UNIVERSITEIT VAN AMSTERDAM

IAS technical report IAS-UVA-14-01

Dec-POMDPs as Non-Observable MDPs

Frans A. Oliehoek[1] **and Christopher Amato**[2]

[1]Intelligent Systems Laboratory Amsterdam, University of Amsterdam.
[1]Department of Computer Science, University of Liverpool.
[2]CSAIL, MIT.

A recent insight in the field of decentralized partially observable Markov decision processes (Dec-POMDPs) is that it is possible to convert a Dec-POMDP to a non-observable MDP, which is a special case of POMDP. This technical report provides an overview of this reduction and pointers to related literature.

IAS
intelligent autonomous systems

Contents

Intelligent Autonomous Systems
Informatics Institute, Faculty of Science
University of Amsterdam
Science Park 904, 1098 XH Amsterdam
The Netherlands

Tel (fax): +31 20 525 7463
`http://isla.science.uva.nl/`

Corresponding author:

F.A. Oliehoek
tel: +31 20 525 8093
`F.A.Oliehoek@uva.nl`
`http://www.fransoliehoek.net`

1 Introduction

A recent insight in the AI literature on decentralized decision-theoretical planning is that *decentralized POMDPs (Dec-POMDPs)*, can be converted to a special case of centralized model. This centralized model is a non-observable MDP (NOMDP) and has parts of joint policies (joint decision rules) as its actions, essentially capturing the planning process itself as a decision problem. Simultaneously, this insight has been (re-)discovered[1] in the decentralized control literature under the name of *the designer approach* [Mahajan and Mannan, 2014].

This document gives an overview of this reduction and provides some pointers to related literature.

2 Background

This report deals with decision making in the context of the decentralized POMDP (Dec-POMDP) framework [Bernstein et al., 2002, Oliehoek, 2012, Amato et al., 2013]. Here we will use a slightly different formalization that will make more explicit all the constraints specified by this model, and how the approach can be generalized (e.g., to deal with different assumptions with respect to communication). We begin by defining the environment of the agents:

Definition 1 (Markov multiagent environment)**.** The *Markov multiagent environment (MME)* is defined as a tuple $\mathcal{M} = \langle \mathcal{D},\mathcal{S},\boldsymbol{\mathcal{A}},T,\boldsymbol{\mathcal{O}},O,R,h,b_0 \rangle$, where

- $\mathcal{D} = \{1,\ldots,n\}$ is the set of n agents.
- $\mathcal{S}$ is a (finite) set of states.
- $\boldsymbol{\mathcal{A}}$ is the set of joint actions.
- T is the transition probability function.
- R is the set of immediate reward functions for all agents.
- $\boldsymbol{\mathcal{O}}$ is the set of joint observations.
- O is the observation probability function.
- h is the horizon of the problem as mentioned above.
- $b_0 \in \triangle(\mathcal{S})$, is the initial state distribution at time $t = 0$.

In this paper we restrict ourselves to collaborative models: a *collaborative MME* is an MME where all the agents get the same reward:

$$\forall_{i,j} \qquad R_i(s,\boldsymbol{a}) = R_j(s,\boldsymbol{a})$$

which will be simply written as $R(s,\boldsymbol{a})$.

An MME is underspecified in that it does not specify the information on which the agents can base their actions, or how they update their information. We make this explicit by defining an agent model.

Definition 2 (agent model)**.** A *model* m_i for agent i is a tuple $m_i = \langle \mathcal{I}_i,I_i,\mathcal{A}_i,\mathcal{O}_i,\mathcal{Z}_i,\pi_i,\iota_i \rangle$, where

- $\mathcal{I}_i$ is the set of *information states (ISs)* (also *internal states,* or *beliefs*),
- I_i is the *current* internal state of the agent,
- $\mathcal{A}_i$,$\mathcal{O}_i$ are as before: the actions taken by / observations that the environment provides to agent i,

[1] The origins of this approach can be traced back to the seventies [Witsenhausen, 1973].

