Definition 3 (Belief bisimilarity). *Two belief states b, c are **bisimilar**, denoted $b \approx c$, if there exists a belief bisimulation relation E such that bEc.*

We note that *belief bisimulation* has a recursive definition. In other words, in order for two belief states to be *bisimilar*, their updated beliefs need to also be *bisimilar*. This in turn implies that their corresponding belief transition functions must also be equal. Therefore, if $\tau(b, a, \omega) E \tau(c, a, \omega)$ (*Condition* (iii) *in Def.* 2), it follows that for any arbitrary belief $g \in B/E$ and action $a \in A, Pr(g|b, a) = Pr(g|c, a)$, and vice versa; where $Pr(g|b, a) = \sum_{b' \in g} \mathcal{T}(b, a)(b')$ and B/E denotes the partition of B into E-equivalence classes. Stochastic bisimulation is the largest belief bisimulation relation.

Unfortunately, the equivalence notion of stochastic bisimulation is too stringent because it requires that the rewards and transition probabilities agree exactly. This is not robust because even small perturbations in rewards or transition probabilities will cause states to appear dissimilar. This motivates the use of a distance metric to evaluate the degree to which two models may be bisimilar. Previously, Ferns *et* al. introduced metrics for computing the *degree* of bisimilarity between two states in an MDPs – and hence the MDPs themselves – with theoretical bounds on the solution quality due to the induced approximations [16, 17]. In general, they used a semimetric as a distance function that quantifies how far apart two states are in the MDP.

Definition 4 (Semimetric). *A semimetric on S is a map $d : S \times S \to [0, \infty)$ s.t. for every triple s, s', and $s'' \in S$, (i) $s = s' \Rightarrow d(s, s') = 0$, (ii) $d(s, s') = d(s', s)$, and (iii) $d(s, s'') \leq d(s, s') + d(s', s'')$*

If the converse of Condition (i) was true, then d would be a proper *metric*. This allows the possibility of the distance between s and s' to be 0 even if s and s' are distinct. Let D be the set of all semimetrics on S that assigns a distance of at most 1. Note that every semimetric d induces an equivalence relation, E, on S, obtained by equating the states assigned a distance of zero by d. For convenience, we will refer to semimetrics as just metrics hereafter.

Definition 5 (Bisimulation metric). *We say that $d \in D$ is a bisimulation relation metric if it measures the bisimulation relation, E, as defined in Def. 1. The bisimulation relation metric d is a bisimulation metric if E is a stochastic bisimulation, $\approx$.*

Ferns et. al construct the bisimulation metrics as a linear combination of a metric on the rewards and a metric on the transition probability distributions.

$$d(s, s') = \max_{a \in A} c_R\, (R(s, a) - R(s', a)) + c_T\, d_p(\mathcal{T}(s, a), \mathcal{T}'(s', a)) \tag{3}$$

where d_p is some probability metric, and c_R and c_T are constants between 0 and 1. The constants represent the respective weights on the absolute difference between reward values and the distance between transition probabilities. The latter is measured using the Kantorovich metric [24], which we detail next. We set $c_T = \gamma$ and $c_R = 1 - \gamma$ where γ is the discount factor of the MDPs.

The Kantorovich metric has been used extensively in recent years as a measure of similarity between 2 probability distributions because it can be elegantly formulated as a linear program computable in polynomial time with several appealing theoretical properties applicable within our context probabilistic concurrency (like in stochastic processes)[2]. In general, behavioral equivalences for probabilistic processes such as MDPs involve a *lifting* operation that converts a relation on states into a relation on distributions of states. This nicely corresponds to the way the Kantorovich metric works – in 2 levels: it considers both (i) the distances between the underlying states, and (ii) the distances between the probability distributions over those states. We detail the linear program used to compute the Kantorovich metric, $T_K(d)$, below:

Definition 6 (Kantorovich metric). *Let C be a block in the partition of states S induced by the equivalence relation, E. Given $d \in \mathcal{D}$, the Kantorovich metric, denoted $T_K(d)$, applied to finite probability distributions P and Q each over S is defined by the following linear program:*

$$T_K(d)(P, Q) = \max_{v_C} \sum_{C \in S/E} (P(C) - Q(C))\, v_C$$

$$\textit{subject to}:\ v_C - v_D \leq \min_{i \in C, j \in D} d(s_i, s_j)\ \ \forall C, D$$

$$0 \leq v_C \leq 1\ \ \forall C$$

and $T_K(d)(P, Q) = 0 \Leftrightarrow P(C) = Q(C), \forall C \in S/E.$

where d is the underlying cost function between two states and $P(C) = \sum_{s \in C} P(s)$. As $d \in D$ is a metric, the solution to the above LP (i.e. the Kantorovich metric distance) is also a metric. Therefore, the bisimulation metric can be expressed in terms of the Kantorovich metric.

