QueryPOMDP: POMDP-Based Communication in Multiagent Systems*

Francisco S. Melo[1], Matthijs T.J. Spaan[2], and Stefan J. Witwicki[1]

[1] INESC-ID/Instituto Superior Técnico
2780-990 Porto Salvo, Portugal
{fmelo,witwicki}@inesc-id.pt
[2] Delft University of Technology
2628 CD, Delft, The Netherlands
m.t.j.spaan@tudelft.nl

Abstract. Decentralized Partially Observable Markov Decision Processes (Dec-POMDPs) provide powerful modeling tools for multiagent decision-making in the face of uncertainty, but solving these models comes at a very high computational cost. Two avenues for side-stepping the computational burden can be identified: structured interactions between agents and intra-agent communication. In this paper, we focus on the interplay between these concepts, namely how sparse interactions impact the communication needs. A key insight is that in domains with local interactions the amount of communication necessary for successful joint behavior can be heavily reduced, due to the limited influence between agents. We exploit this insight by deriving local POMDP models that optimize each agent's communication behavior. Our experimental results show that our approach successfully exploits sparse interactions: we can effectively identify the situations in which it is beneficial to communicate, as well as trade off the cost of communication with overall task performance.

1 Introduction

Decentralized Partially Observable Markov Decision Processes (Dec-POMDPs) provide powerful modeling tools for multiagent decision-making with limited sensing capabilities in stochastic environments. However, the prohibitive computational cost required to compute an optimal decision rule renders them intractable except for the smallest of problems.[1] In the literature, two avenues for side-stepping the computational burden can be identified: *localized interactions between agents*—where the actions of each agent depend on the other agents only in specific, localized situations [1–7]—and *intra-agent communication*—where agents

* This work was funded in part by Fundação para a Ciência e a Tecnologia (INESC-ID multiannual funding) through the PIDDAC Program funds and the project CMU-PT/SIA/0023/2009 under the Carnegie Mellon-Portugal Program. M.S. is funded by the FP7 Marie Curie Actions Individual Fellowship #275217 (FP7-PEOPLE-2010-IEF).

[1] Dec-MDPs are known to be NEXP-complete even in 2-agent scenarios.

M. Cossentino et al. (Eds.): EUMAS 2011, LNAI 7541, pp. 189–204, 2012.

are able to communicate with one another so as to partly mitigate the impact of partial observability [8–15]. In this paper, we focus on the *interplay* between these concepts, namely how sparse interactions impact the communication needs.

A key insight is that in domains with local interactions the amount of communication necessary for successful joint behavior can be heavily reduced, due to the limited influence between agents. Several previous works have implicitly relied on this observation, exploring sparse interactions by having agents share information locally [5, 7, 11, 16, 17]. In this work, we explicitly reason about the benefits of communication/information sharing in scenarios with sparse interactions. Sparse interactions enable, to some extent, decoupling the decision-process of the different agents. We leverage such decoupling to derive local models that optimize each agent's communication behavior, allowing it to overcome partial observability in those situations where decoupled decisions are not possible.

We provide a new way of optimizing communication by proposing a model in which agents need to plan about when to query other agents' local states, which we call QueryPOMDP. We observe that to execute optimal joint policies in fully observable scenarios—policies which can be computed efficiently—agents will generally need to reason about the state of other agents. However, in scenarios where interactions are sparse, this need will be greatly reduced. Our approach thus relies on the interplay between sparse interactions and their impact on the communication needs for executing fully observable policies. Our agents construct a local POMDP model of the environment from the fully observable joint policy of all other agents. Solving this POMDP model allows the agent not only to determine how to solve the task at hand but also to determine when to query the local state of the environment. Our approach thus allows the agents to explicitly reason about communication, without incurring in the prohibitive computational cost of Dec-POMDP models that include communication [18]. Furthermore, in contrast to many methods in the literature [11, 14], QueryPOMDP can properly handle noisy communication channels, and does not require strong independence assumptions [19]. Our empirical analysis on benchmark problems demonstrates the efficacy of QueryPOMDP in balancing communication costs with coordination benefits.

The remainder of this work is organized as follows. First, Section 2 briefly introduces the relevant background regarding Dec-POMDP models, followed by a motivating example which is presented in Section 3. Section 4 describes our proposed model for state querying, and how it can be solved for multiple agents. Experiments are presented in Section 5, followed by a discussion of related research in Section 6. Finally, Section 7 concludes and describes future work.

2 Background

We start by reviewing *Decentralized Partially Observable Markov Decision Processes* (Dec-POMDPs) and related decision theoretic models. An N-agent Dec-POMDP $\mathcal{M}$ can be specified as a tuple $\mathcal{M} = (N, \mathcal{X}, (\mathcal{A}_k), (\mathcal{Z}_k), \mathsf{P}, (\mathsf{O}_k), r, \gamma)$, where:

- $\mathcal{X}$ is the joint state-space;
- $\mathcal{A} = \times_{i=1}^{N} \mathcal{A}_i$ is the set of joint actions, with each $\mathcal{A}_i$ the individual action set for agent $i, i = 1, \ldots, N$;
- Each $\mathcal{Z}_i, i = 1, \ldots, N$, represents the set of possible local observations for agent i;
- $\mathsf{P}(y \mid x, a)$ represents the transition probabilities from joint state x to joint state y when the joint action a is taken;
- Each $\mathsf{O}_i(z_i \mid x, a), i = 1, \ldots, N$, represents the probability of agent i making the local observation z_i when the joint state is x and the last joint action taken was a;
- $r(x, a)$ represents the expected reward received by all agents for taking the joint action a in joint state x;
- The scalar γ is a discount factor.

