Lossless Clustering of Histories in Decentralized POMDPs

Frans A. Oliehoek
Informatics Institute,
University of Amsterdam
Kruislaan 403, 1098 SJ
Amsterdam, The Netherlands
F.A.Oliehoek@uva.nl

Shimon Whiteson
Informatics Institute,
University of Amsterdam
Kruislaan 403, 1098 SJ
Amsterdam, The Netherlands
S.A.Whiteson@uva.nl

Matthijs T.J. Spaan
Institute for Systems and
Robotics
Instituto Superior Técnico
Lisbon, Portugal
mtjspaan@isr.ist.utl.pt

ABSTRACT

Decentralized partially observable Markov decision processes (Dec-POMDPs) constitute a generic and expressive framework for multiagent planning under uncertainty. However, planning optimally is difficult because solutions map local observation histories to actions, and the number of such histories grows exponentially in the planning horizon. In this work, we identify a criterion that allows for lossless clustering of observation histories: i.e., we prove that when two histories satisfy the criterion, they have the same optimal value and thus can be treated as one. We show how this result can be exploited in optimal policy search and demonstrate empirically that it can provide a speed-up of multiple orders of magnitude, allowing the optimal solution of significantly larger problems. We also perform an empirical analysis of the generality of our clustering method, which suggests that it may also be useful in other (approximate) Dec-POMDP solution methods.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*

General Terms

Algorithms, Theory, Performance, Experimentation

Keywords

Planning under uncertainty, cooperative multiagent systems, decentralized POMDPs

1. INTRODUCTION

A fundamental question in artificial intelligence is how an agent should decide which action to take in a specific situation. When uncertainty is involved, this question is particularly challenging. In the last two decades, many researchers have turned to decision-theoretic models for an answer. In particular, the Markov decision process (MDP) has become a popular model for single-agent planning under action uncertainty, i.e., when the agent's actions have stochastic effects. When uncertainty regarding the system state is also present, partially observable MDPs (POMDPs) can be used.

Cite as: Lossless Clustering of Histories in Decentralized POMDPs, Frans A. Oliehoek, Shimon Whiteson, Matthijs T.J. Spaan, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 577–584

In the multiagent case, planning under action and state uncertainty can be formalized as a decentralized POMDP (Dec-POMDP) [4], which models the interaction of cooperative agents. In this model, each agent receives only its own noisy observation, which provides limited information about the state. In this setting POMDP theory does not apply as agents cannot compute the belief state. Moreover, in addition to reasoning about the state, each agent must reason about the the actions of other agents, which are in turn dependent on their observations.

This paper considers the optimal solution of finite-horizon Dec-POMDPs [10, 17, 3, 12, 5]. Such a solution, or *joint policy,* specifies what action to take for each agent and each possible observation history. Since the number of such joint policies is exponential in the number of histories, finding optimal solutions is intractable for all but the smallest problems [4]. To overcome this problem, previous research has investigated lossy compression of the space of histories [8] and observations [6], and lossless policy space compression [5].

In this paper, we reduce the computational costs of solving Dec-POMDPs by clustering histories, an idea first considered by Emery-Montemerlo et al. [8], who cluster histories in Bayesian games (BGs) that model individual stages of the Dec-POMDP. However, their approach uses an ad-hoc heuristic to determine which histories to cluster and consequently finds only approximate solutions. By contrast, we identify a criterion that *guarantees* that two individual histories have the same optimal value, allowing *lossless clustering* and therefore faster optimal solutions of Dec-POMDPs. Comparing to policy space compression [5], clustering histories has potentially more impact, as the policy space is exponentially larger than the history space.

In particular, we cluster histories within the MAA* algorithm [17] applied to BGs, as described in [12]. We demonstrate that in several well-known test problems, our proposed method allows for the optimal solution of significantly longer horizons. For instance, we solve the well-known benchmark decentralized tiger (Dec-Tiger) problem [11] for horizon $h = 5$ (in which case there are $3.82e29$ joint policies). To the best of our knowledge, such results were not obtainable previously. Subsequently we analyze the generality of the proposed clustering. Results suggest that our clustering approach may have a significant impact on other (approximate) algorithms as well.

2. THE DEC-POMDP MODEL

In this section we formally introduce the Dec-POMDP model and describe the planning problem. A *decentralized*

partially observable Markov decision process (Dec-POMDP) with n agents is defined as a tuple $\langle \mathcal{S},\mathcal{A},T,R,\mathcal{O},O \rangle$ where:

- $\mathcal{S}$ is a finite set of states.
- The set $\mathcal{A} = \times_i \mathcal{A}_i$ is the set of joint actions, where $\mathcal{A}_i$ is the set of actions available to agent i. Every time step one joint action $a = \langle a_1,...,a_n \rangle$ is taken.[1]
- T is the transition function, a mapping from states and joint actions to probability distributions over next states: $T : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{P}(\mathcal{S})$.
- R is the reward function, a mapping from states and joint actions to real numbers: $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$.
- $\mathcal{O} = \times_i \mathcal{O}_i$ is the set of joint observations. Every time step one joint observation $o = \langle o_1,...,o_n \rangle$ is received.
- O is the observation function, a mapping from joint actions and successor states to probability distributions over joint observations: $O : \mathcal{A} \times \mathcal{S} \rightarrow \mathcal{P}(\mathcal{O})$.

In a Dec-POMDP, an agent i knows only its own individual actions a_i and observations o_i. The planning problem involves finding the best *policy* for each agent, where a policy is a mapping from the individual histories an agent can observe to its actions. We assume a finite planning *horizon* of h time steps, and an initial *belief* $b^0 \in \mathcal{P}(\mathcal{S})$; this is the initial state distribution at time $t = 0$.[2]

The *action-observation history (AOH) for agent i*, $\vec{\theta}_i^t$, is the sequence of actions taken and observations received by agent i until time step t: $\vec{\theta}_i^t = \left(a_i^0,o_i^1,a_i^1,...,a_i^{t-1},o_i^t\right)$. The *joint action-observation history* is a tuple with the action-observation history for all agents $\vec{\theta}^t = \langle \vec{\theta}_1^t,...,\vec{\theta}_n^t \rangle$. The *observation history (OH) for agent i* is the sequence of observations an agent has received: $\vec{o}_i^t = \left(o_i^1,...,o_i^t\right)$. Similar to action-observation histories, $\vec{o}^t$ denotes a joint observation history. In a similar fashion $\vec{a}_i^t$ and $\vec{a}^t$ denote (joint) action histories.

A *pure* or *deterministic policy*, π_i, for agent i in a Dec-POMDP is a mapping from observation histories to actions, $\pi_i : \vec{\mathcal{O}}_i \rightarrow \mathcal{A}_i$. A pure joint policy π is a tuple containing a pure policy for each agent. We also consider policies that are partially specified w.r.t. time. We can write a policy for agent i as $\pi_i = (\delta_i^0,\ldots,\delta_i^{h-1})$, where δ_i^t is a *decision rule* for stage t: a mapping from length-t observation histories to actions $\delta_i^t : \vec{\mathcal{O}}_i^t \rightarrow \mathcal{A}_i$. Now a partial policy $\varphi_i^t = (\delta_i^0,\ldots,\delta_i^{t-1})$ only specifies actions for the first t stages. A partial joint policy $\varphi^t = \langle \varphi_1^t,\ldots,\varphi_n^t \rangle$ specifies a partial policy for all agents.