The following lemmas are a direct consequence of Def. 5 and Def. 6:

Lemma 1. *Let $M = \langle S, A, T, R \rangle$ and $M' = \langle S, A, T', R' \rangle$ be two MDPs sharing the same set of actions and a common state space S. If $d \in \mathcal{D}$ is a bisimulation metric, then $\forall s, s' \in S$ $d(s, s') = 0$ iff $\forall a \in A$:*

$$R(s, a) - R(s', a) = 0, \quad T_K(d)(\mathcal{T}(s, a), \mathcal{T}'(s', a)) = 0$$

Lemma 2. *If $d \in \mathcal{D}$ satisfies Lemma 1, then*

$$d(s, s') = 0 \Rightarrow s \approx s'\ (i.e.\ s, s'\ are\ bismilar)$$

[2] Commonly used KL divergence is not a proper metric, unlike the Kantorovich metric. The latter is also known by several other names including Monge-Kantorovich, Kantorovich-Rubinstein, Wasserstein, and Earth Movers Distance.

Given Lemmas 1 and 2, the next theorem follows in a straightforward manner.

Theorem 1. *If $d \in \mathcal{D}$ is a bisimulation metric defined on S (i.e. satisfies Lemma 1), then for all s, s' ($\in S$) coming from M, M' respectively,*

$$d(s, s') = 0 \Rightarrow M \approx M' \text{ } (i.e. \text{ } M, M' \text{ } are \text{ } bismilar)$$

The Kantorovich metric leverages a few theoretical results from *fixed-point* theory [16, 17]. It preserves the point-wise partial ordering that the set of probability metrics $\mathcal{D}$ (on S) is equipped with: $\forall d, d' \in \mathcal{D}$ $d \leq d'$ iff $d(s, s') \leq d'(s, s')$. As a result, $T_K : \mathcal{D} \rightarrow \mathcal{D}$ is shown to be continuous given that the partial ordering is $\omega - complete$. We can then define the *bisimulation metric* based on the Kantorovich probability metric as follows:

Definition 7 (Bisimulation metric using Kantorovich metric). *Let $c_R, c_T \geq 0$ with $c_R + c_T \leq 1$. Define a continuous function $F : \mathcal{D} \rightarrow \mathcal{D}$ as,*

$$\begin{aligned} F(d)(s, s') = \quad & \max_{a \in A} c_R \left(R(s, a) - R(s', a)\right) \\ & + c_T \, T_K(d)(\mathcal{T}(s, a), \mathcal{T}'(s', a)) \end{aligned}$$

Then F has a least fixed-point, d^*, and d^* is a bisimulation metric.

The existence of the fixed-point is proven in [16]. Ferns *et* al. also show that $s \approx s' \iff d^*(s, s') = 0$. Note that d^* can be computed to some degree of accuracy by iterative applications of F for a proportional number of steps. This essentially reduces to computing a Kantorovich metric at each iteration for every action and pair of states.

4 MODEL SPACE COMPRESSION FOR N-AGENT I-DIDS

Stochastic bisimulation presents a principled way to establish equivalence relationships between models with different frames without having to solve them first. We seek to incorporate this idea within the context of I-DIDs allowing for model space compression across agents – whom may have different frames – for the first time. Toward this, we generalize the existing 2-agent I-DIDs to N agents.

4.1 GENERALIZATION TO N-AGENTS

We illustrate an example generic two time-slice level $l > 0$ I-DID for agent i situated with 2 other agents j and k in Fig. 3. Notice that we added a model node and a chance node representing the distribution over an agent's actions linked together using a *policy link*, for each other agent.

The subject agent i's reward, transition, and observation functions are impacted by the other agents' actions. Therefore, we note an exponential explosion in the size of the CPTs of the chance nodes S^{t+1} and O_i^{t+1}, and the reward function R_i with increasing numbers of other agents in the setting. As mentioned earlier, in the expansion step of agent i's I-DID, we must update the belief over interactive states of i, which includes the physical states and other agents' models, over time. For simplicity, we assume that given the belief over the physical states and the corresponding distribution over the actions of the other agent, agent i's belief over the other agents' models is conditionally independent and can be factored. Consequently, we can update the models of each other agent independently of other agents' models. The exponential growth in the number of models in the model node over time is further impacted by the number of other agents in the setting. The *model update links* for agents j and k from their corresponding model nodes at time t to time $t + 1$ – denoted by dotted links – are shown in Fig. 1. Models at $t+1$ reflect the updated beliefs of j and k, and their solutions provide the probability distributions for their corresponding action nodes.

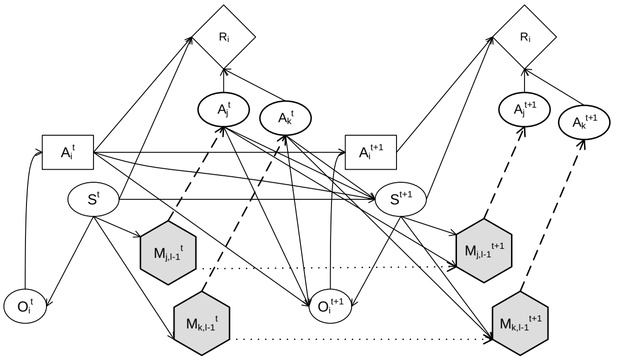

Figure 3: A generic two time-slice level l I-DID for agent i situated with two other agents j and k.