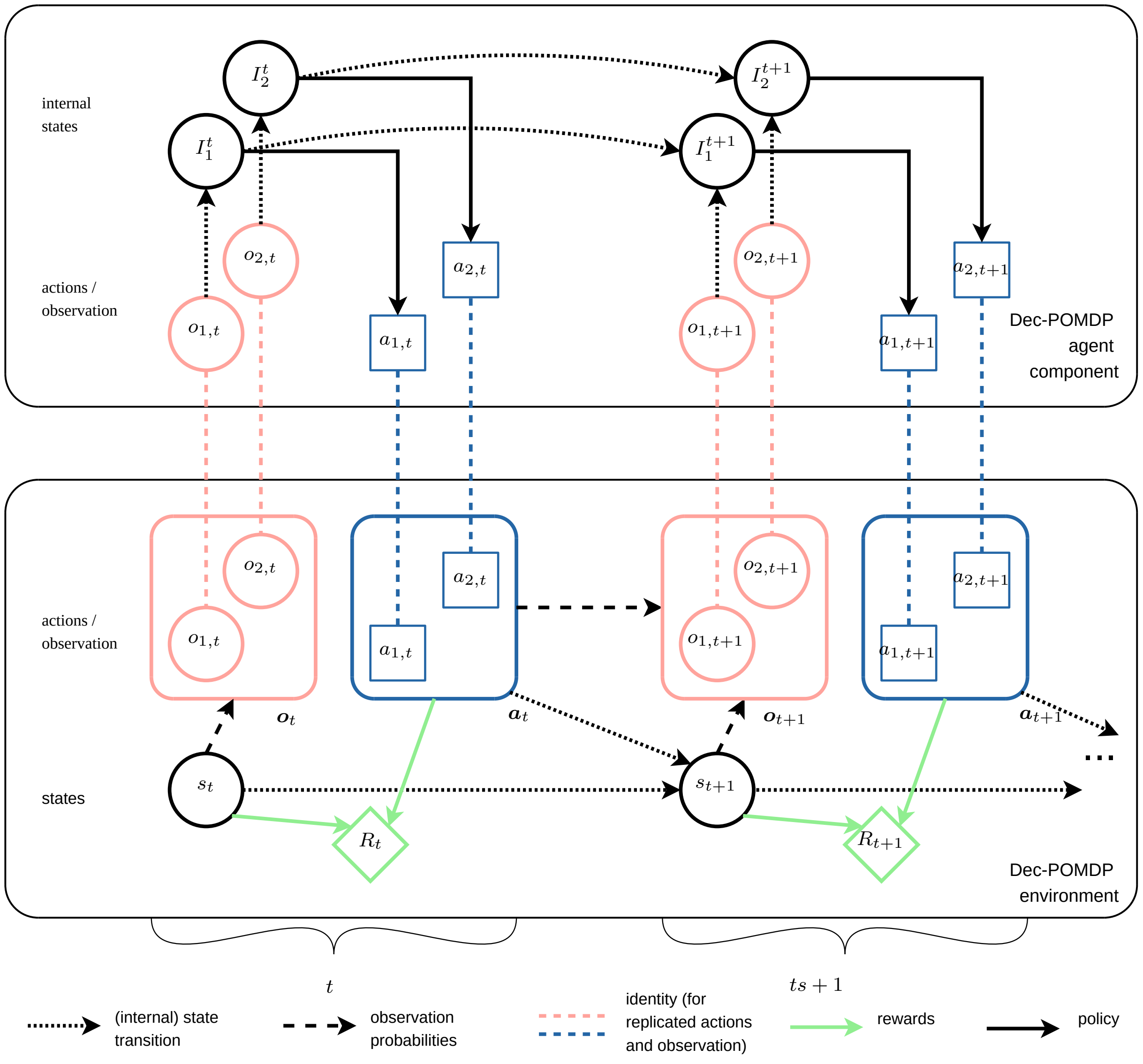

Figure 1: Illustration of the new perspective on the Dec-POMDP for the two-agent case. The process is formed by an environment and an agent component that together generate the interaction over time.

- $\mathcal{Z}_i$ is the set of *auxiliary observations* z_i (e.g., from communication) available to agent i,
- π_i is a (stochastic) action selection policy $\pi_i : \mathcal{I}_i \to \triangle(\mathcal{A}_i)$,
- ι_i the (stochastic) *information state function* (or *belief update function*) $\iota_i : \mathcal{I}_i \times \mathcal{A}_i \times \mathcal{O}_i \times \mathcal{Z}_i \to \triangle(\mathcal{I}_i)$.

This definition makes clear that the MME framework leaves the specification of the auxiliary observations, information states, the information state function, as well as the action selection policy unspecified. As such, the MME by itself is not enough to specify a dynamical process. Instead, it is necessary to specify those missing components for all agents. This is illustrated in Figure 1, which shows how a dynamic multiagent system (in this case, a Dec-POMDP, which we define explicitly below) evolves over time. It makes clear that there is a environment component, the MME, as well as an *agent component* that specifies how the agents update their internal state, which in turn dictates their actions.[2] It is only these two components together that lead to a dynamical process.

Definition 3 (agent component)**.** A *fully-specified agent component* can be formalized as a tuple $\mathcal{AC} = \left\langle \mathcal{D}, \{\mathcal{I}_i\}, \left\{I_i^0\right\}, \{\mathcal{A}_i\}, \{\mathcal{O}_i\}, \{\mathcal{Z}_i\}, \{\iota_i\}, \{\pi_i\} \right\rangle$, where[3]

- $\mathcal{D} = \{1,\ldots,n\}$ is the set of n agents.
- $\{\mathcal{I}_i\}$ are the sets of internal states for each agent.
- $\left\{I_i^0\right\}$ are the initial internal states of each agent.
- $\{\mathcal{A}_i\}$ are the sets of actions.
- $\{\mathcal{O}_i\}$ are the sets of observations.
- $\{\mathcal{Z}_i\}$ are the sets of auxiliary observations, plus a *mechanism for generating them.*
- $\{\iota_i\}$ are the internal state update functions for each agent.
- $\{\pi_i\}$ are the policies, that map from internal states to actions.

Clearly, once the MME and a fully-specified agent component are brought together, we have a dynamical system: a (somewhat more complicated) Markov reward process. The goal in formalizing these components, however, is that we want to *optimize* the behavior of the overall system. That is, we want to optimize the agent component in such a way that the reward is maximized.

As such, we provide a perspective of a whole range of *multiagent decision problems* that can be formalized in this fashion. On the one hand, the *problem designer* 1) selects an optimality criterion, 2) specifies the MME, and 3) may specify a subset of the elements of the agent-component (which determines the 'type' of problem that we are dealing with). On the other hand, the *problem optimizer* (e.g., a planning method we develop) has as its goal to optimize the non-specified elements of the agent component, in order to maximize the value as given by the optimality criterion.

In other words, we can think of a multiagent decision problem as the specification of an MME together with a non-fully specified agent component. A prominent example is the Dec-POMDP:

Definition 4 (Dec-POMDP)**.** A *decentralized POMDP (Dec-POMDP)* is an tuple $\langle OC,\mathcal{M},\mathcal{AC}\rangle$, where

- OC is the optimality criterion,
- $\mathcal{M}$ is an MME, and
- $\mathcal{AC} = \left\langle \mathcal{D}, \cdot, \cdot, \{\mathcal{A}_i\}, \{\mathcal{O}_i\}, \{\mathcal{Z}_i = \emptyset\}_{i\in\mathcal{D}}, \cdot, \cdot \right\rangle$ is a partially specified agent component:

  $\mathcal{AC}$ can be seen to partially specify the model for each agent: for each model m_i contained in the agent component, it specifies that $\mathcal{Z}_i = \emptyset$. That is, there are no auxiliary observations, such that each agent can form its internal state, and thus act, based only on its local actions and observations.

The goal for the problem optimizer for a Dec-POMDP is to specify the elements of $\mathcal{AC}$ that are not specified: $\{\mathcal{I}_i\}, \left\{I_i^0\right\}, \{\iota_i\}, \{\pi_i\}$. That is, the action selection policies need to be optimized and choices need to be made with respect to the representation and updating of information states.

[2] In the most general form, the next internal states would explicitly depend on the taken action too. (Not shown to avoid clutter).

[3] Alternatively, one can think of the agent component as a set of agent models for stage $t = 0$ together with a mechanism to generate the auxiliary observations.

