Universiteit
van
Amsterdam

IAS technical report IAS-UVA-06-02

Dec-POMDPs and extensive form games: equivalence of models and algorithms

Frans Oliehoek and Nikos Vlassis
Intelligent Systems Laboratory Amsterdam,
University of Amsterdam
The Netherlands

In this report we relate different methods for Dec-POMDPs and, more general, POSGs and extensive form games. In particular we show that every POSG can be modeled as an extensive form game. We discuss how one can directly calculate best-response policy in such extensive form games and relate this to existing methods for Dec-POMDPs.

Keywords: Multi-agent systems, Dec-POMDPs, POSGs, planning.

IAS
intelligent autonomous systems

Contents

Intelligent Autonomous Systems
Informatics Institute, Faculty of Science
University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam
The Netherlands
Tel (fax): +31 20 525 7461 (7490)
http://www.science.uva.nl/research/ias/

Corresponding author:
F.A. Oliehoek
tel: +31 20 525 7293
faolieho@science.uva.nl
http://www.science.uva.nl/~faolieho/

1 POSGs

In this report we will treat methods and models for sequential multi-agent decision making under uncertainty. The problem in this setting is that there are multiple decision makers, or *agents*, who can only observe a part of the world they are located in and have to select their actions at different time-steps in order to either reach some goal, minimize some cost or to optimize some payoff.

In this section we treat two models for sequential multi-agent decision making under uncertainty: the *Decentralized partially observable Markov decision process (Dec-POMDP)* model and its generalization the *partially observable stochastic game (POSG)*. Both models do not allow for explicit communication, but there are extensions for the Dec-POMDP that do [5, 6, 17, 22].

1.1 Formal model

The *partially observable stochastic game (POSG)* is the most general framework for partially observable multi-agent systems (MASs) without explicit communication.

Definition 1.1 A POSG is defined as a tuple $\langle \mathcal{S}, \mathcal{A}, T, R, \mathcal{O}, O, h, b^{t=0} \rangle$ where:

- There is a finite set of m agents.
- $\mathcal{S}$ 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 $\mathbf{a} = \langle a_1, ..., a_m \rangle$ is taken. Agents do not observe each other's actions.
- T is the transition function, a mapping from states and joint actions to probability distributions over states: $T : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$.[1]
- R=$\langle R_1, ..., R_m \rangle$ where R_i is the individual reward function for agent i, a mapping from states, joint actions and successor states to real numbers: $R_i : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$. Thus the joint reward function, R, specifies a vector of m real numbers: $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}^m$.
- $\mathcal{O} = \times_i \mathcal{O}_i$ is the set of joint observations, where $\mathcal{O}_i$ is a finite set of observations available to agent i. Every time-step, a joint observation $\mathbf{o} = \langle o_1, ..., o_m \rangle$ is from $\mathcal{O}$ is emitted, each agent i only observes his own component o_i of this joint observation.
- O is the observation function, a mapping from states, joint actions and successor states to probability distributions over joint observations: $O : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathcal{P}(\mathcal{O})$.
- h is the horizon of the problem.
- $b^{t=0} \in \mathcal{P}(\mathcal{S})$, also denoted b^0, is the initial state distribution and is optional.

In a POSG, the goal of an agent is to maximize the expected (discounted) future reward. Therefore the planning problem is to find a conditional plan or *policy* for each agent as to maximize its expected (discounted) future reward.

When all the payoff functions are identical, $\forall_{i,j,s,\mathbf{a}}\ R_i(s, \mathbf{a}) = R_j(s, \mathbf{a})$, we refer to the model as a *partially observable identical payoff stochastic game (POIPSG)* or a *decentralized POMDP (Dec-POMDP)*. In this case we simply write $R(s, \mathbf{a})$.[2]

We use the notation $\mathbf{a}_{\neq i} = \langle a_1, ..., a_{i-1}, a_{i+1}, ..., a_m \rangle$ and $\mathbf{o}_{\neq i} = \langle o_1, ..., o_{i-1}, o_{i+1}, ..., o_m \rangle$ to denote a tuple of respectively actions and observations for all agents but i.

[1] We use $\mathcal{P}(X)$ to denote the infinite set of probability distributions over the finite set X.

[2] So in this case each agent receives a payoff of $R(s, \mathbf{a})$. However, one can also think of one reward $R(s, \mathbf{a})$ that is split equally.

1.2 Histories, sequences and policies

As mentioned above, the goal of an agent in a POSG or Dec-POMDP is to maximize its expected discounted future reward and to do that he has to select a good or optimal policy. This is a conditional plan of what action to perform in what circumstances. Clearly an agent can only condition his plan on what he knows or observes, i.e. on the history. Here we will first formalize two different notions of history.

Definition 1.2 We define *the action-observation history for agent i,* $\vec{\theta}_i$, as the sequence of actions taken by and observations received by agent i. At a specific time-step t, this is:

$$\vec{\theta}_i^t = \left(o_i^0, a_i^0, o_i^1, a_i^1, ..., a_i^t, o_i^t\right).$$

The *joint action-observation history,* $\vec{\theta}$, is the action-observation history for all agents:

$$\vec{\theta}^t = \left\langle \vec{\theta}_1^t, ..., \vec{\theta}_m^t \right\rangle.$$

The set of possible action-observation histories for agent i at time t is $\vec{\Theta}_i^t = \times_t(\mathcal{O}_i \times \mathcal{A}_i)$. The set of all possible action-observation histories for agent i is $\vec{\Theta}_i = \cup_{t=0}^{h-1}\vec{\Theta}_i^t$. Finally the set of all possible joint action-observation histories is given by $\vec{\Theta} = \cup_{t=0}^{h-1}(\vec{\Theta}_1^t \times ... \times \vec{\Theta}_m^t)$.

The action-observation history of an agent corresponds to everything the agent knows. The joint action-observation history corresponds to everything the agents know together. In a POSG, each time-step consists of a state, a joint observation and joint action. So a joint action-observation history specifies the full history of the process except for the states.

We will now look at the second notion of history. This second notion doesn't include the agents' actions.

Definition 1.3 Formally, we define *the observation history for agent i,* $\vec{o}_i$, as the sequence of observations an agent has received. At a specific time-step t, this is:

$$\vec{o}_i^t = \left(o_i^0, o_i^1, ..., o_i^t\right).$$

The *joint observation history,* $\vec{\mathbf{o}}$, is the action-observation history for all agents:

$$\vec{\mathbf{o}}^t = \left\langle \vec{o}_1^t, ..., \vec{o}_m^t \right\rangle.$$

The set of observation histories for agent i at time t is denoted $\vec{\mathcal{O}}_i^t = \times_t \mathcal{O}_i$. Similar to above we also use $\vec{\mathcal{O}}_i$ and $\vec{\mathcal{O}}$.

Now we can formalize the notion of policy. We will start with the simplest case.

Definition 1.4 A *pure-* or *deterministic policy,* π_i, for agent i is a mapping from action-observation histories to actions, $\pi_i : \vec{\Theta}_i \to \mathcal{A}_i$.

Note that when an agent takes its action deterministically, he will be able to infer what action he took from only the observation history. I.e. when an agent takes its actions according to a pure policy, there are other actions he will never take. This means that most of the observation-action histories will never be realized. Therefore it is possible to replace a pure policy by a mapping from observation histories to actions: $\pi_i : \vec{\mathcal{O}}_i \to \mathcal{A}_i$. We will describe this in more detail in section 3.3.

It is also possible for agents to use randomized policies that allow taking an action in some situation with some probability. There are two types of randomized policies:

Definition 1.5 A *stochastic policy*, π_i, for agent i is a mapping from action-observation histories to probability distributions over actions, $\pi_i : \vec{\Theta}_i \rightarrow \mathcal{P}(\mathcal{A}_i)$.

Definition 1.6 A *mixed policy*, μ_i, for agent i is a non-empty set of policies, Π_i, together with a probability distribution over it: $\mu_i : \mathcal{P}(\Pi_i)$. In general the set Π_i can contain any type of policies, but unless stated otherwise we will assume it contains only pure policies.

Similar to previous notation we use $\pi = \langle \pi_1, ..., \pi_m \rangle$ to denote a *joint policy.* Also we use $\pi_{\neq i}$, $\vec{\theta}_{\neq i}$, etc. to denote a tuple of policies, action-observation histories, etc. for all agents except i.

1.3 The 1 agent case: POMDPs

When there is only one agent, a POSG or Dec-POMDP reduces to a regular POMDP. Regular POMDPs have received quite some attention and as a consequence there are some well-known results.

In particular, in a POMDP it is possible to maintain a probability distribution over states, called a *belief* $b \in \mathcal{P}(\mathcal{S})$, instead of remembering the full action-observation history, because such a belief is a sufficient statistic with respect to future rewards. After taking an action and receiving an observation the belief is updated using Bayes rule.

A consequence of this is that a policy is no longer defined as a mapping from action-observation histories to actions, but instead as a mapping from beliefs to actions. Effectively this means that a POMDP can be converted to an MDP over belief states, as we summarize here. Let $\tau = h - t$ denote the number of *time-steps-to-go,* then the standard POMDP Bellman backup is:

$$V^{\tau+1}(b) = \max_{a \in \mathcal{A}} \left[R(b,a) + \gamma \sum_{o \in \mathcal{O}} P(o|a,b) V^{\tau}(b_a^o) \right],$$

$$b_a^o(s') = \frac{P(o|s',a) \sum_{s \in \mathcal{S}} P(s'|s,a) b(s)}{P(o|a,b)}, \tag{1.1}$$

where

$$P(o|a,b) = \sum_{s' \in \mathcal{S}} P(o|s',a) \sum_{s \in \mathcal{S}} P(s'|s,a) b(s). \tag{1.2}$$

Writing this in time-steps t (vs. 'time-to-go') as used in most places in this report, this is:

$$V^{t}(b) = \max_{a \in \mathcal{A}} \left[R(b,a) + \gamma \sum_{o \in \mathcal{O}} P(o|a,b) V^{t+1}(b_a^o) \right].$$

The expected immediate reward for a belief $R(b,a)$ is given by:

$$R(b,a) = \sum_{s \in \mathcal{S}} R(s,a) b(s).$$

Clearly, this is very similar to the definition of a value function of a regular MDP. The problem here is that one can not directly apply value iteration over the continuous belief space, therefore specialized techniques are required [7, 2]. Moreover, this is intractable for all but the smallest problems and therefore approximating methods are required [19, 12].

a	$s_l \rightarrow s_{lR}$	$s_l \rightarrow s_{lL}$	$s_r \rightarrow s_{rR}$	$s_r \rightarrow s_{rL}$	$s_l/s_r \rightarrow s_E$	$s_{lR}/s_{lL}/s_{rR}/s_{rL} \rightarrow s_E$
$\langle a_{Le}, a_F \rangle$	0	1.0	0	1.0	0	n/a
$\langle a_{Ri}, a_F \rangle$	1.0	0	1.0	0	0	n/a
$\langle *, a_Q \rangle$	0	0	0	0	1.0	1.0
$\langle a_Q, * \rangle$	n/a	n/a	n/a	n/a	n/a	1.0
$\langle a_O, a_O \rangle$	n/a	n/a	n/a	n/a	n/a	1.0

Table 1: Transition model for the deaf, the blind and the tiger problem. Not all actions are available at all both time-steps, indicated with n/a. * is a wild-card denoting any action.

2 The deaf, the blind and the tiger

We will introduce a very small problem to illustrate different methods for finding good or optimal policies for Dec-POMDPs. Because we will also show the extensive form of this problem, it has been constrained to make it representable on one page, as a consequence the problem looks slightly artificial. We emphasize that the analysis also holds for more natural (thus larger) problems.

Example 2.1 *The deaf, the blind and the tiger.* This is a variation on the Dec-tiger problem [11]. There are two agents, one deaf and one blind, who can't observe each other's actions. They are located in a labyrinth in which there are two doors. We will call these door 'left' (l) and 'right' (r), but they are assumed to be located in arbitrary places. The doors are heavy and can only be opened by the two agents simultaneously. Behind one of these is a treasure, behind the other a tiger. Agent 1 is good in navigating but deaf, the agent 2 has good ears, but can't navigate as he's blind. The goal of the agents is to open to door to the treasure. To accomplish this goal both agents can select an action at two time-steps.

In the first time-step, agent 1 has the choice to go to door 'left' or to door 'right'. At the same time agent 2 will have to decide if he wants to 'follow' agent 1 or 'quit'. If he leaves, the problem ends, as they will not be able to achieve their goal. Because they wasted their time and split up, leading to a quarrel, they receive a payoff of -2. If agent 2 follows, they will arrive at the door selected by agent 1 with certainty. The cost of this travel is -0.1. On arrival at the selected door agent 1 will knock on the door to provoke the tiger that is potentially behind this door to roar.

In the second time-step, again both have to select an action, however, agent 2 now can also make an observation: either he hears a tiger roaring or not, but this observation is noisy. So at this point, agent 2 knows whether he heard a roar and agent 1 knows which door they are standing in front of. Both have to decide whether they 'open' or 'quit'. The door can only be opened if both agent select 'open', in which case they receive a reward of $+10$ or -10 depending on whether they found the tiger or the treasure. □

The formal Dec-POMDP model of consists of six states plus an end-state s_E. s_l, s_r are the initial states in which the tiger is behind the left and right door and the agents have not navigated to either door yet. Their probabilities (b^0 — the initial belief) are $P(s_l) = 0.55$ and $P(s_r) = 0.45$. In these states the first agent's actions are to navigate to the left door (a_{Le}) or the right door (a_{Ri}), leading to four possible successor states s_{lL}, s_{lR}, s_{rR} and s_{rL}, where the capital letter denotes the door at which the agents are located and the lowercase letter denotes the door behind which the tiger is located. Agent 2's 'follow' action is denoted a_F, other actions are 'quit' (a_Q) and 'open' (a_O). Table 1 shows the transition model.

s	$\mathbf{a}$	s'	$\langle o_\emptyset, o_{Ro}\rangle$	$\langle o_\emptyset, o_{Si}\rangle$	$\langle o_\emptyset, o_\emptyset\rangle$
*	*	s_{lL}	0.85	0.15	0
*	*	s_{rR}	0.7	0.3	0
*	*	s_{lR}	0.03	0.97	0
*	*	s_{rL}	0.03	0.97	0
*	*	*	0	0	1

Table 2: The deaf, the blind and the tiger observation model. * is a wild-card denoting any state or action, such that there is no overlap with an earlier specified $s, \mathbf{a}, s'$ triple.

$\mathbf{a}$	s_l	s_r	s_{lL}	s_{rR}	s_{lR}	s_{rL}
$\langle a_O, a_O\rangle$	n/a	n/a	-10	-10	$+10$	$+10$
$\langle a_Q, a_Q\rangle$	n/a	n/a	-1	-1	-1	-1
$\langle a_Q, a_O\rangle$	n/a	n/a	-2	-2	-2	-2
$\langle *, a_Q\rangle$	-2	-2	-2	-2	-2	-2
$\langle *, a_F\rangle$	-0.1	-0.1	n/a	n/a	n/a	n/a

Table 3: The reward function for the deaf, the blind and the tiger. * is a wild-card denoting any action, such that there is no overlap with any earlier specified joint action.

Apart from states and transitions we also need to specify the observation and reward model. Only agent 2 can make an actual observation in the second time-step, he either hears a roar (o_{Ro}) or silence (o_{Si}). In all other cases the agents receive no-observation ($o_\emptyset$). The two doors and rooms have different isolating properties, so the probability of $P(o_{Ro} \mid s_{rR})$ is different from $P(o_{Ro} \mid s_{lL})$. We assume that observing a roar when the tiger is behind the other door is entirely due to mental pressure and therefore $P(o_{Ro} \mid s_{lR}) = P(o_{Ro} \mid s_{rL})$. The observation and reward model are shown in table 2 and table 3.

3 Extensive form representation

We will now introduce the extensive form representation of a POSG and illustrate it for the deaf and the blind problem. We start with this, because it gives a good intuition of the problem and because it allows for the most straightforward solution methods: normal form and sequence form solving, which we will also discuss here.

3.1 Extensive form games

An extensive form game is given by a tree, in which nodes represent what (chance) moves have been taken and whose root is the start of the game. There are two types of non-terminal nodes: decision nodes for agents, that represent points at which agents can make a move, and chance nodes which represent stochastic transitions. The latter are modeled as decision nodes for the special player 'nature'. Terminal nodes, or *outcome* nodes are the leaves of the game-tree. These specify the payoff for each agent.

In a partial information game, an agent may be uncertain about the true state of the game. This is reflected by the fact that an agent may not be able to discriminate between some nodes in the tree. Such groups of nodes in which the agent has the same information regarding the state are called *information sets*. Formally we define an extensive form as follows:

Definition 3.1 An *extensive form game* is a tuple $G_{e.f.} = \langle \mathcal{N}, E, \mathcal{I}, I, O, n^0 \rangle$, where:

- There is a set of m players or agents. We use i with $1 \leq i \leq m$ to index these. The special agent 'nature' is indexed with $i = 0$.
- $\mathcal{N} = \bigcup_{i=0}^{m} \mathcal{N}_i^d \cup \mathcal{N}^o$ is the set of all nodes. $\mathcal{N}_i^d$ is the set of decision nodes for agent i. We also write $\mathcal{N}^d = \bigcup_{i=0}^{m} \mathcal{N}_i^d$ as the set of all decision nodes. $\mathcal{N}^o$ is the set of outcome nodes.
- $E \subset \mathcal{N}^d \times \mathcal{N}$ is the edges relation specifying transitions from decision nodes to other (decision or outcome) nodes.
- $\mathcal{I} = \bigcup_{i=0}^{m} \mathcal{I}_i$ is the set of all information sets. $\mathcal{I}_i$ is the set of information sets of agent i and I_i is one of these information sets. The special player nature can always discriminate the node he is in, implying that $|\mathcal{I}_0| = |\mathcal{N}_0^d|$.
- $K : \mathcal{N}^d \to I$ is the knowledge or information function that maps decision nodes for an agent to information sets. I.e., $\forall_i \;\; K : \mathcal{N}_i^d \to I_i$.
- $O : \mathcal{N}^o \to \mathbb{R}^m$ is the outcome function, specifying a payoff of an outcome node for each agent.
- $n_{root} \in \mathcal{N}^d$ is the start node.

Strictly speaking, an extensive form game does not define actions; instead an agent i at node $n_i^d \in \mathcal{N}_i^d$ selects an edge and thus a successor node from the set $\{x | E(n_i^d, x)\}$. However, we will assume that there is an action associated with the selection of each edge. As for POSGs we will denote the set of actions for agent i as $\mathcal{A}_i$.

A policy in an extensive form game is very similar to a policy in a POSG or Dec-POMDP as defined in section 1.2. Only now there are no explicit sequences of actions and observations (the action-observation histories), but information sets. Therefore a pure policy is a mapping from information sets to actions and a stochastic policy a mapping from information sets to probabilities over actions. The notion of mixed policies remains the same.