4.2 BISIMULATION METRICS

Incorporating *exact* stochastic bisimulation for model compression in I-DIDs may be impractical because it is sensitive to variations in the numerical values of the parameters of the models as mentioned previously. Therefore, leveraging the notion of stochastic bisimulation for POMDPs from Castro *et* al [4] and the bisimulation metrics for MDPs from Ferns *et* al [16], we generalize the bisimulation metrics to POMDPs. We will operationalize its use as a quantitative measure of similarity between level-0 models (DIDs) ascribed by the subject agent (at level-1) to the others in the model nodes of the I-DID. We will limit the scope of this work to level-1 I-DIDs as we believe this is a good and necessary first step in the generalization to I-DIDs at higher levels. We redefine the bisimulation metrics in the context of belief states in DIDs next:

Definition 8 (Belief bisimulation metric). *We say that $d \in D$ measures the belief bisimulation relation metric if it measures the belief bisimulation relation E as defined in*

Def. 2. We say that d *is a belief bisimulation metric if* E *is a belief bisimulation* $\approx$.

First, we transform the traditional reward function of belief MDPs (Eq. 1) into a binary-valued random variable, $\mathcal{R} : B \times A \rightarrow \Delta(\{0,1\})$ using Cooper's transformation [9]. Specifically, the reward distribution $\mathcal{R}(b,a)$ is:

$$Pr(\mathcal{R}(b,a) = 1|\rho(b,a)) = \frac{\rho(b,a) - \rho_{min}}{\rho_{max} - \rho_{min}} \tag{4}$$

In other words, the probability of selecting and de-selecting the reward $\rho(b,a)$ is given by $Pr(\mathcal{R}(b,a) = 1|\rho(b,a))$ and $1 - Pr(\mathcal{R}(b,a) = 1|\rho(b,a))$ respectively.

We may redefine the bisimulation relation of Def. 2 by replacing the traditional reward function in the first constraint of the definition with the corresponding stochastic reward.

Next, we construct the bisimulation metric for level-0 models – represented as DIDs – as a linear combination of two metrics; one on the stochastic reward and one on the belief transition probabilities. Both metrics can now be defined using the Kantorovich metric and can be computed using a linear program similar to the one in Definition 6.

$$\begin{aligned} d(b,b') = \max_{a \in A}\ & c_R\, T_K(d)(\mathcal{R}(b,a), \mathcal{R}'(b',a)) \\ & + c_T\, T_K(d)(\mathcal{T}(b,a), \mathcal{T}'(b',a)) \end{aligned} \tag{5}$$

where c_R and c_T are as defined previously.

We can also rewrite the lemmas 1 and 2 and theorem 1 in our context next. Let M and M' be two level-0 models sharing the same set of actions and a common belief space B. Let $b, b' \in B$ be the corresponding initial beliefs of the two models respectively.

Lemma 3. *If* $d \in \mathcal{D}$ *is a belief bisimulation metric, then* $\forall b, b' \in B,\ d(b,b') = 0$ *iff* $\forall a \in A$*:*

$$\begin{aligned} & T_K(d)(\mathcal{R}(b,a), \mathcal{R}'(b',a)) = 0,\ and \\ & T_K(d)(\mathcal{T}(b,a), \mathcal{T}'(b',a)) = 0 \end{aligned}$$

Lemma 4. *If* $d \in \mathcal{D}$ *satisfies Lemma 3, then*

$$d(b,b') = 0 \Rightarrow b \approx b'\ (i.e.\ b, b'\ are\ bismilar)$$

Theorem 2. *If* $d \in \mathcal{D}$ *is a belief bisimulation metric defined on* B *(i.e. satisfies Lemma 3), then for all* b, b' $(\in B)$ *coming from* M, M' *respectively,*

$$d(b,b') = 0 \Rightarrow M \approx M'\ (i.e.\ M, M'\ are\ bismilar)$$

Next, we may redefine the belief bisimulation metric based on the Kantorovich metric analogous to Definition 7.

Definition 9 (Belief bisimulation metric). *Let* $c_R, c_T \geq 0$ *with* $c_R + c_T \leq 1$*. Define a continuous fn.* $F : \mathcal{D} \rightarrow \mathcal{D}$ *as,*

$$\begin{aligned} F(d)(b,b') = \ & \max_{a \in A}\ c_R\, T_K(d)(\mathcal{R}(b,a), \mathcal{R}'(b',a)) \\ & + c_T\, T_K(d)(\mathcal{T}(b,a), \mathcal{T}'(b',a)) \end{aligned} \tag{6}$$

Then F *has a least fixed-point,* d^**, and* d^* *is a belief bisimulation metric.*

The proofs for the above lemmas and theorem can be trivially generalized from [16]. The theoretical results concerning fixed-point metrics from [16] also apply here.