An *N-agent Decentralized Markov decision process* (Dec-MDP) is a particular class of Dec-POMDP in which the state is *jointly fully observable.* Formally this can be translated into the following condition: for every joint observation $z \in \mathcal{Z}$, with $\mathcal{Z} = \times_{i=1}^{N} \mathcal{Z}_i$, there is a state $x \in \mathcal{X}$ such that $\mathbb{P}\left[X(t) = x \mid Z(t) = z\right] = 1$, where $X(t)$ is the joint state of the process at time t and $Z(t)$ the corresponding joint observation. Although apparently simpler, optimally solving of a Dec-MDP is in the same complexity class as optimally solving a Dec-POMDP. A *partially observable Markov decision process* (POMDP) is a 1-agent Dec-POMDP and a *Markov decision process* (MDP) is a 1-agent Dec-MDP. Finally, an *N-agent multiagent MDP* (MMDP) is an N-agent Dec-MDP that is *fully observable*, *i.e.*, for every individual observation $z_i \in \mathcal{Z}_i$ there is a state $x \in \mathcal{X}$ such that $\mathbb{P}\left[X(t) = x \mid Z_i(t) = z_i\right] = 1$.

In this partially observable multiagent setting, an individual (non-Markov) policy for agent i is a mapping $\pi_i : \mathcal{H}_i \longrightarrow \Delta(\mathcal{A}_i)$, where $\Delta(\mathcal{A}_i)$ is the space of probability distributions over $\mathcal{A}_i$, and $\mathcal{H}_i$ is the set of all possible finite histories for agent i. The purpose of all agents is to determine a joint policy π that maximizes the total sum of discounted rewards. In other words, considering a distinguished initial state $x^0 \in \mathcal{X}$ that is assumed common knowledge among all agents, the goal of the agents is to maximize

$$V^{\pi} = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r\big(X(t), A(t)\big) \mid X(0) = x^0\right]. \tag{1}$$

For a more detailed introduction to Dec-POMDPs and related models see, for example, [20].

3 A Motivating Example

Multi-robot systems constitute a primary motivation for our work and provide a natural example of the class of problems considered herein. In multi-robot systems, interaction among robots is naturally limited by the robot's physical

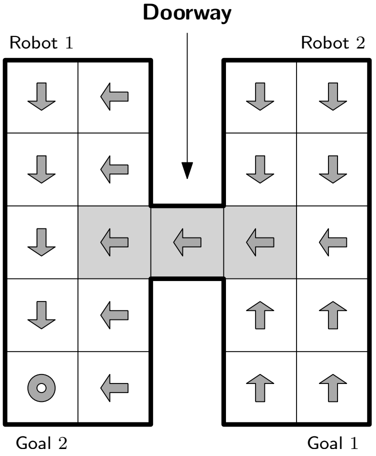


Fig. 1. H-Environment, where two robots need to interact only around the narrow doorway to reach their corresponding goals. The shaded arrows correspond to a possible policy for Robot 2 in the absence of Robot 1.

boundaries (workspace, communication range, etc.) and limited perception capabilities. It is therefore natural to subdivide the overall task into smaller tasks that each robot can execute either autonomously or as part of a small group. Moreover, besides being embedded in a physical environment, robots typically have a way of communicating among themselves.

We motivate our ideas in a simple navigation scenario, depicted in Fig. 1. In this scenario, two robots (Robot 1 and Robot 2) must navigate to their corresponding goal states (marked as Goals 1 and 2). At the same time, they must avoid colliding in the narrow doorway (the central state), since it leads to a large penalty. Each robot has 4 possible actions (namely "Move North", "Move South", "Move East" and "Move West") that move the robot in the corresponding direction. The motion of one robot does not depend on the position or action of the other robot except in the doorway: if the robots collide in the doorway, then their actions have an increasing failure probability. Complicating matters, initially each robot starts uniformly at random in one of the 10 locations on its side of the doorway.

In a fully observable situation, the agents will move toward their respective goals. When reaching the doorway, if the other robot is also close to the doorway one of the two will stop so that the other can safely traverse.[2] It will then resume its trajectory to its goal.

In order for the agents to actually execute the policy just described, they only need to reason about the state of the other agent when reaching the darker area in their starting side of the environment. And then, once one robot is in the doorway, it can just proceed toward its goal, independently of the state of the other robot. Moreover, even if the robots are generally unable to observe the position

[2] Which one stops is determined by the joint policy they adopt.

of the other robot, but they are able to *query it*, they can reasonably assume that the other robot will behave more or less as in the fully observable scenario. This observation is the departing point for the model and approach proposed in this paper and described in the continuation.

4 A Model for State Querying

We depart from an N-agent Dec-MDP model, and address the problem of when communication can be beneficial to improve the performance in such a model. For the purposes of our study, we momentarily focus on the decision processes of all except one agent, which we refer to as agent k. Unlike other communication-based approaches to Dec-MDPs (e.g., [11]), we adopt a relatively general communication model, in which the messages received by an agent are taken as part of its local (noisy) observation. Also, messages received by agent k depend on explicit information-querying actions executed by k.