3. DEC-POMDPS VIA BAYESIAN GAMES

A Bayesian game (BG) [13] is an extension of a normal form game in which the agents can hold some private information which is expressed by their *type*. Emery-Montemerlo et al. [7] proposed to use BGs to approximate Dec-POMDPs. In their method, agents construct and solve a BG for each stage of the process in an on-line fashion. Such modeling is exact when using an optimal payoff function for the BGs [12].

The crucial difficulty in making a decision at some stage t in a Dec-POMDP is that the agents lack a common signal on which to condition their actions and must rely instead on their individual (action-)observation histories. Given b^0 and φ^t, the joint policy followed for stages $0 \ldots t-1$, this situation can be modeled as a BG with identical payoffs. Such a game $BG(b^0,\varphi^t)$ consists of the set of agents $\{1 \ldots n\}$, their joint actions $\mathcal{A}$, the set of their joint types Θ, a probability distribution over these joint types $P(\cdot)$ and a payoff function u that maps a joint type and action to a real number $u(\theta,a)$. A joint type $\theta \in \Theta$ specifies a type for each agent $\theta = \langle \theta_1,\ldots,\theta_n \rangle$. Since the type of an agent represents the private information it holds, types are AOHs $\theta_i \equiv \vec{\theta}_i^t$ and joint types correspond to joint AOHs $\theta \equiv \vec{\theta}^t$. Given φ^t and b^0, the probability distribution over joint AOHs is well-defined and the payoff function is given by $u(\theta,a) \equiv Q^*(\vec{\theta}^t,a)$, the optimal Q-value function of the Dec-POMDP. Although Q^* is intractable to compute, heuristic Q-value functions $\widehat{Q}$ can be computed, for instance by using the underlying MDP's value function.

We can solve such a BG by computing the expected heuristic value $\widehat{V}$ for all joint BG-policies $\beta^t = \langle \beta_1,\ldots,\beta_n \rangle$, where an individual BG-policy maps types (i.e., AOH histories) to actions $\beta_i(\vec{\theta}_i^t) = a_i^t$. This valuation is given by

$$\widehat{V}(\beta^t) = \sum_{\vec{\theta}^t} P(\vec{\theta}^t|\varphi^t,b^0)\widehat{Q}(\vec{\theta}^t,\beta^t(\vec{\theta}^t)), \qquad (1)$$

where $\beta^t(\vec{\theta}^t) = \langle \beta_i(\vec{\theta}_i^t) \rangle_{i=1\ldots n}$ denotes the joint action that results from application of the individual BG-policies to the individual AOH $\vec{\theta}_i^t$ specified by $\vec{\theta}^t$. The solution $\beta^{t,*}$ is the joint BG-policy with the highest expected value. Note that if φ^t is deterministic, the probability of a joint AOH $\vec{\theta}^t = \langle \vec{a}^t,\vec{o}^t \rangle$ is non-zero for only one $\vec{a}^t$ per $\vec{o}^t$. I.e., $\vec{a}^t$ can be reconstructed from $\vec{o}^t,\varphi^t$. Therefore, in effect the BG-policies reduce to decision rules: mappings from OH histories to actions $\beta_i(\vec{o}_i^t) = a_i^t$.

The modeling of a stage of a Dec-POMDP as a BG as outlined above, can be applied in a heuristic policy search scheme called Generalized MAA* (GMAA*), which generalizes MAA* [17] and the method proposed by Emery-Montemerlo et al. [7]. Algorithm 1 shows GMAA*, which maintains an open list P of partial joint policies φ^t and their heuristic values $\widehat{V}(\varphi^t)$. Every iteration the highest ranked φ^t is selected and expanded, i.e., the Bayesian game $BG(\varphi^t,b^0)$ is constructed and all joint BG-policies β^t are evaluated. Consequently these joint BG-policies are used to construct a new set of partial policies

$$\Phi_{\text{new}} := \{\varphi^{t+1} = (\varphi^t,\beta^t)\}$$

and their heuristic values. When the heuristic values are an upper bound to the true values, any lower bounds $\underline{v}^\star$ (i.e.,

Algorithm 1 GMAA*

```
1: v* ← −∞
2: P ← {φ^0 = ()}
3: repeat
4:    φ^t ← SelectHighestRankedPartialJPol(P)
5:    Φ_new ← ConstructAndSolveBG(φ^t, b^0)
6:    if Φ_new contains full policies Π_new ⊆ Φ_new then
7:       π' ← arg max_{π∈Π_new} V(π)
8:       if V(π') > v* then
9:          v* ← V(π') {found new lower bound}
10:         π* ← π'
11:         P ← {φ ∈ P | V̂(φ) > v*}{prune P}
12:      Φ_new ← Φ_new \ Π_new {remove full policies}
13:   P ← (P \ φ^t) ∪ {φ ∈ Φ_new | V̂(φ) > v*}
14: until P is empty
```

[1] Unless stated otherwise, subscripts denote agent indices.
[2] Unless stated otherwise, superscripts denote time indices.

full joint policies) that are found can be used to prune P. When P becomes empty, the optimal policy has been found.

GMAA* as outlined here is MAA* reformulated to work on BGs. The approach of Emery-Montemerlo et al. [7] is similar, but does not backtrack. I.e., rather than constructing all new partial policies $\forall_{\beta^t}\ \varphi^{t+1} = (\varphi^t,\beta^t)$ only the best-ranked partial policy $(\varphi^t,\beta^{t,*})$ is constructed and the open list P will never contain more than 1 partial policy.

4. LOSSLESS CLUSTERING

GMAA* can find the optimal solution for Dec-POMDPs by repeatedly solving BGs for different stages. However, the cost of solving these BGs grows doubly-exponentially with the horizon. In a BG for the last stage, the number of joint policies, and thus the cost of optimally solving it, is

$$O\Big(|\mathcal{A}_*|^{n(|\mathcal{O}_*|^{h-1})}\Big), \tag{2}$$

where $\mathcal{A}_*$ and $\mathcal{O}_*$ denote the largest individual action and observation sets. To counter the exponential growth of the BGs, Emery-Montemerlo et al. [7] proposed to prune AOHs with low probabilities. In subsequent work [8], they replaced this pruning by clustering histories, based on the profiles of the payoff functions of the BGs, thereby increasing the quality of the found policies.