4.3 COMPUTING STOCHASTIC BISIMULATION

In the previous section, we developed a metric which when equal to zero, establishes an exact bisimulation relation between beliefs. This metric also varies smoothly relative to the differences in the reward and transition probabilities. Therefore, we may choose a tolerance parameter $\epsilon \in [0,1]$ and cluster models that are in the ϵ-neighborhoods: all models within a cluster are *ϵ-bisimilar* (denoted by $\approx_\epsilon$). More formally,

Definition 10 (Approximately bisimilar). *If* $d \in \mathcal{D}$ *is a bisimulation metric, then* $b, b' \in B$*,* $d(b,b') \leq \epsilon$ *iff* $\forall a \in A$

$$\begin{aligned} & T_K(d)(\mathcal{R}(b,a), \mathcal{R}'(b',a)) \leq \epsilon,\ and \\ & T_K(d)(\mathcal{T}(b,a), \mathcal{T}'(b',a)) \leq \epsilon \end{aligned} \tag{7}$$

Let $d(b,b') \leq \epsilon \Rightarrow b \approx_\epsilon b'$ and subsequently $M \approx_\epsilon M'$ (i.e. M, M' are ϵ-bisimilar) from Theorem 2.

Consider an $(|S| - 1)$-dimensional belief simplex B, and a partition P of B. We seek to find a *partition* P^*, called the *bisimulation partition*, that divides B into a disjoint set of convex regions (i.e. *blocks*) such that any two arbitrary models within a block in P^* are *ϵ-bisimilar* (as defined in Def. 10). As each block is convex, we may sufficiently represent a block using its finite set of vertex beliefs that make up its *convex hull*. For example, we illustrate a $2D$ belief simplex B in Fig. 4(a).

Algorithm 1 Bisimulation Partitioning

```
   Input: $\epsilon$
1: Let $P = \{B\}$ /* trivial one block partition */
2: while $P \ni B_1, B_2$ s.t. $P \neq split_\epsilon(B_1, B_2, P)$ do
3:     $P = split_\epsilon(B_1, B_2, P)$
4: $P^*$ = the equivalence relation given by $P$
   Output: $P^*$
```

The algorithm for computing the *bisimulation partition* is outlined in Alg. 1. We start with a trivial 1-block partition $P = \{B\}$ (line 1). We split the block if the boundary beliefs are not pairwise ϵ-bisimilar using *barycentric subdivision* (line 3). As we note in Def. 2, the check for ϵ-bisimilarity between two beliefs is recursive. We require that their updated beliefs also be pairwise ϵ-bisimilar, and so on. To that end, we define *stability* of a block $B1$ with respect to block $B2$ as when all pairs of boundary beliefs of $B1$ satisfy Lemma 10 of being carried into block $B2$ for every action $a \in A$ (line 2). We terminate when all blocks in the partition are *stable* with respect to each other. This final partition is a *bisimulation partition* (line 4).

Barycentric Subdivision Barycentric subdivision presents a principled way to *exactly* divide a convex n-dimensional

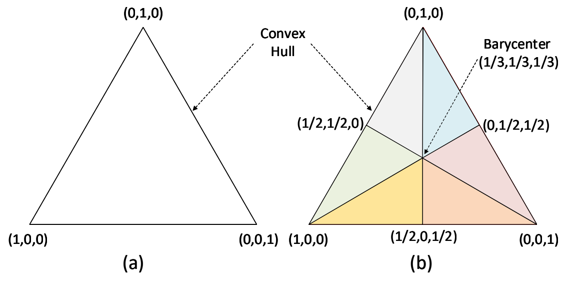

Figure 4: (a) A $2D$ belief simplex whose convex hull is represented using 3 boundary belief points. (b) Six $2D$ sub-simplices resulting from 1 *split*.

simplex into disjoint and convex sub-simplices with the same dimension by connecting the *barycenters* (or centroids) of their faces, *and* guarantee convergence to a *fixed point* [3]. Fig. 4(b) shows how a 2D simplex will look like after one *split* operation. One split operation on an n-dimensional simplex will result in $(n+1)!$ sub-simplices. This is evidently one of the bottlenecks of this approach; it is combinatorial with respect to the dimensionality of the state space. Since our split methodology may not result in a *minimal* partition, we additionally implement a ***merge***$_\epsilon$ method that combines all adjacent blocks – sharing a common $n-1$ dimensional face – which, when merged, continue to preserve the stability constraints.

4.4 SOLVING I-DID USING BISIMULATION

As a first step in our algorithm Solving I-DID (Alg. 2), we compute the final bisimulation partition, P^*, of the belief simplex after ***merge***$_\epsilon$ (line 3). We then solve one representative model m_c per block $C \in P^*$, by randomly picking a candidate model that lies within the block and solving it (lines 4-7). Note that the model can be of *any* other agent. We denote π_C as the solution to the block and assign it to be the solutions for all candidate models of all other agents that lie within that block. Of course, we also transfer the probability mass of all the candidates to their corresponding representatives. The rest of the solution for the I-DID proceeds in the usual manner as in [13].