Throughout this section, we represent the (finite) state-space of the Dec-MDP as a set $\mathcal{X}$ and assume that it can be factorized as $\mathcal{X} = \mathcal{X}_k \times \mathcal{X}_{-k}$, where the elements $x_k \in \mathcal{X}_k$ correspond to agent k's local state. The state at time t, $X(t)$, is thus a pair $\langle X_k(t), X_{-k}(t)\rangle$. We also assume that the observations of each agent do not depend on the actions of the remaining agents, *i.e.*,

$$\mathbb{P}\left[Z_i(t) = z_i \mid X(t), A(t)\right] = \mathbb{P}\left[Z_i(t) = z_i \mid X(t), A_i(t)\right],$$

for all $i = 1, \ldots, N$. Therefore, we can simply write the observation probabilities as $\mathsf{O}_i(z_i \mid x, a_i), i = 1, \ldots, N$.

4.1 Query Actions and Resulting Observations

For the purpose of allowing our agent to reason about communication, we assume that each agent has the ability to *query* the other agents for their local state information. In order to make this explicit, we differentiate between *communication actions* and the remaining actions—henceforth referred as *primitive actions*, and write the set of individual actions for agent k as the cartesian product of the set of communication actions, $\mathcal{A}_k^C$, and the set of primitive actions, $\mathcal{A}_k^P$, *i.e.*, $\mathcal{A}_k = \mathcal{A}_k^C \times \mathcal{A}_k^P$. We also assume that transition probabilities are independent of the communication actions,

$$\mathsf{P}(y \mid x, \langle a_{-k}, (a_k^C, a_k^P)\rangle) = \mathsf{P}(y \mid x, \langle a_{-k}, (b_k^C, a_k^P)\rangle)$$

for any $x, y \in \mathcal{X}$, $a_{-k} \in \mathcal{A}_{-k}$, $a_k^P \in \mathcal{A}_k^P$ and $a_k^C, b_k^C \in \mathcal{A}_k^C$.

We also differentiate between *communication observations*—*i.e.*, observations that result from communication actions—and *primitive observations*, that do not depend on the communication actions. Formally, we write the set of individual observations for agent k as the cartesian product of the set of communication observations, $\mathcal{Z}_k^C$, and primitive observations, $\mathcal{Z}_k^P$, *i.e.*, $\mathcal{Z}_k = \mathcal{Z}_k^C \times \mathcal{Z}_k^P$. Communication observations correspond to either the local state of other agents or

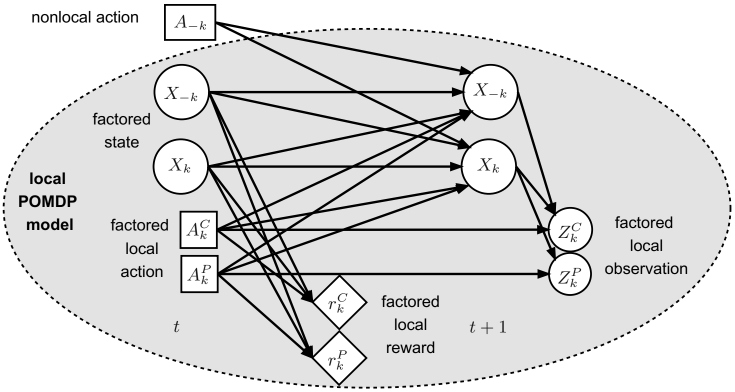


Fig. 2. The factored decision model, from agent k's perspective

the null observation, 0, *i.e.*, $\mathcal{Z}_k^C = \mathcal{X}_{-k} \cup \{0\}$. Moreover, we consider that communication observations do not depend on primitive actions, and that primitive observations do not depend on communication actions. This means that we can decouple the observation probabilities as

$$\mathsf{O}_k\big((z_k^C, z_k^P) \mid x, (a_k^C, a_k^P)\big) = \mathsf{O}_k^C(z_k^C \mid x, a_k^C)\mathsf{O}_k^P(z_k^P \mid x, a_k^P),$$

where

$$\begin{aligned}\mathsf{O}_k^C(z_k^C \mid x, a_k^C) &= \mathbb{P}\left[Z_k^C(t) = z_k^C \mid X(t) = x, A_k^C(t) = a_k^C\right]\\ \mathsf{O}_k^P(z_k^P \mid x, a_k^P) &= \mathbb{P}\left[Z_k^P(t) = z_k^P \mid X(t) = x, A_k^P(t) = a_k^P\right].\end{aligned}$$

Finally, we assume that the reward function can also be decomposed as the sum of two components. The first component, denoted r^C, concerns the *cost of communication* and is independent on the primitive actions of agent k and on the actions of the other agents. The second component, denoted as r^P corresponds to the “regular” (or domain-level) reward defining the overall goal of the agents. It is assumed independent of the communication actions of agent k. Formally, if $a = \langle a_{-k}, a_k\rangle$ and $a_k = (a_k^C, a_k^P)$, this means that the reward r can be written as

$$r(x, a) = r^P(x, \langle a_{-k}, a_k^P\rangle) + r^C(x, a_k^C). \tag{2}$$

Figure 2 depicts a dynamic Bayesian network that summarizes all above considerations.

Following the discussion in Section 3, and for the purpose of its planning process, agent k will treat all remaining agents as if they follow a Markov policy, π_{-k}, that corresponds to the optimal policy for the underlying MMDP. This policy, being Markovian, depends only on the state of the system at time t, $X(t)$, *i.e.*,

$$\mathbb{P}\left[A_{-k}(t) = a_{-k} \mid H(t)\right] = \mathbb{P}\left[A_{-k}(t) = a_{-k} \mid X(t) = x\right] = \pi_{-k}(x, a_{-k}), \tag{3}$$

where $A_{-k}(t)$ denotes the action taken by all agents other than k at time t, $H(t)$ denotes the whole history of the process up to time t and $a_{-k} \in \mathcal{A}_{-k}$. From this perspective, the decision process for agent k can be modeled as a (single-agent) POMDP that we describe in the next section.

