How to Exhibit More Predictable Behaviors

Salomé Lepers, Sophie Lemonnier, Vincent Thomas, Olivier Buffet

How to Exhibit More Predictable Behaviors

A Preprint

Salomé Lepers[1] Sophie Lemonnier[1,2]
Vincent Thomas[1] Olivier Buffet[1]
(1)Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
(2)Université de Lorraine, PErSEUs, F-57045 Metz, France

October 8, 2024

Abstract

This paper looks at predictability problems, *i.e.*, wherein an agent must choose its strategy in order to optimize the predictions that an external observer could make. We address these problems while taking into account uncertainties on the environment dynamics and on the observed agent's policy. To that end, we assume that the observer 1. seeks to predict the agent's future action or state at each time step, and 2. models the agent using a stochastic policy computed from a known underlying problem, and we leverage on the framework of observer-aware Markov decision processes (OAMDPs). We propose action and state predictability performance criteria through reward functions built on the observer's belief about the agent policy; show that these induced *predictable* OAMDPs can be represented by goal-oriented or discounted MDPs; and analyze the properties of the proposed reward functions both theoretically and empirically on two types of grid-world problems.

1 Introduction

In a human-agent collaboration scenario, some properties of the agent behavior can be useful for the human and sometimes allow a better collaboration. Recent papers suggest ways of obtaining such behaviors. In particular, when an agent is aware that it is being observed by a passive human, as in Figure 1, it can control the information disclosed to the observer through its behavior.

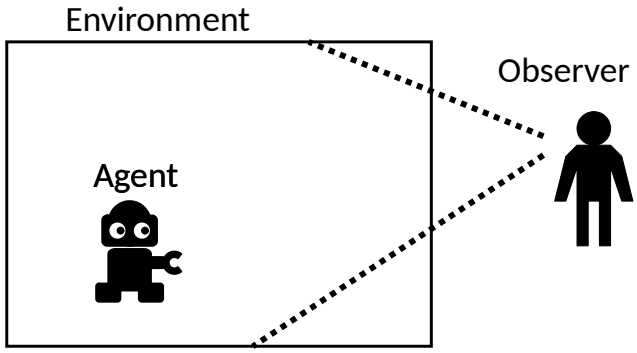


Figure 1: Agent in its environment and a passive observer

Chakraborti et al. [5] build on previous work to derive a taxonomy of these concepts. In particular, they distinguish between 1. transmitting information, with properties such as *legibility* (legible behaviors convey intentions, *i.e.*, actual task at hand, via action choices), *explicability* (explicable behaviors conform to observers' expectations, *i.e.*, they appear to have some purpose), and *predictability* (a behavior is predictable if it is easy to guess the end of an on-going trajectory); or 2. hiding information, as through *obfuscation*, when the agent tries to hide its real goal. They propose a general framework for such problems under the hypothesis that transitions are deterministic, and work mostly with plans (a sequence of actions inducing a state sequence). In their approach, the human is modeled

by the robot as having a model of the robot+environment system (including the robot's possible tasks), and is thus able to predict the robot behavior and adapt to it.

Each of the properties they discuss can be relevant in some situations. They convey different kinds of information to the observer, and can be mutually exclusive. Chakraborti et al. [5] point out that an explicable plan can be unpredictable, *e.g.*, when multiple explicable plans exist. Similarly, Fisac et al. [8] suggest that, if an agent acts legibly, then one can infer its goal but not necessarily how it is going to achieve this goal. *Predictability* is meant to ensure that the agent's behavior conveys this information.

Schadenberg et al. [21] explain that *Predictability* has a real interest when considering human-robot interaction. Their work mainly focuses on how human observers react to a *hand-coded* social robot behavior depending on whether the cause of responsive actions is visible or not. As we do in our experiments, they distinguish the participants' performance in predicting the robot's behavior, called the *behavioral predictability*, and their perception of the predictability of the robot behavior, called the *attributed predictability*. They observe that both predictabilities are not necessarily aligned, and point out that, depending on the scenario, one may want to optimize either the behavioral predictability, for instance with industrial robots, or the behavioral predictability, for instance with social robots. Unlike them, we are interested in *automatically deriving* predictable behaviors, and only consider fully observable settings.

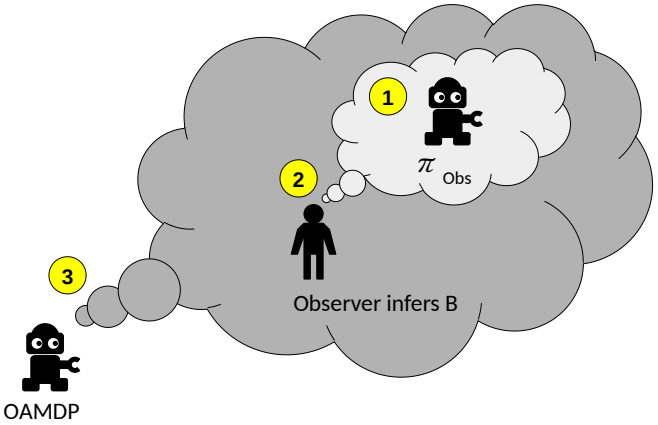


Figure 2: An OAMDP agent (3) assumes that the observer's expectation (2) is that the agent behaves so as to achieve some task (1).

Miura and Zilberstein [19] build a unifying framework while assuming stochastic transitions, namely *observer-aware Markov decision processes* (OAMDPs), adopting a similar approach as Chakraborti et al., as illustrated in Figure 2. Among other things, they work also on legibility, explicability, and predictability. Yet, as we will further discuss in Section 2, the two OAMDP approaches to predictability they consider are not fully satisfying: one amounts to returning an optimal policy for the low-level MDP, and the other reasons on full trajectories, which does not seem appropriate in a stochastic environment (and turns out to be prohibitive).