5 EXPERIMENTS

We implemented our algorithm (shown in Alg. 2) and evaluated its performance against Doshi *et.* al's discriminative model update algorithm (DMU) [13, 28], a state-of-the-art BE technique.

Algorithm 2 Solving I-DID

```
    Inputs: level $l \geq 1$ I-DID or level 0 DID, horizon $T$,
    tolerance parameter $\epsilon$
    Preprocessing:Partitioning
 1: $P^* \leftarrow$ Bisimulation Partitioning $(\epsilon)$ (from Alg. 1)
 2: $P^* \leftarrow \text{merge}_\epsilon(P^*)$
 3: for each $C \in P^*$ do
 4:   Pick $m_c \in \mathcal{M}_{-i,l-1}$ s.t. $m_c \in C$
 5:   Let $\pi_{m_c} \leftarrow$ solution of $m_c$.
 6:   $\pi_C \leftarrow \pi_{m_c}$
 7: for each $j = 1 \dots N$ do
 8:   for each $m_j \in \mathcal{M}_{j,l-1}$ do
 9:     $\pi_{m_j} \leftarrow \pi_C \; s.t \; C \ni m_j$
    Expansion Phase
10: for $t$ from 0 to $T-1$ do
11:   if $l \geq 1$ then
12:     for $j$ from 0 to $N$ do
          Populate $M^{t+1}_{j,l-1}$
13:       for each $m^t_j \in \mathcal{M}^t_{j,l-1}$ do
14:         Let $OPT(m^t_j) \leftarrow \pi^t_{m_j}$
15:         Map the decision node of the solved model,
            $OPT(m^t_j)$, to the corresponding $A_j$
16:       for each $m^t_j \in \mathcal{M}^t_{j,l-1}$ do
17:         for each $a_j \in OPT(m^t_j)$ do
18:           for each $o_j \in O_j$ (part of $m^t_j$) do
19:             Update $j$'s belief, $b^{t+1}_j \leftarrow \tau(b^t_j, a_j, o_j)$
20:             $m^{t+1}_j \leftarrow$ New I-DID (or DID) with $b^{t+1}_j$
21:             $\mathcal{M}^{t+1}_{j,l-1} \overset{\cup}{\leftarrow} \{m^{t+1}_j\}$
22:       Add node $M^{t+1}_{j,l-1}$, and the model update link
23:   Add the nodes and links for $t+1$ time slice
24:   Establish the CPTs for chance and utility nodes
    Solution Phase
25: if $l \geq 1$ then
26:   Represent the model nodes, policy links and the
      model update links as in Fig. 3 to obtain the DID
27: Apply the standard look-ahead and backup method to
    solve the expanded DID
    Output: $\pi_i$
```

5.1 PROBLEM DOMAINS

We experimented on two multiagent problem domains with up to 20 agents: the multiagent tiger problem [19], and a slightly larger, more contemporary, solar energy storage problem inspired by Tesla's *solar city* initiative, modified from [31].

5.1.1 Multiagent Tiger, $Tiger$

In the multiagent tiger problem, N agents are tasked with finding a pot of gold hiding behind one of 2 closed doors.

[3] In the worst case, after ∞ splits, each sub-simplex constitutes 1 belief point which is *bismilar w.r.t.* itself.

Behind the other door is a ferocious tiger. The agents receive a positive reward for opening the door that leads to the gold but get penalized for opening the door that hides the tiger. The agents may open the left door or the right door, or listen. Upon performing the listen action, the agents receive one of 2 observations – growl from the left or growl from the right – indicating the probable location of the tiger. Additionally, the agents hear creaks originating from the direction of the door that was possibly opened by the other agent - creak from the left or creak from right - or silence if no door was opened. Upon performing *open* actions, the tiger's location randomly resets.

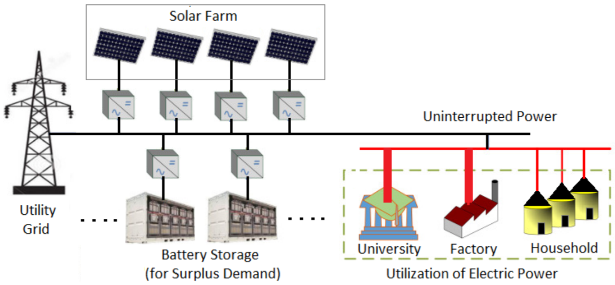

Figure 5: $Solar$ domain with the utility company (subject agent) and 5 consumers (other agents): a university, a factory, and 3 households.