4.2 POMDP Model for a Single Agent

Let $\mathcal{M} = (N, \mathcal{X}, (\mathcal{A}_k), (\mathcal{Z}_k), \mathsf{P}, (\mathsf{O}_k), r, \gamma)$ be a Dec-MDP as described above. Let π_{-k} denote the (state-dependent) joint MMDP policy for all agents other than k. We can now denote the single-agent POMDP model for agent k as a tuple $\mathcal{M}_k = (\mathcal{X}, \mathcal{A}_k, \mathcal{Z}_k, \mathsf{P}_k, \mathsf{O}_k, r_k, \gamma)$, where:

- $\mathcal{X}$ corresponds to the original Dec-MDP state-space.
- $\mathcal{A}_k$ is the individual action-space for agent k.
- $\mathcal{Z}_k$ is the individual observation-space for agent k.
- P_k are the transition probabilities obtained from the original transition probabilities. In particular, given an action $a_k = (a_k^C, a_k^P)$, we have

$$\mathsf{P}_k(y \mid x, a_k) = \sum_{a_{-k} \in \mathcal{A}_{-k}} \pi_{-k}(x, a_{-k}) \mathsf{P}(y \mid x, \langle a_{-k}, a_k^P \rangle).$$

- O_k are the observation probabilities for agent k, that match the original Dec-MDP observation probabilities. In particular, given an action $a_k = (a_k^C, a_k^P)$, we have

$$\mathsf{O}_k(z_k \mid x, a_k) = \mathsf{O}_k^C(z_k^C \mid x, a_k^C) \mathsf{O}_k^P(z_k^P \mid x, a_k^P), \tag{4}$$

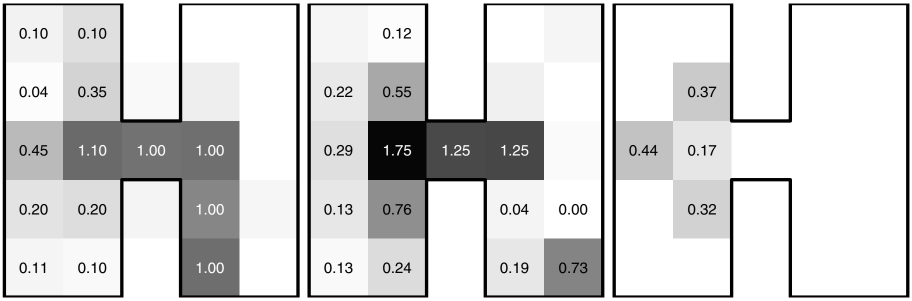
  where $z_k = (z_k^C, z_k^P)$.
- r_k is the reward function obtained from the original Dec-MDP reward function after averaging over the other agents' policy, π_{-k}, *i.e.*,

$$r_k(x, a_k) = \sum_{a_{-k} \in \mathcal{A}_{-k}} \pi_{-k}(x, a_{-k}) r(x, \langle a_{-k}, a_k \rangle).$$

Given this POMDP model, we can use standard POMDP solution techniques to explore the trade-off between the costs and benefits of communication for agent k.

4.3 Results for the H-Environment Example

Continuing the example of Section 3, the application of our model allows us to better understand under which circumstances the benefits of using communication compensate for its costs. For this purpose, we fix the policy of Agent 2 as shown in Fig. 1, which corresponds one possible joint MMDP policy for this environment. As explained above, given such a policy we can construct a POMDP from the point of view of Agent 1, in which it can query Agent 2's states at any time step, at a particular communication cost. For illustration purposes, the initial state of Agent 2 is selected randomly on the right half of the environment.



(a) Det. P, $r^C = 0.0$. (b) Noisy P, $r^C = 0.0$. (c) Noisy P, $r^C = -0.3$.

Fig. 3. Results for the H-environment. (a)-(c) Query frequency in each state for Agent 1, varying in deterministic (Det. P) or noisy (Noisy P) transitions and communication cost.

We test several experimental conditions that include the presence or absence of transition noise and different costs for the communication actions.

We examine in which states Agent 1 queries Agent 2's state. When communication is free (Figs. 3(a) and (b)), Agent 1 queries in all the states it passes through.[3] With a communication cost of 0.3 (Fig. 3(c)), however, it only queries when near to and left of the doorway. In these states it is crucial to know Agent 2's location to avoid potential collisions, an intuition that is exploited automatically by our model. The use of a POMDP model in this context ensures that the agent explicitly reasons about information gathering which, in our setting, translates in weighting the benefits of communication in terms of the overall task against the costs associated with it.

4.4 Computing Policies for Multiple Agents

In the previous section we proposed using a POMDP model to compute the policy for one agent k, treating all other agents as if they were following the optimal joint policy for the underlying MMDP. Given this POMDP model for agent k we can compute the corresponding optimal policy using any preferred POMDP solution technique. We use this approach to better understand the communication needs of one agent in a simple multiagent navigation scenario, and to determine in which situations the cost of communication outweighs its value.