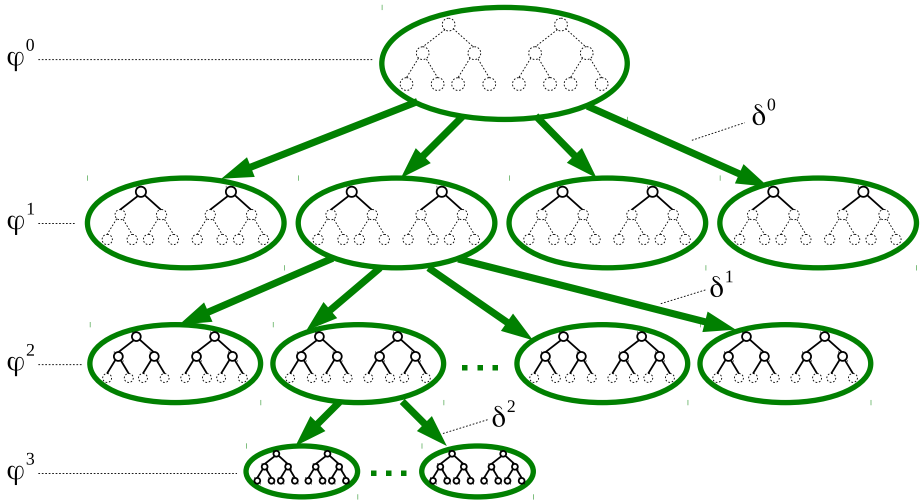

Figure 2: A hypothetical MAA* search tree.

These choices are typically done differently in the case of a finite and infinite horizon: maximizing the undiscounted and discounted sum of cumulative rewards are typically considered as the optimality criteria for the finite and infinite-horizon cases, respectively.

Also, internal states in Dec-POMDPs are often represented as nodes in a tree or finite-state controller for each agent which are updated by local observations. In contrast, the agent component for an MPOMDP includes auxiliary observations for all other agents and the internal states are the set of joint beliefs.

3 The Finite-Horizon Case

Here we give an overview of the reduction to an NOMDP in the finite-horizon case. In this case, the typical choice for the information states is to simply use the observation histories, and this is also what we will consider in this section.

3.1 The Plan-Time MDP and Optimal Value Function

Multiagent A* (MAA*) and its extensions [Szer et al., 2005, Oliehoek et al., 2013] provide a heuristic search framework for Dec-POMDPs. The core idea is that it is possible to search the space of partially specified joint policies, by ordering them in a search tree as illustrated in Figure 2. In particular, MAA* searches over *past joint policies* $\boldsymbol{\varphi}_t = \langle \varphi_{1,t}, \ldots, \varphi_{n,t} \rangle$, where each *individual* past joint policy is a sequence of decision rules: $\varphi_{i,t} = (\delta_{i,0}, \ldots, \delta_{i,t-1})$ that in turn map length-k observation histories $\vec{o}_{i,k} = (o_{i,1}, \ldots, o_{i,k})$ to actions: $\delta_{i,k}(\vec{o}_{i,k}) = a_{i,k}$.

This perspective gives rise to a reinterpretation of the planning process as a decision-making process in itself. In particular, it is possible to interpret the search-tree of MAA* as special type of MDP, called *plan-time MDP* [Oliehoek, 2010, 2013]. That is, each node in Figure 2 (i.e., each past joint policy $\boldsymbol{\varphi}_t$) can be interpreted as a state and each edge (i.e., each joint decision rule $\boldsymbol{\delta}_t$) then corresponds to an action. In this plan-time MDP, the transitions are deterministic, and the rewards $\check{R}(\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t)$ are the expected reward for stage t :

$$\begin{aligned} \check{R}(\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t) &= \mathbf{E}\left[R(s_t,\boldsymbol{a}_t) \mid b_0,\boldsymbol{\varphi}_t\right] \\ &= \sum_{s_t} \sum_{\vec{\boldsymbol{o}}_t} \Pr(s_t,\vec{\boldsymbol{o}}_t | b_0,\boldsymbol{\varphi}_t) R(s_t,\boldsymbol{\delta}_t(\vec{\boldsymbol{o}}_t)). \end{aligned}$$

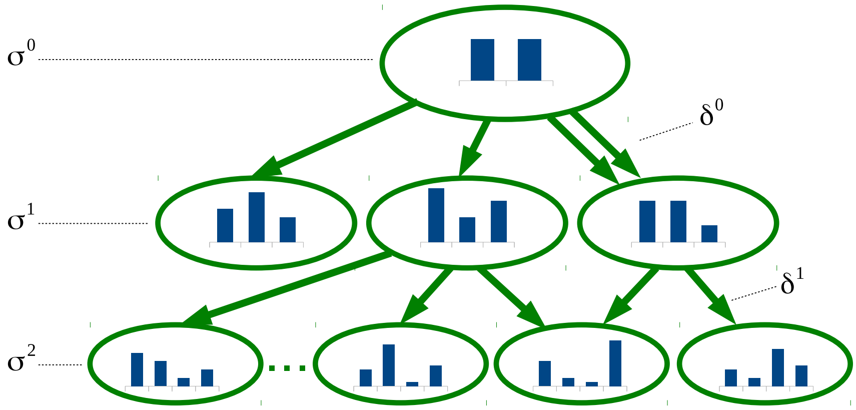

Figure 3: A hypothetical MAA* search tree based on plan-time sufficient statistics. Illustrated is that two joint decision rules from the same node (here: the root node) can map to the same next-stage statistic σ_1, and that two $\boldsymbol{\delta}_1$ from different nodes (the two rightmost σ_1) can lead to the same next-stage statistic σ_2.