5.1.2 Solar Energy Storage, $Solar$

The solar energy storage problem, on the other hand, is slightly more contemporary and trending right now. It is the consequence of the fact that much of the renewable energy generation is intermittent: wind or solar power generation peaks are often around times of low demand. Therefore, companies like Tesla offer massive batteries that store electricity during the day when the supply is abundant and discharge it, on demand, even after the sun goes down. Utility companies that own solar farms use these batteries for when there is surplus demand. We consider the problem faced by our subject agent (i.e. utility companies) in deciding how much battery storage it needs to buy to fully sustain the demand from $N-1$ different consumers as illustrated in Fig. 5. Let C_{max} be the maximum total electricity storage capacity in the batteries procured by the utility company. There is a fixed per-unit cost for energy procurement and a fixed positive return for per-unit sale to each consumer. The state space constitutes the difference between supply and expected demand in terms of percentage of C_{max}. Each consumer agent may have a different rate of electricity consumption depending on their own usage and the amount of electricity produced in-house (using solar panels or Tesla Energy's home batteries). The utility company may choose to draw 0, 1, or 2 units of electricity generated from its batteries. Similarly, each consumer may draw up to 2 units of electricity from the grid at a time. The state space is not fully observable to the agent because the expected demand in the next step is uncertain. We assume the existence of a data-driven demand forecast model that generates the observation probabilities indicating the possible demand in the next step. A two-time slice level l I-DID for the $Solar$ problem involving 3 agents is shown in Fig.6.

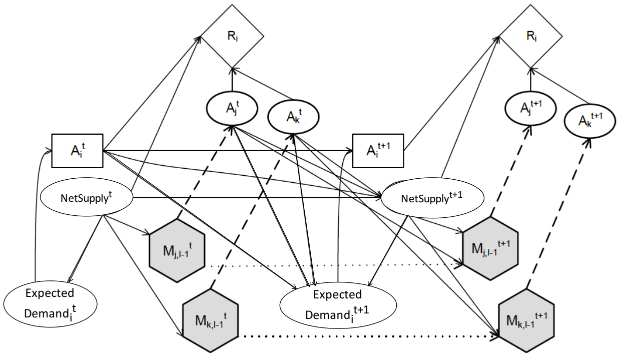

Figure 6: A generic two time-slice level l I-DID for the $Solar$ problem for the utility company agent i situated with two other consumer agents j and k.

Table 1: Domain dimensions and input parameters.

Domain	$\lvert\mathcal{M}^0_{-i}\rvert$	Dimension
Tiger	1000	$\lvert S\rvert = 2, \lvert A\rvert = 3, \lvert\Omega_i\rvert = 6, \lvert\Omega_{j\neq i}\rvert = 2$
Solar	2000	$\lvert S\rvert = 6, \lvert A\rvert = 3, \lvert\Omega_i\rvert = 3, \lvert\Omega_{j\neq i}\rvert = 3$

We summarize the domain parameters in Table 1. We compare run time performances and average reward over 10 trials of our approach I-DID BIS against DMU. Next, for the $Tiger$ problem, we scale in the number of agents and demonstrate significant savings in terms of the number of models solved for varying tolerance values. Our computing configuration included an Intel 2.7GHz processor, 32GB RAM and Linux.

5.2 VALIDATION

First we focus on generating solutions for an N-agent I-DID using our bisimulation approach I-DID BIS for both the multiagent tiger (denoted $Tiger$) and multiagent solar energy storage (denoted $Solar$) problem domains, and comparing their average expected utility over 10 trials against the solutions generated by DMU. We expect that the solution quality approaches that of DMU validating the correctness of our solutions. We vary the number of agents $N \in \{3, 5, 8\}$ and the horizons $T \in \{3, 5, 10, 15\}$ while fixing the candidate model space for each other agent $|\mathcal{M}_{j\neq i}| = 1000$ and 2000, and the tolerance parameter $\epsilon = 0.1$ and 0.14 for $Tiger$ and $Solar$ problems respectively. Expectedly, we note in Table 2 that the solution quality in terms of average reward over 10 trials for I-DID BIS was equal to that of DMU in 8 out of the 13 runs with different input parameter settings. In the remaining runs, the average expected utility of I-DID BIS solutions was slightly smaller compared to that of DMU, but not statistically significantly. This empirically verifies the correctness of our approach.

Table 2: Performance Comparison: I-DID BIS vs DMU

Domain	N	T	I-DID BIS		DMU		Speedup
			Time (in sec)	Avg Reward	Time (in sec)	Avg Reward	
Tiger	3	5	0.0071	14	2.611	14	368
		10	0.486	77.5	187.992	77.5	387
		15	27.301	135.5	11351.7	135.5	416
	5	5	0.806	77.5	5.953	88.5	7
		10	13.291	108.5	390.878	113	29
		15	76.474	135.5	22993.62	143.5	301
	8	5	9.166	86	17.115	86	2
		10	24.174	108	609.261	108	25
Solar	3	5	2.09	8.5	13.561	9	6
		10	18.983	23.5	437.22	23.5	23
	5	5	4.41	9	21.877	9	5
		10	53.847	106	693.22	108.5	13
	8	3	7.23	8.5	24.55	8.5	3