We now want to extend these ideas and actually compute the policy for *all* agents in the Dec-MDP. The idea of using POMDP models to plan in multiagent scenarios has been previously explored in the Dec-POMDP literature [21, 22]. The general difficulty with these approaches arises from the fact that each agent has only a local observation of the joint state of the world. This implies that,

[3] We note that, due to the transition noise, an agent can remain in the same state more than one consecutive time-step, and hence the values > 1.

when planning for agent k, the POMDP model necessary to properly capture the behavior of all agents other than k can either be prohibitively large, require agent k to reason about how the other agents reason about agent k's state, leading to infinitely nested beliefs, or both [21, 22].

In our approach, we rely on the intuition discussed in Section 3, according to which the use of *active communication* allied with *sparse interactions* may actually alleviate the difficulties associated with planning in multiagent systems with partial observability. We plan for each agent k while treating all other agents as if following the *optimal joint MMDP policy.* In scenarios where interactions are sparse, the general behavior of the agents is expected to roughly follow the MMDP policy, as discussed in Section 3 and in those situations where coordination is necessary, agents can resort to communication, but weighting the benefits of such communication with the associated costs.

Several previous works have already studied the benefits of exploiting communication and structured interactions separately (see, for example, [5, 6, 8]).[4] The novelty in our approach lies precisely on the fact that we can explicitly exploit the interplay between these two aspects (communication and sparse interactions) to attain efficient planning in multiagent problems. Section 5 describes the application of our approach in several navigation scenarios of different dimensions. Our results empirically show that our approach is indeed able to make effective use of communication and attain a performance that indeed approaches that observed in fully observable settings.

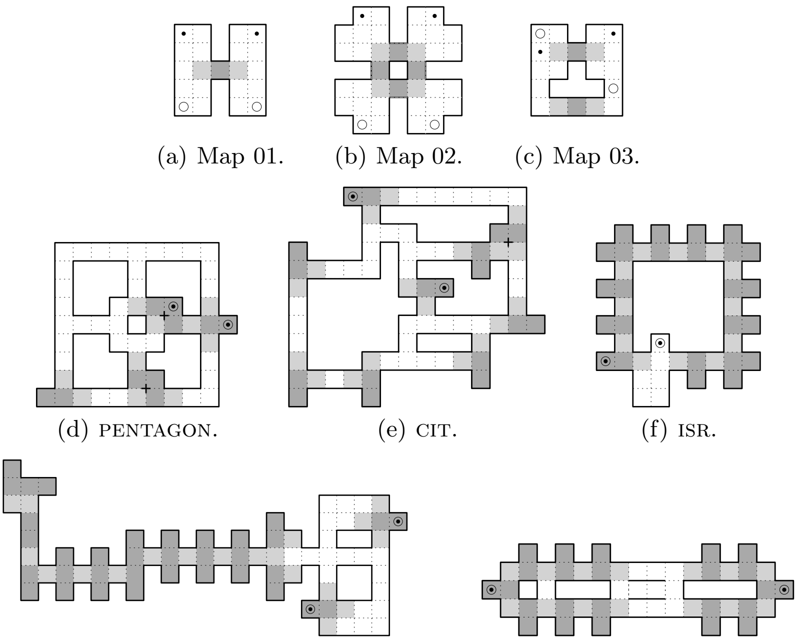
5 Experiments

In this section we illustrate the application of our method to several navigation scenarios from the POMDP and Dec-POMDP literature. We use robot navigation scenarios (Fig. 4), since our model is particularly suited for modeling multi-robot problems. Furthermore, results can be easily visualized and interpreted in this class of problems.

Experimental Setup. In each of the test scenarios, each of two (identical) robots departs from one of the locations marked with a dot, and must reach the state marked with a circle that is furthest from its initial state. Each robot has 4 actions that move the robot in one of the four possible direction with probability 0.8 and fail with probability 0.2, plus a fifth "NoOp" action.

All agents have full local state observability. The shaded regions correspond to areas inside of which the agents are able to successfully communicate, *i.e.*, when an agent queries another agent, it incurs a cost of -0.1 and successfully observes the local state of the queried agent with a probability of 0.8. With a probability of 0.2 it receives no observation about the state of the other. In the white cells, an agent is never able to perceive the state of the other, but still incurs a penalty of -0.1 if it attempts to communicate. In other words, the agents can always

[4] We refer to Section 6 for a detailed discussion of related approaches.



(a) Map 01. (b) Map 02. (c) Map 03.

(d) PENTAGON. (e) CIT. (f) ISR.

(g) SUNY. (h) MIT.

Fig. 4. Environments used in the experiments

attempt to communicate, incurring in a penalty of -0.1 ($r^C(x) = -0.1$ for all $x \in \mathcal{X}$), but only in the shaded areas does communication succeed (with high probability). The darker cells correspond to states where the agents receive a penalty of -20 when standing there simultaneously, in which case the rate of action failure is also increased to 0.4 for both agents. When an agent reaches its goal position, it receives a reward of 10 and moves to a rewardless absorbing state. Throughout the experiments, we used $\gamma = 0.95$.

For each of the test scenarios, following the approach in Section 4, we compute the optimal MMDP joint policy that we use to determine a POMDP model describing the decision process for each individual agent. This POMDP is then solved using the Perseus approximate solver [23]. We test our QueryPOMDP policy for 100 independent trials of 250 steps each and measure the obtained performance in terms of total discounted reward. We also test the performance of other sets of agents that communicate at different (but fixed) frequencies (see Table 1(a)):

- "Never Comm" agents *never communicate*. These agents observe only their local state, and each follows the optimal policy for the underlying single-agent MDP obtained by disregarding the other agent in the environment;
- "Always Comm" agents *communicate at every time-step*, incurring the corresponding penalty. As QueryPOMDP agents, they are subject to

Table 1. (a) Main differences between the groups of agents used. (b) Total discounted reward for each set of agents in each of the test-scenarios. Entries in *italic* in the same column are not statistically different.