Here, we also consider clustering of AOHs. In contrast, however, we do not consider a lossy clustering scheme based on the heuristic payoff function $\widehat{Q}$ of the BGs. Rather, we introduce a criterion for clustering AOHs based on the belief they induce over histories of the other agents and over states. Subsequently we show that clustering histories that satisfy this criterion is *lossless*: the solution for the clustered BG can be used to construct the solution for the original BG and the values of the two BGs are identical. Thus, the criterion allows for clustering of AOHs in BGs that represent Dec-POMDPs without compromising solution quality, i.e., optimality is preserved.

4.1 Probabilistic Equivalence Criterion

A particular stage t of a Dec-POMDP can be represented as a BG. For such a BG we can cluster two individual histories $\vec{\theta}_{i,a},\vec{\theta}_{i,b}$ when they satisfy the *probabilistic equivalence criterion* as we define here.

Criterion 1 (Probabilistic Equivalence). *Two AOHs $\vec{\theta}_{i,a},\vec{\theta}_{i,b}$ for agent i are probabilistically equivalent (PE) when the following holds:*

$$\forall_{\vec{\theta}_{\neq i}}\forall_s \quad P(s,\vec{\theta}_{\neq i}|\vec{\theta}_{i,a}) = P(s,\vec{\theta}_{\neq i}|\vec{\theta}_{i,b}). \tag{3}$$

Remark 1. Alternatively, the criterion can be rewritten to the following two:

$$\forall_{\vec{\theta}_{\neq i}} \quad P(\vec{\theta}_{\neq i}|\vec{\theta}_{i,a}) = P(\vec{\theta}_{\neq i}|\vec{\theta}_{i,b}), \tag{4}$$

$$\forall_{\vec{\theta}_{\neq i}}\forall_s \quad P(s|\vec{\theta}_{\neq i},\vec{\theta}_{i,a}) = P(s|\vec{\theta}_{\neq i},\vec{\theta}_{i,b}). \tag{5}$$

These equations give a natural interpretation: the first says that the probability distribution over the other agents' AOHs must be identical for both $\vec{\theta}_{i,a},\vec{\theta}_{i,b}$. The second demands that the resulting joint beliefs are identical.

Remark 2. The above probabilities are not well defined without the initial state distribution b^0 and past joint policy φ^t. However, since we consider clustering of histories within a particular BG (for some stage t) and because this BG is constructed for a particular b^0,φ^t, they are implicitly specified. Therefore we drop these arguments, clarifying the notation.

Remark 3. Probabilities as defined in (3) appear similar to beliefs in POMDPs, but are substantially different because they are not sufficient statistics. In fact, only a "multi-agent belief" specified over states and *future policies* of other agents has been shown to be a sufficient statistic in Dec-POMDPs [10]. Our notion of PE is specified over states and AOHs given only a *past* joint policy. Thus establishing conditions for equivalence in Dec-POMDPs is a non-trivial extension over the POMDP case.

Probabilistic equivalence has a convenient property: if it holds for a particular pair of histories, then it will also hold for all *identical extensions* of those histories, i.e., the property propagates forwards regardless of the policies the other agents use.

Definition 1. Given two AOHs $\vec{\theta}^t_{i,a},\vec{\theta}^t_{i,b}$, their respective extensions $\vec{\theta}^{t+1}_{i,a} = (\vec{\theta}^t_{i,a},a_i,o_i)$ and $\vec{\theta}^{t+1}_{i,b} = (\vec{\theta}^t_{i,b},a'_i,o'_i)$ are called *identical extensions* if and only if $a_i = a'_i$ and $o_i = o'_i$.

Lemma 1 (Propagation of PE). *Given $\vec{\theta}^t_{i,a},\vec{\theta}^t_{i,b}$ that are PE, regardless of the policy the other agents use $\beta^t_{\neq i}$, identical extensions are also PE:*

$$\forall_{a^t_i}\forall_{o^{t+1}_i}\forall_{\beta^t_{\neq i}}\forall_{s^{t+1}}\forall_{\vec{\theta}^{t+1}_{\neq i}} \quad P(s^{t+1},\vec{\theta}^{t+1}_{\neq i}|\vec{\theta}^t_{i,a},a^t_i,o^{t+1}_i,\beta^t_{\neq i}) = P(s^{t+1},\vec{\theta}^{t+1}_{\neq i}|\vec{\theta}^t_{i,b},a^t_i,o^{t+1}_i,\beta^t_{\neq i}) \tag{6}$$

Proof. See appendix. □

4.2 Proof of Lossless Clustering

Here we prove that if the identified PE criterion holds for two AOHs in a BG, we can cluster them together without loss in value for any agent. Theorem 2 shows that clustering can be performed when an agent commits to taking the same action for two histories. Since a rational agent will commit if two histories are *best-response equivalent* (BRE), Lemma 3 identifies two conditions for BRE. Both of these conditions follow from the PE criterion: the first by definition and the second by Lemma 4. Theorem 5 combines these pieces to show that lossless clustering is possible when the PE criterion holds.

Theorem 2 (Reduction through commitment). *Given that in a Bayesian game B agent i is committed to select a policy that assigns the same action for two of its types θ^a_i,θ^b_i, i.e., to select a policy β_i such that*

$$\beta_i(\theta^a_i) = \beta_i(\theta^b_i), \tag{7}$$

then the BG can be reduced to a smaller one without loss in value for any of the agents. I.e., the two types can be substituted by a new type θ^c_i such that

$$\forall_{\theta_{\neq i}}\ P(\theta^c_i,\theta_{\neq i}) = P(\theta^a_i,\theta_{\neq i}) + P(\theta^b_i,\theta_{\neq i}) \tag{8}$$

$$\forall_j\forall_a \quad u(\langle\theta^c_i,\theta_{\neq i}\rangle,a) = \frac{P(\theta^a_i,\theta_{\neq i})u(\langle\theta^a_i,\theta_{\neq i}\rangle,a) + P(\theta^b_i,\theta_{\neq i})u(\langle\theta^b_i,\theta_{\neq i}\rangle,a)}{P(\theta^a_i,\theta_{\neq i}) + P(\theta^b_i,\theta_{\neq i})}. \tag{9}$$

The result is a new BG B' in which the expected value is the same as in the original BG: $V^{B'} = V^B$.

PROOF. See appendix. □

This theorem tells us that given that agent i is committed to taking the same action for its types θ_i^a,θ_i^b, we can reduce the Bayesian game B to a smaller one B' and translate the joint BG-policy β' found for B' back to a joint BG-policy β in B. This does not necessarily mean that $\beta = (\beta_i,\beta_{\neq i})$ is also a solution (Bayesian Nash-equilibrium) for B, because the best-response of agent i against $\beta_{\neq i}$ may not select the same action for θ_i^a,θ_i^b. Rather β_i is the best-response against $\beta_{\neq i}$ *given that the same action needs to be taken for* θ_i^a,θ_i^b. For instance, when θ_i^a,θ_i^b are BRE as we detail below.