3.2 Extensive form of POSGs

Definition 3.2 The *extensive form of a POSGs* is an extensive form game and thus a tree. Every trace from root to leaf has the following structure:

$$n_0, (n_1, ..., n_m, n_0)^{h-1}, n_1, ..., n_m, n^o$$

where n_0 are decision nodes of nature, n_i with $i \geq 1$ are decision nodes of the agents and n^o are outcome nodes. The root n_{root} is the first n_0 node. Because of this structure, each node n has an associated time-step in the POSG, we denote this time-step as t_n.

The structure of an extensive form POSG is illustrated in figure 1. Shown in the figure is that nature's nodes can be split in transition probability nodes, n_0^T, and observation probability nodes, n_0^O, so the structure of a trace becomes $n_0^T, n_0^O, (n_1, ..., n_m, n_0^T, n_0^O,)^{h-1}, n_1, ..., n_m, n^o$, where the first n_0^O corresponds with an initial observation, which is usually omitted in POSGs.

Definition 3.3 A *path, $\sigma(n)$, in an extensive form of a POSG* is the path from the root n_{root} to node n. This path determines all the actions taken by the agents and nature and therefore corresponds with a sequence of joint actions, joint observations and states. For a particular decision node n_1 for agent 1 with $t_{n_1} = k$, this sequence has the following form:

$$\sigma(n_1) \equiv \left(n_{root}, s^0, \mathbf{o}^0, \mathbf{a}^0, s^1, \mathbf{o}^1, \mathbf{a}^1, ..., s^k, \mathbf{o}^k \right).$$

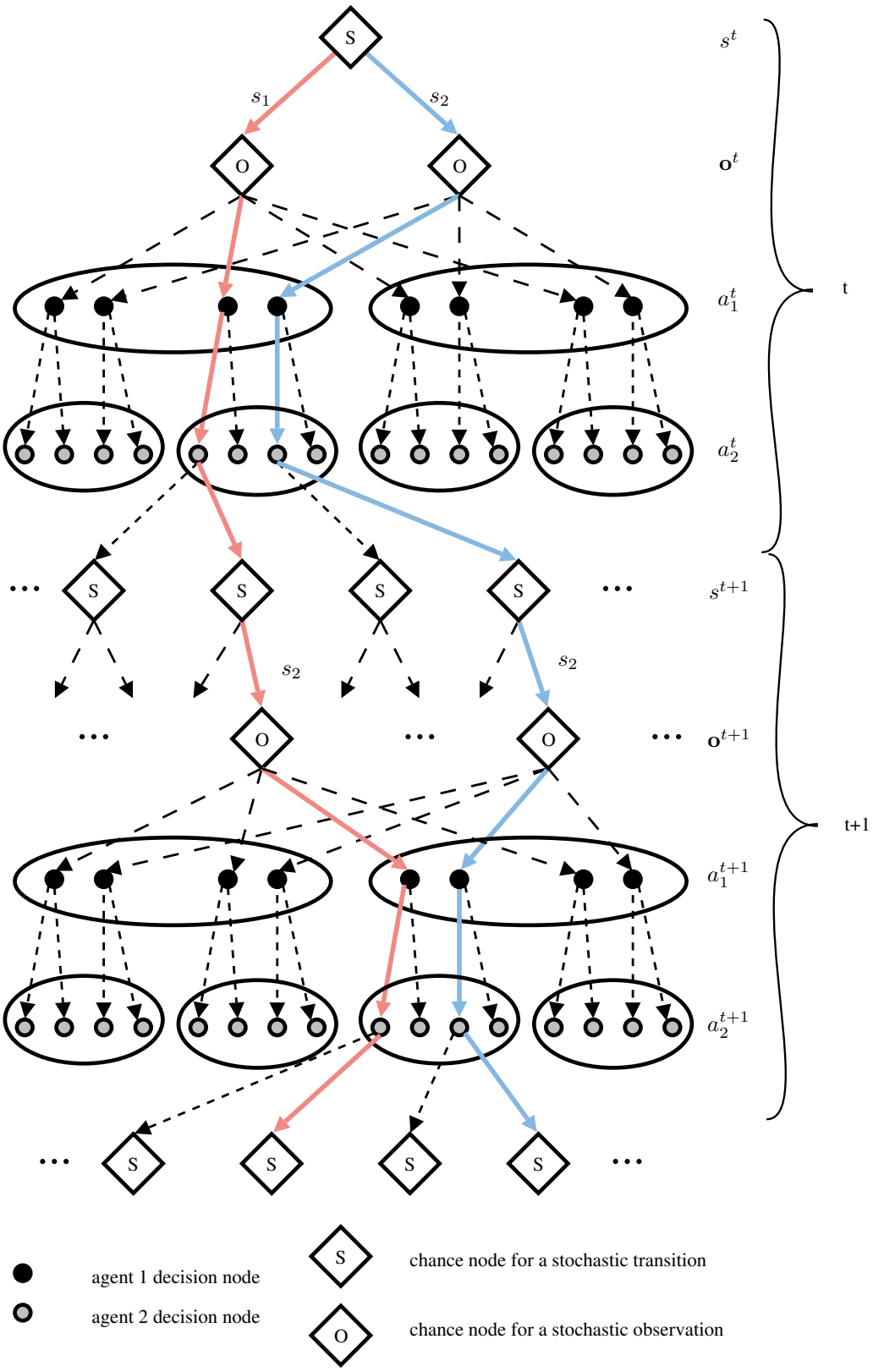

Figure 1: General illustration of the structure of an extensive form of a POSG. Also indicated are two different paths in the tree that specify the same joint action-observation history.

We write $s^t \in \sigma(n)$ if the path specifies state s at time-step t. Similarly we also write $\mathbf{a}^t \in \sigma(n), a_i^t \in \sigma(n), \mathbf{o}^t \in \sigma(n), o_i^t \in \sigma(n)$ if $t_n \geq t$ (such that the path consists of at least t time-steps) and the path $\sigma(n)$ specifies the particular (joint) action/observation at time-step t. As a consequence we also write $\vec{\theta}^t \in \sigma(n)$ and $\vec{\mathbf{o}}^t \in \sigma(n)$ if the path specifies the particular (joint) action-observation history or (joint) observation history.

Definition 3.4 The *outcomes in an extensive form of a POSG* correspond to the sums of rewards obtainable in the POSG. The sum of rewards for agent i specified by a full path $\sigma(n^o)$ from root to an outcome node n^o, which stretches over h time-steps, is:

$$O_i(n^o) \equiv O_i(\sigma(n^o)) = \sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t), \tag{3.1}$$

where $s^t, \mathbf{a}^t$ are the state and joint action specified by the path $\sigma(n^o)$ at time t.

Example 3.1 The extensive form of the deaf and the blind problem is shown in figure 2. The figure clearly illustrates the complexity of even this very small problem. Because the problem is a Dec-POMDP the outcome nodes specify only one outcome. In accordance with equation 3.1, the outcomes shown are the sum of rewards received along a path. E.g. the outcome for the 'good door opened' is -0.1 for the first time-step plus $+10$ for the second time-step.

Also clearly shown is, how information sets correspond to action-observation histories. In the deaf and the blind problem, both agents have three information sets. Agent one has the initial information set I_1^0 which corresponds to an empty action-observation history $\vec{\theta}_1 = \emptyset$, the information set I_1^{Le} 'left' corresponding to $\vec{\theta}_1 = \langle a_{Le}, o_\emptyset \rangle$ and I_1^{Ri} 'right' corresponding to $\vec{\theta}_1 = \langle a_{Ri}, o_\emptyset \rangle$. Likewise, for agent 2, we have I_2^0 ($\vec{\theta}_2 = \emptyset$), I_2^{Ro} 'roar' ($\vec{\theta}_2 = \langle a_F, o_{Ro} \rangle$) and I_2^{Si} 'silence' ($\vec{\theta}_2 = \langle a_F, o_{Si} \rangle$). □

3.3 Normal form solving

The normal (or strategic-) form of a game is a representation in terms of pure policies and expected outcomes for combinations of these pure policies for different players. The expected utility of a joint pure policy is given by the payoffs of the outcome nodes the joint pure policy can realize, weighted by their probabilities (induced by nature). Formally:

Definition 3.5 The expected payoff $V_i(\pi)$ for agent i of a joint policy π in an extensive form game, is the sum of the outcomes of all full paths it can realize, weighted by their probability. I.e., the value for agent i is given by [9]:

$$V_i(\pi) = \sum_{n^o \in \mathcal{N}^o} O_i(n^o) \cdot \nu(n^o) \cdot P(\sigma(n^o)|\pi) \tag{3.2}$$

where $\nu(n^o) \equiv \nu(\sigma(n^o))$ is nature's component of the probability that $\sigma(n^o)$ is realized (the product of the probabilities of the chance moves specified along the path) and $P(\sigma(n^o)|\pi)$ is the agents' component of this probability, specified by the joint policy. In case of a pure joint policy, this component is given by:

$$P(\sigma(n^o)|\pi) = \begin{cases} 1 & , \pi \text{ is consistent with } \sigma(n^o) \\ 0 & , \text{otherwise.} \end{cases}$$

Consistent means that when the path is given by $\sigma(n^o) = \left(s^0, \mathbf{o}^0, \mathbf{a}^0, s^1, \mathbf{o}^1, \mathbf{a}^1, ..., s^k, \mathbf{o}^k\right)$, the joint policy specifies all the joint actions in the path. I.e.: $\pi(\left(\mathbf{o}^0\right)) = \mathbf{a}^0$, $\pi(\left(\mathbf{o}^0, \mathbf{o}^1\right)) = \mathbf{a}^1$, etc.

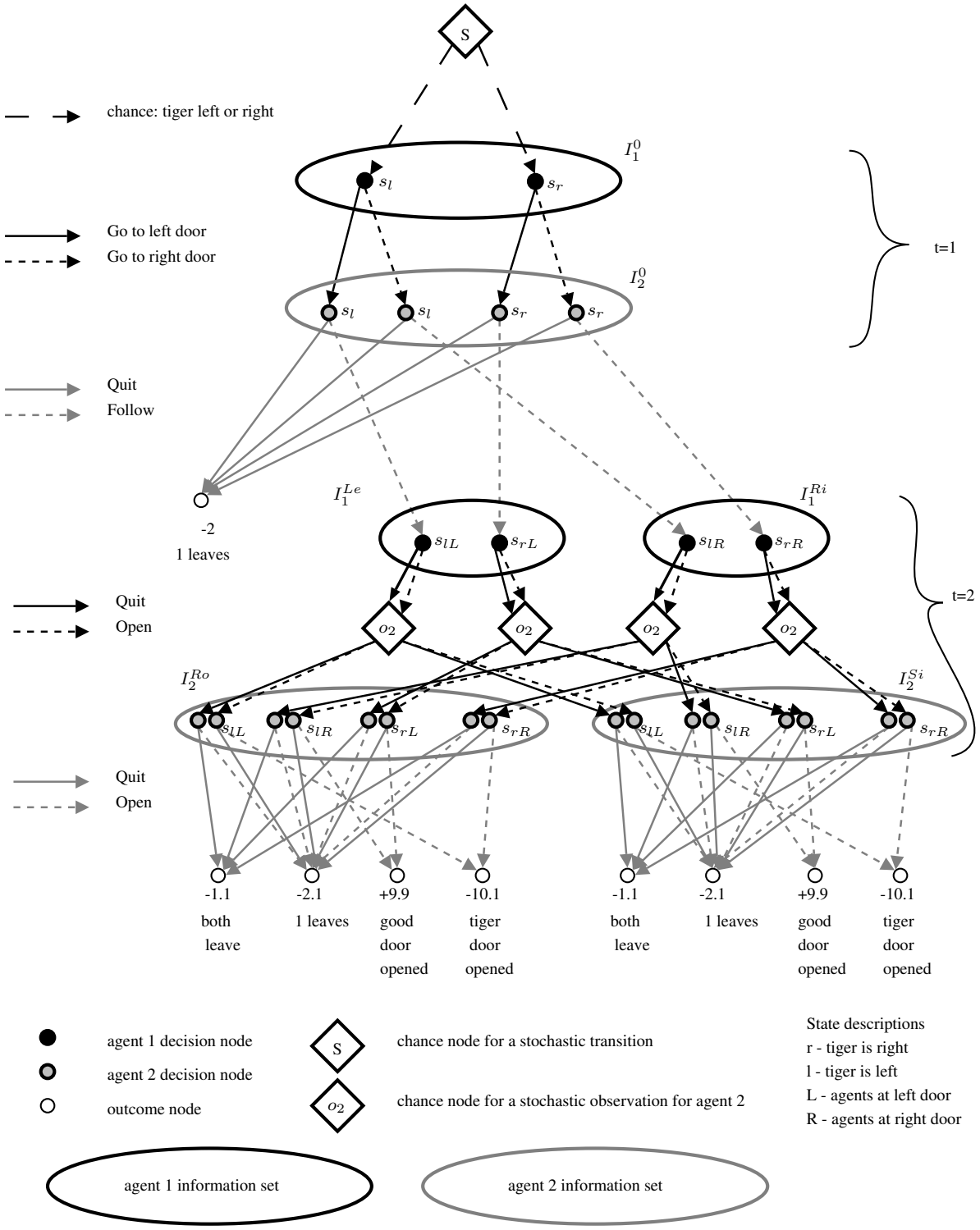

Figure 2: Extensive form for the deaf, the blind and the tiger problem. The nodes are annotated with their states. Agent 1 never receives an observation and therefore has no chance nodes for his observations. Except for the start node, the transitions are also deterministic, therefore there is no chance nodes for these transitions. The ovals show which nodes are grouped together in information sets.

	$I_2^0 : a_F$ $I_2^{Ro} : a_O$ $I_2^{Si} : a_O$	$I_2^0 : a_F$ $I_2^{Ro} : a_Q$ $I_2^{Si} : a_O$	$I_2^0 : a_F$ $I_2^{Ro} : a_O$ $I_2^{Si} : a_Q$	$I_2^0 : a_F$ $I_2^{Ro} : a_Q$ $I_2^{Si} : a_Q$	$I_2^0 : a_Q$ $I_2^{Ro} : *$ $I_2^{Si} : *$
$I_1^0 : a_{Le}$ $I_1^{Le} : a_O$ $I_1^{Ri} : *$	-1.1	$+2.478$	-5.678	-2.1	-2
$I_1^0 : a_{Le}$ $I_1^{Le} : a_Q$ $I_1^{Ri} : *$	-2.1	-1.619	-1.581	-1.1	-2
$I_1^0 : a_{Ri}$ $I_1^{Le} : *$ $I_1^{Ri} : a_O$	$+0.9$	$+3.222$	-4.422	-2.1	-2
$I_1^0 : a_{Ri}$ $I_1^{Le} : *$ $I_1^{Ri} : a_Q$	-2.1	-1.702	-1.298	-1.1	-2

Table 4: The (reduced) normal form representation of the deaf, the blind and the tiger problem. In reduced normal form pure strategies specifying the same behavior are been merged. (the 'full' normal form would be 8×8). The actions on which these merged policies differ are indicated with a *, which therefore can be interpreted as a wild-card.

Definition 3.6 In an extensive form of a POSG, nature's component of the probability $\sigma(n^o)$ is realized, is given by:

$$\nu(n^o) = b^0(s^{t=0}) \prod_{t=0}^{h-2} P(s^{t+1}|s^t, \mathbf{a}^t) \cdot P(\mathbf{o}^{t+1}|\mathbf{a}^t, s^{t+1}), \tag{3.3}$$

where $s^t, \mathbf{a}^t, \mathbf{o}^t$ are the state and joint action, observation specified by the path $\sigma(n^o)$, i.e., $\forall_t\ s^t, \mathbf{a}^t, \mathbf{o}^t \in \sigma(n^o)$.

In general, the normal form gives the expected outcome for every joint policy for each player. In the two agent case, this can be represented as a matrix $\mathbf{R}$ showing the outcome for both agents for each joint policy (for identical payoffs; Dec-POMDPs) or by two separate 'payoff matrices' $\mathbf{R}_i$ for each agent i (general payoffs; POSGs). This generalizes to multi-dimensional arrays for more than 2 agents. The entries r_i of the payoff matrix $\mathbf{R}_i$ for agent i are given by $V_i(\pi)$ according to equation 3.2.

Table 4 shows the normal form of the deaf, the blind and the tiger. Again, because we are dealing with a Dec-POMDP, there is only 1 outcome specified. To calculate the expected outcome of a joint policy, the outcome nodes that are realizable under the joint policy are taken, weighted by their probability, induced by the nature transitions along the path.

Example 3.2 As an example, the policy pair $\pi_1 = I_1^0 : a_{Ri}$, $I_1^{Le} : *$, $I_1^{Ri} : a_O$, $\pi_2 = I_2^0 : a_F$, $I_2^{Ro} : a_Q$, $I_2^{Si} : a_O$ can reach 4 outcome nodes: when both agents select a_O, they can receive $+9.9$ or -10.1, depending on whether the state was s_{lR} or s_{rR}. When agent 2 hears a roar, he will select a_Q, leading to -2.1 for both s_{lR} and s_{rR}. This leads to the following expected outcome of the joint policy:

$$P(s_l)P(o_{Si}|s_{lR}) \cdot 9.9 + P(s_r)P(o_{Si}|s_{rR}) \cdot (-10.1) + ...$$
$$P(s_l)P(o_{Ro}|s_{lR}) \cdot (-2.1) + P(s_r)P(o_{Ro}|s_{rR}) \cdot (-2.1) = +3.222$$

Calculation of other entries is illustrated in appendix B.1. □

How the normal form is solved to select a policy for each agent depends on the type of outcomes (zero-sum, general- or identical payoff) and the number of agents ($m = 2$ or $m > 2$). For 2-agent zero-sum games, the normal form can be converted to a linear program (LP) of the same size [21] and solved by linear programming [18, 20]. 2-Agent general sum games can be converted to a linearly complementary problem (LCP) [8] and solved with, for example, the Lemke-Howson algorithm [10]. When the number of agents is higher than two, any of the methods mentioned in [16] can be used for general payoff cases.