(a) Different methods used.

Agents	**Comm.**	**Succ.**	**Failed**
	Freq.	**Comm.**	**Comm.**
QueryPOMDP	Variable	POMDP	POMDP
Never Comm	Never	–	Indiv. MDP
Always Comm	1 step	MMDP	Indiv. MDP
Comm $k = 2$	2 steps	MMDP	Indiv. MDP
Comm $k = 3$	3 steps	MMDP	Indiv. MDP
Comm $k = 4$	4 steps	MMDP	Indiv. MDP

(b) Experimental results.

Environment	Map 1	Map 2	Map 3	CIT	ISR	MIT	Pent.	SUNY
# States	441	1,296	400	4,900	1,849	2,401	2,704	5,476
QueryPOMDP	5.132	3.598	6.156	*5.260*	*6.755*	*2.964*	6.444	*5.328*
Never Comm	−1.834	0.900	1.917	*5.306*	6.663	*2.959*	5.641	*5.283*
Always Comm	1.961	2.248	3.276	3.286	4.779	1.116	5.038	3.297
Comm $k = 2$	−0.069	1.097	3.001	4.306	5.839	2.141	5.578	4.294
Comm $k = 3$	−0.127	1.707	1.564	4.666	6.114	2.426	5.246	4.646
Comm $k = 4$	−0.785	1.289	3.295	4.324	5.760	2.106	5.448	4.317
MMDP	5.787	5.253	6.608	*5.305*	*6.817*	3.182	7.606	*5.297*

communication errors/limitations and, as such, are not always able to perceive the state of the other agent. When communication fails, the agent observes only its local state and adopts the individual MDP policy. When communication succeeds, it adopts the underlying MMDP policy.

- "Comm $k = 2, 3, 4$" agents query the state of the other agent every k steps. Except for the different communication frequency, they are otherwise similar to "Always Comm" agents.

Comparisons between these different agents will allow us to analyze (i) the impact that communication costs can have on performance, if communication is not optimized; and (ii) the impact that communication can have in mitigating partial observability. As discussed ahead, direct comparison against other methods such as the one in [11] is not very informative, as these do not trade-off communication costs with task performance.

Results and Discussion. The performance of the 6 agent groups in terms of total discounted reward is summarized in Table 1(b). As a reference against which to assess the quality of our computed policy we also provide the results for the MMDP optimal policy in the different environments, providing a performance upper bound. The QueryPOMDP approach performs very favorably, outperforming all other policies and coming close to the MMDP upper bound in several of the tested scenarios.

The results in Table 1(b) prompt several interesting observations. First, comparing the performance of the MMDP policy against that of the group that never communicates provides an important indication of how critical coordination is in a given scenario. NeverComm agents act individually, disregarding the existence of other agents in the environment. In environments where coordination is critical, NeverComm agents will perform poorly. MMDP agents, on the other hand, always act in a perfectly coordinated manner, in which coordination does not come at a cost. In an environment where little coordination is needed, the difference between these two groups is going to be small. In contrast, scenarios that require significant coordination will cause the performance of the two groups to significantly differ.

From Table 1(b), we can see that coordination is critical in the smaller environments (Maps 1-3). In the larger environments, such as cit, mit and suny, coordination is less critical. The results in the smaller environments illustrate the impact of effective communication in mitigating the effects of partial observability. Our method is actually able to attain a performance very close to that of the MMDP agents, even paying for communication. Additionally, our approach uses communication efficiently, since the performance of all other communicating agents is significantly inferior. In contrast, in cit, mit and suny, non-communicating agents actually attain optimal performance. The difference in performance to the communicating groups can be explained by the communication penalty. Again, in these scenarios, our approach is able to manage communication needs, as it performs similarly to non-communicating agents.

A second observation is that the MMDP performance is an upper bound on the optimal Dec-MDP performance. This means that in those scenarios where our approach performs close to or as well as the MMDP group, we can immediately conclude that it is also performing close to or as well as the optimal Dec-MDP policy. A general comparison of the performance of our method against that of the MMDP group indicates that our method, if not optimal, must be very close to optimal in most scenarios tested. This, in turn, indicates that approximating the behavior of our agents with that of MMDP agents does provide a solid basis for planning.

We also applied our approach to a benchmark problem from the Dec-POMDP literature, namely the firefighting problem with 3 houses and 3 fire levels [24]. In this scenario, the QueryPOMDP agents are allowed to communicate at a cost, but no observation results out of it, since there is no communication in the original problem. Allowing for no shared information among agents renders our version of the firefighting problem effectively equivalent to the original problem and thus enables a meaningful comparison between the two methods.

Applying our method in the firefighting problem provides useful insights into two important aspects of our method. First, on the trade-off between communication costs and benefits, our method should figure out that communication is

Table 2. Results of QueryPOMDP in the Firefighting problem [24]

Problem	Dimension	QueryPOMDP	Optimal
Firefighters [24]	432	-7.679	-7.176

useless in this setting and effectively not use it. Second, concerning the general applicability of our method, the results should shed some light on whether the proposed approximation provides meaningful information in scenarios with local interactions sense, *i.e.*, if each agent, by assuming the other agents to behave according to the MMDP policy, are still able to make good decisions.