Our objective in this paper is to propose a more satisfying approach to predictability by working not with complete trajectories, but with actions or states at each time step. This implies reasoning on dynamic variables, which requires introducing a variant of the OAMDP formalism. Moreover, we also consider not only discounted problems, but also stochastic shortest-path (*i.e.*, goal-oriented) problems.

Section 2 provides background on Markov decision processes and observer-aware MDPs. Our approach to action and state predictability, through dedicated reward functions, is described in Section 3, along with proofs that well-defined problems are induced. Experiments are then presented in Section 4, where we generate and interprete policies on two types of grid-world problems, comparing with standard MDP solution policies, and then, in Section 5, where human observers are confronted with these policies on some of these problems.

2 Background

2.1 Markov Decision Processes

A Markov decision process (MDP) [3] is specified through a tuple $\langle \mathcal{S}, \mathcal{A}, T, R, \gamma, \mathcal{S}_\mathcal{T} \rangle$ where:

- $\mathcal{S}$ is a set of states;
- $\mathcal{A}$ is a set of actions;

- $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0; 1]$, the transition function, gives the probability $T(s, a, s')$ that action a performed in state s will lead to state s';
- $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$, the reward function, gives the immediate reward $R(s, a, s')$ received upon transition (s, a, s').
- $\gamma \in [0, 1]$ is a discount factor; and
- $\mathcal{S}_{\mathcal{T}} \subset \mathcal{S}$ is a set of terminal states: for all $s, a \in \mathcal{S} \times \mathcal{A}$, $T(s, a, s) = 1$ and $R(s, a, s) = 0$.

Then, a (stochastic) *policy* $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ maps states to distributions over actions, $\pi(a|s)$ denoting the probability to perform a when in s. When a policy is deterministic, $\pi(s)$ denotes the only possible action in s. Assuming $\gamma < 1$, the value of a policy π is the sum of discounted rewards on an infinite horizon:

$$V^{\pi}(s) \stackrel{\text{def}}{=} \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(S_t, A_t) | S_0 = s\right],$$

and an optimal policy π^* is such that, for all s, $V^{\pi *}(s) = \max_{\pi} V^{\pi}(s)$. The *value iteration* (VI) algorithm [3] approximates V^*, the value function common to all optimal policies, by iterating the following computation (where k is the current iteration):

$$V_{k+1}(s) \leftarrow \max_{a} \sum_{s'} T(s, a, s') \cdot \left(R(s, a, s') + \gamma V_k(s')\right).$$

Calculations stop when the *Bellman residual* is below a threshold:

$$\underbrace{\max_{s} |V_{k+1}(s) - V_k(s)|}_{\text{Bellman residual}} \leq \frac{1-\gamma}{\gamma}\epsilon.$$

Then, an ϵ-optimal policy is obtained by acting greedily with respect to the solution value function V_k, *i.e.*, using

$$\pi_k(s) \leftarrow \arg\max_{a} \sum_{s'} T(s, a, s') \cdot \left(R(s, a, s') + \gamma V_k(s')\right).$$

The same dynamic programming operator and ϵ-greedy selection apply when $\gamma = 1$ if $\mathcal{S}_{\mathcal{T}}$ is not empty.[1] Such problems are called *Shortest Stochastic Path* problems (SSPs) [4, 11]. SSPs are more general than MDPs because any MDP can be turned into an SSP with, at any time step, a $1 - \gamma$ probability to transition to a terminal state [4, Sec. 7.3].

Let us call *proper* a policy π that reaches $\mathcal{S}_{\mathcal{T}}$ with probability 1 from any state. We will from now on make the assumptions that, in our SSPs:

(A1) for any policy π and any state s, π reaches $\mathcal{S}_{\mathcal{T}}$ with probability 1 from s iff $V^{\pi}(s) > -\infty$; and

(A2) at least one proper policy π exists (*i.e.*, $\forall s$, $V^{\pi}(s) > -\infty$).

In particular, the first assumption holds if, for all $(s, a, s') \in (\mathcal{S} \setminus \mathcal{S}_{\mathcal{T}}) \times \mathcal{A} \times (\mathcal{S} \setminus \mathcal{S}_{\mathcal{T}})$, $R(s, a, s') < 0$.

2.2 Observer-Aware Markov Decision Processes

As introduced by Miura and Zilberstein, an observer-aware MDP (OAMDP) [19] models a situation wherein an agent attempts to maximize an observer's information regarding some target random variable, called *type*, under some model of the observer's evolving belief about this type. Formally, an OAMDP is described by an 8-tuple $\langle \mathcal{S}, \mathcal{A}, T, \gamma, \mathcal{S}_{\mathcal{T}}, \Theta, B, R \rangle$, where:

- $\langle \mathcal{S}, \mathcal{A}, T, \gamma, \mathcal{S}_{\mathcal{T}} \rangle$ is a reward-less discounted MDP ($\gamma < 1$);
- Θ is a finite set of *types* representing a characteristic of the agent such as possible goals, intentions or capabilities;
- $B : H^* \to \Delta^{|\Theta|}$ gives the assumed belief of the observer given a history ($H = \mathcal{S} \times \mathcal{A}$);
- $R : \mathcal{S} \times \mathcal{A} \times \Delta^{|\Theta|} \to \mathbb{R}$ is the reward function.

[1] No stopping criterion provides guarantees about the solution quality in general SSPs (cf. [10]). Here, we simply stop the algorithm when the Bellman residual is below some threshold $\eta \ll \epsilon$ and assume that V is ϵ-close to V^*.