For identical payoff normal forms, such as the normal form of a Dec-POMDP, the solution is given by simply selecting the joint policy with the highest expected outcome. This means that for every Dec-POMDP there is at least one optimal pure joint policy:

Theorem 3.1 *For every finite horizon Dec-POMDP there is at least one optimal pure joint policy.*

Proof The entries of the normal form for the Dec-POMDP specify the expected cumulative rewards for all pure joint policies. At least one of these entries will be maximal. Assume that π is a policy specifying such an entry. As all agents receive the same payoff, no agent will have an incentive to deviate from π. Also π gives the maximal expected cumulative reward, therefore π is an optimal joint policy. □

In non-reduced normal form, policies are specified as mappings from information sets, i.e. action-observation histories to actions. At time-step t, there are $(|\mathcal{A}_i| \cdot |\mathcal{O}_i|)^t$ of these sequences for agent i. As a consequence there are a total of

$$\sum_{t=0}^{h-1} (|\mathcal{A}_i| \cdot |\mathcal{O}_i|)^t = \frac{(|\mathcal{A}_i| \cdot |\mathcal{O}_i|)^h - 1}{(|\mathcal{A}_i| \cdot |\mathcal{O}_i|) - 1}$$

of such sequences for agent i. When we let $|\mathcal{A}_*|$ and $|\mathcal{O}_*|$ denote the largest individual action and observation sets, the space complexity of the normal form which is equal to the number of joint policies is given by:

$$O\left(\left[|\mathcal{A}_*|^{\left(\frac{(|\mathcal{A}_*|\cdot|\mathcal{O}_*|)^h-1}{(|\mathcal{A}_*|\cdot|\mathcal{O}_*|)-1}\right)}\right]^m\right) = O\left(|\mathcal{A}_*|^{\frac{m\left[(|\mathcal{A}_*|\cdot|\mathcal{O}_*|)^h-1\right]}{(|\mathcal{A}_*|\cdot|\mathcal{O}_*|)-1}}\right).$$

This representation, however, suffers from two types of 'redundancies' as we illustrate using figure 3. The first redundancy occurs in pure policies and was briefly discussed in section 1.2. When a policy π_i deterministically specifies an action say a_i for a particular action-observation history $\vec{\theta}_i$, this means that some parts of the tree of will not be reached. Clearly, it is unnecessary to specify actions for these unreached parts. E.g., in figure 3, if agent 1 selects a policy π_1 that specifies $\bar{a}_1$ at decision point 1, this policy will not have to specify an action at decision point 2. This redundancy can be exploited by defining a pure policy π_i as mapping from the *observation history* $\vec{o}_i$, the sequence of all observations received by the agent, to actions.

This is exactly what is done in the *reduced* normal form, which reduces the size of the representation to:

$$O\left(|\mathcal{A}_*|^{\frac{m\left[(|\mathcal{O}_*|)^h-1\right]}{(|\mathcal{O}_*|)-1}}\right)$$

Clearly, constructing the reduced normal form corresponds with brute-force joint policy evaluation as mentioned in [11], as it simply calculates the expected outcome of all pure joint policies. (For a detailed proof of this correspondence see appendix A.1.) This means that reduced normal form solving, like brute-force policy evaluation, is intractable for all but the smallest problems, as the complexity of brute-force policy evaluation is:

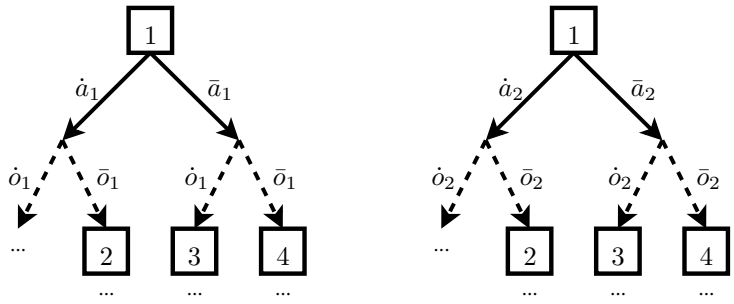

Figure 3: Partial trees of action-observation histories for two agents. Left: tree of agent 1's action-observation histories. Right: the same for agent 2. The squares represent points at which the agents can take decisions.

$$O\left(\left(|\mathcal{A}_*|^{\frac{m\left[|\mathcal{O}_*|^h-1\right]}{|\mathcal{O}_*|-1}}\right)\cdot\left(|\mathcal{S}|\cdot|\mathcal{O}_*|^m\right)^h\right), \tag{3.4}$$

Here, $(|\mathcal{S}|\cdot|\mathcal{O}_*|^m)^h$ is the cost of evaluating one joint policy.

The second redundancy arises when reasoning about all pure policies, in this case different policies share sub-trees. E.g. in figure 3, if π_1 is a policy that specifies $\bar{a}_1$ at decision point 3 and 4 and π_1' is a policy that specifies $\dot{a}_1$ at those decision points, then both policies specify $\dot{a}_1$ for decision point 1. Intuitively, this means that it should be possible to represent the policies more compactly. This is indeed possible, as we will discuss in the next section.

3.4 Sequence form

In [8, 9] a representation called *sequence form* for solving extensive form games is introduced. The sequence form essentially translates the above intuitions into an appropriate data structure for representing policies: Since policies are essentially trees, sequence form represents sets of policies using their common sub-trees. As the name implies, sequence form is based on 'sequences'. These are very much related to the paths and histories which we already discussed. Formally:

Definition 3.7 A *sequence for agent* i, σ_i, is the portion of a path that is under agents i's control and observation. More specific, agents i's sequence to a particular node n, $\sigma_i(n)$, consists of:

- All agent i's actions and observations up to n_{prec}, agent i's decision node preceding n in path $\sigma(n)$.
- The action specified at n_{prec}.

Therefore a sequence can be summarized as a tuple $\langle I_i, a_i\rangle$, where $K(n_{prec}) = I_i$ is agent i's information set preceding n in $\sigma(n)$ and a_i is the action specified for I_i by $\sigma(n)$. When we need to refer to the k-th sequence of agent i we will write $\sigma_{i,k}$.

Because a sequence $\sigma_i(n)$ specifies the components of a path $\sigma(n)$ that agent i can observe and control, this can be used to express agent i's contribution to the realization of a path $P(\sigma(n)|\pi)$ by defining *realization weights* over these sequences:

Definition 3.8 The *realization weight of a sequence* σ_i, denoted $\rho_i(\sigma_i)$, is the probability that agent i will take the moves specified by σ_i, *given that the appropriate information sets are reached.*

Of course not all arbitrary assignments of realization weights to an agent's sequences represent a valid (possibly randomized) policy. In particular, the realization weights of continuations of a sequence must sum up to the probability of that sequence. Let $\sigma_i(I_i)$ be a sequence for player i that can lead to a particular information set I_i. Let $\sigma_i(I_i) \circ a_1, ..., \sigma_i(I_i) \circ a_n$ be sequences that are continuations of $\sigma_i(I_i)$, that specify taking action $a_1, ..., a_n$ at information set I_i. The constraints for realization weights tell us that:

$$\rho_i(\sigma_i(I_i)) = \rho_i(\sigma_i(I_i) \circ a_1) + ... + \rho_i(\sigma_i(I_i) \circ a_n).$$

When the realization weights of $\sigma_i(I_i)$ and $\sigma_i(I_i) \circ a_i$ are known, the probability of taking action a_i at information set I_i is:

$$P(a_i|I_i, \rho_i) = \frac{\rho_i(\sigma_i(I_i) \circ a_i)}{\rho_i(\sigma_i(I_i))}.$$

So in this way, a set of realization weights satisfying the proper constraints corresponds to a stochastic policy.

The other way around, it is also possible to find the realization weights, given a particular stochastic (joint) policy. We write $\rho_i^\pi(\sigma_i) = \rho_i^{\pi_i}(\sigma_i)$ for the realization weight of σ_i as specified by joint policy $\pi = \langle \pi_i, \pi_{\neq i} \rangle$. For the extensive form of a POSG, in which an information set corresponds with an action-observation history, we can write the realization weight of a sequence as follows:

$$\begin{aligned} \rho_i^\pi(\sigma_i) = \rho_i^\pi(\langle I_i, a_i^t \rangle) &= \rho_i^\pi(\langle (o_i^0, a_i^0, ..., o_i^t), a_i^t \rangle) \\ &= P^\pi(a_i^t|(o_i^0, a_i^0, ..., o_i^{t-1}, a_i^{t-1}, o_i^t)) \\ &\quad \cdot P^\pi(a_i^{t-1}|(o_i^0, a_i^0, ..., o_i^{t-1})) \cdot .... \cdot P(a_i^0|o_i^0) \\ &= \prod_{t'=0}^{t} P^\pi(a_i^{t'}|\vec{\theta}_i^{t'}), \end{aligned} \tag{3.5}$$

where P^π denotes the probability according to $\pi = \langle \pi_i, \pi_{\neq i} \rangle$.

In sequence form, the expected outcome of a joint policy is defined as follow:

Definition 3.9 The expected value for agent i of a joint policy in sequence form is

$$V_i(\pi) = \sum_{n^o \in \mathcal{N}^o} O_i(n^o) \cdot \nu(n^o) \cdot \prod_{i=1}^{m} \rho_i^\pi(\sigma_i(n^o)), \tag{3.6}$$

where $\nu(n^o)$ is the product of probabilities of nature's moves along the path as before (eq. 3.3).

This is the equivalent of equation 3.2 that was used for the normal form, generalized to stochastic policies specified by realization weights. As before, in the two agent case, this can be rewritten to matrix form, similar to the normal form,[3] but with rows and columns corresponding to sequences of the agents rather than pure policies. Let $\mathbf{R}$ be the sequence form payoff matrix for agent 1, then an entry r_{lk} corresponds with the expected value of agent 1's l-th sequence $\sigma_{1,l}$ against $\sigma_{2,k}$, and is given by:

$$r_{lk} = \sum_{n^o \in \mathcal{N}^o \text{ s.t. } \sigma_1(n^o)=\sigma_{1,l} \wedge \sigma_2(n^o)=\sigma_{2,k}} \nu(n^o) \cdot O_1(n^o).$$

Here, the summation is over outcome nodes consistent with the sequences l and k. As these sequences completely specify the joint action-observation history, the consistent outcome nodes n^o specify paths $\sigma(n^o)$ that only differ on the actual states. Other nodes will not have to be

		I_2^0		I_2^{Ro}		I_2^{Si}	
		a_Q	a_F	a_Q	a_O	a_Q	a_O
I_1^0	a_{Le}	-2	0	0	0	0	0
	a_{Ri}	-2	0	0	0	0	0
I_1^{Ri}	a_Q	0	0	-0.365	-0.696	-0.735	-1.404
	a_O	0	0	-0.696	-3.018	-1.404	$+3.918$
I_1^{Le}	a_Q	0	0	-0.529	-1.010	-0.571	-1.090
	a_O	0	0	-1.010	-4.588	-1.090	$+3.488$

Table 5: Sequence form of the deaf and the blind problem. The rows are sequences for the deaf (agent 1), columns for the blind (agent 2). The sequences are grouped per action-observation history or, equivalently, information set.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \rho_1(\sigma_{root}) \\ \rho_1(\langle I_1^0, a_{Ri} \rangle) \\ \rho_1(\langle I_1^0, a_{Le} \rangle) \\ \rho_1(\langle I_1^{Ri}, a_Q \rangle) \\ \rho_1(\langle I_1^{Ri}, a_O \rangle) \\ \rho_1(\langle I_1^{Le}, a_Q \rangle) \\ \rho_1(\langle I_1^{Le}, a_O \rangle) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Table 6: Sequence form constraint matrix and equation for the deaf (agent 1).

considered and, as a consequence, many of the matrix entries are zero. The complete sequence form of the deaf and the blind problem is shown in table 5.

Example 3.3 As an example, the entry for the sequences $\langle (I_1^{Ri}, a_O), (I_2^{Ro}, a_O) \rangle$ is the summation over all the outcome nodes n^o that are consistent with these sequences weighted by their probability induced by nature $\nu(n^o)$. Which for this specific combination of sequences gives:

$$\begin{aligned} P(s_l) \cdot P(o_{Ro}|s_{lR}) \cdot 9.9 + P(s_r) \cdot P(o_{Ro}|s_{rR}) \cdot (-10.1) &= \\ 0.55 \cdot 0.03 \cdot 9.9 + 0.45 \cdot 0.7 \cdot (-10.1) &= -3.018. \end{aligned}$$

Note that for all the combinations of the information sets $I_1^{Le}, I_1^{Ri}, I_2^{Si}$ and I_2^{Ro}, there is a symmetry between sequences that specify $a_1 = a_O, a_2 = a_Q$ and sequences that specify $a_1 = a_Q, a_2 = a_O$. This is because the payoffs specified by all the outcome nodes that are consistent with these sequences are the same (namely -2.1) and the probabilities induced by nature are also the same. □

The agents will have to choose the realization weights of their sequences in agreement with the relevant constraints, as illustrated by table 6 and 7 that shows the constraint matrix for respectively agent 1 and 2. When both agents have selected such realization weights, this specifies a joint policy. When $\mathbf{R}$ is the payoff matrix for agent 1, this agent's value (given by eq. 3.6) can be written as:

$$V_1(\pi) = \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \cdot r_{lk}. \tag{3.7}$$

[3]Again, this representation generalizes to a multi-dimensional array in the case of more than 2 agents.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \rho_2(\sigma_{root}) \\ \rho_2(\langle I_2^0, a_Q \rangle) \\ \rho_2(\langle I_2^0, a_F \rangle) \\ \rho_2(\langle I_2^{Ro}, a_Q \rangle) \\ \rho_2(\langle I_2^{Ro}, a_O \rangle) \\ \rho_2(\langle I_2^{Si}, a_Q \rangle) \\ \rho_2(\langle I_2^{Si}, a_O \rangle) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Table 7: Sequence form constraint matrix and equation for the blind (agent 2).

Example 3.4 As an example, if π_1 specifies $\rho(\langle I_1^{Ri}, a_O \rangle) = 1$ and π_2 specifies $\rho(\langle I_2^{Si}, a_O \rangle) = \rho(\langle I_2^{Ro}, a_Q \rangle) = 1$,[4] this corresponds with the optimal joint policy $\pi_1 = I_1^0 : a_{Ri}$, $I_1^{Le} : *$, $I_1^{Ri} : a_O$, $\pi_2 = I_2^0 : a_F$, $I_2^{Ro} : a_Q$, $I_2^{Si} : a_O$ which we encountered before. Its expected value according to equation 3.7 is:

$$-0.696 + 3.918 = +3.222,$$

which is exactly what we calculated before. □

Solving the sequence form means finding optimal realization weights for all agents. As for normal form games, the question of how to solve this sequence form, depends on the number of players and the type of outcomes it specifies. Research has mainly addressed the two agent case. We will focus on identical payoff games here.

For identical payoff games (e.g. Dec-POMDPs), optimally solving the sequence form takes worst-case exponential time, assuming EXP $\neq$ NEXP. This can be seen as follows. The size of the sequence form payoff matrix is the number of joint action-observation histories times the number of joint actions:

$$O\left(|\mathcal{A}_*|^m \cdot \left(\frac{(|\mathcal{A}_*| \cdot |\mathcal{O}_*|)^h - 1}{(|\mathcal{A}_*| \cdot |\mathcal{O}_*|) - 1}\right)^m\right)$$

which is exponential in the size of the Dec-POMDP. Would there be a polynomial algorithm, then solving a Dec-POMDP is in EXP. However, solving a Dec-POMDP is NEXP-complete [1]. Therefore, assuming EXP $\neq$ NEXP, there can be no polynomial time algorithm for optimally solving an identical payoff game in sequence form. This means that, although sequence form is exponentially smaller than normal form and thus offers exponential space savings, the worst-case time complexity is equal to that of constructing the full normal form (and thus to brute force policy evaluation[5]). Currently, the only known algorithm for optimally solving the sequence form of an identical payoff game is evaluating all combinations of pure policies, yielding all pure joint policies, but there might be methods with better lower-bounds.

Another possibility for solving the sequence form of identical payoff games is to apply *alternating maximization.* In this procedure, an arbitrary joint policy is used as initialization. Then one agent is selected whose policy is improved, while keeping the policies of the other agents fixed. The agent improves his policy by calculating a best-response: it assigns a realization weight of 1 to those sequences that maximize the sum specified by equation 3.7, respecting the constraints for the realization weights. E.g. in the two player case where $\mathbf{R}$ is the common payoff matrix, agent 1 will perform the following maximization:

$$\pi_1' = \arg\max_{\pi_1} \sum_l \rho_1^{\pi}(\sigma_{1,l}) \sum_k \rho_2^{\pi}(\sigma_{2,k}) \cdot r_{lk} \tag{3.8}$$

[4] Note that, because of the constraints, this also implies that $\rho(I_1^0, a_{Ri}) = 1$ and that $\rho(I_2^0, a_F) = 1$.

[5] In fact, brute-force policy evaluation has the best space complexity, as it requires storing only the best policy found so far.

		I_2^0		I_2^{Ro}		I_2^{Si}	
		a_Q	a_F	a_Q	a_O	a_Q	a_O
I_1^0	a_{Le}	-2	0	0	0	0	0
	a_{Ri}	-2	0	0	0	0	0
I_1^{Ri}	a_Q	0	0	-0.365	-0.696	-0.735	-1.404
	a_O	0	0	-0.696	-3.018	-1.404	$+3.918$
I_1^{Le}	a_Q	0	0	-0.529	-1.010	-0.571	-1.090
	a_O	0	0	-1.010	-4.588	-1.090	$+3.488$

Table 8: Illustration of the calculation of a best-response for the agent 1 (the deaf) against the policy $\pi_2 = I_2^0 : a_F$, $I_2^{Ro} : a_Q$, $I_2^{Si} : a_O$ of the second agent (the deaf). The dark columns can not be realized under π_2.

The procedure of this maximization is illustrated in table 8. Next, another agent is selected to improve its policy, etc. This will lead to a Nash-equilibrium, but it might not be the best one. I.e., it is only guaranteed to find a locally optimal solution.

4 Direct calculation of best-response policies (DCBRP)

In section 3.4 we showed how an agent could select a best-response policy using sequence form. It is also possible to calculate a best-response for an extensive form game more directly, as is shown for poker games in [13, 14]. We will refer to this method as *direct calculation of best-response policies (DCBRP)* for extensive form games.

The approach is to transform the extensive form game to a POMDP for a protagonist agent. The solution of this POMDP gives an optimal (deterministic) best-response policy for the protagonist agent with regards to the policies of the other agents. The transformation to a POMDP is accomplished by converting all decision nodes for the protagonist agent and all outcome nodes to states for the POMDP. The deterministic transitions from the game-tree are converted to stochastic transitions in the POMDP, where the transition probabilities are defined through the fixed policies of the other agents, and any observations that are implicit in the extensive form are made explicit in the observation model. Figure 4 shows a POMDP model for the first agent in 'the deaf and the blind'. Comparing it to figure 2 gives an intuition behind this transformation.

Now we will further formalize the transformation to a POMDP model. First we will discuss the transformation for arbitrary extensive form games, then for extensive forms of POSGs. We will denote the states in the POMDP model as p in order not to confuse them with states for POSGs (notated as s).