The transitions are deterministic: the next past joint policy is determined completely by $\boldsymbol{\varphi}_t$ and $\boldsymbol{\delta}_t$, such that we can define the transitions as:

$$\check{T}(\boldsymbol{\varphi}_{t+1}|\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t) = \begin{cases} 1 & \text{if } \boldsymbol{\varphi}_{t+1} = \langle \boldsymbol{\varphi}_t \circ \boldsymbol{\delta}_t \rangle, \\ 0 & \text{otherwise.} \end{cases}$$

Here $\langle \boldsymbol{\varphi}_t \circ \boldsymbol{\delta}_t \rangle$ defines the concatenation of $\boldsymbol{\varphi}_t$ and $\boldsymbol{\delta}_t$ (see Oliehoek 2012 for details). Given that we have just defined an MDP, we can write down its optimal value function:

$$V_t^*(\boldsymbol{\varphi}_t) = \max_{\boldsymbol{\delta}_t} Q_t^*(\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t) \tag{3.1}$$

where Q^* is defined as

$$Q_t^*(\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t) = \begin{cases} \check{R}(\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t) & \text{for the last stage } t = h-1, \\ \check{R}(\boldsymbol{\varphi}_t,\boldsymbol{\delta}_t) + V_{t+1}^*(\langle \boldsymbol{\varphi}_t \circ \boldsymbol{\delta}_t \rangle) & \text{otherwise.} \end{cases} \tag{3.2}$$

This means that, via the notion of the plan-time MDP, we can define an optimal value function for the Dec-POMDP. It is informative to contrast the formulation of an *optimal* value function here to that of the value function *of a particular policy* (as given by, e.g., Oliehoek [2012]). Where the latter only depends on the history of observations, the optimal value function depends on the entire past joint policy. As a consequence, even though this optimal formulation admits a dynamic programming algorithm, it is not helpful, as this (roughly speaking) would result in brute force search through all joint policies [Oliehoek, 2010].

3.2 Plan-Time Sufficient Statistics

The problem in using the optimal value function defined by (3.1) is that it is too difficult to compute: the number of past joint policies is too large to be able to compute it for most problems. However, it turns out that it is possible to replace the dependence on the past joint policy by a so-called *plan-time sufficient statistic:* a distribution over histories and states [Dibangoye et al., 2013, Nayyar et al., 2013, Oliehoek, 2013]. This is useful, since many past joint policies can potentially map to the same statistic, as indicated in Figure 3.

Definition 5 (Sufficient statistic for deterministic past joint policies)**.** A sufficient statistic for a tuple $(b_0,\boldsymbol{\varphi}_t)$—with $\boldsymbol{\varphi}_t$ deterministic—is a distribution over joint observation histories and states: $\sigma_t(s_t,\vec{\boldsymbol{o}}_t) \triangleq \Pr(s_t,\vec{\boldsymbol{o}}_t|b_0,\boldsymbol{\varphi}_t)$ [Oliehoek, 2013].

Such a statistic is sufficient to predict the immediate reward:

$$\check{R}(\sigma_t,\boldsymbol{\delta}_t) = \sum_{s_t}\sum_{\vec{\boldsymbol{o}}_t} \sigma_t(s_t,\vec{\boldsymbol{o}}_t)R(s_t,\boldsymbol{\delta}_t(\vec{\boldsymbol{o}}_t)),$$

as well as the next statistic (a function of σ_t and $\boldsymbol{\delta}_t$). Let $\vec{\boldsymbol{o}}_{t+1} = (\vec{\boldsymbol{o}}_t,\boldsymbol{o}_{t+1})$ be a joint observation history that extends $\vec{\boldsymbol{o}}_t$ with $\boldsymbol{o}_{t+1}$, then the updated statistic is given by

$$\sigma_{t+1}(s_{t+1},\vec{\boldsymbol{o}}_{t+1}) = [U_{ss}(\sigma_t,\boldsymbol{\delta}_t)](s_{t+1},\vec{\boldsymbol{o}}_{t+1}) = \sum_{s_t} \Pr(s_{t+1},\boldsymbol{o}_{t+1}|s_t,\boldsymbol{\delta}_t(\vec{\boldsymbol{o}}_t))\sigma_t(s_t,\vec{\boldsymbol{o}}_t). \tag{3.3}$$

This means that we can define the optimal value function for a Dec-POMDP as

$$V_t^*(\sigma_t) = \max_{\boldsymbol{\delta}_t} Q_t^*(\sigma_t,\boldsymbol{\delta}_t), \tag{3.4}$$

where

$$Q_t^*(\sigma_t,\boldsymbol{\delta}_t) = \begin{cases} \check{R}(\sigma_t,\boldsymbol{\delta}_t) & \text{for the last stage } t = h-1, \\ \check{R}(\sigma_t,\boldsymbol{\delta}_t) + V_{t+1}^*(U_{ss}(\sigma_t,\boldsymbol{\delta}_t)) & \text{otherwise.} \end{cases} \tag{3.5}$$

Since many $\boldsymbol{\varphi}_t$ may potentially map to the same statistic σ_t, the above formulation can enable a more compact representation of the optimal value function. Moreover, it turns out that this value function satisfies the same property as POMDP value functions:

Theorem 1 (PWLC of Optimal Value Function)**.** *The optimal value function given by (3.4) is piecewise linear and convex (PWLC).*

Proof. For the last stage, $V_{h-1}{}^*(\sigma)$ is clearly PWLC. The value is given by

$$V_{h-1}^*(\sigma) = \max_{\boldsymbol{\delta}} \sum_{(s,\vec{\boldsymbol{o}})} \sigma_{h-1}(s,\vec{\boldsymbol{o}})R(s,\boldsymbol{\delta}(\vec{\boldsymbol{o}}))$$

By defining $R_{\boldsymbol{\delta}}$ as the vector with entries $R(s,\boldsymbol{\delta}(\vec{\boldsymbol{o}}))$ we can rewrite this as a maximization over inner products: $\max_{\boldsymbol{\delta}} \sum_{(s,\vec{\boldsymbol{o}})} \sigma(s,\vec{\boldsymbol{o}})R_{\boldsymbol{\delta}}(s,\vec{\boldsymbol{o}}) = \max_{\boldsymbol{\delta}} \sigma \cdot R_{\boldsymbol{\delta}}$ which shows that $V_{h-1}^*(\sigma)$ is PWLC.