In most of the cases they consider, Miura and Zilberstein derive B by relying on Baker et al.'s "BST"[2] Bayesian belief update rule [2], *i.e.*, considering that, again from the agent's viewpoint, the observer models the agent's behavior for a given type through an MDP by

1. using a corresponding reward function R^{θ}_{MDP};
2. solving the discounted MDP $\langle \mathcal{S}, \mathcal{A}, T, R^{\theta}_{\text{MDP}}, \gamma, \mathcal{S}_{\mathcal{T}} \rangle$ (where all components but R^{θ}_{MDP} come from the OAMDP definition) to obtain $V^{\theta,*}_{\text{MDP}}$;
3. building a stochastic "softmax" policy such that, $\forall (s,a)$,

$$\pi^{\theta}_{\text{MDP}}(a|s) = \frac{e^{\frac{1}{\tau} Q^{\theta,*}_{\text{MDP}}(s,a)}}{\sum_{a'} e^{\frac{1}{\tau} Q^{\theta,*}_{\text{MDP}}(s,a')}}, \text{ where}$$
$$Q^{\theta,*}_{\text{MDP}}(s,a) = \sum_{s'} T(s,a,s') \cdot \Big(r(s,a,s') + \gamma V^{\theta,*}_{\text{MDP}}(s') \Big),$$

   and temperature $\tau > 0$ allows tuning the policy's optimality (thus the agent's assumed rationality for the observer).

With $\pi_{\text{MDP}} \equiv (\pi^{\theta}_{\text{MDP}})_{\theta \in \Theta}$ in hand, the observer's belief function about the type can then be obtained through Bayesian inference.

Miura and Zilberstein [19] use the OAMDP framework to formalize various observer-aware problems from the literature, including legibility, explainability, and predictability. For predictability, which we now focus on, they mention two approaches. The first one builds on Dragan et al.'s idea to "model the predictability of a trajectory as simply proportional to the value (negative cost) of a trajectory" [7], which, in the OAMDP setting, translates into 1. having a single type θ^0, and 2. optimizing the underlying reward function $R^{\theta^0}_{\text{MDP}}$, *i.e.*, acting greedily wrt $Q^{\theta^0,*}_{\text{MDP}}$ (rather than following $\pi^{\theta^0}_{\text{MDP}}$). The second approach builds on Fisac et al.'s t-predictability [8], which maximizes $Pr(a_{t+1}, \dots, a_T | a_1, \dots, a_t)$ in deterministic settings, by using a type for each possible trajectory—*i.e.*, exponentially many types—over a finite temporal horizon.

In the following, we propose an alternative approach to predictability and discuss its properties.

3 Contribution

As a preliminary contribution, while Miura and Zilberstein consider only discounted OAMDPs, we introduce OASSPs (thus, using $\gamma = 1$). This mainly raises the question: Under which conditions do proper policies exist in the induced SSP? We will discuss this issue in the context of predictability.

3.1 Predictable Observer-Aware MDPs

Both approaches to predictability mentioned by Miura and Zilberstein are inspired by work in deterministic settings, reasoning on trajectories. Because both the softmax policy π_{MDP} and the dynamics of the system can be stochastic, we instead propose to try predicting either actions or states, both alternatives (*action* and *state predictability*) possibly leading to different solutions. Yet, OAMDP types θ are static variables (as types in Bayesian games [12, 9]), while actions and states are dynamic. This leads us to introduce pOAMDPs (predictable OAMDPs), where we instead talk of a (dynamic) *target variable*, also noted θ_t, which is now a function of the current transition: $\theta_t = \phi(s_t, a_t, s_{t+1})$. This 1. does not allow encoding problems where the target variable is static and hidden (*i.e.*, is a type), *e.g.*, legibility or explicability, but 2. still allows (a) defining and solving the observer's MDP (because the type does not influence the system dynamics), and (b) using the BST belief update (because of the Markovian nature of the target variables).

The following sections describe respectively, for both the action and state predictabilities: 1. how to derive B and solve the pOAMDP given a reward function R, and 2. the reward functions proposed to formalize predictability, along with properties of the resulting decision problems.

3.2 Belief Function and Properties of pOAMDPs

For action predictability, $\Theta = \mathcal{A}$, $\phi(s,a,s') = a$, and B is

$$B : \begin{array}{ccc} H^* & \to & \Delta^{|\mathcal{A}|}, \\ (s_0, a_0, \dots, s_t) & \mapsto & \pi_{\text{MDP}}(A_t | s_t). \end{array}$$

[2]The acronym stands for the authors' initial letters.

For state predictability, $\Theta = \mathcal{S}$, $\phi(s, a, s') = s'$, and B is

$$B : \begin{array}{ccc} H^* & \to & \Delta^{|\mathcal{S}|}, \\ (s_0, a_0, \ldots, s_t) & \mapsto & \sum_{a'} \pi_{\text{MDP}}(a'|s_t) \cdot T(s_t, a', S_{t+1}). \end{array}$$

In both cases, since B depends only on the current state, s_t, we can denote the belief about target variable θ under s_t as $B(s_t) = b_{s_t}(\theta)$ and redefine the pOAMDP reward function (not the observer's one) as $R'(s_t, a_t) \stackrel{\text{def}}{=} R(s_t, a_t, b_{s_t}(\theta))$ instead of $R(s_t, a_t, B(s_0, a_0, \ldots, s_t))$.

The agent's sequential decision-making problem can then be expressed as an MDP $\langle \mathcal{S}, \mathcal{A}, T, R', \gamma, \mathcal{S}_{\mathcal{T}} \rangle$ solvable with an algorithm such as value iteration. The solving complexity is thus the complexity of solving both the observer MDP and the MDP induced by the pOAMDP. In contrast, in the case of OAMDPs [19], one generally cannot obtain such an MDP, and solving the pOAMDP requires specific algorithms in which the action choice is linked to the whole state-action history (so that the tree of possible futures that needs to be accounted for grows exponentially).