4.1 General extensive form games

We will now discuss DCBRP for general extensive form games in which there are m agents, but in which these do not necessarily take actions simultaneously each time-step (or even in a fixed order). For such a game, the probability of reaching a particular next decision node n_i' of the protagonist agent i from a decision node n_i after action a_i, is determined by all other decision nodes that are between them. E.g. suppose action a_i leads to a node n_j for agent j and that the latter's action a_j leads to n_i', then the probability is given by:

$$P(n_i'|n_i, a_i) = P(a_j|n_j),$$

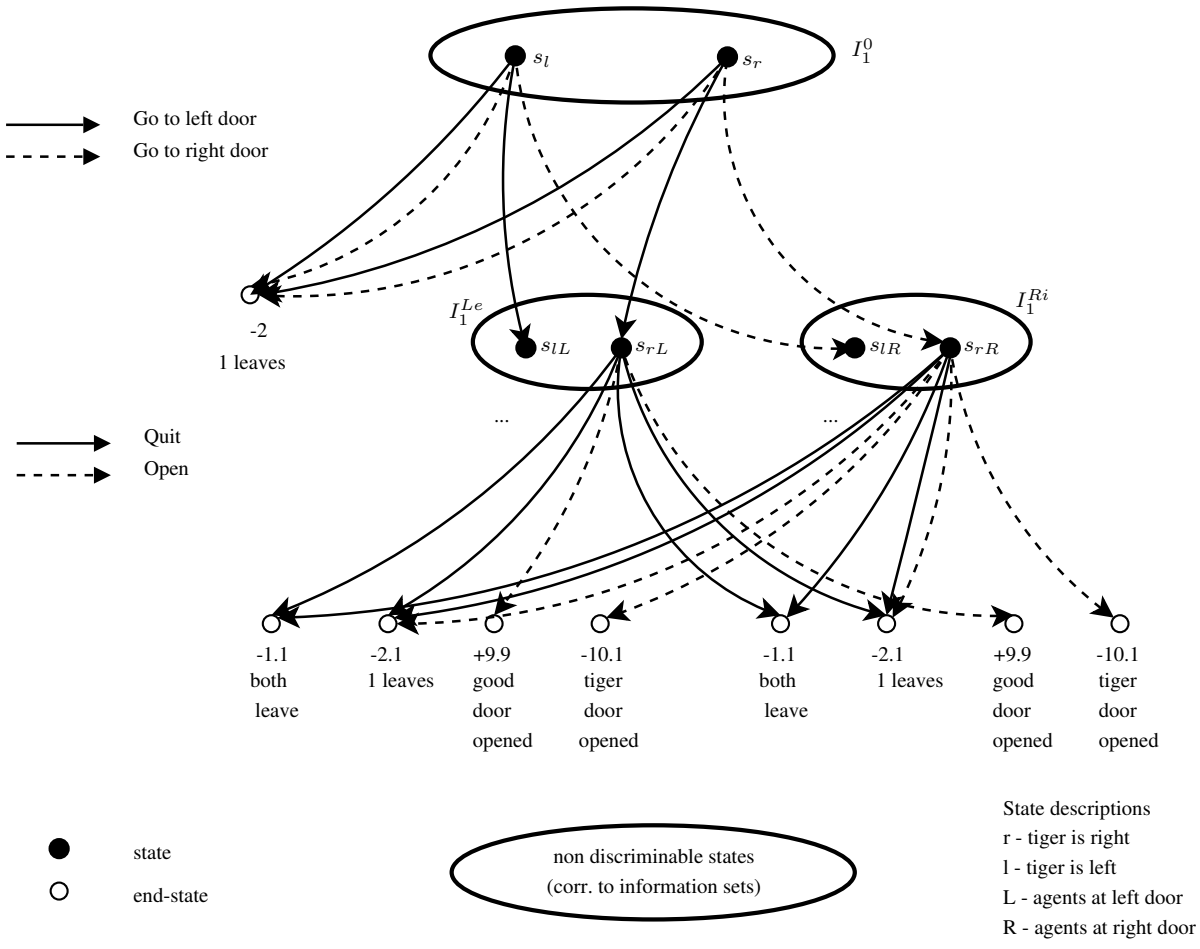

Figure 4: A POMDP model for the deaf and the blind for first agent (the deaf). For clarity, the transitions from states s_{lL} and s_{lR} are omitted.

where $P(a_j|n_j)$ is 0 or 1 if agent j has a deterministic policy. So in this case, the POMDP model would specify $P(p'|p, a_i) = P(a_j|n_j)$, where p', p are the POMDP states that correspond with n'_i, n_i. In general, however, there can be any number of nodes. In this case the probability can be expressed by sequences in the sub-tree induced by n_i: [6]

$$P(p'|p, a_i) = \begin{cases} \nu(n'_i) \cdot \prod_{j \neq i} \rho^{\pi}_j(\sigma_j(n'_i)) & , a^0_i \in \sigma(n'_i) \\ 0 & , \text{otherwise} \end{cases} \tag{4.1}$$

where $a^0_i \in \sigma(n'_i)$ indicates that action a_i must be the first action in $\sigma(n'_i)$, the path from n_i to n'_i.

Equation 4.1 is also valid when n'_i is an outcome node and the initial state distribution, $b^0(p)$, is given by the probability that the first decision nodes of agent i are reached. E.g. the initial transitions in figure 2 determine the initial belief over POMDP states (which are shown in figure 4). As a result the transition model is completely specified.

The observation and reward model for a general extensive form game are trivial. The observation the protagonist agent receives is the information set of the new node and he receives this observation with probability 1:

$$P(I(n'_i)|a_i, p') = P(I(n'_i)|p') = 1$$

where p' is the state representing node n'_i.

The reward of reaching state p' is non-zero only when the corresponding node is an outcome node:

$$R(p') \quad = \quad \begin{cases} O(n'_i) & , n'_i \in \mathcal{N}^o \\ 0 & , \text{otherwise} \end{cases}$$

as is illustrated in figure 4. The more commonly used reward function form $R(s, a)$ can be found by applying one backup step.

4.2 DCBRP for extensive form POSGs

In the previous subsection we specified the transition probabilities for the POMDP formed from a general extensive form game. Because a general extensive form game can have any structure, $P(p'|p, a_i)$ could only be defined by paths. Now, however, we consider extensive form POSGs which have a well-defined and fixed structure. Therefore this probability can be defined more explicitly.

Let p be the POMDP state that represents node n_i, and let s and $\vec{\theta}$ be the state and joint action-observation history that n_i specifies. The path $\sigma(n_i)$ differs from $\vec{\theta}$ in that it also assigns states for each time-step, so there are maximally $|\mathcal{S}|^t$ (where $t = t_{n_i}$, the time-step of the n_i) nodes (and thus as much POMDP states p) that specify the same action-observation history $\vec{\theta}$. [7]

Similarly, p' represents n'_i, the descendant decision node for agent i in the next time-step that specifies state s' and joint action-observation history $\left(\vec{\theta}, \mathbf{a}, \mathbf{o}\right)$. Also the joint action $\mathbf{a} = \langle \mathbf{a}_{\neq i}, a_i \rangle$ specifies action a_i for the protagonist agent i. Now we can define the probability of the transition from p to p' as:

$$P(p'|p, a_i) = P(\mathbf{o}|s', \langle \mathbf{a}_{\neq i}, a_i \rangle) P(s'|s, \langle \mathbf{a}_{\neq i}, a_i \rangle) P(\mathbf{a}_{\neq i}|\vec{\theta}).$$

[6] I.e., the sub-tree with n_i as its root. Therefore the path $\sigma(n'_i) = (n_i, ..., n'_i)$ and a sequence $\sigma_j(n'_i)$ represents agent j' s components of that path.

[7] Alternatively, it is also possible to specify p as the POMDP state representing the group of $|\mathcal{S}|^{t-1}$ nodes that specify the same $\vec{\theta}$ and state s^t. Here, for ease of explanation, we will assume the former and more straightforward specification in which there is a one to one correspondence to nodes in the game-tree.

The observation agent i receives in such a transition is specified to be one of the observations $o_i \in \mathcal{O}_i$ from the POSG model. Therefore we can directly use these observations instead of 'observing the information set' as in section 4.1. Let n'_i specify joint observation $\mathbf{o} = \langle \mathbf{o}_{\neq i}, o_i \rangle$, then:

$$P(o_i|p') = 1.$$

Other observations o'_i have probability 0.

The reward model can also be simplified.[8]We can specify the reward function as:

$$R(p') = R(s, \mathbf{a}),$$

where s is the POSG state specified by (n_i and thus by) p, the predecessor POMDP state of p'. $\mathbf{a}$ is the joint action that leads to p', i.e., $\mathbf{a}$ is the last joint action in $\sigma(n'_i)$. Effectively, this means that in each state p the rewards of the preceding transition are received. I.e. the rewards are delayed one time-step. This is not uncommon though and additionally allows rewards that are dependent on the next state: $R(s, a, s')$.

Alternatively, we can also specify:

$$R(p, a_i) = R(s, \langle \pi_{\neq i}(\vec{o}_{\neq i}), a_i \rangle),$$

where $\vec{o}_{\neq i}$ is specified by p, or when the other agents are allowed to have stochastic policies:

$$\begin{aligned} R(p, a_i) &= E\left[R(s, \left\langle \pi_{\neq i}(\vec{\theta}_{\neq i}), a_i \right\rangle\right] \\ &= \sum_{\mathbf{a}_{\neq i}} R(s, \langle \mathbf{a}_{\neq i}, a_i \rangle) P(\mathbf{a}_{\neq i} | \pi_{\neq i}, \vec{\theta}_{\neq i}). \end{aligned}$$

4.3 Solving the POMDP

In general solving a finite POMDP is PSPACE-hard [15]. However, when constructing a POMDP given the fixed policies of the other agents, there is exactly one belief for each information set of the protagonist agent.[9] As, for a finite extensive form game, the game-tree is finite, this means that the number of beliefs is finite and linear in the size of the extensive form. Therefore it is possible to construct a (fully observable) MDP over belief states by generating all possible beliefs, their transitions and rewards. This MDP can then be solved exactly using value iteration. [13, 14]

The construction of the belief MDP is straightforward. The chance of reaching a next belief is equal to the chance of receiving the observation that leads to that belief, i.e.:

$$P(b'|b, a) = P(o_i|a_i, b),$$

where a_i and o_i are the action and observation leading to belief b' and $P(o_i|a_i, b)$ is the probability of receiving observation o_i after action a_i from belief b, defined as:

$$P(o_i|a_i, b) = \sum_{p'} P(o_i|p') \sum_{p} P(p'|p, a_i) b(p),$$

which, because of the way the observations are defined reduces to:

$$P(o_i|a_i, b) = \sum_{p' \ s.t. \ P(o_i|p')=1} \sum_{p} P(p'|p, a_i) b(p).$$

[8]This is different from what is shown in figure 4. However, it is easy to imagine the reward being divided per time-step.

[9]An information set exactly and uniquely specifies the action-observation history for the agent.

The reward of a particular belief b is also trivially defined as:

$$R(b) = \sum_p R(p)b(p),$$

Now the only ingredient left is the belief update:

$$b_{a_i}^{o_i}(p') = \frac{P(o_i|p')\sum_p P(p'|p,a_i)b(p)}{P(o_i|a_i,b)},$$

which is defined completely in terms of preceding equations.

5 JESP

Joint equilibrium based search for policies (JESP) [11] is a method for Dec-POMDPs that also calculates a local optimum by applying the methodology of alternating maximization which we discussed in section 3.4. The variant DP-JESP which we will consider here (and refer to as simply 'JESP' hereafter) combines this with the direct calculation of best-responses as discussed in section 4.

5.1 JESP's dynamic program

Instead of first constructing the complete sequence form and using equation 3.8 to perform the maximization thus calculating a best-response, JESP uses a function 'best-response' which we prove to be equivalent to the direct best-response calculation from section 4.

As mentioned JESP is a dynamic programming algorithm that also performs alternating maximization. That means that it fixes the policies of all but one agent, and calculates a best-response for this agent. In order to do this, the agent calculates a value function over beliefs. In this case, however, a belief over states s is insufficient: in order to predict the actions of the other agents, the probability of their observations is required as well. Therefore JESP maintains beliefs over states s and observation histories of the other agents $\vec{o}_{\neq i}$.

In the following we assume two agents. We calculate a best-response for agent 1, so agent 2's policy is fixed. This means that a belief for agent 1 specifies the probabilities of states and observation histories of agent 2: $b_1^t(\langle s, \vec{o}_2^t \rangle)$. We also assume that agent 2's policy is deterministic. In this case the value function for agent 1 is given by:

$$V^t(b_1^t) = \max_{a_1 \in \mathcal{A}_1} \left[R(b_1^t, a_1) + \sum_{o_1 \in \mathcal{O}_1} P(o_1|b_1^t, a_1) V^{t+1}(b_1^{t+1}) \right]. \tag{5.1}$$

In this equation

$$R(b_1^t, a_1) = \sum_{\langle s^t, \vec{o}_2^t \rangle} b_1^t(\langle s^t, \vec{o}_2^t \rangle) R(s^t, \langle a_1, \pi_2(\vec{o}_2^t) \rangle) \tag{5.2}$$

is the expected immediate reward of performing action a_1 under belief b_1^t. The updated belief resulting from b_1^t after action a_1 and observing o_1 and given by:

$$b_1^{t+1}(\langle s^{t+1}, \vec{o}_2^{t+1} \rangle) = \frac{1}{P(o_1|b_1^t, a_1)} \sum_{\langle s^t, \vec{o}_2^t \rangle} b_1^t(\langle s^t, \vec{o}_2^t \rangle) \cdot P(\langle s^{t+1}, \vec{o}_2^{t+1} \rangle, o_1 | \langle s^t, \vec{o}_2^t \rangle, \langle a_1, \pi_2(\vec{o}_2^t) \rangle),$$

Because the observation history is formed by concatenation, the probability of $\vec{o}_2^{t+1}$ is non-zero only when $\vec{o}_2^{t+1} = \left(\vec{o}_2^t, o_2\right)$. Therefore we can write:

$$b_1^{t+1}(\langle s^{t+1}, \left(\vec{o}_2^t, o_2\right)\rangle) = \frac{1}{P(o_1|b_1^t, a_1)} \sum_{s^t} b_1^t(\langle s^t, \vec{o}_2^t\rangle) \cdot P(s^{t+1}, \langle o_1, o_2\rangle | s^t, \langle a_1, \pi_2(\vec{o}_2^t)\rangle),$$

in the above equations, $P(o_1|b_1^t, a_1)$ is a normalizing constant:

$$P(o_1|b_1^t, a_1) = \sum_{s^{t+1}} \sum_{o_2 \in \mathcal{O}_2} P(\langle o_1, o_2\rangle \,|\, \langle a_1, \pi_2(\vec{o}_2^t)\rangle, s^{t+1}) \sum_{\langle s^t, \vec{o}_2^t\rangle} b_1^t(\langle s^t, \vec{o}_2^t\rangle) \cdot P(s^{t+1}|s^t, \langle a_1, \pi_2(\vec{o}_2^t)\rangle).$$

5.2 The relation with DCBRP for extensive form games

Direct calculation of best-response policies for extensive form games as discussed in section 4 defines a standard POMDP and therefore uses the standard POMDPs expressions from section 1.3. However, although DCBRP solves a standard POMDP, the states p over which this POMDP is defined, correspond with POSG states s and observation histories of other agents $\vec{o}_2^t$. Therefore it is possible to show that JESP's definition of the value function and the belief update is equivalent to the definitions used by DCBRP:

Theorem 5.1 *The function used in JESP to calculate a best-response for Dec-POMDPs is equivalent to direct calculation of best-response policies for extensive form games when applied to the extensive form representation of the same Dec-POMDP. I.e., they calculate the same value function.*

Proof See appendix A.2. □

Now we that we established that the value function is identical for both methods, the only difference could be in how the methods actually perform the dynamic programming. This is also almost identical: JESP also generates all reachable beliefs and than performs value iteration. The only difference is that JESP specifies to perform this value iteration ordered, i.e. from $t = h-1, ..., 0$ (or equivalently expressed in time to go: $\tau = 1, ..., h-1$). This is an implementational optimization one would also apply for DCBRP.

6 Immediate reward sequence form

In this section we introduce the *immediate reward sequence form (IRSF)*. This is a model very similar to the regular sequence form, but distributing the rewards over time-steps. We use this model to make a link between JESP and the sequence form approaches as discussed. Suppose agent 2 from the deaf and the blind has the following policy: $\pi_2 = I_2^0 : a_F$, $I_2^R : a_Q$, $I_2^S : a_O$. In this case, agent 1 will calculate a best-response policy by calculating the JESP value function:

$$V^t(b_1^t) = \max_{a_1 \in \mathcal{A}_1} \left[R(b_1^t, a_1) + \sum_{o_1 \in \mathcal{O}_1} P(o_1|b_1^t, a_1) V^{t+1}(b_1^{t+1}) \right],$$

Were the belief b_1^t is a distribution over states and observation histories of agent 2: $b_1^t \in \mathcal{P}(\mathcal{S} \times \vec{\mathcal{O}}_2^t)$. In this way, all possible beliefs given the fixed policy of agent 2 are evaluated per time-step.

In contrast, the sequence form performs the maximization of equation 3.8. Which, in its substituted form, is:

			$t=0$		$t=1$			
			I_2^0		I_2^{Ro}		I_2^{Si}	
			a_Q	a_F	a_Q	a_O	a_Q	a_O
$t=0$	I_1^0	a_{Le}	-2	-0.1	0	0	0	0
		a_{Ri}	-2	-0.1	0	0	0	0
$t=1$	I_1^{Ri}	a_Q	0	0	-0.332	-0.663	-0.669	-1.337
		a_O	0	0	-0.663	-2.985	-1.337	$+3.985$
	I_1^{Le}	a_Q	0	0	-0.481	-0.962	-0.519	-1.038
		a_O	0	0	-0.962	-4.540	-1.038	$+3.540$

Table 9: The IR sequence form of the deaf and the blind. The matrix contains two sub-matrices for $t=0$ and $t=1$.

$$\pi_1^{BR} = \arg\max_{\pi_1} \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \sum_{\substack{n^o \in \mathcal{N}^o: \\ \sigma_1(n^o)=\sigma_{1,l} \wedge \\ \sigma_2(n^o)=\sigma_{2,k}}} \nu(n^o) \cdot O_1(n^o).$$

This maximization clearly is not performed in a time-step based manner. Therefore it is not possible to directly relate JESP and alternating maximization using the sequence form. In the next sub-section IRSF is introduced that then will be used to clarify the relation to JESP.

6.1 IRSF definition

In order to get closer to the JESP approach, we rewrite eq. 3.7, so that it performs a summation over time-steps. The complete deduction can be found in appendix B.2.