5.3 RUN TIME FOR SOLVING I-DIDS

Table 2 also shows how our algorithm stacks up against DMU in terms of computation time. We expect better run times in I-DID BIS because it only solves one model per block in the bisimulation partition whereas DMU requires that all initial models of the other agents be solved first before we start noticing its benefits. However, in I-DID BIS, there is a one-time overhead for computing the bisimulation partition. We test for varying ϵ, increasing number of agents N and planning horizons T for the 2 problem domains described earlier. Despite the overhead, we observe that our algorithm, I-DID BIS, takes orders of magnitude lesser time for solving the I-DID compared to DMU indicating that the benefits of solving lesser number of models outweigh the cost of computing the partition. We show the speedup of I-DID BIS with respect to DMU in Table 2. As expected, increasing N and T imply increasing run times.

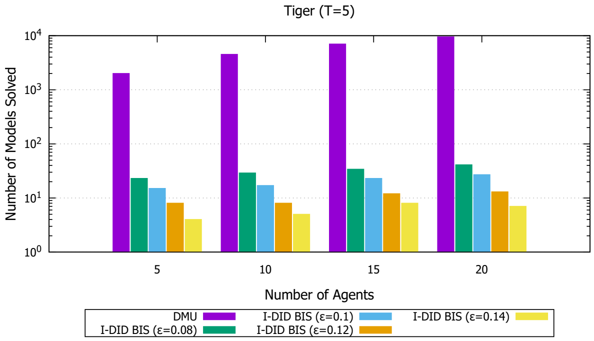

Figure 7: Savings in terms of the total number of models of the other agents solved for different ϵ values in the $Tiger$ domain with up to 20 agents. We also compare with DMU.

5.4 SCALABILITY

Next, we scale in the number of agents and illustrate in Fig. 7, the number of models solved for varying tolerance parameter values $\epsilon = \{0.08, 0.1, 0.12, 0.14\}$ for the $Tiger$ domain. We fix the planning horizon to $T = 5$ and the number of initial models of the other agents in our subject agent's I-DID to be $|\mathcal{M}_{j \neq i}| = 500$. As expected, we observe that the number of equivalence classes – and therefore the number of models solved – reduces with increasing ϵ. Again, because DMU ends up having to solve all initial models of the others, we note a significant increase in the number of models solved compared to I-DID BIS. Consequently, the time taken to solve the I-DID is expected to be orders of magnitude higher.

We note that we reached the memory cap on how much we can scale I-DIDs within HUGIN EXPERT [2], a well-known API for solving multistage influence diagrams. A general hurdle is that further scalability of ID-based graphical models is also limited by the absence of state-of-the-art techniques for solving DIDs within commercial implementations such as HUGIN EXPERT that predominantly rely on solving the entire DID in main memory. Although newer versions of HUGIN use limited memory IDs, a more scalable approach for solving multistage IDs would help drive further scalability of I-DID solutions.

6 CONCLUSION

In conclusion, we directly address the curse of dimensionality in I-DIDs due to exponential growth in the number of models over time. We successfully defined, implemented, and tested a metric to quantitatively measure the *similarity* between any two models in the space of models that the subject agent ascribes to the other agents in a partially observable setting within the context of an I-DID. Using such a metric, we were able to partition the belief space into equivalence classes using barycentric subdivision without having to solve the models first. This is the first time this has ever been done. Toward this, we generalized I-DIDs to N-agents and compared the performance of our model space compression technique against a baseline I-DID solver that uses the current state-of-the-art BE-based technique on two multiagent domains exhibiting partial observability with up to 20 agents. Our approach could benefit from a better partitioning technique because barycentric subdivision leads to a combinatorial explosion in the number of partitions generated after every split with the size of the state space. In any case, we think that this is a good first step in the right direction toward scaling I-DID solutions.

Acknowledgements

This research is supported in part by a grant from ONR N-00-0-141310870 and a grant from NSF IIA-1444182.

References

[1] Brandenburger Adam and Eddie Dekel. Hierarchies of beliefs and common knowledge. *International Journal of Game Theory*, 1993.

[2] Stig K. Andersen, Kristian G. Olesen, Finn V. Jensen, and Frank Jensen. Hugin: A shell for building bayesian belief universes for expert systems. In *IJCAI*, 1989.

[3] Robert J. Aumann. Interactive epistemology i: Knowledge. *International Journal of Game Theory*, 28(3):263–300, 1999.

[4] Pablo Samuel Castro, Prakash Panangaden, and Doina Precup. Equivalence relations in fully and partially observable markov decision processes. In *IJCAI*, volume 9, pages 1653–1658, 2009.

[5] Muthukumaran Chandrasekaran, Prashant Doshi, and Yifeng Zeng. Approximate solutions of interactive dynamic influence diagrams using epsilon-behavioral equivalence. In *11th International Symposium on Artificial Intelligence and Mathematics*, 2010.