We now consider a BG for a stage of a Dec-POMDP and demonstrate when the best-response for two histories is the same. In a general BG, a best-response β_i^* for agent i's type θ_i against some fixed policy profile $\beta_{\neq i}$ is given by

$$\beta_i^*(\theta_i) = \arg\max_{a_i} \sum_{\theta_{\neq i}} P(\theta_{\neq i}|\theta_i) u_i(\langle\theta_i,\theta_{\neq i}\rangle,\langle a_i,\beta_{\neq i}(\theta_{\neq i})\rangle).$$

LEMMA 3 (BEST-RESPONSE EQUIVALENCE). *When for two types $\theta_{i,a}$,$\theta_{i,b}$ it holds that*

$$\forall_{\theta_{\neq i}} \quad P(\theta_{\neq i}|\theta_{i,a}) = P(\theta_{\neq i}|\theta_{i,b}) \tag{10}$$

and

$$\forall_a \forall_{\theta_{\neq i}} \quad u(\theta_{i,a},\theta_{\neq i},a) = u(\theta_{i,b},\theta_{\neq i},a), \tag{11}$$

then the best-response policy for agent i will always select the same action for $\theta_{i,a}$,$\theta_{i,b}$.

PROOF. We can simply derive

$$\begin{aligned}\beta_i^*(\theta_{i,a}) &= \arg\max_{a_i} \sum_{\theta_{\neq i}} P(\theta_{\neq i}|\theta_{i,a}) u(\theta_{i,a},\theta_{\neq i},a_i,a_{\neq i}) \\ &= \arg\max_{a_i} \sum_{\theta_{\neq i}} P(\theta_{\neq i}|\theta_{i,b}) u(\theta_{i,b},\theta_{\neq i},a_i,a_{\neq i})\end{aligned}$$

which is equal to $\beta_i^*(\theta_{i,b})$. □

Since we want to show that two PE histories can be clustered under the optimal policy, we need to show (11) holds and thus that their optimal Q-values are the same.

LEMMA 4 (Q^* EQUIVALENCE). *When two histories in a BG for a Dec-POMDP $\vec{\theta}_{i,a}$, $\vec{\theta}_{i,b}$ satisfy Criterion 1, then they have equal Q-values according to the optimal finite-horizon Q-value function*

$$\forall_{\vec{\theta}^t_{\neq i}} \forall_a \quad Q^*(\vec{\theta}^t_{i,a},\vec{\theta}^t_{\neq i},a) = Q^*(\vec{\theta}^t_{i,b},\vec{\theta}^t_{\neq i},a). \tag{12}$$

PROOF. The proof is by induction backwards in time (i.e., from the last time step $t = h-1$ to the first $t = 0$). However, to prove the induction step we employ Lemma 1, which ensures propagation forward through time of the PE criterion on identical extensions. See appendix for details. □

THEOREM 5 (LOSSLESS CLUSTERING). *When two histories $\vec{\theta}_{i,a}$,$\vec{\theta}_{i,b}$ are PE, then they are best-response equivalent and can be clustered as one history without loss in value.*

PROOF. Given that PE implies BRE, we can apply Theorem 2 to prove that $\vec{\theta}_{i,a}$,$\vec{\theta}_{i,b}$ can be clustered without loss in value. Proof that PE in fact does imply BRE is as follows. The criterion itself entails (10). (11) for the BG constructed using the optimal Q-value function follows from Lemma 4. □

Algorithm 2 $\Phi_{\text{new}} = \text{ConstructAndSolveBG-Cluster}(\varphi^t,b^0)$

```
1: BG^t ← ConstructBG(φ^t,b^0)
2: BG^t ← ClusterBG(BG^t)
3: for all joint BG-policies β^t do
4:    φ^{t+1} ←(φ^t,β^t)
5:    V̂(φ^{t+1}) ← V^{0...t-1}(φ^t) + V̂(β^t)
6:    Φ_new ← Φ_new ∪ φ^{t+1}
```

Algorithm 3 $BG = \text{ClusterBG}(BG)$

```
1: for each agent i do
2:    for each individual type θ_i ∈ BG.Θ_i do
3:       for each individual type θ'_i ∈ BG.Θ_i do
4:          if P(θ'_i) = 0 then
5:             BG.Θ_i ← BG.Θ_i \ θ'_i {Remove θ'_i from BG:}
6:             continue with next θ'_i ∈ BG.Θ_i
7:          isEquivalent ← true
8:          for all ⟨s,θ_{≠i}⟩ do
9:             if P(s,θ_{≠i}|θ_i) ≠ P(s,θ_{≠i}|θ'_i) then
10:               isEquivalent ← false
11:               break
12:         if isEquivalent then
13:            BG.Θ_i ← BG.Θ_i \ θ'_i {Remove θ'_i from BG:}
14:            for each a ∈ A do
15:               for all θ_{≠i} do
16:                  { take the lowest upper bound }
                     u(θ_i,θ_{≠i},a) ← min(u(θ_i,θ_{≠i},a),u(θ'_i,θ_{≠i},a))
17:                  P(θ_i,θ_{≠i}) ← P(θ_i,θ_{≠i}) + P(θ'_i,θ_{≠i})
18:                  P(θ'_i,θ_{≠i}) ← 0
```

5. GMAA*-CLUSTER

Knowledge of which individual histories can be clustered together without loss of value may potentially be employed by many algorithms. In this paper, we focus on its application within the GMAA* framework.

Emery-Montemerlo et al. [8] showed how clustering can be incorporated at every stage in their algorithm: when the BG for a stage t is constructed, first a clustering of the individual histories (types) is performed and only afterward the (reduced) BG is solved. The same thing can be done within GMAA*, leading to an algorithm we dub GMAA*-Cluster. GMAA*-Cluster replaces the function ConstructAndSolveBG from Algorithm 1 with Algorithm 2. The actual clustering is performed by Algorithm 3, which performs a pairwise comparison of all types of each agent to see if they satisfy the criterion and eliminates individual types with zero probability. If there is a large number of states, some efficiency may be gained by first checking (4) and then checking (5), rather than looping over all $\langle s,\theta_{\neq i}\rangle$ as is done in line 8. Also note that the algorithm shown assumes that the heuristic used as the payoff function u is admissible (i.e., is an upper bound to the optimal value). Therefore, rather then using (9), we can take the lowest upper bound in line 16. In general this might increase the tightness of the heuristic, which can have a great effect on the performance [12].

Because PE of AOHs propagates forwards (i.e., identical extensions of PE histories are also PE), we do not have to construct all $|\mathcal{O}_i|^t$ possible AOHs at every stage t (given the past policy φ_i^t of agent i). Instead of clustering this exponentially growing set of types, we can simply extend the already clustered types of the previous stage's BG, as shown in Algorithm 4. This way, we bootstrap the clustering at each stage and spend significantly less time clustering. If the typeset Θ_i' at the previous stage $t-1$ was much smaller than the set of all histories $|\Theta_i'| \ll |\mathcal{O}_i|^{t-1}$, then the new typeset Θ_i is also much smaller: $|\Theta_i| \ll |\mathcal{O}_i|^t$.