For other stages, we expand

$$Q_t^*(\sigma,\boldsymbol{\delta}) = \sigma \cdot R_{\boldsymbol{\delta}} + V_{t+1}^*(U_{ss}(\sigma,\boldsymbol{\delta}))$$

if we assume, per induction hypothesis, that the next-stage value function is PWLC, then it is representable as the maximum over a set of vectors $\mathcal{V}_{t+1}^*$, and we can write:

$$\begin{aligned}
Q_t^*(\sigma,\boldsymbol{\delta}) &= \sigma \cdot R_{\boldsymbol{\delta}} + \max_{v\in\mathcal{V}_{t+1}^*} U_{ss}(\sigma,\boldsymbol{\delta}) \cdot v \\
&= \sigma \cdot R_{\boldsymbol{\delta}} + \max_{v\in\mathcal{V}_{t+1}^*} \sum_{(s',\vec{\boldsymbol{o}}')} \left[U_{ss}(\sigma,\boldsymbol{\delta})(s',\vec{\boldsymbol{o}}')\right] v(s',\vec{\boldsymbol{o}}') \\
&= \sigma \cdot R_{\boldsymbol{\delta}} + \max_{v\in\mathcal{V}_{t+1}^*} \sum_{(s',\vec{\boldsymbol{o}}')} \left[\sum_{(s,\vec{\boldsymbol{o}})} \Pr(s',\vec{\boldsymbol{o}}'|s,\vec{\boldsymbol{o}},\boldsymbol{\delta}(\vec{\boldsymbol{o}}))\sigma(s,\vec{\boldsymbol{o}})\right] v(s',\vec{\boldsymbol{o}}') \\
&= \sigma \cdot R_{\boldsymbol{\delta}} + \max_{v\in\mathcal{V}_{t+1}^*} \sum_{(s,\vec{\boldsymbol{o}})} \sigma(s,\vec{\boldsymbol{o}}) \sum_{(s',\vec{\boldsymbol{o}}')} \Pr(s',\vec{\boldsymbol{o}}'|s,\vec{\boldsymbol{o}},\boldsymbol{\delta}(\vec{\boldsymbol{o}}))v(s',\vec{\boldsymbol{o}}') \\
&= \sigma \cdot R_{\boldsymbol{\delta}} + \max_{v\in\mathcal{V}_{t+1}^*} \sum_{(s,\vec{\boldsymbol{o}})} \sigma(s,\vec{\boldsymbol{o}})g_{\boldsymbol{\delta}}^v(s,\vec{\boldsymbol{o}})
\end{aligned}$$

where we defined the back-projection of $v \in \mathcal{V}^*_{t+1}$, given $\boldsymbol{\delta}$, as:

$$g_{\boldsymbol{\delta},v}(s,\vec{\boldsymbol{o}}) \triangleq \sum_{(s',\vec{\boldsymbol{o}}')} \Pr(s',\vec{\boldsymbol{o}}'|s,\vec{\boldsymbol{o}},\boldsymbol{\delta}(\vec{\boldsymbol{o}}))v(s',\vec{\boldsymbol{o}}'). \tag{3.6}$$

As a result, we see that $Q^*_t(\sigma,\boldsymbol{\delta}) = \sigma \cdot R_{\boldsymbol{\delta}} + \max_{v\in\mathcal{V}^*_{t+1}} \sigma \cdot g_{\boldsymbol{\delta},v} = \max_{v\in\mathcal{V}^*_{t+1}} \sigma \cdot [R_{\boldsymbol{\delta}} + g_{\boldsymbol{\delta},v}]$,which means that $Q^*_t(\sigma,\boldsymbol{\delta})$ is PWLC. Therefore, V^*_t via (3.4) is a maximum over PWLC functions and hence is itself PWLC. □

3.3 A NOMDP Formulation

The PWLC property of the optimal value function seems to imply that we are actually dealing with a kind of POMDP. This intuition is correct [Nayyar et al., 2011, Dibangoye et al., 2013, MacDermed and Isbell, 2013]. In particular, it is possible to make a reduction to special type of POMDP: a non-observable MDP *(NOMDP)* [Boutilier et al., 1999], which is a POMDP with just one 'NULL' observation.

Definition 6 (Plan-time NOMDP)**.** The *plan-time NOMDP* $\mathcal{M}_{PT}$ for a Dec-POMDP $\mathcal{M}$ is a tuple $\mathcal{M}_{PT}(\mathcal{M}) = \left\langle \check{\mathcal{S}},\check{\mathcal{A}},\check{T},\check{R},\check{\mathcal{O}},\check{O},\check{h},\check{b}_0 \right\rangle$, where:

- $\check{\mathcal{S}}$ is the set of augmented states, each $\check{s}_t = \langle s_t,\vec{\boldsymbol{o}}_t \rangle$.
- $\check{\mathcal{A}}$ is the set of actions, each $\check{a}_t$ corresponds to a joint decision rule $\boldsymbol{\delta}_t$ in the Dec-POMDP.
- $\check{T}$ is the transition function:

$$\check{T}(\langle s_{t+1},\vec{\boldsymbol{o}}_{t+1}\rangle \mid \langle s_t,\vec{\boldsymbol{o}}_t\rangle, \boldsymbol{\delta}_t) = \begin{cases} \Pr(s_{t+1},\boldsymbol{o}_{t+1}|s_t,\boldsymbol{\delta}_t(\vec{\boldsymbol{o}}_t)) & \text{if } \vec{\boldsymbol{o}}_{t+1} = (\vec{\boldsymbol{o}}_t,\boldsymbol{o}_{t+1}), \\ 0 & \text{otherwise.} \end{cases}$$

- $\check{R}$ is the reward function: $\check{R}(\langle s_t,\vec{\boldsymbol{o}}_t\rangle, \boldsymbol{\delta}_t) = R(s_t,\boldsymbol{\delta}_t(\vec{\boldsymbol{o}}_t))$.
- $\check{\mathcal{O}} = \{NULL\}$ is the observation set which only contains the $NULL$ observation.
- $\check{O}$ is the observation function that specifies that $NULL$ is received with probability 1 (irrespective of the state and action).
- The horizon is just the horizon of $\mathcal{M}$: $\check{h} = h$.
- $\check{b}_0$ is the initial state distribution. Since there is only one $\vec{\boldsymbol{o}}_0$ (i.e., the empty joint observation history), it can directly be specified as

$$\forall s_0 \qquad \check{b}_0(\langle s_0,\vec{\boldsymbol{o}}_0\rangle) = b_0(s_0).$$

Since a NOMDP is a special case of POMDP, all POMDP theory and solution methods apply. In particular, it should be clear that the belief in this plan-time NOMDP corresponds exactly to the plan-time sufficient statistic from Definition 5. Moreover, it can be easily shown that the optimal value function for this plan-time NOMDP is identical to the formulation equations (3.4) and (3.5).