3.3 pOAMDP Reward Function

Reward Definition When in state s, to predict the next target variable's value (action or state) as well as possible, the observer should pick one of the most likely values according to her model of the agent's behavior. This means picking an action in $\arg\max_{a \in \mathcal{A}} b_s(a)$ (or a state in $\arg\max_{s' \in \mathcal{S}} b_s(s')$). We will assume that the observer samples her prediction uniformly from this set, and thus define $\text{pred}(\theta|s) \stackrel{\text{def}}{=} \frac{1}{|\arg\max_{\theta \in \Theta} b_s(\theta)|}$ if $\theta \in \arg\max_{\theta \in \Theta} b_s(\theta)$, and 0 otherwise. Note: From now on, we focus on action predictability, only highlighting some points for state predictability.

Then, considering an SSP (thus with $\gamma = 1$), we would like to minimize the expected number of prediction errors made by the observer along a trajectory. For a single transition (s, a, s'), assuming the above model of observer prediction, the probability of a bad action prediction is $1 - \text{pred}(a|s)$. Because we are in a maximization rather than a minimization setting, and generalizing the formula to both action and state predictabilities, this leads to defining the reward function as:

$$R^{\Theta}_{\text{pred}}(s, a, s') \stackrel{\text{def}}{=} \text{pred}(\phi(s, a, s')|s) - 1.$$

Then, in any state s, $-V^*(s)$ gives the expected number of future prediction errors.

Valid SSPs? An important question is whether this reward function induces a valid SSP, which requires ensuring that assumptions (A1) and (A2) are satisfied.

Proposition 1. *Let us assume that (i) $\gamma = 1$, (ii) the MDP considered by the observer is a valid SSP, and (iii) $R^{\mathcal{A}}_{pred}$ is the pOAMDP reward function. Then the pOAMDP is a well-defined problem as its induced SSP satisfies assumptions (A1) and (A2).*

Proof. (A1) Let π be a policy, and (if it exists) $\mathcal{S}' \subseteq (\mathcal{S} \setminus \mathcal{S}_{\mathcal{T}})$ be a connex subset of states under π, *i.e.*, once reached, all states are visited infinitely often. Let $s' \in \mathcal{S}'$ be a state in which an optimal policy π^*_{MDP} of the observer SSP would leave $\mathcal{S}'$. Then, $\pi^*_{\text{MDP}}(s') \neq \pi(s')$, so that $\text{pred}(\pi(s')|s') < 1$ and $R^{\mathcal{A}}_{\text{pred}}(s', \pi(s'), s'') < 0$ for any s''. As a consequence, states in $\mathcal{S}'$ being visited infinitely often, for any $s \in \mathcal{S}'$, $V^{\pi}(s) = -\infty$. On the other hand, if, for some state $s \in \mathcal{S}$, π reaches $\mathcal{S}_{\mathcal{T}}$ with probability 1, then, trivially, $V^{\pi}(s) > -\infty$. This proves that (A1) holds.

(A2) Let us point out that whether a policy is proper or not depends on the reachability of terminal states, not on the reward function. Since the observer SSP satisfies assumption (A2) and only differs from the pOASSP in its rewards function, the induced SSP also satisfies assumption (A2). □

This result does not hold for state predictability.

Proposition 2. *Let us assume that (i) $\gamma = 1$, (ii) the MDP considered by the observer is a valid SSP, and (iii) $R^{\mathcal{S}}_{pred}$ is the pOAMDP reward function. Then the pOAMDP may be an ill-defined problem as its induced SSP satisfies assumption (A2), but may not satisfy assumption (A1).*

Proof. The proof that assumption (A2) holds is the same as for action predictability.

To prove that assumption (A1) may not hold, let us consider an OAMDP with:

- $\mathcal{S} = \{s_0, s_G\}$, with s_0 initial and s_G terminal;
- $\mathcal{A} = \{a_1, a_2\}$;
- the transition function described in Figure 3;

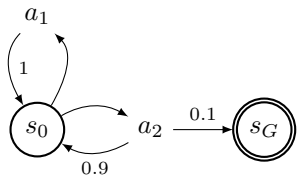


Figure 3: Transition function of an ill-defined p-OASSP for state predictability, with transition probabilities as edge labels.

- $r_{\text{MDP}}(s_0, a, s') = 1$ for all a and s'; and
- the state-predictability reward function.

In this setting, $\pi_{\text{MDP}}(s_0)$ is deterministic since there is a single optimal action when in s_0: a_2. Then, due to the transition function, when applying π_{MDP} in s_0, the most probable next state is s_0, meaning that the observer should bet on s_0. As a consequence, we can evaluate the policies π_{a_1} and π_{a_2} that respectively always pick a_1 and a_2 when in s_0:

$$V^{\pi_{a_1}}(s_0) = T(s_0, a_1, s_0) \cdot \left(R^{\mathcal{S}}_{\text{pred}}(s_0, a_1, s_0) + V^{\pi_{a_1}}(s_0)\right) \tag{1}$$
$$= 1 \cdot (0 + V^{\pi_{a_1}}(s_0)) \tag{2}$$
$$= 0,^{\text{[One could argue that the value is undefined.]}} \tag{3}$$

and

$$V^{\pi_{a_2}}(s_0) = T(s_0, a_2, s_0) \cdot \left(R^{\mathcal{S}}_{\text{pred}}(s_0, a_2, s_0) + V^{\pi_{a_2}}(s_0)\right) \tag{4}$$
$$+ T(s_0, a_2, s_G) \cdot \left(R^{\mathcal{S}}_{\text{pred}}(s_0, a_2, s_G) + V^{\pi_{a_2}}(s_G)\right) \tag{5}$$
$$= 0.9 \cdot (0 + V^{\pi_{a_2}}(s_0)) + 0.1 \cdot (-1 + 0) \tag{6}$$
$$= 0.9 \cdot V^{\pi_{a_2}}(s_0) - 0.1 \tag{7}$$
$$= -1. \tag{8}$$

π_{a_1} is thus the only optimal policy for state-predictability, although it never reaches the terminal state, which breaks assumption (A1). □