Let $\sigma_{1,l}^t = \left\langle \vec{\theta}_1^t, a_1^t \right\rangle$ be the l-th $t+1$-step sequence for agent 1, i.e., a sequence containing $t+1$ actions $a_1 \in \mathcal{A}_1$. Similarly, let $\sigma_{2,k}^t = \left\langle \vec{\theta}_2^t, a_2^t \right\rangle$ be the k-th $t+1$-step sequence for agent 2.

Then we can write the expected value for agent i of a joint policy as

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_l \rho_1^\pi(\sigma_{1,l}^t) \sum_k \rho_2^\pi(\sigma_{2,k}^t) \cdot r_{lk}^t, \tag{6.1}$$

where

$$r_{lk}^t = \sum_{s^t} R_i(s^t, \mathbf{a}^t) \cdot \nu(s^t, \vec{\theta}^t), \tag{6.2}$$

gives the expected immediate reward, weighted by $\nu(s^t, \vec{\theta}^t)$, which is nature's component of realizing state s^t *and* the joint action-observation history $\vec{\theta}^t = \left\langle \vec{\theta}_1^t, \vec{\theta}_2^t \right\rangle$, specified by the sequences. This probability is defined as:

$$\nu(s^t, \vec{\theta}^t) = \nu(s^t, \vec{\theta}^t \mid b^0) = \sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}), \tag{6.3}$$

Definition 6.1 The *immediate reward sequence form (IRSF) of a POSGs with horizon h* is given by h matrices for each agent. Let the matrices for agent 1 be $\mathbf{R}^0, \ldots, \mathbf{R}^{h-1}$, then the entries of a matrix $\mathbf{R}^t$ are given by equation 6.2.

Of course, it is also possible to merge the t different matrices into 1 larger one. Table 9 shows the IR sequence form for the deaf and the blind problem in 1 table.

Example 6.1 As an example, we will again consider the optimal policy $\pi_1 = I_1^0 : a_{Ri}$, $I_1^{Le} : *$, $I_1^{Ri} : a_O$, $\pi_2 = I_2^0 : a_F$, $I_2^{Ro} : a_Q$, $I_2^{Si} : a_O$ which we encountered before. Its expected value according to equation 6.1 then is:

$$-0.1 + (-0.663 + 3.985) = +3.222,$$

which is exactly what we calculated before. □

6.2 JESP vs. IRSF

Here we will relate the best-response function used by JESP to calculating a best-response in immediate reward sequence form. We start by noting that, in fact, a JESP belief is a conditional probability distribution:

$$b(\langle s^t, \vec{o}_{\neq i}^t \rangle) = P(s^t, \vec{o}_{\neq i}^t \mid \vec{\theta}_i^t, \pi_{\neq i}), \tag{6.4}$$

because $\pi_{\neq i}$ is a tuple of pure policies, we only need to consider $\vec{o}_{\neq i}^t$, the observation history of other agents, and not $\vec{\theta}_{\neq i}^t$, their full action-observation history. This corresponds to the fact that we can ignore certain columns (or rows) when calculating a best-response in sequence form. In the light of equation 6.1 it means that the summation is only over particular k (or l) because the $\rho_2^\pi(\sigma_{2,k}^t)$ is 0 for non-specified actions (thus sequences). In the light of equation 6.2 this means that the joint action-observation history $\vec{\theta}^t = \left\langle \vec{\theta}_1^t, \vec{\theta}_2^t \right\rangle$ specified by the sequences within this summation, is always consistent with π_2. Therefore it would be possible to rewrite these equation as follows, when calculating a best-response (for agent 1):

$$V_1(\pi_1 \mid \pi_2) = \sum_{t=0}^{h-1} \sum_l \rho_1^{\pi_1}(\sigma_{1,l}^t) \sum_{k \ s.t. \ \rho_2^{\pi_2}(\sigma_{2,k}^t)=1} r_{lk}^t, \tag{6.5}$$

where we now can write:

$$r_{lk}^t = \sum_{s^t} R_1(s^t, \mathbf{a}^t) \cdot \nu(s^t, \vec{\theta}_1^t, \vec{o}_2^t \mid \pi_2, b^0),$$

because we only need agent 2's observation history to calculate the probability of the joint sequence when we know the pure policy π_2. More elaborately put:

$$\begin{aligned} \nu(s^t, \vec{\theta}^t) &= \nu(s^t, \vec{\theta}_1^t, \vec{\theta}_2^t \mid b^0) \\ &= \nu(s^t, \vec{\theta}_1^t, \vec{o}_2^t \mid \pi_2, b^0). \end{aligned}$$

When restricting π_1 also to be a pure policy, we can write $\nu(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0)$. Which can be decomposed further to:

$$\nu^{\vec{\theta}^t}(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0) = \nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) \cdot \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0),$$

Here , we write $\nu^{\vec{\theta}^t}$ instead of ν to stress that it is nature's probability component of joint action-observation history $\vec{\theta}^t$. The components are given by:

$$\nu^{\vec{\theta}^t}(\vec{o}_1^{t'} \mid \pi_1, \pi_2, b^0) = \prod_{t'=0}^{t-1} P(o_1^{t'+1} | b_{\pi_2}^{\vec{\theta}_1^{t'}}, \pi_1(\vec{o}_1^{t'})),$$

where

$$P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)) = \sum_{s^{t+1}} \sum_{o_2^{t+1}} \sum_{\langle s^t, \vec{o}_2^t \rangle} P(s^{t+1}, o_1^{t+1}, o_2^{t+1} \mid s^t, \langle \pi_1(\vec{o}_1^t), \pi_2(\vec{o}_2^t) \rangle) b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t),$$

with $b_{\pi_2}^{\vec{\theta}_1^t} \in \mathcal{P}(\mathcal{S} \times \vec{\mathcal{O}}_2^t)$ is the belief agent 1 has at time-step t induced by his action-observation history and the knowledge he has about agent 2's policy. I.e. a belief as is used in JESP and defined in eq. 6.4.

Given the above definition of $\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)$, $\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0)$ is given by (proof in appendix A.3):

$$\begin{aligned}\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) &= \frac{\nu^{\vec{\theta}^t}(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0)}{\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)} \\ &\equiv b_{\pi_2}^{\vec{\theta}^t}(s^t, \vec{o}_2^t)\end{aligned}$$

So by substitution in the equation for $V_1(\pi_1 \mid \pi_2)$ (eq. 6.5) we get:

$$V_1(\pi_1 \mid \pi_2) = \sum_{t=0}^{h-1} \sum_l \rho_1^{\pi_1}(\sigma_{1,l}^t) \sum_{\substack{k \text{ s.t.} \\ \rho_2^{\pi_2}(\sigma_{2,k}^t)=1}} \sum_{s^t} R_1(s^t, \mathbf{a}^t) \nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0). \tag{6.6}$$

Now we will relate this to JESP's best-response calculation. Realizing that sequences correspond to tuples: $\sigma_{1,l}^t = \left\langle \vec{\theta}_1^t, a_1 \right\rangle$, and that, for a deterministic policy π_2, $\rho_2^{\pi_2}(\sigma_{2,k}^t) = 1 \leftrightarrow \sigma_{2,k}^t = \left\langle \vec{o}_2^t, \pi_2(\vec{o}_2^t) \right\rangle$, we can rewrite equation 6.6 to:

$$\begin{aligned}V_1(\pi_1 \mid \pi_2) = \sum_{t=0}^{h-1} \sum_{\langle \vec{\theta}_1^t, a_1 \rangle} \rho_1^{\pi_1}(\left\langle \vec{\theta}_1^t, a_1 \right\rangle) \sum_{\vec{o}_2^t} \sum_{s^t} R_1(s^t, \left\langle a_1, \pi_2(\vec{o}_2^t) \right\rangle) \cdot \\ \nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) \cdot \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0),\end{aligned}$$

Here $\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)$ can be moved to the front:

$$\begin{aligned}V_1(\pi_1 \mid \pi_2) = \sum_{t=0}^{h-1} \sum_{\langle \vec{\theta}_1^t, a_1 \rangle} \rho_1^{\pi_1}(\left\langle \vec{\theta}_1^t, a_1 \right\rangle) \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) \\ \sum_{s^t} \sum_{\vec{o}_2^t} R_1(s^t, \left\langle a_1, \pi_2(\vec{o}_2^t) \right\rangle) \cdot \nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0).\end{aligned}$$

Now combining this with equation 5.2 and substituting $\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0)$ with $b_{\pi_2}^{\vec{\theta}^t}(s^t, \vec{o}_2^t)$ gives a definition based on JESP beliefs:

$$V_1(\pi_1 | \pi_2) = \sum_{t=0}^{h-1} \sum_{\langle \vec{\theta}_1^t, a_1 \rangle} \rho_1^{\pi_1}(\left\langle \vec{\theta}_1^t, a_1 \right\rangle) \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) R_1(b_{\pi_2}^{\vec{\theta}_1^t}, a_1),$$

with the expected immediate reward as defined in JESP (eq.5.2):

$$R_1(b_{\pi_2}^{\vec{\theta}_1^t}, a_1) = \sum_{s^t} \sum_{\vec{o}_2^t} R(s^t, \left\langle a_1, \pi_2(\vec{o}_2^t) \right\rangle) b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t)$$

The calculation of a best-response is given by a maximization:

$$V_1^{BR}(\pi_2) = \max_{\pi_1} \sum_{t=0}^{h-1} \sum_{\langle \vec{\theta}_1^t, a_1 \rangle} \rho_1^{\pi_1}(\left\langle \vec{\theta}_1^t, a_1 \right\rangle) \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) R_1(b_{\pi_2}^{\vec{\theta}_1^t}, a_1).$$

θ	$\langle\theta_{1,1},\theta_{2,1}\rangle$		$\langle\theta_{1,1},\theta_{2,2}\rangle$		$\langle\theta_{1,2},\theta_{2,1}\rangle$		$\langle\theta_{1,2},\theta_{2,2}\rangle$	
$P(\theta)$	0.3		0.2		0.2		0.3	
	$\dot{a}_2$	$\bar{a}_2$	$\dot{a}_2$	$\bar{a}_2$	$\dot{a}_2$	$\bar{a}_2$	$\dot{a}_2$	$\bar{a}_2$
$\dot{a}_1$	3	2	2	1	3	2	0	1
$\bar{a}_1$	1	0	2	3	1	2	3	4

Table 10: A Bayesian game with identical payoffs.

Because we are calculating a pure policy π_1, for each observation history the realization weight of one action is 1. Therefore this can be rewritten to:

$$V_1^{BR}(\pi_2) = \max_{\pi_1} \sum_{t=0}^{h-1} \sum_{\vec{o}_1^t} \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) R_1(b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)).$$

It is intuitively clear that this is a non-recursive definition of V^0 according to 5.1. (For proof see appendix A.4.) Therefore we can conclude that the difference between alternating maximization using the immediate reward sequence form and JESP (and thus DCBRP) is that JESP decomposes nature's probability component in two parts, while IRSF doesn't. We conjecture that this is a minor difference and that the methods can be regarded as equivalent.

7 Bayesian Game approximation

In this section we will discuss the method for solving POSGs as outlined in [3, 4] and relate it to methods encountered earlier.

7.1 Bayesian games and POSGs

The approach is based on dividing the POSG in Bayesian games; one for each time-step.

A Bayesian game (BG) is defined as a tuple $\langle\Theta, \mathcal{A}, P, u\rangle$, with:

- m agents.
- $\Theta = \Theta_1 \times ... \times \Theta_m$, is the *type profile space* or *joint type space.* Θ_i is the *type space* for agent i. A *type*, $\theta_i \in \Theta_i$, defines the private information an agent i holds. $\theta \in \Theta$ denotes a *type profile* and $\theta_{\neq i}$ denotes the type of all agents but agent i.
- $\mathcal{A}$ is the set of joint actions.
- $P(\Theta)$ is a probability distribution over the type profile space. $P(\theta)$ is the probability of a specific type profile θ. The probabilities $P(\theta_i)$ and $P(\theta_{\neq i}|\theta_i)$ can also be extracted from $P(\Theta)$.
- $u = \langle u_1, ..., u_n\rangle$ is the collection of utility functions. $u_i(\mathbf{a}, \theta)$ gives the utility for agent i of joint action $\mathbf{a}$ under type profile θ. We also write $u_i(\langle a_i, a_{\neq i}\rangle, \langle\theta_i, \theta_{\neq i}\rangle)$.

Example 7.1 We will illustrate to concept of Bayesian games with a small example. Table 10 shows a Bayesian game with two agents. Both have two actions and two types, $\theta_{1,2}$ denotes the second type of agent 1. This leads to 4 type profiles, indicated in the top row, and their probabilities, shown in the bottom row.

For each type profile, the agents play a normal form game. In this example, we assume identical payoffs, so there is only one value listed per type profile and joint action. □

Now we can model a POSG as a sequence of Bayesian games (one for each time-step). We will consider the game for the t-th time-step of the POSG. Let θ_1^t denote a type of agent 1 in this game. Because the type of an agent in a Bayesian game corresponds to some private information he is holding, this corresponds to the action-observation history of the agent in the POSG, i.e.:

$$\theta_1^t \equiv \vec{\theta}_1^t.$$

In [3, 4] the BG approach for POSGs is actually only applied to Dec-POMDPs. At each time-step t, starting with $t = 0$, the agents solve the BG which, for the identical payoff case[10] (i.e. Dec-POMDPs) gives a conditional policy for time-step t, π_i^t, mapping from types (thus action-observation histories) to actions: $\pi_i^t : \vec{\Theta}_i^t \rightarrow \mathcal{A}_i$. Next, each agent i performs action, $\pi_i^t(\vec{\theta}_i^t)$, where $\vec{\theta}_i^t$ is agent i's true action-observation history. Thus the agents execute joint action $\pi^t(\vec{\theta}^t)$.

Clearly, this joint action $\pi^t(\vec{\theta}^t)$ is optimal when u^t, the utility function of the t-th time-step BG is optimal. I.e. when

$$u^t(\mathbf{a}, \vec{\theta}^t) = Q^*(\mathbf{a}, \vec{\theta}^t),$$

which is the expected value of performing $\mathbf{a}$ and following the optimal joint policy thereafter. Because Q^* is not known[11], [3, 4] employ heuristics such as $\mathrm{Q_{MDP}}$ to define the utility functions u^t.

The big advantage of assuming a heuristic utility function is available, is that this enables making a single forward pass (from $t = 0, ..., h-1$) solving the Bayesian games at each time-step. Normally one would have to consider all time-steps simultaneously: The expected value $Q(\mathbf{a}, \vec{\theta}^t)$ of performing $\mathbf{a}$ when the joint observable history is $\vec{\theta}^t$, depends on the actions you're taking at a later time, as well as on the actions taken earlier. This is why JESP and alternating maximization in sequence form have to perform a maximization over all time-steps before the agent is alternated.

An additional benefit is that performing a forward sweep through time allows the use of the knowledge of the policy at time-step t when reasoning about the next time-step $t+1$: The BGs will only have to include the action-observation histories that are consistent with the policies calculated for previous time-steps.

7.2 BG vs. IRSF

In the previous section we mentioned that at each time-step t the BG is solved to find a joint policy π^t, but did not explain exactly how this is done. Here we will explain this, directly relating this approach to the immediate reward sequence form.

In order to solve a Bayesian game it can be converted to normal form. The normal form is the matrix giving the expected outcome of all pure joint policies π. For an identical payoff game, this means that the optimal joint policy is simply the one with the highest payoff. However, a pure policy is a mapping from types to actions and as each type corresponds with an action-observation history, of which there are exponentially many, constructing this normal form is intractable.

Another approach, as used in [3, 4], is to convert the BG to sequence form. In a standard extensive form game, a sequence for an agent is defined as an information set of that agent plus an action to take at that information set. In a Bayesian game this corresponds with a type and an action to take for that type. In a Bayesian game used to approximate a time-step of a POSG a type is an action-observation history, which corresponds to an information set in the extensive form of the POSG. Therefore the notions coincide nicely.

[10] Formally, $\forall_{i,j} u_i = u_j = u$

[11] Calculating Q^* would require optimally solving the Bayesian games for all future time-steps $h-1, h-2, ..., t$.

		$\theta_{2,1}$		$\theta_{2,2}$	
		$\dot{a}_2$	$\bar{a}_2$	$\dot{a}_2$	$\bar{a}_2$
$\theta_{1,1}$	$\dot{a}_1$	0.9	0.6	0.4	0.2
	$\bar{a}_1$	0.3	0.0	0.4	0.6
$\theta_{1,2}$	$\dot{a}_1$	0.6	0.4	0.0	0.3
	$\bar{a}_1$	0.2	0.4	0.9	1.2

Table 11: Sequence form for the Bayesian game of example 7.1.

Example 7.2 We will illustrate the sequence form representation using the Bayesian game of example 7.1. Table 11 shows its sequence form. Here sequence $\langle \theta_{2,2}, \dot{a}_2 \rangle$ played against $\langle \theta_{1,1}, \dot{a}_1 \rangle$ corresponds with joint action $\langle \dot{a}_1, \dot{a}_2 \rangle$ for type profile $\langle \theta_{1,1}, \theta_{2,2} \rangle$, which gives an outcome of 1 as can be seen in table 10. However as $P(\langle \theta_{1,1}, \theta_{2,2} \rangle) = 0.2$, the outcome is weighted by this probability, therefore the entry has value 0.2.

The payoff of a joint pure policy is the sum of all entries the joint policy specifies. E.g. for the following joint policy:

$$\begin{aligned} \theta_{1,1} &\rightarrow a_{1,1}, \\ \theta_{1,2} &\rightarrow a_{1,2}, \\ \theta_{2,1} &\rightarrow a_{2,1}, \\ \theta_{2,2} &\rightarrow a_{2,2}, \end{aligned}$$

is given by $0.9 + 0.2 + 0.2 + 1.2 = 2.5$ □

However, because in the BG for a time-step of a POSG the types correspond with action-observation histories, a sequence for such a BG (type and an action to take for that type) is identical to a sequence for a POSG (an action-observation history and an action to take). This leads to the following observation:

> *The time-step t Bayesian game, used to approximate a POSG has the same form as $\mathbf{R}^t$, the t-th payoff matrix of the immediate reward sequence form.*

Meaning that the sequences are identical, but the entries differ. The entries of the IRSF are expected immediate rewards:

$$r^t(\vec{\theta}^t, \mathbf{a}) = r^t(\left\langle \vec{\theta}_1^t, a_1 \right\rangle, \left\langle \vec{\theta}_2^t, a_2 \right\rangle) = \sum_{s^t} R_i(s^t, \mathbf{a}^t) \cdot \nu(s^t, \vec{\theta}^t)$$

In contrast, the entry of the BG is a heuristic representing the optimal action value:

$$\begin{aligned} u^*(\left\langle \vec{\theta}_1^t, a_1 \right\rangle, \left\langle \vec{\theta}_2^t, a_2 \right\rangle) &\overset{def.}{\approx} Q^*(\left\langle \vec{\theta}_1^t, a_1 \right\rangle, \left\langle \vec{\theta}_1^t, a_2 \right\rangle) \\ &= r^t(\vec{\theta}^t, \mathbf{a}) + \sum_{\mathbf{o}} P(\mathbf{o}|\vec{\theta}^t, \mathbf{a}) V(\vec{\theta}^{t+1}, \pi^*(\vec{\theta}^{t+1})). \end{aligned}$$

However, to calculate this 'optimal heuristic', the optimal policy itself is needed.