4 The Infinite-Horizon Case

In this section we treat the infinite-horizon case. In this case, it is no longer possible to use observation histories as information states, since there are an infinite number of them. As such, we discuss a formulation with more abstract (but finitely many) information states. For instance, one could use the typical choice of a finite state controller where the information states are represented as nodes in the controller. We essentially follow the approach by MacDermed and Isbell [2013], but we make a distinction between the general concept of a what we will refer to as a *Dec-POMDP with information state abstraction (ISA-Dec-POMDP)* and the *bounded belief Dec-POMDP (BB-Dec-POMDP)* introduced by MacDermed and Isbell [2013] as a specific instantiation of that framework.

4.1 ISA-Dec-POMDP

Similar to the transformation of finite-horizon Dec-POMDPs into NOMDPs described above, infinite-horizon Dec-POMDPs can also be transformed into NOMDPs. The basic idea is to replace the observation histories by (a finite number of) abstract information states:

Definition 7 (ISA-Dec-POMDP)**.** A *Dec-POMDP with information state abstraction (ISA-Dec-POMDP)* is a Dec-POMDP framework together with the specification of the sets $\{\mathcal{I}_i\}$ of information states (ISs).

For an ISA-Dec-POMDP, using the notation of the agent components defined above, there are two optimizations that need to be performed jointly:

1. the optimization of the joint action selection policy $\boldsymbol{\pi} = \langle \pi_1, \dots, \pi_n \rangle$, and
2. the optimization of the joint information state function $\iota = \langle \iota_1, \dots, \iota_n \rangle$.

Essentially, an ISA-Dec-POMDP can be thought of as having chosen FSCs to represent the policies (although different choices may be made to actually represent the ISs, and IS functions) where the nodes are represented by internal states, controller transitions are defined by the internal state update functions and the action selection probabilities are defined by the policies.

In this section, we only consider the setting in which we search a stationary policy such that $\boldsymbol{\pi} = \boldsymbol{\delta}$ is a (induced, via the individual decision rules) mapping from (joint) information states to (joint) actions. In this paper, we will only consider deterministic $\boldsymbol{\pi}$ and ι, but extensions to stochastic rules are straightforward.

4.2 Plan-Time Sufficient Statistics

Here, we show that *given the information state update functions*, we can replicate the approach taken for the finite-horizon case. First, let $I = \langle I_1, \dots, I_n \rangle$ denote a joint information state. This allows us to redefine the plan-time sufficient statistic as follows:

Definition 8 (Plan-time ISA sufficient statistic)**.** The plan-time sufficient statistic for an ISA-Dec-POMDP is $\sigma_t(s,I) \triangleq \Pr(s,I|\boldsymbol{\delta}_0,\dots,\boldsymbol{\delta}_{t-1})$.

Again, this statistic can be updated using Bayes' rule. In particular $\sigma'(s',I')$ is given by

$$\forall_{(s',I')} \quad [U_{ss}(\sigma,\boldsymbol{\delta})]\,(s',I') = \sum_{(s,I)} \Pr(s',I'|s,I,\boldsymbol{\delta}(I))\sigma(s,I) \tag{4.1}$$

where—using $\iota(I'|I,\boldsymbol{a},\boldsymbol{o}) = \prod_{i\in\mathcal{D}} \iota_i(I'_i|I_i,a_i,o_i)$ for the joint internal state update probability—the probability of transitioning to (s',I') is given by:

$$\Pr(s',I'|s,I,\boldsymbol{a}) = \Pr(s'|s,\boldsymbol{a}) \sum_{\boldsymbol{o}} \iota(I'|I,\boldsymbol{a},\boldsymbol{o}) \Pr(\boldsymbol{o}|\boldsymbol{a},s'), \tag{4.2}$$

It is easy to show that, *for a given set $\{\iota_i\}$ of internal state update functions*, one can construct a plan-time NOMDP analogous to before:

Definition 9 (Plan-time ISA-NOMDP)**.** Given the internal state update functions, an ISA-Dec-POMDP can be converted to a *plan-time ISA-NOMDP* $\mathcal{M}_{PT-ISA-NOMDP}$ for a Dec-POMDP $\mathcal{M}$ is a tuple $\mathcal{M}_{PT-ISA-NOMDP}(\mathcal{M}) = \left\langle \check{\mathcal{S}},\check{\mathcal{A}},\check{T},\check{R},\check{\mathcal{O}},\check{O},\check{h},\check{b_0} \right\rangle$, where:

- $\check{\mathcal{S}}$ is the set of augmented states, each $\check{s} = \langle s,I \rangle$.
- $\check{\mathcal{A}}$ is the set of actions, each $\check{a}$ corresponds to a joint decision rule $\boldsymbol{\delta}$ (which is a joint (stationary) action selection policy) in the ISA-Dec-POMDP.
- $\check{T}$ is the transition function:

$$\begin{aligned}\check{T}(\left\langle s,I' \right\rangle \mid \langle s,I \rangle, \boldsymbol{\delta}) &= \Pr(s',I'|s,\boldsymbol{a} = \boldsymbol{\delta}(I)) \\ &= \Pr(s'|s,\boldsymbol{a}) \sum_{\boldsymbol{o}} \iota(I'|I,\boldsymbol{a},\boldsymbol{o}) \Pr(\boldsymbol{o}|\boldsymbol{a},s')\end{aligned}$$

- $\check{R}$ is the reward function: $\check{R}(\langle s,I \rangle, \boldsymbol{\delta}) = R(s,\boldsymbol{\delta}(I))$.
- $\check{\mathcal{O}} = \{NULL\}$ is the observation set which only contains the $NULL$ observation.
- $\check{O}$ is the observation function that specifies that $NULL$ is received with probability 1 (irrespective of the state and action).
- The horizon is just the (infinite) horizon of $\mathcal{M}$: $\check{h} = h$.
- $\check{b_0}$ is the initial state distribution. Since the initial joint information state I_0 can be selected, it can directly be specified as

$$\forall s_0 \qquad \check{b_0}(\langle s_0,I_0 \rangle) = b_0(s_0).$$

4.3 Optimizing the Information State Update Functions

In the previous setting, we showed that the NOMDP formulation can still be applied in the infinite-horizon case, given that the information state update function is specified. However, in the infinite-horizon setting, the selection of those internal state update functions becomes part of the optimization task.