This negative result does not prevent from using $R^{\mathcal{S}}_{\text{pred}}$ in OA-SSPs, in particular:

- if linearly combined with another reward function that necessarily induces a valid SSP, *e.g.*, $R^{\mathcal{A}}_{\text{pred}}$ or a non-positive reward function (such as $R(s, a, s') = -1$ for any (s, a, s')); and
- in case of deterministic dynamics; indeed, if a policy π induces an absorbing subset of states $\mathcal{S}' \subseteq \mathcal{S} \setminus \mathcal{S}_{\mathcal{T}}$, then $T(s, \pi(s)) \neq T(s, \pi_{\text{MDP}}(s))$ for some $s \in \mathcal{S}'$, so that $R^{\mathcal{S}}_{\text{pred}}(s, \pi(s), T(s, \pi(s))) < 0$, meaning that π has a negative infinite value at least in s (so that assumption (A1) holds).

In the case of (discounted) MDPs, we will rely on the same reward definition. The interpretation of $-V^*(s)$ is similar if one sees the problem as an equivalent SSP with a $1 - \gamma$ termination probability at each time step.

The next two sections study this approach to action and state predictability on simple examples 1. *in silico*, observing and analyzing the policies obtained through our approach, and compared with simple MDP policies; and 2. *in vivo*, *i.e.*, confronting actual human observers with several policies.

4 Generating and Interpreting Policies

These first experiments aim at illustrating and better understanding the policies induced by the proposed reward function, and in particular at determining whether they can be considered as predictable. The code will be made available under an open license.

4.1 Protocol

To describe the two types of pOAMDPs considered in our experiments, let us just detail the corresponding MDPs, both set in 4-connected grid worlds, that the observer will take into account:

- an SSP, named *maze*, in which the agent wants to reach a terminal goal state; and
- a discounted MDP (with no terminal state), named *firefighter*, in which the agent uses water sources to extinguish fires.

To facilitate the analysis, most problems have deterministic dynamics.

Maze problem A *maze* (cf. Figure 4) contains walls (in dark grey), normal cells (in white), slippery cells (in cyan), and terminal cells (pink disks). The starting cell is marked by a circle. More formally, in this SSP:

- each state s in $\mathcal{S}$ indicates the (x, y) coordinates of the agent in a normal, slippery, or terminal cell;
- $\mathcal{S}_{\mathcal{T}}$ is a non-empty (but also possibly non-singleton) subset of $\mathcal{S}$;
- $\mathcal{A} = \{up, down, left, right\}$;
- $T(s, a, s')$ encodes the agent's moves: an agent in a normal cell moves in the direction indicated by its action if no wall prevents it; in a slippery cell, the agent has a probability p (0.5 in our experiments) of making a 2-cell rather than 1-cell move (if possible); in a terminal cell, the agent does not move;
- R_{MDP}, the reward function, returns
  - a default penalty of -0.04 for each move,
  - -1 when the agent hits a wall,
  - $+1$ upon reaching a terminal state s_f, and
  - 0 when the agent stays in the terminal state.

This SSP trivially satisfies assumptions (A1) and (A2).

Firefighter problem Similar grids are used for the *firefighter* problem, but with terminal cells replaced by fires and water sources (cf. Figure 6). The agent now has a water tank, which is emptied upon reaching a (never extinguished) fire, and filled upon reaching a (never emptied) water source. More formally, in this $\gamma = 0.99$-discounted MDP:

- each state s in $\mathcal{S}$ is represented by a triplet (x, y, w) with (x, y) the agent's coordinates and w a boolean encoding whether its water tank is full;
- $\mathcal{A} = \{up, down, left, right\}$;
- $T(s, a, s')$ is similar to the *maze* problem, except that w becomes false upon reaching a fire, and true upon reaching a water source;
- R_{MDP}, the reward function, returns
  - a default penalty of -0.04 for each move,
  - -1 when the agent hits a wall, and
  - $+1$ when the agent reaches a fire while carrying water ($w =$true).

Optimal MDP policies consist in endlessly going back and forth between a water source and a fire.

Baseline Policies The pOAMDP solution policy, denoted $\pi^{\Theta}_{\text{pred}}$, will be compared with near-optimal solutions of the observer MDP obtained as follows. We solve the observer MDP until convergence to an ϵ-optimal value function. Then, in each state s, let $\psi(s) \stackrel{\text{def}}{=} \{a \in \mathcal{A} \mid Q^*(s, a) \leq V^*(s, a) - 2\epsilon\}$. This set necessarily contains all optimal actions. With this, we can first define $\pi_{\text{MDP-S}}$, a *stochastic* policy that, in each state s, samples actions uniformly from $\psi(s)$.

In practice, algorithms will often be biased, having a preference order over actions. We thus also consider the policies that, in each state s, deterministically pick the preferred action given a predefined order. These *biased* (and deterministic) policies are denoted $\pi_{\text{MDP-B}}$, not distinguishing them from each other.

pOAMDP Model For both types of problems and for each grid environment, a pOAMDP is derived using the previously proposed reward function for predictability R^{Θ}_{pred}. The baseline policy $\pi_{\text{MDP-S}}$ described above serves to identify the observer's possible predictions. Since each pOAMDP can be considered as an MDP, pOAMDPs are solved by using again the value iteration algorithm with an appropriate discount factor (details in the next section), resulting in the R^{Θ}_{pred} policy. Note that our approach does not make use of the softmax policy, thus making its temperature parameter τ irrelevant.