The performance of our approach is summarized in Table 2, corresponding to the total reward obtained over a 6-step run, averaged over $1,000$ independent Monte Carlo trials. For comparison, we also provide the optimal value for the 6-step horizon, reported in [25]. As expected, the QueryPOMDP agents learn not to use communication. Moreover, although the firefighting problem does not strictly adhere to the setting considered in this paper, it still exhibits some level of independence that our approach is successfully able to leverage—the difference in obtained performance is statistically not significant.

Summarizing, our results show that, in scenarios with sparse interactions like the ones analyzed, our agents behave approximately as MMDP agents, effectively using communication to mitigate the effects of partial observability.

6 Related Work

In the Dec-POMDP literature, early approaches introduced the idea of transition and reward independence as forms of simplified interactions [26]. Further examples of models with sparse interactions include *interaction-driven Markov games* (IDMGs) [5,17], *distributed POMDPs with coordination locales* [7], *transition-decoupled POMDPs* [6], *factored Dec-POMDPs* [4], and models relying on event-driven interactions [3,27].

Our representation is closest to IDMGs [5], which leverage independence between different agents in a Dec-POMDP to decouple the decision process in significant portions of the joint state-space. In those situations in which agents interact, IDMGs rely on communication to bring down the the computational complexity of the joint decision process. The use of communication to overcome partial observability sets this approach apart from other approaches that also exploit local interactions. However, communication is assumed to *always* take place and to be error-free [5]. In our case, we add explicit query actions to the agents action repertoires, enabling them to ask another agent's state, under environment-specific constraints. For instance, two robots may only be able to share information when they are physically close. We further assume that communication is not error-free and comes at a cost that must be considered.

Explicit communication in multiagent planning was already addressed in [18], where the proposed Com-MTDP model allows agents to explicitly reason about communication in Dec-POMDP scenarios. However, being a generalization of Dec-POMDPs, it shares the discouraging computational complexity of the latter model. The actual process of communication has been investigated in [28]. Roth et al. [11] propose to exploit a factored Dec-MDP model and policy representation, in which agents query other agents' local states when this knowledge is required for choosing their local actions. Although this work already seeks to optimize communication, this optimization is conducted parallel with the underlying decision process. Therefore, the cost of communication does not directly translate in the agent's task performance, as in our proposed approach, rendering the tradeoff between communication costs and benefits unclear. Another closely related work is that of Wu et al. [14] where communication is used as a means to decrease the planning complexity in Dec-POMDP models. Like in our proposed approach, this work considers that communication may not always be available. However, unlike our approach, this work does not consider explicitly optimization of communication. Finally, Mostafa and Lesser [16] do optimize communication, while considering the presence of communication limitations. However, this optimization is also conducted parallel with the underlying decision process, without directly impacting in the agent's task performance. Also, none of the aforementioned methods considers noisy communication channels.

A key point in our approach is that, although we use the MMDP policy in our planning, its computation is significantly more efficient than computing a centralized policy for the actual partially observable decision problem. The fact that we plan individually for each agent is somewhat related to several works that use round-robin policy optimization to individually optimize the policy of different agents in Dec-POMDP settings. One of the early examples is the JESP algorithm [21], which also models agents individually as POMDPs, but does not use communication. Round-robin policy optimization has been used to learn communication primitives in Dec-POMDPs whose base models are transition and observation independent [12], but which are coupled through the communication actions agents can choose to execute. In that case, however, agents have to learn when sending a particular message will be beneficial for team performance, which is far from trivial given that the policy of the receiving agent does not exploit the information provided by incoming messages. In our case, however, agents can opt to query other agents' states, and it is much easier to determine when doing so improves performance. Secondly, we consider a much richer model where agents also "physically" influence each other, instead of only through communication.

7 Conclusions

In this paper, we analyzed the interplay between sparse interactions and communication in multiagent planning. We observed that, in scenarios where interactions among agents are sparse (i.e., intra-agent action coordination is only infrequently necessary), the distributed execution of an MMDP policy seldom requires full-state information. As such, if each agent is (individually) allowed to query other agents for their local state information when necessary, it may be possible to partly mitigate partial state observability and leverage more efficient planning approaches.

Relying on this insight, we proposed the use of a POMDP model to analyze the communication needs of an agent in a Dec-MDP scenario where the interaction between the agents is sparse. Our model accommodates communication costs and failures—the agent must explicitly reason about these factors in its decision process. QueryPOMDP allows agents to optimize communication, explicitly trading-off its costs with its benefits in terms of the underlying task.

We used our approach to optimize communication in the simple scenario of Fig. 1, where our approach was successfully able to capture the intuition that the fundamental states for coordination are those around the doorway. We further explored the usefulness of this approach in computing policies for larger and more general Dec-MDPs. We built POMDP models for each agent by considering the other agents to behave as if in an MMDP, and use the obtained POMDP optimal policies. Our results show that our agents are able to effectively using communication to mitigate the effects of partial observability, behaving approximately as MMDP agents. One important avenue of future work is to generalize these techniques beyond Dec-MDPs, to scenarios in which agents can query other agents' observations instead of states.