[6] Muthukumaran Chandrasekaran, Prashant Doshi, Yifeng Zeng, and Yingke Chen. Team behavior in interactive dynamic influence diagrams with applications to ad hoc teams (extended abstract). In *Autonomous Agents and Multi-Agent Systems Conference (AAMAS)*, pages 1559–1560, 2014.

[7] Muthukumaran Chandrasekaran, Prashant Doshi, Yifeng Zeng, and Yingke Chen. Can bounded and self-interested agents be teammates? application to planning in ad hoc teams. *Autonomous Agents and Multi-Agent Systems*, 31(4):821–860, July 2017.

[8] Ross Conroy, Yifeng Zeng, and Jing Tang. Approximating value equivalence in interactive dynamic influence diagrams using behavioral coverage. In *IJCAI*, 2016.

[9] Gregory F Cooper. A method for using belief networks as influence diagrams. In *Proceedings of the Fourth Workshop on Uncertainty in Artificial Intelligence (UAI)*, pages 55–63, 1988.

[10] Prashant Doshi. Decision making in complex multiagent settings: A tale of two frameworks. *AI Magazine*, 33(4):82–95, 2012.

[11] Prashant Doshi, Muthukumaran Chandrasekaran, and Yifeng Zeng. Epsilon-subjective equivalence of models for interactive dynamic influence diagrams. In *WIC/ACM/IEEE WI-IAT*, 2010.

[12] Prashant Doshi, Xia Qu, Adam Goodie, and Diana Young. Modeling recursive reasoning in humans using empirically informed interactive POMDPs. In *International Autonomous Agents and Multiagent Systems Conference (AAMAS)*, pages 1223–1230, 2010.

[13] Prashant Doshi and Yifeng Zeng. Improved approximation of interactive dynamic influence diagrams using discriminative model updates. In *AAMAS*, pages 907–914, 2009.

[14] Prashant Doshi, Yifeng Zeng, and Qiongyu Chen. Graphical models for interactive pomdps: Representations and solutions. *JAAMAS*, 18(3):376–416, 2009.

[15] Edmund Durfee and Shlomo Zilberstein. *Multiagent Systems*, chapter Multiagent Planning, Control and Execution, pages 485–546. MIT Press, second edition, 2013.

[16] Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. In *Proceedings of Uncertainty in Artificial Intelligence (UAI-04)*, July 2004.

[17] Norm Ferns, Prakash Panangaden, and Doina Precup. Bisimulation metrics for continuous markov decision processes. *SIAM Journal on Computing*, 40(6):1662–1714, 2011.

[18] R. Givan, T. Dean, and M. Greig. Equivalence notions and model minimization in markov decision processes. *Artificial Intelligence*, 147:163–223, 2003.

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

[20] J.R. Munkres. *Elements of Algebraic Topology*. Advanced book classics. Avalon Publishing, 1984.

[21] Brenda Ng, Carol Meyers, Kofi Boakye, and John Nitao. Towards applying interactive POMDPs to real-world adversary modeling. In *Innovative Applications in Artificial Intelligence (IAAI)*, pages 1814–1820, 2010.

[22] David Pynadath and Stacy Marsella. Minimal mental models. In *AAAI*, pages 1038–1044, 2007.

[23] Richard Seymour and Gilbert L. Peterson. A trust-based multiagent system. In *IEEE International Conference on Computational Science and Engineering*, pages 109–116, 2009.

[24] L. Nisonovich Vaserstein. Markov processes over denumerable products of spaces, describing large systems of automata. *Problemy Peredachi Informatsii*, 5(3):64–72, 1969.

[25] Fangju Wang. An I-POMDP based multi-agent architecture for dialogue tutoring. In *International Conference on Advanced ICT and Education (ICAICTE-13)*, pages 486–489, 2013.

[26] Mark P. Woodward and Robert J. Wood. Learning from humans as an i-pomdp. *CoRR*, abs/1204.0274, 2012.

[27] Michael Wunder, Michael Kaisers, John Yaros, and Michael Littman. Using iterated reasoning to predict opponent strategies. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 593–600, 2011.

[28] Yifeng Zeng and Prashant Doshi. Exploiting model equivalences for solving interactive dynamic influence diagrams. *JAIR*, 43:211–255, 2012.

[29] Yifeng Zeng, Prashant Doshi, Yingke Chen, Yinghui Pan, Hua Mao, and Muthukumaran Chandrasekaran. Approximating behavioral equivalence for scaling solutions of i-dids. *Knowledge and Information Systems*, 49(2):511–552, 2016.

[30] Yifeng Zeng, Prashant Doshi, Yinghui Pan, Hua Mao, Muthukumaran Chandrasekaran, and Jian Luo. Utilizing partial policies for identifying equivalence of behavioral models. In *AAAI*, pages 1083–1088, 2011.

[31] Ronghuo Zheng, Ying Xu, Nilanjan Chakraborty, and Katia P Sycara. A crowdfunding model for green energy investment. In *IJCAI*, pages 2669–2676, 2015.