The above is possible only because we perform an exact,

Algorithm 4 BG^t = ConstructExtendedBG(BG^{t-1},β^{t-1})

```
1: t ← BG^{t-1}.t + 1
2: pBG ← BG^{t-1}
3: pPol ← β^{t-1}
4: for each agent i do
5:    BG^t.Θ_i = ConstructExtendedTypeSet(i)
6: for each joint type θ = (θ^{t-1},a^{t-1},o^t) ∈ BG^t.Θ do
7:    for each state s^t ∈ S do
8:       Compute P(s^t|θ)
9:    P(θ) ← P(o^t|θ^{t-1},a^{t-1})P(θ^{t-1})
10:   for each a ∈ A do
11:      q ← ∞
12:      for each history θ⃗^t represented by θ do
13:         q ← min(q,Q̂(θ⃗^t,a)) { if Q* ≤ Q̂ we can take the lowest
            upper bound }
14:      u(θ,a) ← q
```

value preserving, clustering for which Lemma 1 tells us that identical extensions will also be clustered without loss in value. When performing the same procedure in a lossy clustering scheme (e.g., as in [8]) errors might accumulate and thus it might be better to re-cluster from scratch at every stage. Since lossy clustering is beyond the scope of this paper, we only consider bootstrapped clustering.

Optimally solving a BG takes exponential time w.r.t. the number of types, as there are $O(|\mathcal{A}_*|^{n|\Theta_*|})$ joint BG-policies. Clustering involves a pairwise comparison of all types of each agent and each of these comparisons needs to check $O(|\Theta_*|^{n-1}|\mathcal{S}|)$ numbers for equality to verify (3). The total cost of clustering can therefore be written as

$$O(n\,|\Theta_*|^2\,|\Theta_*|^{n-1}\,|\mathcal{S}|),$$

which is only polynomial in the number of types, and the number of types itself can be much less than the number of histories when using bootstrapped clustering. When clustering decreases the number of types $|\Theta_*|$, it can therefore significantly reduce the overall time needed. However, when no clustering is possible, we will incur some overhead.

From an implementation perspective, it is important to avoid reconstructing flat Dec-POMDP policies, as they can cause an exponential blow-up in space requirements. Instead, we maintain a pointer to the previous joint policy $\varphi^t = (\varphi^{t-1},\beta^{t-1})$. The current implementation keeps all constructed φ^{t-1} in memory, but reference counting can be used discard all φ^{t-1} which are no longer being pointed to by any φ^tin the policy pool.

6. EXPERIMENTS

In our experiments, we first compare optimal solving of several problems with and without clustering, followed by an analysis of the generality of lossless clustering, including for larger horizons for which optimal solutions are infeasible to compute.

We evaluated on a range of standard benchmarks problems. The most well-known are the Dec-Tiger [11] and BroadcastChannel [10] problem. Dec-Tiger considers two agents that have to coordinate to open the door to the treasure, rather than to the tiger. At the start of the problem, the tiger is behind the left or right door with 50% probability. Agents can either open either door or listen. Every stage each agent noisily hears the tiger behind the left or right door, but the observation conveys information only when both agents selected 'listen' in the previous stage. Opening the door resets the problem, but the agents do not observe this. In BroadcastChannel two agents have to transmit messages over a communication channel, but when both agents transmit at the same time a collision occurs which is noisily observed by the agents. Other problems we used are GridSmall with two observations [1]; Cooperative Box Pushing [15], a larger two-robot benchmark; Recycling Robots [2]; FireFighting [12], and Hotel 1 [16].

All timing results mentioned in this paper are CPU times with a resolution of 0.01s. The timings exclude time needed to parse the problem and compute the heuristic (which can be amortized).

6.1 Optimal solutions using clustering

For all problems we compared GMAA* against GMAA*-Cluster using the Q_{BG} or Q_{MDP} heuristic [12], depending on problem size and planning horizon. Regardless of the particular heuristic, both methods compute an optimal policy, but we expect GMAA*-Cluster to be more efficient when lossless clustering is possible in the domain. The obtained results are shown in Table 1, which details the optimal value V^* and the running times $T_{\text{GMAA*}}$ for GMAA* and $T_{cluster}$ for GMAA*-Cluster. Entries marked '−' indicate that no solution was found within 8 hours. Furthermore, the table lists the number of *joint* types in the BGs constructed for the last stage without clustering, $|BG^t|$, and with, $|cBG^t|$. The former is constant while the latter is an average, as the algorithm can form BGs for different past policies, leading to clusterings of different sizes. For the Dec-Tiger problem, the solution time needed by GMAA*-Cluster is more than 3 orders of magnitude less for horizon $h = 4$. For $h = 5$ this test problem has $3.82e29$ joint policies. To our knowledge, no other method has been able to optimally solve $h = 5$ Dec-Tiger. GMAA*-Cluster, however, is able to solve Dec-Tiger for $h = 5$ in reasonable time.

For the FireFighting problem, no lossless clustering is possible at any stage, and as such, we incur some overhead for the clustering. This is clearly shown for $h = 4$. For horizon 3, GMAA*-Cluster is actually a bit faster. Analysis revealed that for this horizon the cost of attempting to cluster is negligible. GMAA*-Cluster is faster because constructing the BGs using bootstrapping from the previous BG takes less time than constructing a BG from scratch.

For GridSmall, Cooperative Box Pushing, and Hotel 1 we see results comparable to those for Dec-Tiger: substantial clustering is possible, resulting in significant speedups. Because the solution of BGs takes time exponential in their size, even small reductions in size yield a big increase in efficiency. Therefore, the substantial amounts of clustering found in these problems, allow optimal solutions for longer horizons than have have been presented before.

For BroadcastChannel, GMAA*-Cluster achieves an even more dramatic increase in performance, allowing the solving of up to horizon $h = 25$. Analysis reveals that the BGs constructed for all stages are fully clustered: they contain only one type for each agent. Consequently, the time needed to solve each BG does not grow with the horizon. Total time, however, still increases super-linearly due to more backtracking. The Recycling Robots problem can also be clustered to a relatively constant number of approximately 9 joint types per stage, allowing for optimal solving to high horizons. Both the BroadcastChannel and Recycling Robots problem run out of (2GB of) memory for higher horizons.