References

1. Allen, M., Zilberstein, S.: Agent influence as a predictor of difficulty for decentralized problem-solving. In: Proc. 22nd AAAI Conf. Artificial Intelligence, pp. 688–693 (2007)
2. Becker, R., Zilberstein, S., Lesser, V., Goldman, C.: Transition-independent decentralized Markov decision processes. In: Proc. Int. Conf. Auton. Agents and Multiagent Systems, pp. 41–48 (2003)
3. Becker, R., Lesser, V., Zilberstein, S.: Decentralized Markov decision processes with event-driven interactions. In: Proc. Int. Conf. Auton. Agents and Multiagent Systems, pp. 302–309 (2004)
4. Oliehoek, F., Spaan, M., Whiteson, S., Vlassis, N.: Exploiting locality of interaction in factored Dec-POMDPs. In: Proc. Int. Conf. Auton. Agents and Multiagent Systems (2008)
5. Spaan, M., Melo, F.: Interaction-driven Markov games for decentralized multiagent planning under uncertainty. In: Proc. Int. Conf. Auton. Agents and Multiagent Systems, pp. 525–532 (2008)
6. Witwicki, S., Durfee, E.: Influence-based policy abstraction for weakly-coupled Dec-POMDPs. In: Int. Conf. Automated Planning and Scheduling (2010)
7. Varakantham, P., Kwak, J., Taylor, M., Marecki, J., Scerri, P., Tambe, M.: Exploiting coordination locales in distributed POMDPs via social model shaping. In: Proc. 19th Int. Conf. Automated Planning and Scheduling, pp. 313–320 (2009)
8. Goldman, C., Zilberstein, S.: Optimizing information exchange in cooperative multiagent systems. In: Proc. 2nd Int. Conf. Autonomous Agents and Multiagent Systems, pp. 137–144 (2003)
9. Goldman, C., Zilberstein, S.: Communication-based decomposition mechanisms for decentralized MDPs. J. Artificial Intelligence Res. 32, 169–202 (2008)

10. Roth, M., Simmons, R., Veloso, M.: Decentralized communication strategies for coordinated multiagent policies. In: Multi-Robot Systems: From Swarms to Intelligent Automata, pp. 93–106 (2005)
11. Roth, M., Simmons, R., Veloso, M.: Exploiting factored representations for decentralized execution in multiagent teams. In: Proc. Int. Conf. Auton. Agents and Multiagent Systems, pp. 469–475 (2007)
12. Spaan, M., Gordon, G., Vlassis, N.: Decentralized planning under uncertainty for teams of communicating agents. In: Proc. Int. Conf. Auton. Agents and Multiagent Systems (2006)
13. Tasaki, M., Yabu, Y., Iwanari, Y., Yokoo, M., Tambe, M., Marecki, J., Varakantham, P.: Introducing communication in Dis-POMDPs with locality of interaction. In: IEEE/WIC/ACM Int. Conf. Web Intelligence and Intelligent Agent Technology, vol. 2, pp. 169–175 (2008)
14. Wu, F., Zilberstein, S., Chen, X.: Multi-agent online planning with communication. In: Proc. Int. Conf. Automated Planning and Scheduling, pp. 321–329 (2009)
15. Xuan, P., Lesser, V., Zilberstein, S.: Communication decisions in multiagent cooperation: Model and experiments. In: Proc. 5th Int. Conf. Autonomous Agents, pp. 616–623 (2001)
16. Mostafa, H., Lesser, V.: Offline planning for communication by exploiting structured interactions in decentralized MDPs. In: IEEE/WIC/ACM Int. Conf. Web Intelligence and Intelligent Agent Technology, pp. 193–200 (2009)
17. Melo, F., Veloso, M.: Decentralized MDPs with sparse interactions. Artificial Intelligence 175(11), 1757–1789 (2011)
18. Pynadath, D., Tambe, M.: The communicative multiagent team decision problem: Analyzing teamwork theories and models. J. Artificial Intelligence Res. 16, 389–423 (2002)
19. Becker, R., Carlin, A., Lesser, V., Zilberstein, S.: Analyzing myopic approaches for multi-agent communications. Computational Intelligence 25(1), 31–50 (2009)
20. Seuken, S., Zilberstein, S.: Formal models and algorithms for decentralized decision making under uncertainty. Auton. Agents and Multi-Agent Systems (2008)
21. Nair, R., Tambe, M., Yokoo, M., Pynadath, D., Marsella, S.: Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In: Proc. 18th Int. Joint Conf. Artificial Intelligence, pp. 705–711 (2003)
22. Doshi, P., Gmytrasiewicz, P.: On the difficulty of achieving equilibrium in interactive POMDPs. In: Proc. 21st AAAI Conf. Artificial Intelligence, pp. 1131–1136 (2006)
23. Spaan, M.T.J., Vlassis, N.: Perseus: Randomized point-based value iteration for POMDPs. J. Artificial Intelligence Res. 24, 195–220 (2005)
24. Oliehoek, F., Spaan, M., Vlassis, N.: Optimal and approximate Q-value functions for decentralized POMDPs. J. Artificial Intelligence Res. 32, 289–353 (2008)
25. Spaan, M., Oliehoek, F., Amato, C.: Scaling up optimal heuristic search in Dec-POMDPs via incremental expansion. In: Proc. Int. Joint Conf. Artificial Intelligence, pp. 2027–2032 (2011)
26. Becker, R., Zilberstein, S., Lesser, V., Goldman, C.: Solving transition independent decentralized Markov decision processes. J. Artificial Intelligence Res. 22, 423–455 (2004)
27. Mostafa, H., Lesser, V.: A compact mathematical formulation for problems with structured agent interactions. In: Proc. AAMAS MSDM Workshop (2011)
28. Goldmann, C., Allen, M., Zilberstein, S.: Learning to communicate in a decentralized environment. J. Auton. Agents and Multiagent Systems 15(1), 47–90 (2007)