A Proofs

A.1 Equivalence brute force and normal form solving

We will show that constructing and solving the normal form of a Dec-POMDP is equivalent to brute force joint policy evaluation. In order to prove this, we will need the following lemmas:

Lemma A.1 *The expected cumulative reward (value) of a decision node n_i^d in the extensive form of a POSG for some agent i under deterministic joint policy π, is given by:*

$$V_i^{l,\pi}(n_i^d) = \sum_{t=l}^{h-1} \sum_{\left(s^{l+1},\mathbf{o}^{l+1},\ldots,s^{h-1},\mathbf{o}^{h-1}\right)} R_i(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=l}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t)),$$

where l is the time-step of n_i^d, i.e. $t_{n_i^d} = l$ and s^l and $\mathbf{o}^l$ are specified by n_i^d.

Proof The node n_i^d specifies a sub-game of the full extensive form game: it is the root of a sub-tree of the full game-tree. Let $\mathcal{N}'^o \subseteq \mathcal{N}^o$ be the subset of outcome nodes within this sub-game (the outcome nodes reachable from n_i^d). The expected future reward starting from n_i^d, is the cumulative reward over the time-steps $l, ..., h-1$:

$$O_i'(n'^o) \equiv O_i'(\sigma(n'^o)) = \sum_{t=l}^{h-1} R_i(s^t, \mathbf{a}^t),$$

where $\sigma(n'^o)$ is the path from root (n_i^d) to outcome node n^0 in the sub-game. Similarly, nature's component of the probability of a path in the sub-game is adapted:

$$\nu'(n'^o) = \prod_{t=l}^{h-2} P(s^{t+1}|s^t, \mathbf{a}^t) \cdot P(\mathbf{o}|\mathbf{a}^t, s^{t+1}).$$

Using equation 3.2, we can now specify the value of a joint policy starting at n_i^d as

$$V_i^{l,\pi}(n_i^d) = \sum_{\sigma(n'^o) \in \sigma(\mathcal{N}'^o)} O_i'(n'^o) \cdot P(\sigma(n'^o)|\pi) \cdot \nu'(n'^o).$$

Because the joint policy is fixed and the summation is only over paths that are consistent with the joint policy, we can rewrite this as:

$$\begin{aligned} V_i^{l,\pi}(n_i^d) &= \sum_{\left(s^{l+1},\mathbf{o}^{l+1},\ldots,s^{h-1},\mathbf{o}^{h-1}\right)} O_i'(n'^o) \cdot \nu'(n'^o) \\ &= \sum_{\left(s^{l+1},\mathbf{o}^{l+1},\ldots,s^{h-1},\mathbf{o}^{h-1}\right)} \sum_{t=l}^{h-1} R_i(s^t, \pi(\vec{\mathbf{o}}^t)) \prod_{t=l}^{h-2} P(s^{t+1}|s^t, \pi(\mathbf{o}^t)) \\ &\quad \cdot P(\mathbf{o}^{t+1}|\pi(\vec{\mathbf{o}}^t), s^{t+1}) \end{aligned}$$

By contracting the conditional probabilities and swapping the sum-operators, we get the lemma. □

Lemma A.2 *In the extensive form of a POSG, the expected cumulative reward (value) of a decision node n_i^d for some agent i under deterministic joint policy π, can be expressed as the value $V_{i,\pi}^l(s, \vec{\mathbf{o}})$ of the state s and joint observation history $\vec{\mathbf{o}}$ specified by n_i^d. I.e., when n_i^d is a node for time-step $t = l$, then:*

$$\begin{aligned} V_i^{l,\pi}(n_i^d) &= V_i^{l,\pi}(s^l, \vec{\mathbf{o}}^l) \\ &= \sum_{t=l}^{h-1} \sum_{\substack{(s^{l+1},\mathbf{o}^{l+1},\ldots \\ ,s^{h-1},\mathbf{o}^{h-1})}} R_i(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=l}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t)) \end{aligned}$$

Proof We start with $V^{l}_{i,\pi}(s^l, \vec{\mathbf{o}}^l)$. At time l, $\langle s^l, \vec{\mathbf{o}}^l \rangle$ specifies a set $\mathcal{N}_{s^l,\vec{\mathbf{o}}^l}$ of decision nodes n^d_i: all n^d_i such that $s^l \in \sigma(n^d_i)$ and $\vec{\mathbf{o}}^l \in \sigma(n^d_i)$. Therefore we get:

$$V^{l,\pi}_i(s^l, \vec{\mathbf{o}}^l) = \sum_{n^d_i \in \mathcal{N}_{s^l,\vec{\mathbf{o}}^l}} P(\sigma(n^d_i)|s^l, \vec{\mathbf{o}}^l) V^l_{i,\pi}(n^d_i), \tag{A.1}$$

where $\mathcal{N}_{s^l,\vec{\mathbf{o}}^l} = \{n_i |\ s^l \in \sigma(n^d_i) \wedge \vec{\mathbf{o}}^l \in \sigma(n^d_i) \wedge t_{n^d_i} = l\}$. Next we turn to $V^l_{i,\pi}(n^d_i)$. Since n^d_i specifies a sub-tree with itself as the root, according to lemma A.1, the expected value is given by:

$$V^{l,\pi}_i(n^d_i) = \sum_{t=l}^{h-1} \sum_{\substack{(s^{l+1},\mathbf{o}^{l+1},\ldots \\ ,s^{h-1},\mathbf{o}^{h-1})}} R_i(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=l}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1}|s^t, \pi(\vec{\mathbf{o}}^t)),$$

where $s^l, \mathbf{o}^l$ are specified by n^d_i. However, as all nodes over which equation A.1 sums, specify the same $s^l, \mathbf{o}^l$, the value for these nodes is also equal. That is,

$$\forall_{n_i, n'_i \in \mathcal{N}_{s^l,\vec{\mathbf{o}}^l}} \quad V^{l,\pi}_i(n_i) = V^{l,\pi}_i(n'_i).$$

Therefore, when selecting n'^d_i as an arbitrary node from $\mathcal{N}_{s^l,\vec{\mathbf{o}}^l}$, equation A.1 reduces to:

$$\begin{aligned} V^{l,\pi}_i(s^l, \vec{\mathbf{o}}^l) &= V^{l,\pi}_i(n'^d_i) \sum_{n^d_i \in \mathcal{N}_{s^l,\vec{\mathbf{o}}^l}} P(\sigma(n^d_i)|s^l, \vec{\mathbf{o}}^l) \\ &= V^{l,\pi}_i(n'^d_i). \end{aligned}$$

Proving this lemma. □

Another way to interpret at this lemma is by noting that the states are Markov and that, because the joint action-observation history is identical, the actions executed by the agents must be the same. Now, we are ready to prove the main theorem.

Theorem A.1 *For a finite horizon Dec-POMDP, constructing and solving the normal form is equivalent to brute force search for the optimal joint policy. I.e. for each joint policy calculating the value using:*

$$V(\pi) \equiv V_i(\pi) = \sum_{\sigma(n^o) \in \sigma(\mathcal{N}^o)} O_i(n^o) \cdot P(\sigma(n^o)|\pi) \cdot \nu(n^o)$$

is equivalent with calculating the value of each joint policy using:

$$V^{0,\pi}(b^0) = \sum_{s \in \mathcal{S}} b^0(s) V^{0,\pi}(s, \vec{\mathbf{o}}^0),$$

where $\vec{\mathbf{o}}^0 = \langle \vec{o}_\emptyset, \vec{o}_\emptyset \rangle$ *and*

$$V^t_\pi(s, \vec{\mathbf{o}}^t) = R(s, \pi(\vec{\mathbf{o}}^t)) + \sum_{s' \in \mathcal{S}} P(s'|s, \pi(\vec{\mathbf{o}}^t)) \sum_{\mathbf{o} \in \mathcal{O}} P(\mathbf{o}|s^{t+1}, \pi(\vec{\mathbf{o}}^t)) \cdot V^{t+1}_\pi(s^{t+1}, \vec{\mathbf{o}}^{t+1}). \tag{A.2}$$

Proof Because $\forall_{i,j} \forall_{n^o}\ O_i(n^o) = O_j(n^o)$ in the extensive form of a Dec-POMDP, we immediately have $\forall_{i,j} \forall_\pi\ V_i(\pi) = V_j(\pi)$ which we define as $V(\pi)$. So we can use V_i of an arbitrary agent i. We start by splitting $V_i(\pi)$ in the first nature step (representing the initial state distribution) and the rest:

$$V_i(\pi) = V^\pi_i(n_{root}) = \sum_{n^d_1} P(n^d_1|n_{root}) V^{0,\pi}_i(n^d_1),$$

which by lemma A.2 reduces to:

$$V_i^{0,\pi}(n_{root}) = \sum_{s^0} P(s^0) \sum_{\vec{\mathbf{o}}^0} P(\vec{\mathbf{o}}^0|s^0) V_i^{0,\pi}(s^0, \vec{\mathbf{o}}^0).$$

When there is no initial observation (meaning $\vec{\mathbf{o}}^0 = \langle \vec{o}_\emptyset, ..., \vec{o}_\emptyset \rangle$), we can write this as:

$$V_i^{0,\pi}(n_{root}) = \sum_{s^0} P(s^0) V_i^{0,\pi}(s^0, \vec{\mathbf{o}}^0),$$

which we can also write as

$$V_i^{0,\pi}(b^0) = \sum_{s^0 \in \mathcal{S}} b(s^0) V_i^{0,\pi}(s^0, \vec{\mathbf{o}}^0).$$

So now, the only thing we need to do is to show that the definition of $V_i^{0,\pi}(s^0, \vec{\mathbf{o}}^0)$ according to lemma A.1 and A.2 coincide with equation A.2. From lemma A.1 and A.2 we get:

$$V_i^{0,\pi}(s^0, \vec{\mathbf{o}}^0) = \sum_{t=0}^{h-1} \sum_{\substack{(s^1, \mathbf{o}^1, ..., \\ s^{h-1}, \mathbf{o}^{h-1})}} R_i(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t)) \tag{A.3}$$

Because the index i in this expression only binds R_i, and we know that $\forall_{i,j} \forall_{s,\mathbf{a}}\ R_i(s, \mathbf{a}) = R_j(s, \mathbf{a})$, we can safely remove this index. When we also split the expression in a first step and the rest we get:

$$V^{0,\pi}(s^0, \vec{\mathbf{o}}^0) = \sum_{\substack{(s^1, \mathbf{o}^1, ..., \\ s^{h-1}, \mathbf{o}^{h-1})}} R(s^0, \pi(\vec{\mathbf{o}}^0)) \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t)) +$$
$$\sum_{t=1}^{h-1} \sum_{\substack{(s^1, \mathbf{o}^1, ..., \\ s^{h-1}, \mathbf{o}^{h-1})}} R(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t))$$

Here, $R(s^0, \pi(\vec{\mathbf{o}}^0))$ is a constant as both $s^0, \pi(\vec{\mathbf{o}}^0)$ are fixed for all paths over which is summed. Also, the probabilities of continuations of $s^0, \pi(\vec{\mathbf{o}}^0)$sum to 1, i.e.

$$\sum_{\substack{(s^1, \mathbf{o}^1, ..., \\ s^{h-1}, \mathbf{o}^{h-1})}} \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t)) = 1$$

Therefore we get:

$$V^{0,\pi}(s^0, \vec{\mathbf{o}}^0) = R(s^0, \pi(\vec{\mathbf{o}}^0)) + \sum_{t=1}^{h-1} \sum_{\substack{(s^1, \mathbf{o}^1, ..., \\ s^{h-1}, \mathbf{o}^{h-1})}} R(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t))$$

In this equation, we can also split the summation over paths into a summation over a first step and the rest:

$$V^{0,\pi}(s^0, \vec{\mathbf{o}}^0) = R(s^0, \pi(\vec{\mathbf{o}}^0)) + \sum_{t=1}^{h-1} \sum_{(s^1, \mathbf{o}^1)} \sum_{\substack{(s^2, \mathbf{o}^2, ..., \\ s^{h-1}, \mathbf{o}^{h-1})}} R(s^t, \pi(\vec{\mathbf{o}}^t)) \cdot \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \pi(\vec{\mathbf{o}}^t))$$

which can be rewritten as

$$V^{0,\pi}(s^0,\vec{\mathbf{o}}^{\,0}) = R(s^0,\pi(\vec{\mathbf{o}}^{\,0})) + \sum_{t=1}^{h-1}\sum_{s^1,\mathbf{o}^1} P(s^1,\mathbf{o}^1|s^0,\pi(\vec{\mathbf{o}}^{\,0})) \sum_{\substack{(s^2,\mathbf{o}^2,\ldots,\\ s^{h-1},\mathbf{o}^{h-1})}} R(s^t,\pi(\vec{\mathbf{o}}^{\,t}))\cdot\prod_{t=1}^{h-2} P(s^{t+1},\mathbf{o}^{t+1}|s^t,\pi(\vec{\mathbf{o}}^{\,t}))$$

and thus

$$V^{0,\pi}(s^0,\vec{\mathbf{o}}^{\,0}) = R(s^0,\pi(\vec{\mathbf{o}}^{\,0})) + \sum_{s^1,\mathbf{o}^1} P(s^1,\mathbf{o}^1|s^0,\pi(\vec{\mathbf{o}}^{\,0})) \sum_{t=1}^{h-1}\sum_{\substack{(s^2,\mathbf{o}^2,\ldots,\\ s^{h-1},\mathbf{o}^{h-1})}} R(s^t,\pi(\vec{\mathbf{o}}^{\,t}))\cdot\prod_{t=1}^{h-2} P(s^{t+1},\mathbf{o}^{t+1}|s^t,\pi(\vec{\mathbf{o}}^{\,t}))$$

which finally, with lemma A.2, reduces to:

$$V^{0,\pi}(s^0,\vec{\mathbf{o}}^{\,0}) = R(s^0,\pi(\vec{\mathbf{o}}^{\,0})) + \sum_{s^1,\mathbf{o}^1} P(s^1,\mathbf{o}^1|s^0,\pi(\vec{\mathbf{o}}^{\,0}))\cdot V^{1,\pi}(s^1,\mathbf{o}).$$

Realizing that the derivation of this equation from A.3 didn't depend on the fact that the initial value function ($V^{0,\pi}$) was considered, means that $V^{0,\pi}$ and $V^{1,\pi}$ can be replaced by $V^{t,\pi}$ and $V^{t+1,\pi}$ respectively, giving the theorem. □

A.2 Equivalence of JESP and DCBRP

Here we prove the equivalence of JESP [11] and DCBRP applied to an extensive form representation of a Dec-POMDP. We will assume that we are calculating a best-response policy for agent i and that the fixed policy of the other agents is given by $\pi_{\neq i}$. We first prove two lemmas in order to prove the main theorem.

Lemma A.3 *The functional form of the value function used by JESP and DCBRP are equivalent. I.e.*

$$V^{t,JESP} \equiv V^{t,DCBRP},$$

where $V^{t,JESP}$ is the JESP value function given by:

$$V^{t,JESP}(b_i^t) = \max_{a_i\in\mathcal{A}_i}\left[R(b_i^t,a_i) + \sum_{o_i\in\mathcal{O}_i} P(o_i|b_i^t,a_i)V^{t+1}(b_i^{t+1})\right], \tag{A.4}$$

with

$$R(b_i^t,a_i) = \sum_{\langle s^t,\vec{o}_{\neq i}^{\,t}\rangle} b_i^t(\langle s^t,\vec{o}_{\neq i}^{\,t}\rangle)R(s^t,\langle a_i,\pi_{\neq i}(\vec{o}_{\neq i}^{\,t})\rangle) \tag{A.5}$$

and $V^{t,DCBRP}$ is the standard POMDP value function, given by:

$$V^{t,DCBRP}(b^t) = \max_{a\in\mathcal{A}}\left[R(b^t,a) + \sum_{o\in\mathcal{O}} P(o|a,b^t)V^{t+1}(b^{t+1})\right],$$

with $R(b^t, a) = \sum_p R(p, a)b^t(p)$, as the POMDP is defined over states p that correspond with nodes for the protagonist agent.

Proof Because the states over which DCBRP defines the POMDP are denoted p, and the actions, observations and beliefs belong to the protagonist agent i we can rewrite $V^{t,DCBRP}$ to:

$$V^{t,DCBRP}(b_i^t) = \max_{a_i \in \mathcal{A}_i} \left[R(b_i^t, a_i) + \sum_{o_i \in \mathcal{O}_i} P(o_i|a_i, b_i^t) V^{t+1}(b_i^{t+1}) \right],$$

with $R(b_i^t, a_i) = \sum_p R(p, a_i)b(p)$. This equation is identical to eq. A.4, therefore we only need to show equivalence of the two definitions of immediate reward.

The states p are defined as either the decision nodes for agent i or outcome nodes. At time-step t, the set of nodes n_i (with $t_{n_i} = t$) that correspond with the states p^t is given by all paths $(n_{root}, s^0, \mathbf{o}^0, \mathbf{a}^0, ..., s^t, \mathbf{o}^t)$. Therefore a belief over states $b(p)$ actually corresponds to a belief over paths $b(n_{root}, s^0, \mathbf{o}^0, \mathbf{a}^0, ..., s^t, \mathbf{o}^t)$. As the policy of the other agents are deterministic policies and agent i knows his own action-observation history, this reduces to a belief over states and observation histories of other agents, $b(\left\langle \left(s^0, ..., s^t\right), \vec{o}_{\neq i}^t \right\rangle)$.[12] So we get:

$$R(b_i^t, a_i) = \sum_{\left\langle (s^0,...,s^t), \vec{o}_{\neq i}^t \right\rangle} b_i^t(\langle \left(s^0, ..., s^t\right), \vec{o}_{\neq i}^t \rangle) R(s^t, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle)$$

which, because R is a function of the current state s^t only, reduces to:

$$R(b_i^t, a_i) = \sum_{\langle s^t, \vec{o}_{\neq i}^t \rangle} b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle) R(s^t, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle) \tag{A.6}$$

with $b_i^t(\left\langle s^t, \vec{o}_{\neq i}^t \right\rangle) = \sum_{(s^0,...,s^{t-1})} b_i^t(\left\langle \left(s^0, ..., s^{t-1}, s^t\right), \vec{o}_{\neq i}^t \right\rangle)$. Equation A.6 is identical to the expected reward of a belief in JESP (eq. A.5). □

Lemma A.4 *The belief update performed by JESP is the same as performed by DCBRP for the extensive form of a Dec-POMDP. I.e., when b_i^{t+1}, is the updated belief resulting from b_i^t after action a_i and observing o_i:*

$$b_i^{t+1,JESP} \equiv b^{t+1,DCBRP},$$

where the JESP belief update is given by

$$b_i^{t+1,JESP}(\langle s^{t+1}, (\vec{o}_{\neq i}^t, \mathbf{o}_{\neq i}) \rangle) = \frac{1}{P(o_i|a_i, b_i^t)} \sum_{s^t} b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle) P(s^{t+1}, \langle o_i, \mathbf{o}_{\neq i} \rangle \mid s^t, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle), \tag{A.7}$$

with $P(o_i|b_i^t, a_i)$ is a normalizing constant. The DCBRP belief update, is the standard POMDP belief updates given by:

$$b^{t+1,DCBRP}(p') = \frac{P(o|p', a) \sum_p P(p'|p, a)b^t(p)}{P(o|a, b^t)},$$

with $P(o|a, b) = \sum_{p'} P(o|p', a) \sum_p P(p'|p, a)b(p)$ is the normalizing constant.