One idea to address this, dating back to Meuleau et al. [1999] and extended to Dec-POMDPs by MacDermed and Isbell [2013] (discussed in the next section), is to make searching the space of deterministic internal state update functions part of the problem. This can be done by defining a cross-product MDP in which “a decision is the choice of an action and of a next node” — Meuleau et al. [1999]. That is, selection of the ι_i function can be done by introducing $|\mathcal{O}_i|$ new ‘information-state action’ variables (say, $a_i^\iota = \{a_i^{\iota,1}, \dots, a_i^{\iota,|\mathcal{O}_i|}\}$) that specify, for each observation $o_i \in \mathcal{O}_i$, the next internal state. In other words, we can define augmented actions $\bar{a}_i = \langle a_i,a_i^\iota \rangle$ that define the information state function via

$$\iota_i(I_i'|I_i,\bar{a}_i,o_i = k) = \iota_i(I_i'|I_i,a_i,o_i = k) = \begin{cases} 1 & I_i' = a_i^{\iota,k} \\ 0 & \text{otherwise.} \end{cases}$$

Letting $\bar{\boldsymbol{\delta}}$ denote the (joint) decision rules that map information states to such augmented (joint) actions, we can define a plan-time model where the transition function no longer depends on a pre-specified information state update function

$$\check{T}(\left\langle s,I' \right\rangle \mid \langle s,I \rangle, \bar{\boldsymbol{\delta}}) = \Pr(s'|s,\boldsymbol{a}) \sum_{\boldsymbol{o}} \iota(I'|I,\bar{\boldsymbol{a}},\boldsymbol{o}) \Pr(\boldsymbol{o}|\boldsymbol{a},s')$$

As such, it is possible to jointly optimize over action selection policies and information state update functions in a single augmented model at the cost of an increased action space.

4.4 BB-Dec-POMDPs: Alternating Phase 'Optimal Belief Compression'

The simple method described above leads to a much larger action set (of size $|\mathcal{A}_i| \cdot |\mathcal{I}_i|^{|\mathcal{O}_i|}$) for each agent. In order to avoid this increase, MacDermed and Isbell [2013] introduce the *bounded belief Dec-POMDP*[4] *(BB-Dec-POMDP)*, which is a ISA-Dec-POMDP that encodes the selection of optimal $\{\iota_i\}$ by splitting each problem stage into two stages: one for selection of the domain-level actions ('belief expansion phase') and one for selection of the information-state update actions a_i^ι ('belief compression phase').

In particular, the belief compression phase stores the observations the agent receives as part of the state. As a result, the agents do not need to select a next information state for every possible observation, but only for the observation actually encoded by this belief compression state. This limits the growth of the number of actions needed per agents. For further details of this formulation we refer to the original paper by MacDermed and Isbell [2013].

5 Efficient Point-Based Backups for NOMDPs

A bottleneck in the solution of all of the resulting NOMDP formulations (in both the finite and infinite-horizon case) is that the actions correspond to decision rules, and the set of decision rules is large (exponential in the number of information states). To address this problem, MacDermed and Isbell [2013] propose a modification of the point-based POMDP backup operator [Shani et al., 2013] (which they apply in the point-based method Perseus [Spaan and Vlassis, 2005] to solve the NOMDP).

In more detail, the modification aims at mitigating the bottleneck of maximizing over (exponentially many) decision rules in

$$V^*(\sigma) = \max_{\boldsymbol{\delta}} Q^*(\sigma,\boldsymbol{\delta})$$

Since the value function of a NOMDP is PWLC, the next-stage value function can be represented using a set of vectors $v \in \mathcal{V}$, and we can write

$$\begin{aligned} V^*(\sigma) &= \max_{\boldsymbol{\delta}} \sum_{(s,I)} \sigma(s,I) \left(R(s,\boldsymbol{\delta}(I)) + \max_{v\in\mathcal{V}} \sum_{(s',I')} \Pr(s',I'|s,I,\boldsymbol{\delta}) v(s',I') \right) \\ &= \max_{v\in\mathcal{V}} \max_{\boldsymbol{\delta}} \sum_{(s,I)} \sigma(s,I) \underbrace{\left(R(s,\boldsymbol{\delta}(I)) + \sum_{(s',I')} \Pr(s',I'|s,I,\boldsymbol{\delta}(I)) v(s',I') \right)}_{v_{\boldsymbol{\delta},v}(s,I)}. \end{aligned}$$

The key insight is that, in the last expression, the vector $v_{\boldsymbol{\delta},v}$ constructed from v for $\boldsymbol{\delta}$ (i.e, the bracketed part) only depends on $\boldsymbol{\delta}(I)$ (i.e., on that part of $\boldsymbol{\delta}$ that specifies the actions for I only). That is, we can simplify $v_{\boldsymbol{\delta},v}$ to $v_{\boldsymbol{a},v}$. As such it is possible to rewrite this value as a maximization of solutions to collaborative Bayesian games (CBG) [e.g. Oliehoek, 2010, p.19], one for each $v \in \mathcal{V}$:

$$V^*(\sigma) = \max_{v\in\mathcal{V}} \text{CBG-value}(\sigma,v).$$

The value of each CBG is given by

$$\text{CBG-value}(\sigma,v) = \max_{\boldsymbol{\delta}} \sum_{I} \sigma(I) Q^v(I,\boldsymbol{\delta}(I)),$$

[4] The term 'bounded belief' refers to the finite number of internal states (or 'beliefs') considered.

where $Q^v(I,\boldsymbol{a}) = \sum_s v_{\boldsymbol{a},v}(s,I)$ is the payoff function of the CBG.