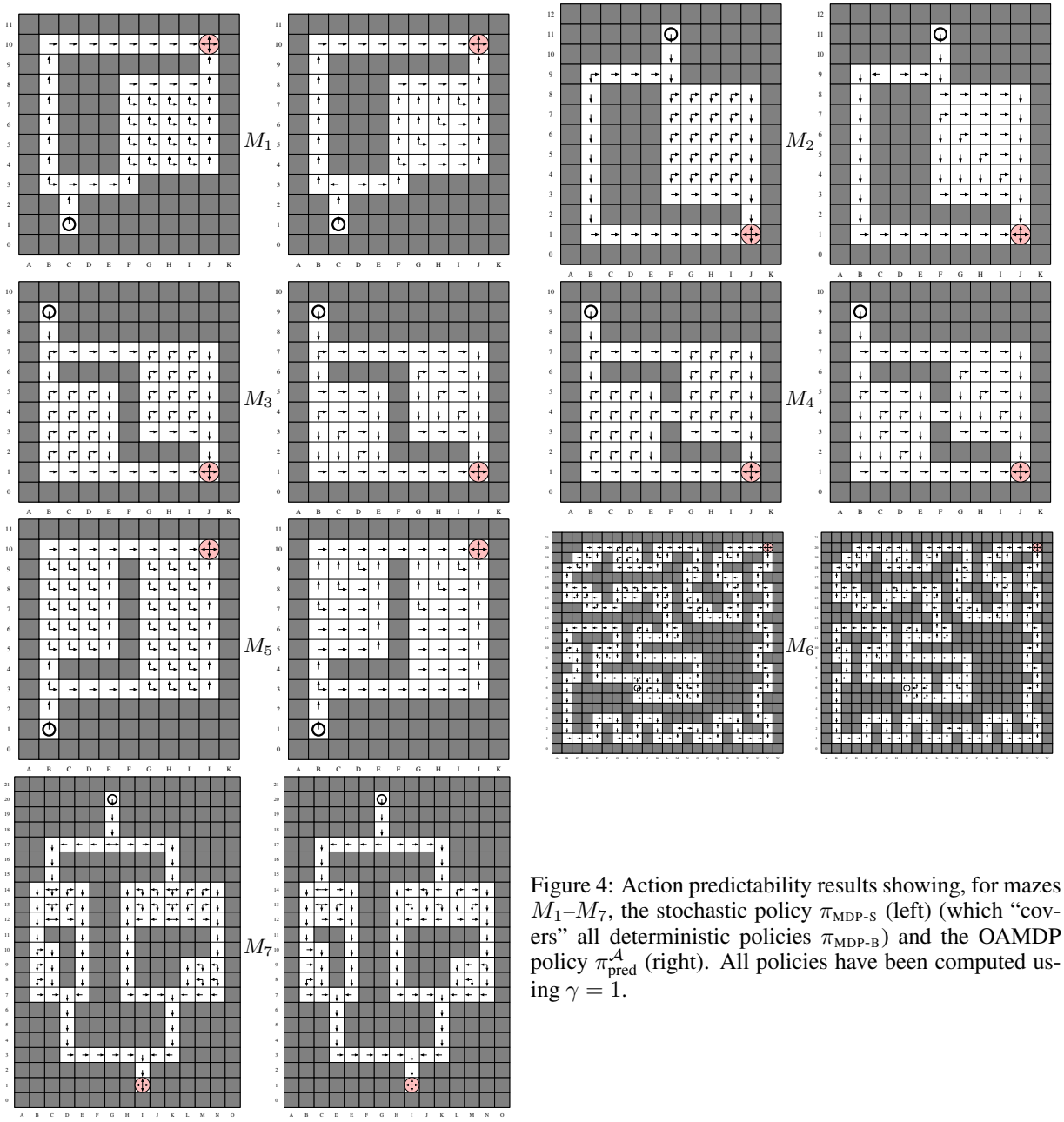


Figure 4: Action predictability results showing, for mazes M_1–M_7, the stochastic policy $\pi_{\text{MDP-S}}$ (left) (which "covers" all deterministic policies $\pi_{\text{MDP-B}}$) and the OAMDP policy $\pi^{\mathcal{A}}_{\text{pred}}$ (right). All policies have been computed using $\gamma = 1$.

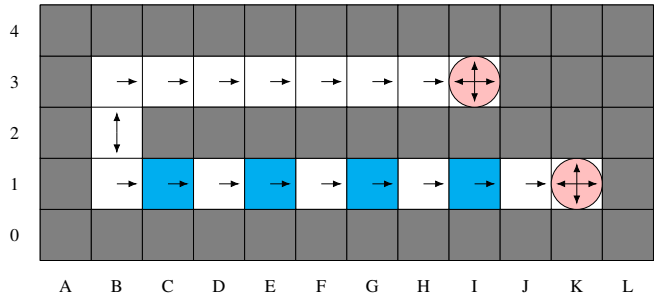
4.2 Results

The figures present both stochastic MDP policies $\pi_{\text{MDP-S}}$ (which also "cover" all deterministic policies $\pi_{\text{MDP-B}}$), and pOAMDP policies $\pi^{\Theta}_{\text{pred}}$, the arrows indicating all ϵ-optimal actions.

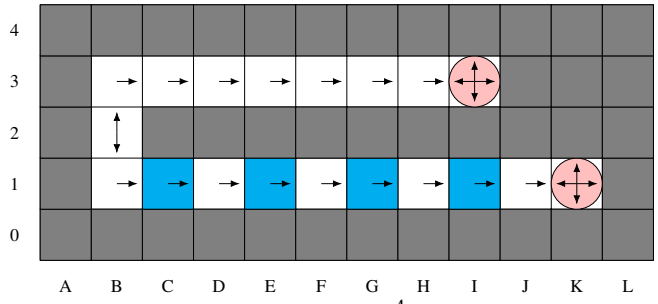
4.2.1 Maze problem

Grids used The mazes mainly consist of corridors and (empty) rooms. For action predictability, we expect the pOAMDP policies to prefer corridors over rooms (which allow for more possible optimal actions). Figure 4 shows mazes M_1–M_7, which have been used for action predicatility (including experiments with humans discussed in Section 5). They all consist in a number of corridors and rooms, have a starting state s_0 (marked by a circle), and overall increase in complexity from M_1 to M_7. The maze M_8 in Figure 5 consists of 2 corridors that lead to a terminal state. One of these corridors contains slippery cells, but the average traversal time is the same for both. This maze's goal is to observe differences between $R^{\mathcal{A}}_{\text{pred}}$ and $R^{\mathcal{S}}_{\text{pred}}$.