[12] As mentioned in a footnote in section 4.2 it is possible to specify p as the POMDP state representing the group of $|\mathcal{S}|^{t-1}$ nodes that specify the same $\vec{\theta}$ and state s^t. In this case the belief immediately reduced to $b(\langle s^t, \vec{o}_{\neq i}^t \rangle)$, a belief over the current state and observation histories of other agents.

Proof Again, we start by realizing that in DCBRP all actions, observations and beliefs are for agent i, we get by substitution:

$$b_i^{t+1}(p') = \frac{P(o_i|p', a_i)\sum_p P(p'|p, a_i)b^t(p)}{P(o_i|a_i, b_i^t)}.$$

As in lemma A.3, a belief $b^t(p)$ reduces to a belief over states and observation histories of other agents, $b^t(\left\langle \left(s^0, ..., s^t\right), \vec{o}_{\neq i}^t \right\rangle)$:

$$b_i^{t+1}(\left\langle \left(s^0, ..., s^{t+1}\right), \vec{o}_{\neq i}^{t+1} \right\rangle) = \frac{1}{P(o_i|a_i, b_i^t)} P(o_i| \left\langle \left(s^0, ..., s^{t+1}\right), \vec{o}_{\neq i}^{t+1} \right\rangle, a_i)$$
$$\sum_{\left\langle (s^0,...,s^t), \vec{o}_{\neq i}^t \right\rangle} P(\left\langle \left(s^0, ..., s^{t+1}\right), \vec{o}_{\neq i}^{t+1} \right\rangle \mid \langle \left(s^0, ..., s^t\right), \vec{o}_{\neq i}^t \rangle, a_i) \cdot b_i^t(\langle \left(s^0, ..., s^t\right), \vec{o}_{\neq i}^t \rangle),$$

which because of the Markov property reduces to:

$$b_i^{t+1}(\left\langle s^{t+1}, \vec{o}_{\neq i}^{t+1} \right\rangle) = \frac{1}{P(o_i|a_i, b_i^t)} P(o_i| \left\langle s^{t+1}, \vec{o}_{\neq i}^{t+1} \right\rangle, a_i)$$
$$\sum_{\langle s^t, \vec{o}_{\neq i}^t \rangle} P(\left\langle s^{t+1}, \vec{o}_{\neq i}^{t+1} \right\rangle \mid \langle s^t, \vec{o}_{\neq i}^t \rangle, a_i) \cdot b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle),$$

with $b_i^t(\left\langle s^t, \vec{o}_{\neq i}^t \right\rangle) = \sum_{(s^0,...,s^{t-1})} b_i^t(\left\langle \left(s^0, ..., s^{t-1}, s^t\right), \vec{o}_{\neq i}^t \right\rangle)$.[13]

Because $P(\left\langle s^{t+1}, \vec{o}_{\neq i}^{t+1} \right\rangle \mid \left\langle s^t, \vec{o}_{\neq i}^t \right\rangle, a_i) > 0$ only when $\vec{o}_{\neq i}^{t+1} = (\vec{o}_{\neq i}^t, \mathbf{o}_{\neq i})$ for some joint observation other agents $\mathbf{o}_{\neq i}$. ($\vec{o}_{\neq i}^{t+1}$ and $\vec{o}_{\neq i}^t$ need to specify the same action-observation history for the first t time-steps) we can write this as:

$$\begin{aligned}
b_i^{t+1}(\langle s^{t+1}, (\vec{o}_{\neq i}^t, \mathbf{o}_{\neq i}) \rangle) &= \frac{1}{P(o_i|a_i, b_i^t)} P(o_i| \langle s^{t+1}, (\vec{o}_{\neq i}^t, \mathbf{o}_{\neq i}) \rangle, a_i) \\
&\quad \sum_{s^t} P(\langle s^{t+1}, (\vec{o}_{\neq i}^t, \mathbf{o}_{\neq i}) \rangle \mid s^t, \vec{o}_{\neq i}^t, a_i) \cdot b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle) \\
&= \frac{1}{P(o_i|a_i, b_i^t)} P(o_i|s^{t+1}, \mathbf{o}_{\neq i}, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle) \\
&\quad \sum_{s^t} P(s^{t+1}, \mathbf{o}_{\neq i} \mid s^t, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle) \cdot b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle) \\
&= \frac{1}{P(o_i|a_i, b_i^t)} \sum_{s^t} P(o_i|s^{t+1}, \mathbf{o}_{\neq i}, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle) \\
&\quad P(s^{t+1}, \mathbf{o}_{\neq i} \mid s^t, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle) \cdot b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle),
\end{aligned}$$

which reduces to:

$$b_i^{t+1}(\langle s^{t+1}, (\vec{o}_{\neq i}^t, \mathbf{o}_{\neq i}) \rangle) = \frac{1}{P(o_i|a_i, b_i^t)} \sum_{s^t} P(s^{t+1}, \langle o_i, \mathbf{o}_{\neq i} \rangle \mid s^t, \langle a_i, \pi_{\neq i}(\vec{o}_{\neq i}^t) \rangle) \cdot b_i^t(\langle s^t, \vec{o}_{\neq i}^t \rangle),$$

which equals, $b_i^{t+1,JESP}$, the belief update in JESP (eq. A.7). □

[13] When p is specified as the POMDP state representing the group of $|\mathcal{S}|^{t-1}$ nodes that specify the same $\vec{\theta}$ and state s^t, i.e. when $b(p) = b(\langle s^t, \vec{o}_{\neq i}^t \rangle)$, this is immediately given by Bayes rule.

Theorem A.2 *The function 'OptimalPolicyDP' used in JESP for Dec-POMDPs is equivalent to direct calculation of best-response policies for extensive form games when applied to the extensive form representation of the same Dec-POMDP.*

Proof The belief update performed by both methods is identical (lemma A.4). Also the functional form of the value function defined by both methods is equivalent (lemma A.3). Therefore, for a given Dec-POMDP, the value function calculated by OptimalPolicyDP is identical to the value function calculated by DCBRP. □

A.3 Decomposition of nature's prob. component in IRSF

Here we show how the nature probability component $\nu(s^t, \vec{\theta}^t)$ can be split up in two parts $\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)$ and $\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0)$, that directly correspond to JESP's dynamic program. We start with the definition for $\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)$ and then show that under that definition, $\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0)$ corresponds to a belief as defined in JESP.

Definition A.1 Let nature's probability component $\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)$ of realizing $\vec{o}_1^t$ be defined as

$$\nu^{\vec{\theta}^{t'}}(\vec{o}_1^{t'} \mid \pi_1, \pi_2, b^0) = \prod_{t=0}^{t'-1} P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)),$$

where

$$P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)) = \sum_{s^{t+1}} \sum_{o_2^{t+1}} \sum_{\langle s^t, \vec{o}_2^t \rangle} P(s^{t+1}, o_1^{t+1}, o_2^{t+1} \mid s^t, \langle \pi_1(\vec{o}_1^t), \pi_2(\vec{o}_2^t) \rangle) b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t),$$

with $b_{\pi_2}^{\vec{\theta}_1^t}$ is the belief agent 1 has at time-step t induced by his action-observation history and the knowledge he has about agent 2's policy.

Now, let $\nu^{\vec{\theta}^t}(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0)$ be defined as in equation 6.3, i.e.:

$$\nu(s^t, \vec{\theta}^t | b^0) = \nu^{\vec{\theta}^t}(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0) = \sum_{(s^0, \ldots, s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}),$$

as was deduced in appendix B.2.

Lemma A.5 *When $\nu^{\vec{\theta}^t}(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0)$ and $\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)$ are defined as in equation 6.3 and definition A.1, then:*

$$\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) \equiv b_{\pi_2}^{\vec{\theta}^t}(s^t, \vec{o}_2^t)$$

Proof Per definition of condition probability we have

$$\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) = \frac{\nu^{\vec{\theta}^t}(s^t, \vec{o}_1^t, \vec{o}_2^t \mid \pi_1, \pi_2, b^0)}{\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0)}.$$

So by substitution we immediately get:

$$\nu^{\vec{\theta}^t}(s^t, \vec{o}_2^t \mid \vec{o}_1^t, \pi_1, \pi_2, b^0) =$$

$$\frac{\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \pi(\vec{\mathbf{o}}^{t'}))}{\prod_{t'=0}^{t-1} P(o_1^{t'+1} \mid b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t))} =$$

$$\frac{\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \vec{\mathbf{o}}^{t'}, \pi)}{\prod_{t'=0}^{t-1} \left[\sum_{s^{t+1}} \sum_{o_2^{t+1}} \sum_{\langle s^t, \vec{o}_2^t \rangle} P(s^{t+1}, o_1^{t+1}, o_2^{t+1} \mid s^t, \pi(\vec{\mathbf{o}}^t)) b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t)\right]} =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \frac{\prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \vec{\mathbf{o}}^{t'}, \pi)}{\prod_{t'=0}^{t-1} \left[\sum_{s^{t+1}} \sum_{o_2^{t+1}} \sum_{\langle s^t, \vec{o}_2^t \rangle} P(s^{t+1}, o_1^{t+1}, o_2^{t+1} \mid s^t, \pi(\vec{\mathbf{o}}^t)) b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t)\right]} =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} \frac{P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \vec{\mathbf{o}}^{t'}, \pi)}{\sum_{s^{t+1}} \sum_{o_2^{t+1}} \sum_{\langle s^t, \vec{o}_2^t \rangle} P(s^{t+1}, o_1^{t+1}, o_2^{t+1} \mid s^t, \pi(\vec{\mathbf{o}}^t)) b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t)} =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} \frac{P(s^{t'+1}, o_1^{t'+1}, o_2^{t'+1} \mid s^{t'}, \vec{o}_1^{t'}, \vec{o}_2^{t'}, \pi)}{\sum_{s^{t'}} \sum_{s^{t'+1}} \sum_{\vec{o}_2^{t'}} \sum_{o_2^{t'+1}} P(s^{t'+1}, o_1^{t'+1}, o_2^{t'+1} \mid s^{t'}, \vec{o}_1^{t'}, \vec{o}_2^{t'}, \pi) P(s^{t'}, \vec{o}_2^{t'} \mid \pi, \vec{o}_1^{t'})} =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} \frac{P(s^{t'+1}, o_1^{t'+1}, o_2^{t'+1} \mid s^{t'}, \vec{o}_1^{t'}, \vec{o}_2^{t'}, \pi)}{\sum_{s^{t'+1}} \sum_{o_2^{t'+1}} P(s^{t'+1}, o_1^{t'+1}, o_2^{t'+1} \mid \pi, \vec{o}_1^{t'})} =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} \frac{P(s^{t'+1}, o_1^{t'+1}, o_2^{t'+1} \mid s^{t'}, \vec{o}_1^{t'}, \vec{o}_2^{t'}, \pi)}{P(o_1^{t'+1} \mid \pi, \vec{o}_1^{t'})} =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, o_2^{t'+1} \mid s^{t'}, \vec{o}_1^{t'}, o_1^{t'+1}, \vec{o}_2^{t'}, \pi) =$$

$$\sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \left[P(s^t, o_2^t \mid s^{t-1}, \left(o_1^0, ..., o_1^t\right), \left(o_2^0, ..., o_2^{t-1}\right), \pi) \cdot \ldots \cdot P(s^1, o_2^1 \mid s^0, \left(o_1^0, o_1^1\right), \left(o_2^0\right), \pi)\right] =$$

$$\sum_{s^0} b^0(s^0) \sum_{(s^1,\ldots,s^{t-1})} P(s^1, ..., s^t, o_2^1, ..., o_2^t | s^0, \left(o_1^0, ..., o_1^t\right), \pi) =$$

$$\sum_{s^0} b^0(s^0) P(s^t, o_2^1, ..., o_2^t | s^0, \vec{o}_1^t, \pi) =$$

$$P(s^t, \vec{o}_2^t | \vec{o}_1^t, \pi) = P(s^t, \vec{o}_2^t | \vec{\theta}_1^t, \pi_2) \equiv b_{\pi_2}^{\vec{\theta}_1^t}(s^t, \vec{o}_2^t)$$

Which proves the lemma. □

A.4 JESP recursive vs. iterative formulation of V

Lemma A.6 *The iterative formulation of JESP's value function is given by:*

$$V_1^{BR}(\pi_2) = \max_{\pi_1} \sum_{t=0}^{h-1} \sum_{\vec{o}_1^t} \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) R_1(b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)).$$

and therefore is equivalent to $V_1^{0,*}(b_1^0)$ given by:

$$V_1^{t,*}(b_1^t) = \max_{a_1 \in \mathcal{A}_1} \left[R_1(b_1^t, a_1) + \sum_{o_1 \in \mathcal{O}_1} P(o_1 | b_1^t, a_1) V^{t+1}(b_1^{t+1}) \right].$$

Proof We will simply expand $V_1^{0,*}(b_1^0)$:

$$\begin{aligned}
V_1^{0,*}(b_1^0) &= \max_{a_1^0 \in \mathcal{A}_1} \left[R(b_1^0, a_1^0) + \sum_{o_1^1 \in \mathcal{O}_1} P(o_1^1 | b_1^0, a_1^0) \max_{a_1^1 \in \mathcal{A}_1} \left(R(b_1^1, a_1^1) + \sum_{o_1^2 \in \mathcal{O}_1} P(o_1^2 | b_1^1, a_1^1) \right. \right. \\
&\quad \left. \left. \left[... \left(... \sum_{o_1^2 \in \mathcal{O}_1} P(o_1^{h-1} | b_1^{h-2}, a_1^{h-2}) \max_{a_1^{h-1} \in \mathcal{A}_1} \left[R(b_1^{h-1}, a_1^{h-1}) \right] \right) ... \right] \right) \right] \\
&= \max_{\pi_1} \left[R(b_{\pi_2}^{\vec{\theta}_1^0}, \pi_1(\vec{\theta}_1^0)) + \sum_{o_1^1 \in \mathcal{O}_1} P(o_1^1 | b_{\pi_2}^{\vec{\theta}_1^0}, \pi_1(\vec{\theta}_1^0)) \left(R(b_{\pi_2}^{\vec{\theta}_1^1}, \pi_1(\vec{\theta}_1^1)) + \right. \right. \\
&\quad \sum_{o_1^2 \in \mathcal{O}_1} P(o_1^2 | b_{\pi_2}^{\vec{\theta}_1^1}, \pi_1(\vec{\theta}_1^1)) \left[... \left(... \sum_{o_1^{h-1} \in \mathcal{O}_1} P(o_1^{h-1} | b_{\pi_2}^{\vec{\theta}_1^{h-2}}, \pi_1(\vec{\theta}_1^{h-2})) \right. \right. \\
&\quad \left. \left. \left. \left[R(b_{\pi_2}^{\vec{\theta}_1^{h-1}}, \pi_1(\vec{\theta}_1^{h-1}) \right] \right) ... \right] \right) \right],
\end{aligned}$$

where $b_{\pi_2}^{\vec{\theta}_1^t}$ is the belief corresponding to the action-observation history $\vec{\theta}_1^t$.

We now write a bit shorter: $R^0, ..., R^{h-1}$and $p(o^0), ..., p(o^{h-1})$. Note they still depend on the beliefs and therefore differ within each time-step. We now write:

$$\begin{aligned}
V_1^{0,*}(b_1^0) &= \max_{\pi_1} \left[R^0 + \sum_{o^1} p(o^1) \left(R^1 + \sum_{o^2} p(o^2) \left[... \sum_{o^{h-1}} p(o^{h-1}) \left(R^{h-1} \right) ... \right] \right) \right] \\
&= \max_{\pi_1} \left[R^0 + \sum_{o^1} \left(p(o^1) R^1 + p(o^1) \sum_{o^2} p(o^2) \left[... p(o^{h-1}) \sum_{o^{h-1}} p(o^{h-1}) R^{h-1} \right] \right) \right] \\
&= \max_{\pi_1} \left[R^0 + \sum_{o^1} p(o^1) R^1 + \sum_{o^1} \sum_{o^2} p(o^1) p(o^2) R^2 + ... + \right. \\
&\quad \left. \sum_{o^1} \sum_{o^2} ... \sum_{o^{h-1}} p(o^1) p(o^2) .... p(o^{h-1}) R^{h-1} \right].
\end{aligned}$$

Here

$$\begin{aligned}
\sum_{o^1} \sum_{o^2} p(o^1) p(o^2) &= \sum_{o_1^1 \in \mathcal{O}_1} \sum_{o_1^2 \in \mathcal{O}_1} P(o_1^1 | b_{\pi_2}^{\vec{\theta}_1^0}, \pi_1(\vec{\theta}_1^0)) P(o_1^2 | b_{\pi_2}^{\vec{\theta}_1^1}, \pi_1(\vec{\theta}_1^1)) \\
&= \sum_{\langle o_1^1, o_1^2 \rangle} P(\left(o_1^1, o_1^2\right) | b_{\pi_2}^{\vec{\theta}_1^0}, b_{\pi_2}^{\vec{\theta}_1^1}, \pi_1)
\end{aligned}$$

and more general

$$\sum_{o_1^1, ..., o_1^t} p(o_1^1) ... p(o_1^t) R^t = \sum_{\vec{o}_1^t} \prod_{t'=0}^{t-1} P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{\theta}_1^t)).$$

So we can write

$$\begin{aligned} V_1^{0,*}(b_1^0) &= \max_{\pi_1} \sum_{t=0}^{h-1} \left[\sum_{\vec{o}_1^t} \left(\prod_{t'=0}^{t-1} P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{\theta}_1^t)) \right) R(b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{\theta}_1^t)) \right] \\ &= \max_{\pi_1} \sum_{t=0}^{h-1} \left[\sum_{\vec{o}_1^t} \left(\prod_{t'=0}^{t-1} P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)) \right) R(b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)) \right], \end{aligned}$$

if we want to find a deterministic best-response policy.