Note that the results reported here are a vast improvement

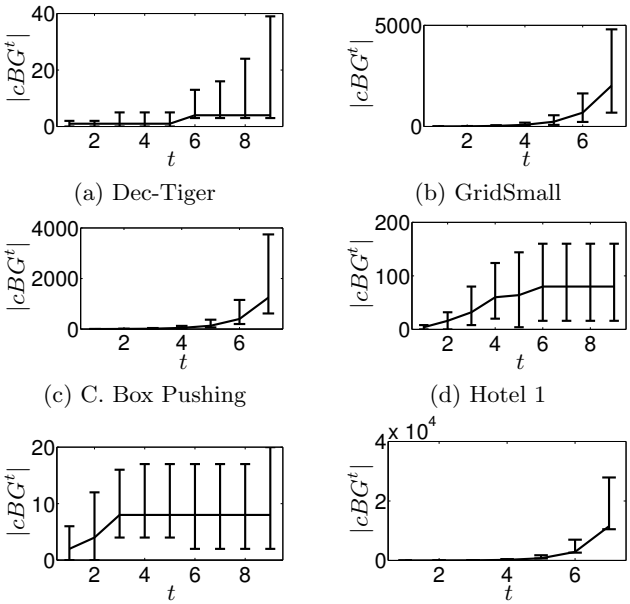
Dec-Tiger (Q_{BG})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	−4.0000	≤ 0.01	≤ 0.01	4	4.00
3	5.1908	0.02	≤ 0.01	16	9.00
4	4.8028	3,069.4	1.50	64	23.14
5	7.0265	–	130.82	256	40.43
BroadcastChannel (Q_{MDP})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	2.0000	≤ 0.01	≤ 0.01	4	1.00
3	2.9900	≤ 0.01	≤ 0.01	16	1.00
4	3.8900	3.22	≤ 0.01	64	1.00
5	4.7900	–	≤ 0.01	256	1.00
6	5.6900	–	≤ 0.01	1024	1.00
7	6.5900	–	≤ 0.01	4096	1.00
8	7.4900	–	≤ 0.01	16384	1.00
9	8.3900	–	≤ 0.01	65536	1.00
10	9.2900	–	≤ 0.01	$2.62e5$	1.00
15	13.7900	–	≤ 0.01	$2.68e8$	1.00
20	18.3132	–	0.08	$2.75e11$	1.00
25	22.8815	–	1.67	$2.81e14$	1.00
GridSmall (Q_{BG})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	0.9100	≤ 0.01	≤ 0.01	4	4.00
3	1.5504	4.21	0.71	16	12.00
4	2.2416	–	30.17	64	25.00
Cooperative Box Pushing (Q_{MDP})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	17.6000	0.05	≤ 0.01	25	4.00
3	66.0810	–	4.55	625	25.00
Recycling Robots (Q_{MDP})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	6.8000	≤ 0.01	≤ 0.01	4	4.00
3	9.7647	0.02	≤ 0.01	16	9.00
4	11.7264	23052.5	0.02	64	8.67
5	13.7643	–	0.10	256	9.00
6	15.5760	–	0.19	1024	9.00
7	17.2126	–	0.67	4096	9.00
8	18.6839	–	1.28	16384	9.00
9	20.0085	–	2.72	65536	9.00
10	21.2006	–	4.92	$2.62e5$	9.00
11	22.2734	–	9.83	$1.05e6$	9.00
12	23.2390	–	17.11	$4.19e6$	9.00
13	24.1080	–	30.61	$1.68e7$	9.00
14	24.8901	–	50.12	$6.71e7$	9.00
15	25.5940	–	81.46	$2.68e8$	9.00
Hotel 1 (Q_{BG})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	9.5000	≤ 0.01	0.02	16	4.00
3	15.7047	–	0.07	256	16.00
4	20.1125	–	1.37	4096	32.00
FireFighting $\langle n_h = 3, n_f = 3\rangle$ (Q_{BG})					
h	V^*	T_{GMAA^*}(s)	$T_{cluster}$(s)	$\lvert BG^t\rvert$	$\lvert cBG^t\rvert$
2	−4.3825	0.03	0.03	4	4.00
3	−5.7370	0.91	0.70	16	16.00
4	−6.5789	5605.3	5823.5	64	64.00

Table 1: Results of GMAA* on several problems. Listed are the run times of regular GMAA* and GMAA*-Cluster, and the size of the BGs solved at each time step, with and without clustering.

over the current state-of-the-art methods for optimally solving Dec-POMDPs. For instance, reported results for policy compression [5] solve the Dec-Tiger and BroadcastChannel problems only up to horizon 4 (in 534s resp. 4.59s).

6.2 General clustering performance

The reduction in BG-size in GMAA*-Cluster leads to significant gains in efficiency, showing that heuristically high-ranked partial policies allow for much clustering. We also investigated the general applicability of our clustering method



D-T (a)	GS (b)	CBP (c)	H1 (d)	RR (e)	FF (f)
65,536	8.127	$4.90e6$	$8.59e8$	32,768	1.42

(g) Median clustering ratios $\frac{|BG^t|}{|cBG^t|}$ for last time step.

Figure 1: Empirical clustering performance given random joint policies, for several problems, based on 1,000 independent samples. Plots (a)–(f) show the median size of the Bayesian games at each stage after clustering $|cBG^t|$, and the errorbars show the 0.25 and 0.75-quantile. Table (g) shows their median clustering ratio $\frac{|BG^t|}{|cBG^t|}$ for the last time step tested.

by testing how much clustering can be done in BGs constructed for random past policies. If substantial clustering is possible on random policies, not just those considered by GMAA*-Cluster, then our approach may be useful for a much broader set of methods. The results are shown in Figure 1, which for different stages shows the median number of joint types $|cBG^t|$ in the Bayesian games (constructed for 1,000 random past policies) after clustering.

The FireFighting problem, which could not be clustered when searching for an optimal policy, does allow for some clustering given randomly selected policies (Figure 1(g)). In both the Recycling Robots and the Hotel 1 problem the growth in BG size appears to stabilize, while in Dec-Tiger, GridSmall, and Cooperative Box Pushing $|cBG^t|$ keeps growing in the planning horizon. Even so, $|BG^t|$ grows faster, resulting in high clustering ratios for these problems also.

These experiments imply that our clustering technique can provide significantly smaller policy representations without loss of value at a relatively low computational cost, for the benefit of optimal and approximate algorithms alike. Also this technique gives insight into how many future policies an agent should consider: if at some stage and given a past policy an agent has only k types, this means that it maximally needs to consider k future policies from that situation. Some state-of-the-art approximate Dec-POMDP solvers (e.g. IMBDP [15, 6]) have a parameter controlling

the number of future policies considered, but until now there has been no principled way of estimating good values for this parameter. As such, we expect that this clustering technique can have a substantial impact on new and existing, exact and approximate algorithms.

6.3 Discussion

Our empirical results show that lossless clustering offers dramatic performance gains on a diverse set of problems. However, since some domains cannot be clustered in this way, it remains unclear in exactly what types of problems lossless clustering is effective. This is a hard question, as it requires an analysis of the subclasses of Dec-POMDPs, a matter about which relatively few results are known. Most research has focused on analysis of methods, rather than of properties of Dec-POMDP problems, notable exceptions being [14, 9]. Although a detailed analysis is beyond the scope of this paper, we offer some observations based on our empirical results.

As noted before, the BroadcastChannel problem exhibits full clustering, which we can explain as follows. When constructing a BG for $t = 1$, there is only one joint type for the previous BG, so we know what the joint action was. The crucial property of BroadcastChannel is that the (joint) observation tells us nothing about the new state, but only about what joint action was taken (e.g., 'collision' if both agents chose to 'send'). As a result, the observation does not convey any information and the different individual histories can be clustered. In a BG constructed for stage $t = 2$, there will again be only one joint type in the previous game. Therefore, given the past policy, the actions of the other agents can be perfectly predicted. Again the observation will convey no information so this process repeats. Consequently, the problem can be considered a special form of a non-observable Dec-POMDP; lossless clustering automatically exploits this property.