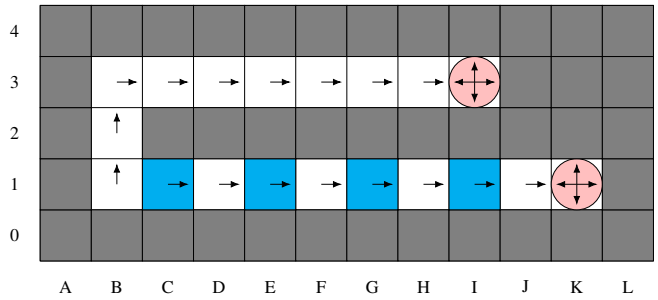
Each SSP is solved with $\gamma = 1$ and $\epsilon = 0.001$. As expected, when crossing a room of size $n \times m$ from one corner to the opposite corner, $\pi_{\text{MDP-S}}$ randomly picks one of the $\binom{n+m}{n}$ optimal paths, while the only two possible $\pi_{\text{MDP-B}}$ policies follow the walls (clockwise or counterclockwise).



(a) Stochastic policy $\pi_{\text{MDP-S}}$ ($\gamma = 1$)



(b) pOAMDP policy $\pi^{\mathcal{A}}_{\text{pred}}$ ($\gamma = 1$)



(c) pOAMDP policy $\pi^{\mathcal{S}}_{\text{pred}}$ ($\gamma = 1$)

Figure 5: Results for maze M_8

Note: In the following, we mainly focus on action predictability because, here, solution policies turn out to be identical for state predictability. This is favored in deterministic environments, where predicting the next state is often equivalent to predicting the next action.

Analysis of $\pi^{\mathcal{A}}_{\text{pred}}$ and $\pi^{\mathcal{S}}_{\text{pred}}$ We observe several interesting behaviors with $R^{\mathcal{A}}_{\text{pred}}(s, a, s')$:

1. The pOAMDP agent will plan a longer path through a narrow corridor, where its next action will be easy to predict, rather than a shorter path going through one or multiple rooms as illustrated on M_1 and M_6.
2. In rooms, $\pi_{\text{MDP-S}}$ has two optimal actions except along the two walls near the exit, with a single optimal action. The pOAMDP agent behaves thus more predictably by going towards the closest of these two exit walls and following it, as visible in M_1–M_7.
3. In M_3, the pOAMDP agent can choose between (i) a corridor leading to a room, and (ii) a room leading to a corridor. When $\gamma = 1$, the pOAMDP agent has no preference. When $\gamma < 1$ (policy not shown here), the pOAMDP agent prefers to go through a corridor first because the discount puts more importance on early rewards (see cell $(B, 7)$).
4. In M_4, adding a door compared to M_3 makes for more uncertainty in the left room, so that the agent prefers going towards the right room.
5. In Figure 5, cell $(B, 2)$, $\pi^{\mathcal{A}}_{\text{pred}}$ has no preference between going up and down as, in both cases, there is no ambiguity about optimal actions afterwards.

Quantitative results in the first column of Table 1 are obtained by computing the value of each policy wrt $R^{\mathcal{A}}_{\text{pred}}$ and displaying $-V^{\pi}_{R^{\mathcal{A}}_{\text{pred}}}(s_0)$. They show that $\pi_{\text{MDP-S}}$'s expected number of errors per trajectory is worse than for the two

Table 1: Results for Maze problems M_1–M_7 with actual human observers against 3 agents: $\pi_{\text{MDP-S}}$, $\pi_{\text{MDP-B}}$, π_{pred}, indicating: [#Err.p] the predicted average number of errors when evaluating the policy using $R^{\mathcal{A}}_{\text{pred}}$; [#Err.h] the actual average number of errors per trajectory with human observers; [#steps] the number of time steps to reach the goal; [time/step] the average response time of the human observer per time step.

		#steps	#Err.p	#Err.h	time/step (ms)
π_1	M_1	15.0	2.9	3.4±1.7	561.3± 188.2
	M_2	13.0	3.3	3.9±1.5	600.7± 279.2
	M_3	15.0	2.9	2.9±1.6	490.8± 167.5
	M_4	15.0	3.1	3.3±1.3	574.5± 188.2
	M_5	16.0	3.3	3.1±1.6	506.4± 124.2
	M_6	84.0	10.5	12.8±2.1	481.7± 101.6
	M_7	29.0	2.6	2.7±1.5	437.5± 100.8
	$\bigoplus_i M_i$	189.0	28.5	30.5± -	499.1± -
π_2	M_1	15.0	2.0	1.1±0.9	379.3± 110.6
	M_2	13.0	2.1	1.0±0.8	350.7± 94.0
	M_3	15.0	2.1	1.0±0.9	370.9± 120.9
	M_4	15.0	2.1	0.9±0.8	334.1± 86.3
	M_5	16.0	2.5	0.6±0.7	361.4± 96.0
	M_6	84.0	10.3	10.1±2.5	436.8± 81.5
	M_7	29.0	2.8	2.5±0.8	392.5± 109.7
	$\bigoplus_i M_i$	189.0	24.0	16.4± -	400.1± -
π_3	M_1	17.0	1.5	1.1±0.9	311.0± 74.8
	M_2	13.0	2.0	1.5±1.1	364.4± 125.9
	M_3	15.0	2.0	1.7±1.1	353.3± 94.7
	M_4	15.0	2.0	0.5±0.7	346.3± 82.8
	M_5	16.0	2.0	1.4±1.1	371.8± 143.0
	M_6	86.0	2.7	2.3±1.6	310.4± 62.6
	M_7	29.0	1.7	1.8±1.3	376.2± 94.2
	$\bigoplus_i M_i$	192.0	13.8	9.7± -	335.1± -

other agent policies, in particular when large rooms exist. Also, $\pi^{\mathcal{A}}_{\text{pred}}$ has significantly better results than the two other policies on problems M_6 & M_7, which have multiple rooms and are more complex.