For each $v \in \mathcal{V}$, the maximization over $\boldsymbol{\delta}$ can now be performed more effectively using a variety of methods [Oliehoek et al., 2010, Kumar and Zilberstein, 2010, Oliehoek et al., 2012]. MacDermed and Isbell [2013] propose a method based on integer programming, and empirically found that the integer program can be optimally solved via a linear program relaxation in many cases.[5] We note that the maximizing $\boldsymbol{\delta}$ directly induces a vector $v_{\boldsymbol{\delta}}$, which is the result of the point-based backup. As such, this modification can also be used by other point-based POMDP methods.

6 Conclusions

This paper gives an overview of the reduction from a decentralized problem to a centralized one. In particular, we can transform a Dec-POMDP to a NOMDP. This mapping is exact, but does not immediately make the problem easier to solve. However, it does allow the application of sophisticated POMDP solution methods, which can lead to improvements in performance [Dibangoye et al., 2013, MacDermed and Isbell, 2013].

While this document has restricted itself to the Dec-POMDP model without communication, a similar reduction (but now to an actual POMDP) is possible in the context of k-steps delayed communication [Oliehoek, 2010, Nayyar et al., 2011, Oliehoek, 2013], and can be further generalized to exploit all common information [Nayyar et al., 2014]. Moreover, we conjecture that generalization to many different communication and knowledge updating protocols may be possible, by suitably adjusting the set of auxiliary observations and the mechanism that generates them.

References

C. Amato, G. Chowdhary, A. Geramifard, N. K. Ure, and M. J. Kochenderfer. Decentralized control of partially observable Markov decision processes. In *Proceedings of the Fifty-Second IEEE Conference on Decision and Control*, pages 2398–2405, 2013.

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

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

J. S. Dibangoye, C. Amato, O. Buffet, and F. Charpillet. Optimally solving Dec-POMDPs as continuous-state MDPs. In *Proc. of the International Joint Conference on Artificial Intelligence*, 2013.

A. Kumar and S. Zilberstein. Point-based backup for decentralized POMDPs: Complexity and new algorithms. In *Proc. of the International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 1315–1322, 2010.

L. C. MacDermed and C. Isbell. Point based value iteration with optimal belief compression for Dec-POMDPs. In *Advances in Neural Information Processing Systems 26*, pages 100–108. 2013.

[5] That is, they found that in a great percentage of cases, the solution of the LP relaxation was integral, which establishes its optimality.

A. Mahajan and M. Mannan. Decentralized stochastic control. *Annals of Operations Research*, pages 1–18, 2014.

N. Meuleau, K. Kim, L. P. Kaelbling, and A. R. Cassandra. Solving POMDPs by searching the space of finite policies. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 417–426, 1999.

A. Nayyar, A. Mahajan, and D. Teneketzis. Optimal control strategies in delayed sharing information structures. *IEEE Trans. Automat. Contr.*, 56(7):1606–1620, 2011.

A. Nayyar, A. Mahajan, and D. Teneketzis. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Transactions on Automatic Control*, 58: 1644–1658, July 2013.

A. Nayyar, A. Mahajan, and D. Teneketzis. The common-information approach to decentralized stochastic control. In G. Como, B. Bernhardsson, and A. Rantzer, editors, *Information and Control in Networks*, volume 450 of *Lecture Notes in Control and Information Sciences*, pages 123–156. Springer International Publishing, 2014.

F. A. Oliehoek. *Value-Based Planning for Teams of Agents in Stochastic Partially Observable Environments.* PhD thesis, Informatics Institute, University of Amsterdam, Feb. 2010.

F. A. Oliehoek. Decentralized POMDPs. In M. Wiering and M. van Otterlo, editors, *Reinforcement Learning: State of the Art*, volume 12 of *Adaptation, Learning, and Optimization*, pages 471–503. Springer Berlin Heidelberg, Berlin, Germany, 2012.

F. A. Oliehoek. Sufficient plan-time statistics for decentralized POMDPs. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 302–308, 2013.

F. A. Oliehoek, M. T. J. Spaan, J. Dibangoye, and C. Amato. Heuristic search for identical payoff Bayesian games. In *Proceedings of the Ninth International Conference on Autonomous Agents and Multiagent Systems*, pages 1115–1122, May 2010.

F. A. Oliehoek, S. Whiteson, and M. T. J. Spaan. Exploiting structure in cooperative Bayesian games. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, pages 654–664, Aug. 2012.

F. A. Oliehoek, M. T. J. Spaan, C. Amato, and S. Whiteson. Incremental clustering and expansion for faster optimal planning in decentralized POMDPs. *Journal of Artificial Intelligence Research*, 46:449–509, 2013.

G. Shani, J. Pineau, and R. Kaplow. A survey of point-based POMDP solvers. *Autonomous Agents and Multi-Agent Systems*, 27(1):1–51, 2013. doi: 10.1007/s10458-012-9200-2.

M. T. J. Spaan and N. Vlassis. Perseus: Randomized point-based value iteration for POMDPs. *Journal of AI Research*, 24:195–220, 2005.

D. Szer, F. Charpillet, and S. Zilberstein. MAA*: A heuristic search algorithm for solving decentralized POMDPs. In *Proc. of Uncertainty in Artificial Intelligence*, pages 576–583, 2005.

H. Witsenhausen. A standard form for sequential stochastic control. *Mathematical Systems Theory*, 7(1):5–11, 1973.

Acknowledgements

We thank Shimon Whiteson for his useful comments. F.O. is funded by NWO Innovational Research Incentives Scheme Veni #639.021.336.

IAS reports

This report is in the series of IAS technical reports. The series editor is Bas Terwijn (`B.Terwijn@uva.nl`). Within this series the following titles appeared:

A. Visser *UvA Rescue Technical Report: a description of the methods and algorithms implemented in the UvA Rescue code release* Technical Report IAS-UVA-12-02, Informatics Institute, University of Amsterdam, The Netherlands, September 2012.

A. Visser *A survey of the architecture of the communication library LCM for the monitoring and control of autonomous mobile robots* Technical Report IAS-UVA-12-01, Informatics Institute, University of Amsterdam, The Netherlands, September 2012.

Olaf Booij and Zoran Zivkovic *The Planar two point algorithm* Technical Report IAS-UVA-09-05, Informatics Institute, University of Amsterdam, The Netherlands, September 2009.

All IAS technical reports are available for download at the ISLA website,
`http://www.science.uva.nl/research/isla/MetisReports.php`.