In the FireFighting and GridSmall little or no clustering was possible. Analysis revealed that given most (and especially given sensible) joint policies, each observation history implicated a different belief over states violating (5). For longer horizons it is less likely that every history induces such a different belief and we expect more clustering, which is confirmed by Figure 1(g).

The other problems are harder to analyze. In Dec-Tiger a key property is that opening the door resets the problem. Such resets invalidate the history, allowing for clustering. Another factor is that the observations are conditionally independent given only the new state. I.e., $P(o|a,s') = P(o_1|s')P(o_2|s')$, which means that all information regarding the history of the other agent is obtained through estimation of the state.

Of course, the criterion for clustering is quite strict and there will also be many problems in which little or no lossless clustering is possible. In the future, we plan to consider approximations for such cases. In particular, one idea is to cluster approximately PE histories, e.g., if Kullback-Leibler divergence is below some threshold. Another idea is to cluster histories that induce the same *individual belief* over states:

$$P(s|\vec{\theta}_i) = \sum_{\vec{\theta}_{\neq i}} P(s,\vec{\theta}_{\neq i}|\vec{\theta}_i). \tag{13}$$

Such individual beliefs literally summarize the criterion and may therefore perform quite well in practice. Further investigation is needed to determine for which classes of problems such approximations might work.

7. CONCLUSIONS

This paper introduced a method for lossless clustering of action-observation histories in Dec-POMDPs, which can be applied in GMAA* policy search for Dec-POMDPs via Bayesian games. Rather than applying an ad-hoc clustering of these BGs, we identified a probabilistic equivalence criterion that guarantees that, given a particular past joint policy φ^t, two action-observation histories $\vec{\theta}_i^t$ of agent i at stage t have the same optimal Q-values and therefore can be clustered without loss in solution quality. Empirical evaluation of GMAA* demonstrated that for several domains speedups of multiple orders of magnitude are achieved by clustering. We also investigated the amount of clustering possible for random past policies φ^t, the result of which suggests that our clustering methods may also be exploited in other algorithms, such as IMBDP [15]. As such, we expect that the proposed clustering method may have a significant impact on both exact and approximate solutions of Dec-POMDPs.

APPENDIX

Here we provide (sketches of) the proofs.

PROOF OF LEMMA 1. Assume an arbitrary $a_i^t, o_i^{t+1}, \beta_{\neq i}^t, s^{t+1}$ and $\vec{\theta}_{\neq i}^{t+1} = (\vec{\theta}_{\neq i}^t, a_{\neq i}^t, o_{\neq i}^{t+1})$. We have that

$$P(s^{t+1},\vec{\theta}_{\neq i}^{t+1},o_i^{t+1}|\vec{\theta}_{i,a}^t,a_i^t,\beta_{\neq i}^t) = \sum_{s^t} P(o_i^{t+1},o_{\neq i}^{t+1}|a_i^t,a_{\neq i}^t,s^{t+1})$$
$$P(s^{t+1}|s^t,a_i^t,a_{\neq i}^t)P(a_{\neq i}^t|\vec{\theta}_{\neq i}^t,\beta_{\neq i}^t)P(s^t,\vec{\theta}_{\neq i}^t|\vec{\theta}_{i,a}^t)$$
$$= P(s^{t+1},\vec{\theta}_{\neq i}^{t+1},o_i^{t+1}|\vec{\theta}_{i,b}^t,a_i^t,\beta_{\neq i}^t)$$

Because we assumed an arbitrary $s^{t+1},\vec{\theta}_{\neq i}^{t+1},o_i^{t+1}$ it holds for all, which means we can conclude

$$P(s^{t+1},\vec{\theta}_{\neq i}^{t+1}|\vec{\theta}_{i,a}^t,a_i^t,o_i^{t+1},\beta_{\neq i}^t) = P(s^{t+1},\vec{\theta}_{\neq i}^{t+1}|\vec{\theta}_{i,b}^t,a_i^t,o_i^{t+1},\beta_{\neq i}^t)$$

Finally, because $a_i^t, o_i^{t+1}, \beta_{\neq i}^t, s^{t+1}, \vec{\theta}_{\neq i}^{t+1}$ were all arbitrarily chosen we can conclude (6). □

PROOF OF THEOREM 2. We show that the expected value of any joint policy $(\beta_i,\beta_{\neq i})$ that satisfies condition (7) is the same in both B and B'.

$$V(\beta_i,\beta_{\neq i}) = \sum_\theta P(\theta)u(\theta,\beta(\theta)),$$
$$= \sum_{\theta_{\neq i}}\sum_{\theta_i} P(\theta_i,\theta_{\neq i})u(\langle\theta_i,\theta_{\neq i}\rangle,\langle\beta_i(\theta_i),\beta_{\neq i}(\theta_{\neq i})\rangle),$$

using short-hand $a = \langle\beta_i(\theta_i),\beta_{\neq i}(\theta_{\neq i})\rangle$, $V^B(\beta_i,\beta_{\neq i})$

$$= \sum_{\theta_{\neq i}}\Bigg[\overbrace{P(\theta_i^a,\theta_{\neq i})u(\theta_i^a,\theta_{\neq i},a) + P(\theta_i^b,\theta_{\neq i})u(\theta_i^b,\theta_{\neq i},a)}^{\left(P(\theta_i^a,\theta_{\neq i})+P(\theta_i^b,\theta_{\neq i})\right)u(\theta_i^c,\theta_{\neq i},a)}$$
$$+ \sum_{\theta_i\in\Theta_i\setminus\{\theta_i^a,\theta_i^b\}} P(\theta_i,\theta_{\neq i})u(\theta_i,\theta_{\neq i},a)\Bigg]$$
$$= \sum_{\theta_{\neq i}}\Big[P(\theta_i^c,\theta_{\neq i})u(\langle\theta_i^c,\theta_{\neq i}\rangle,a) + \ldots\Big] = V^{B'}(\beta_i,\beta_{\neq i})$$

which is the expected value of $(\beta_i,\beta_{\neq i})$ as computed in the reduced BG. □

PROOF OF LEMMA 4. The proof is by induction. The base case is given by the last stage $t = h - 1$ of the Dec-POMDP. In this case we have that

$$\forall_a \forall_{\vec{\theta}^t_{\neq i}} \quad Q^*(\vec{\theta}_{i,a},\vec{\theta}_{\neq i},a) = \sum_{s\in\mathcal{S}} R(s,a)P(s|\vec{\theta}_{\neq i},\vec{\theta}_{i,a}) = \sum_{s\in\mathcal{S}} R(s,a)P(s|\vec{\theta}_{\neq i},\vec{\theta}_{i,b}) = Q^*(\vec{\theta}_{\neq i},\vec{\theta}_{i,b},a)$$

because of (5) in 1. For stages $0 \leq t < h - 1$ the optimal Q-value function is given by

$$Q^*(\vec{\theta}^t,a) = R(\vec{\theta}^t,a) + \sum_{o^{t+1}\in\mathcal{O}} P(o^{t+1}|\vec{\theta}^t,a)Q^*(\vec{\theta}^{t+1},\pi^*(\vec{\theta}^{t+1})).$$