As the nature probability component is defined as

$$\nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) = \prod_{t'=0}^{t-1} P(o_1^{t+1} | b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)),$$

substitution gives

$$V_1^{0,*}(b_1^0) = \max_{\pi_1} \sum_{t=0}^{h-1} \sum_{\vec{o}_1^t} \nu^{\vec{\theta}^t}(\vec{o}_1^t \mid \pi_1, \pi_2, b^0) R(b_{\pi_2}^{\vec{\theta}_1^t}, \pi_1(\vec{o}_1^t)),$$

proving the lemma. □

B Calculations and derivations

B.1 Calculation of normal form

Table 12 shows the normal form of the deaf and the blind problem. In non-trivial cases, the performed calculation is shown using some new abbreviations (to fit it on the page). The abbreviations have the following meaning: $l = P(s_l)$ and $r = P(s_r)$ are the state probabilities (given by b^0), RlL, SlL, RlR, etc are observation probabilities with the following meaning: $SlL = P(o_{Si}|s = s_{lL})$, $RlR = P(o_{Ro}|s = s_{lR})$, etc.

B.2 Deduction of IR sequence form

Here we show the deduction of the immediate reward sequence form from the regular sequence form. In sequence form the value for agent i, $V_i(\pi)$, of a joint policy π is given by:

$$V_i(\pi) = \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \cdot \mathrm{r}_{lk}$$

where

$$\mathrm{r}_{lk} = \sum_{n^o \in \mathcal{N}^o \; s.t. \; \sigma_1(n^o)=\sigma_{1,l} \wedge \sigma_2(n^o)=\sigma_{2,k}} \nu(n^o) \cdot O_i(n^o).$$

where

$$O_i(n^o) \equiv O_i(\sigma(n^o)) = \sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t),$$

Let $n^o \in \mathcal{N}^o : \sigma_1(n^o) = \sigma_{1,l} \wedge \sigma_2(n^o) = \sigma_{2,k}$ the set of paths consistent with sequence l and k, be denoted by C_{jk}. This means that the value is given by :

$$V_i(\pi) = \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \cdot \sum_{\sigma(n^o) \in C_{jk}} \nu(n^o) \cdot O_i(n^o). \tag{B.1}$$

	$I_2^0 : a_F$ $I_2^{Ro} : a_O$ $I_2^{Si} : a_O$	$I_2^0 : a_F$ $I_2^{Ro} : a_Q$ $I_2^{Si} : a_O$	$I_2^0 : a_F$ $I_2^{Ro} : a_O$ $I_2^{Si} : a_Q$	$I_2^0 : a_F$ $I_2^{Ro} : a_Q$ $I_2^{Si} : a_Q$	$I_2^0 : a_Q$ $I_2^{Ro} : *$ $I_2^{Si} : *$
$I_1^0 : a_{Le}$ $I_1^{Le} : a_O$ $I_1^{Ri} : *$	$-10.1l\ RlL+$ $-10.1l\ SlL+$ $9.9r\ RrL+$ $9.9r\ SrL =$ $\mathbf{-1.100}$	$-2.1l\ RlL+$ $-10.1l\ SlL+$ $-2.2r\ RrL+$ $9.9r\ SrL =$ $\mathbf{+2.478}$	$-10.1l\ RlL+$ $-2.1l\ SlL+$ $9.9r\ RrL+$ $-2.1r\ SrL =$ $\mathbf{-5.678}$	$\mathbf{-2.1}$	$\mathbf{-2}$
$I_1^0 : a_{Le}$ $I_1^{Le} : a_Q$ $I_1^{Ri} : *$	$\mathbf{-2.1}$	$-1.1l\ RlL+$ $-2.1l\ SlL+$ $-1.1r\ RrL+$ $-2.1r\ SrL =$ $\mathbf{-1.619}$	$-2.1l\ RlL+$ $-1.1l\ SlL+$ $-2.1r\ RrL+$ $-1.1r\ SrL =$ $\mathbf{-1.581}$	$\mathbf{-1.1}$	$\mathbf{-2}$
$I_1^0 : a_{Ri}$ $I_1^{Le} : *$ $I_1^{Ri} : a_O$	$9.9l\ RlR+$ $9.9l\ SlR+$ $-10.1r\ RrR+$ $-10.1r\ SrR =$ $\mathbf{+0.900}$	$-2.1l\ RlR+$ $9.9l\ SlR+$ $-2.1r\ RrR+$ $-10.1r\ SrR =$ $\mathbf{+3.222}$	$9.9l\ RlR+$ $-2.1l\ SlR+$ $-10.1r\ RrR+$ $-2.1r\ SrR =$ $\mathbf{-4.422}$	$\mathbf{-2.1}$	$\mathbf{-2}$
$I_1^0 : a_{Ri}$ $I_1^{Le} : *$ $I_1^{Ri} : a_Q$	$\mathbf{-2.1}$	$-1.1l\ RlR+$ $-2.1l\ SlR+$ $-1.1r\ RrR+$ $-2.1r\ SrR =$ $\mathbf{-1.769}$	$-2.1l\ RlR+$ $-1.1l\ SlR+$ $-2.1r\ RrR+$ $-1.1r\ SrR =$ $\mathbf{-1.431}$	$\mathbf{-1.1}$	$\mathbf{-2}$

Table 12: Normal form representation of the deaf and the blind problem. In order to fit the table to the page (the full normal form would be 8×8), pure strategies specifying the same behavior have been collapsed. The actions where these collapsed policies differ are indicated with a *, which therefore can be interpreted as a wild-card.

Now, as we know that the outcomes of the extensive form of a POSG are in fact the sum of the rewards at each time-step, we want to rewrite this to express the value as a sum over time-steps.

The summations over sequences in equation B.1 are in fact summations over sequences of different lengths: some sequences specify 1 action, e.g. a sequence $\left\langle \vec{\theta}_1^0, a_1 \right\rangle = \langle (o_1^0), a_1^0 \rangle$, some specify h actions, e.g. $\left\langle \vec{\theta}_1^{h-1}, a_1 \right\rangle = \left\langle (o_1^0, a_1^0, ..., o_1^{h-1}), a_1^{h-1} \right\rangle$. However, only the paths leading to an outcome node that are consistent with a choice of two sequences (one for each agent) specify an outcome. We will assume here that this is the case for h-step sequences (sequences that specify h actions). When our goal — expressing the value of the policy as a sum over time-steps — is achieved, generalization follows trivially.

First make the obvious substitutions in equation B.1, starting with the outcome $O_i(n^o)$:

$$V_i(\pi) = \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \cdot \sum_{\sigma(n^o) \in C_{jk}} \nu(n^o) \cdot \sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t).$$

Also as nature's probability component $\nu(n^o) = \nu(\sigma(n^o))$ is given by:

$$\begin{aligned} \nu(\sigma(n^o)) &= b^0(s^0) \prod_{t=0}^{h-2} P(s^{t+1}|s^t, \mathbf{a}^t) \cdot P(\mathbf{o}^{t+1}|\mathbf{a}^t, s^{t+1}) \\ &= b^0(s^0) \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1}|s^t, \mathbf{a}^t), \end{aligned}$$

where the states, joint actions and observations, $s^t, \mathbf{a}^t, \mathbf{o}^t$ are specified by the path $\sigma(n^o)$. Therefore we get:

$$V_i(\pi) = \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \cdot \sum_{\sigma(n^o) \in C_{jk}} \sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t) \cdot b^0(s^0) \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1}|s^t, \mathbf{a}^t).$$

Because the two chosen sequences fully specify the joint observable history, the set of consistent paths C_{jk} can only vary on the actual states that are chosen, so we can write:

$$V_i(\pi) = \sum_l \rho_1^\pi(\sigma_{1,l}) \sum_k \rho_2^\pi(\sigma_{2,k}) \sum_{(s^0,...,s^{h-1})} \sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t) \cdot b^0(s^0) \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1}|s^t, \mathbf{a}^t),$$

here the reward $\sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t)$ is 0 when either $\sigma_{1,l}$ or $\sigma_{2,k}$ is not a h-step sequence.[14] (If one of the sequences contains less steps, $\mathbf{o}^{h-1}$and $\mathbf{a}^{h-1}$ are not specified.)

Let the joint realization weight for a horizon t joint sequence be given by:

$$\begin{aligned} \rho^\pi(\langle (\mathbf{o}^0, \mathbf{a}^0, ..., \mathbf{o}^{t-1}), \mathbf{a}^{t-1} \rangle) &= \rho^\pi(\left\langle \vec{\theta}^{t-1}, \mathbf{a}^{t-1} \right\rangle) \\ &= \rho_1^\pi(\langle (o_1^0, a_1^0, ..., o_1^{t-1}), a_1^{t-1} \rangle) \cdot \rho_2^\pi(\langle (o_2^0, a_2^0, ..., o_2^{t-1}), a_2^{t-1} \rangle) \end{aligned}$$

This allows us to write the summation over sequences as a summation over joint horizon h sequences:

$$V_i(\pi) = \sum_{\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle} \rho^\pi(\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle) \sum_{(s^0,...,s^{h-1})} \sum_{t=0}^{h-1} R_i(s^t, \mathbf{a}^t) \cdot b^0(s^0) \prod_{t=0}^{h-2} P(s^{t+1} \mathbf{o}^{t+1}|s^t, \mathbf{a}^t).$$

[14] Remember we assumed that only h-step paths and thus joint sequences specify an outcome.

Bringing the summation over time to the front gives:

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_{\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle} \rho^\pi(\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle) \sum_{(s^0,\ldots,s^{h-1})} R_i(s^t, \mathbf{a}^t) \cdot b^0(s^0) \prod_{t=0}^{h-2} P(s^{t+1}, \mathbf{o}^{t+1} | s^t, \mathbf{a}^t).$$

Then noting that the reward at time step t, $R_i(s^t, \mathbf{a}^t)$, is independent from the exact continuation $(s^{t+1}, \mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \ ..., s^{h-1}, \mathbf{o}^{h-1}, \mathbf{a}^{h-1})$ gives:

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_{\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle} \sum_{\left\langle \mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \ldots, \mathbf{o}^{h-1}, \mathbf{a}^{h-1} \right\rangle} \rho^\pi(\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle)$$
$$\sum_{(s^0,\ldots,s^t)} \sum_{(s^{t+1},\ldots,s^{h-1})} R_i(s^t, \mathbf{a}^t) \cdot b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}) \prod_{t'=t}^{h-2} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}),$$

which becomes:

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_{\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle} \sum_{(s^0,\ldots,s^t)} R_i(s^t, \mathbf{a}^t) \left[b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}) \right]$$
$$\sum_{(\mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \ldots, \mathbf{o}^{h-1}, \mathbf{a}^{h-1})} \rho^\pi(\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle) \sum_{(s^{t+1},\ldots,s^{h-1})} \prod_{t'=t}^{h-2} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}).$$

Because a realization weight is a product over time-steps (as in equation 3.5), we can express the realization weight of a h-step joint sequence as follows:

$$\rho^\pi(\left\langle \vec{\theta}^{h-1}, \mathbf{a}^{h-1} \right\rangle = \rho^\pi(\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle \cdot \prod_{t'=t}^{h-2} P(\mathbf{a}^{t'+1} | \vec{\theta}^{t'+1}, \pi)$$

This allows us to further rewrite to:

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_{\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle} \rho^\pi(\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle \sum_{(s^0,\ldots,s^t)} \left[b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1} \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}) \right] R_i(s^t, \mathbf{a}^t)$$
$$\sum_{\substack{(\mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \\ \ldots, \mathbf{o}^{h-1}, \mathbf{a}^{h-1})}} \left[\prod_{t'=t}^{h-2} P(\mathbf{a}^{t'+1} | \vec{\theta}^{t'+1}, \pi) \right] \sum_{\substack{(s^{t+1}, \\ \ldots, s^{h-1})}} \prod_{t'=t}^{h-2} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}),$$

where the last part

$$\sum_{\substack{(\mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \\ \ldots, \mathbf{o}^{h-1}, \mathbf{a}^{h-1})}} \left[\prod_{t'=t}^{h-2} P(\mathbf{a}^{t'+1} | \vec{\theta}^{t'+1}, \pi) \right] \sum_{\substack{(s^{t+1}, \\ \ldots, s^{h-1})}} \prod_{t'=t}^{h-2} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'})$$
$$= \sum_{\substack{(s^{t+1}, \mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \\ \ldots, s^{h-1}, \mathbf{o}^{h-1}, \mathbf{a}^{h-1})}} \left[\prod_{t'=t}^{h-2} P(\mathbf{a}^{t'+1} | \vec{\theta}^{t'+1}, \pi) \right] \prod_{t'=t}^{h-2} P(s^{t'+1}, \mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'})$$
$$= \sum_{\substack{(s^{t+1}, \mathbf{o}^{t+1}, \mathbf{a}^{t+1}, \\ \ldots, s^{h-1}, \mathbf{o}^{h-1}, \mathbf{a}^{h-1})}} \prod_{t'=t}^{h-2} P(s^{t'+1}, \mathbf{o}^{t'+1}, \mathbf{a}^{t'+1} | s^{t'}, \mathbf{a}^{t'}, \vec{\theta}^{t'+1}, \pi)$$
$$= 1.$$

So we get:

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_{\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle} \rho^\pi(\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle \sum_{(s^0,\ldots,s^t)} \left[b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}\mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}) \right] R_i(s^t, \mathbf{a}^t).$$

Finally, we more conveniently notate this as:

$$V_i(\pi) = \sum_{t=0}^{h-1} \sum_{\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle} \rho^\pi(\left\langle \vec{\theta}^t, \mathbf{a}^t \right\rangle \sum_{s^t} R_i(s^t, \mathbf{a}^t) \cdot \nu(s^t, \vec{\theta}^t),$$

where

$$\nu(s^t, \vec{\theta}^t) = \sum_{(s^0,\ldots,s^{t-1})} b^0(s^0) \prod_{t'=0}^{t-1} P(s^{t'+1}\mathbf{o}^{t'+1} | s^{t'}, \mathbf{a}^{t'}),$$

is nature's component of realizing $\vec{\theta}^t$ *and* s^t.

References

[1] Daniel S. Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of Markov decision processes. *Math. Oper. Res.*, 27(4):819–840, 2002.

[2] Anthony Rocco Cassandra. *Exact and approximate algorithms for partially observable Markov decision processes*. PhD thesis, Brown University, 1998. Adviser-Leslie Pack Kaelbling.

[3] Rosemary Emery-Montemerlo, Geoff Gordon, Jeff Schneider, and Sebastian Thrun. Approximate solutions for partially observable stochastic games with common payoffs. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 136–143, Washington, DC, USA, 2004. IEEE Computer Society.

[4] Rosemary Emery-Montemerlo, Geoff Gordon, Jeff Schneider, and Sebastian Thrun. Game theoretic control for robot teams. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1175–1181, 2005.

[5] Claudia V. Goldman and Shlomo Zilberstein. Optimizing information exchange in cooperative multi-agent systems. In *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 137–144, New York, NY, USA, 2003. ACM Press.

[6] Claudia V. Goldman and Shlomo Zilberstein. Decentralized control of cooperative systems: Categorization and complexity analysis. *Journal of Artificial Intelligence Research (JAIR)*, 22:143–174, 2004.

[7] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artif. Intell.*, 101(1-2):99–134, 1998.

[8] Daphne Koller, Nimrod Megiddo, and Bernhard von Stengel. Fast algorithms for finding randomized strategies in game trees. In *Proc. of the 26th ACM Symposium on Theory of Computing (STOC)*, pages 750–759, 1994.

[9] Daphne Koller and Avi Pfeffer. Representations and solutions for game-theoretic problems. *Artificial Intelligence*, 94(1-2):167–215, 1997.

[10] C. E. Lemke and J. T. Howson. Equilibrium points in bimatrix games. *Journal of the Society for Industrial and Applied Mathematics*, 12:413–423, 1964.

[11] Ranjit Nair, Milind Tambe, Makoto Yokoo, David V. Pynadath, and Stacy Marsella. Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 705–711, 2003.

[12] Andrew Y. Ng and Michael I. Jordan. Pegasus: A policy search method for large mdps and pomdps. In *Proc. of Uncertainty in Artificial Intelligence*, pages 406–415, 2000.

[13] Frans Oliehoek, Matthijs T. J. Spaan, and Nikos Vlassis. Best-response play in partially observable card games. In *Benelearn 2005: Proceedings of the 14th Annual Machine Learning Conference of Belgium and the Netherlands*, pages 45–50, February 2005.

[14] Frans A. Oliehoek. Game theory and AI: a unified approach to poker games. Master's thesis, University of Amsterdam, September 2005.

[15] Christos H. Papadimitriou and John N. Tsitsiklis. The complexity of Markov decision processes. *Mathematics of Operations Research*, 12(3):441–451, 1987.

[16] Ryan Porter, Eugene Nudelman, and Yoav Shoham. Simple search methods for finding a Nash equilibrium. *Games and Economic Behavior*, (to appear).

[17] David V. Pynadath and Milind Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of AI research (JAIR)*, 16:389–423, 2002.

[18] R. Tyrrell Rockafellar. *Convex analysis*. Princeton, N.J., Princeton University Press, 1970.

[19] Matthijs T. J. Spaan and Nikos Vlassis. Perseus: Randomized point-based value iteration for POMDPs. *Journal of Artificial Intelligence Research*, 24:195–220, 2005.

[20] R.J. Vanderbei. *Linear Programming: Foundations and Extensions*. Kluwer Academic Publishers, 1996.

[21] J. von Neumann and O. Morgenstern. *The Theory of Games and Economic Behavior*. Princeton University Press, 1947. 2nd edition.

[22] P. Xuan, V. Lesser, and S. Zilberstein. Communication decisions in multi-agent cooperation: Model and experiments. In *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 616–623. ACM Press, 2001.

Acknowledgements

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.

IAS reports

This report is in the series of IAS technical reports. The series editor is Stephan ten Hagen (`stephanh@science.uva.nl`). Within this series the following titles appeared:

G. Pavlin and J. Nunnink and F. Groen *Inference meta models: A new perspective on belief propagation with bayesian networks.* Technical Report IAS-UVA-06-01, Informatics Institute, University of Amsterdam, The Netherlands, March 2006.

Z. Zivkovic and O. Booij *How did we built our hyperbolic mirror omni-directional camera - practical issues and basic geometry.* Technical Report IAS-UVA-05-04, Informatics Institute, University of Amsterdam, The Netherlands, December 2005.

A. Diplaros and T. Gevers and N. Vlassis *An efficient spatially constrained EM algorithm for image segmentation.* Technical Report IAS-UVA-05-03, Informatics Institute, University of Amsterdam, The Netherlands, December 2005.

All IAS technical reports are available for download at the IAS website, `http://www.science.uva.nl/research/ias/publications/reports/`.