The induction hypothesis is as follows: If at $t+1$ the criteria hold for any two $\vec{\theta}^{t+1}_{i,a},\vec{\theta}^{t+1}_{i,b}$, then they have equal Q-values:

$$\forall_{\vec{\theta}^{t+1}_{\neq i}} \forall_{a^{t+1}} \quad Q^*(\vec{\theta}^{t+1}_{i,a},\vec{\theta}^{t+1}_{\neq i},a^{t+1}) = Q^*(\vec{\theta}^{t+1}_{i,b},\vec{\theta}^{t+1}_{\neq i},a^{t+1}). \tag{14}$$

Assume some stage $0 \leq t < h - 1$. Assume that the criteria hold for $\vec{\theta}^t_{i,a},\vec{\theta}^t_{i,b}$. Assume an arbitrary $a = \langle a_i,a_{\neq i}\rangle$ and $\vec{\theta}^t_{\neq i}$. Now we need to show that

$$R(\vec{\theta}^t_{i,a},\vec{\theta}^t_{\neq i},a) + \sum_{o^{t+1}\in\mathcal{O}} P(o^{t+1}|\vec{\theta}^t_{i,a},\vec{\theta}^t_{\neq i},a)Q^*(\vec{\theta}^{t+1}_a,\pi^*(\vec{\theta}^{t+1}_a)) = R(\vec{\theta}^t_{i,b},\vec{\theta}^t_{\neq i},a) + \sum_{o^{t+1}\in\mathcal{O}} P(o^{t+1}|\vec{\theta}^t_{i,b},\vec{\theta}^t_{\neq i},a)Q^*(\vec{\theta}^{t+1}_b,\pi^*(\vec{\theta}^{t+1}_b)) \tag{15}$$

To prove the equality of (15), we have to show that: 1) The immediate rewards are equal. This clearly is the case (similar to the proof of the last stage). 2) Equal observation probabilities. This is also evident given that the criterion holds. (if the underlying state distribution is the same the next joint observation probabilities are also identical.) 3) The relevant next-stage Q-values are identical. I.e.:

$$\forall_{o^{t+1}} \forall_{a^{t+1}} \quad Q^*(\vec{\theta}^{t+1}_a,a^{t+1}) = Q^*(\vec{\theta}^{t+1}_b,a^{t+1}). \tag{16}$$

To prove this, we show that the identically extended histories are PE, and that therefore the induction hypothesis applies: We can rewrite the demonstrandum (16) to

$$\forall_{o^{t+1}_i} \forall_{o^{t+1}_{\neq i}} \forall_{a^{t+1}} \quad Q^*(\vec{\theta}^{t+1}_{i,a} = (\vec{\theta}^t_{i,a},a_i,o^{t+1}_i),\vec{\theta}^{t+1}_{\neq i},a^{t+1}) = Q^*(\vec{\theta}^{t+1}_{i,b} = (\vec{\theta}^t_{i,b},a_i,o^{t+1}_i),\vec{\theta}^{t+1}_{\neq i},a^{t+1}).$$

This is proven (by application of the induction hypothesis) if we can show that the criterion holds for $\vec{\theta}^{t+1}_{i,a},\vec{\theta}^{t+1}_{i,b}$. Since $\vec{\theta}^{t+1}_{i,a},\vec{\theta}^{t+1}_{i,b}$ are identical extensions of PE histories $\vec{\theta}^t_{i,a},\vec{\theta}^t_{i,b}$, they themselves are PE by application of Lemma 1. □

Acknowledgments

We would like to thank Alan Carlin and Christopher Amato for making available the Recycling Robots and Cooperative Box Pushing domains. The research reported here is part of the Interactive Collaborative Information Systems (ICIS) project, supported by the Dutch Ministry of Economic Affairs, grant nr: BSIK03024. This work was partially supported by Fundação para a Ciência e a Tecnologia (ISR/IST pluriannual funding) through the POS_Conhecimento Program that includes FEDER funds and through grant PTDC/EEA-ACR/73266/2006.

A. REFERENCES

[1] C. Amato, D. S. Bernstein, and S. Zilberstein. Optimal fixed-size controllers for decentralized POMDPs. In *Multi-Agent Sequential Decision Making in Uncertain Domains (AAMAS Workshop)*, May 2006.

[2] C. Amato, D. S. Bernstein, and S. Zilberstein. Optimizing memory-bounded controllers for decentralized POMDPs. In *Proc. of Uncertainty in Artificial Intelligence*, July 2007.

[3] R. Aras, A. Dutech, and F. Charpillet. Mixed integer linear programming for exact finite-horizon planning in decentralized POMDPs. In *Int. Conf. on Automated Planning and Scheduling*, 2007.

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

[5] A. Boularias and B. Chaib-draa. Exact dynamic programming for decentralized POMDPs with lossless policy compression. In *Int. Conf. on Automated Planning and Scheduling*, 2008.

[6] A. Carlin and S. Zilberstein. Value-based observation compression for DEC-POMDPs. In *AAMAS*, 2008.

[7] R. Emery-Montemerlo, G. Gordon, J. Schneider, and S. Thrun. Approximate solutions for partially observable stochastic games with common payoffs. In *AAMAS*, 2004.

[8] R. Emery-Montemerlo, G. Gordon, J. Schneider, and S. Thrun. Game theoretic control for robot teams. In *IEEE Int. Conf. on Robotics and Automation*, 2005.

[9] C. V. Goldman and S. Zilberstein. Decentralized control of cooperative systems: Categorization and complexity analysis. *Journal of Artificial Intelligence Research*, 22, 2004.

[10] E. A. Hansen, D. S. Bernstein, and S. Zilberstein. Dynamic programming for partially observable stochastic games. In *AAAI*, 2004.

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

[12] F. A. Oliehoek, M. T. J. Spaan, and N. Vlassis. Optimal and approximate Q-value functions for decentralized POMDPs. *Journal of Artificial Intelligence Research*, 32, 2008.

[13] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. The MIT Press, July 1994.

[14] D. V. Pynadath and M. Tambe. Multiagent teamwork: Analyzing the optimality and complexity of key theories and models. In *AAMAS*, 2002.

[15] S. Seuken and S. Zilberstein. Improved memory-bounded dynamic programming for decentralized POMDPs. In *Proc. of Uncertainty in Artificial Intelligence*, July 2007.

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

[17] D. Szer, F. Charpillet, and S. Zilberstein. MAA*: A heuristic search algorithm for solving decentralized POMDPs. In *Proc. of Uncertainty in Artificial Intelligence*, 2005.