In most of these problems, $\pi^{\mathcal{A}}_{\text{pred}}$ and $\pi^{\mathcal{S}}_{\text{pred}}$ exhibit identical behaviors. This is not the case in maze M_8 (Figure 5), as $\pi^{\mathcal{S}}_{\text{pred}}$ prefers going up in cell $(B, 1)$, which goes against the observer's predictions, to follow the path with no slippery cells (as slippery cells induce state uncertainties).

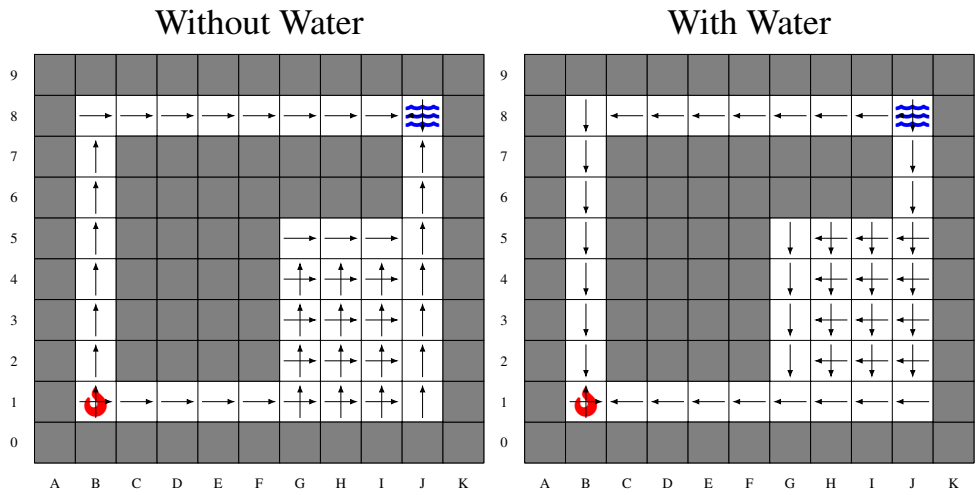
4.2.2 Firefighter problem

Grids used The following grids where used to test the reward functions:

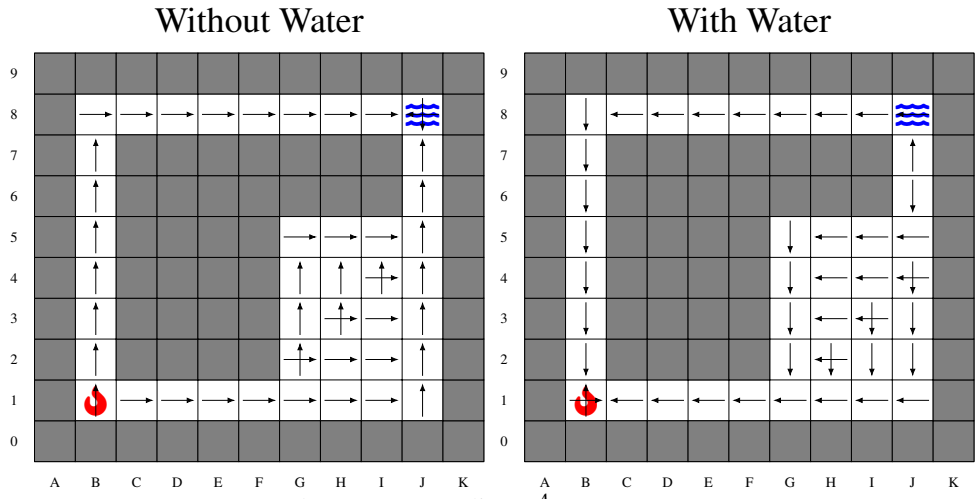
1. the grid in Figure 6 contains 1 fire and 1 water source linked by a room and by a corridor;
2. the grid in Figure 7 is a room with 2 fires and 2 water sources;
3. the grid in Figure 8 contains 2 fires and 2 water sources; a part of the map is a room and the other part is a corridor.

The underlying MDPs are not SSPs anymore, so that we use $\gamma = 0.99$-discounted pOAMDPs.

Analysis of $\pi^{\mathcal{A}}_{\text{pred}}$ A behavior similar to the maze problem can be observed. In Figure 6, $\pi^{\mathcal{A}}_{\text{pred}}$ prefers the corridor over the open room. In such rooms, $\pi^{\mathcal{A}}_{\text{pred}}$, as $\pi_{\text{MDP-B}}$ (see Figure 11, page 16), tries to reach a wall and walk along it (Figures 6 and 8). In Figure 7, the pOAMDP agent tries to be more predictable by walking along the wall or by reaching Row 5 or Column F to reduce the number of optimal paths to reach the fire in the middle. In Figure 8, the pOAMDP agent prefers the fire located in $(B, 1)$ and the water source located in $(E, 8)$ even if another water source ou fire spot is closer. This is particularly visible on the "without water" side of the figure, where $\pi^{\mathcal{A}}_{\text{pred}}$ goes from $(G, 5)$ to $(E, 8)$ to refill.



(a) Stochastic policy $\pi_{\text{MDP-S}}$ ($\gamma = 0.99$)



(b) pOAMDP policy $\pi^{\mathcal{A}}_{\text{pred}}$ ($\gamma = 0.99$)

Figure 6: Results for firefighter problem F1

5 Experimenting with Humans

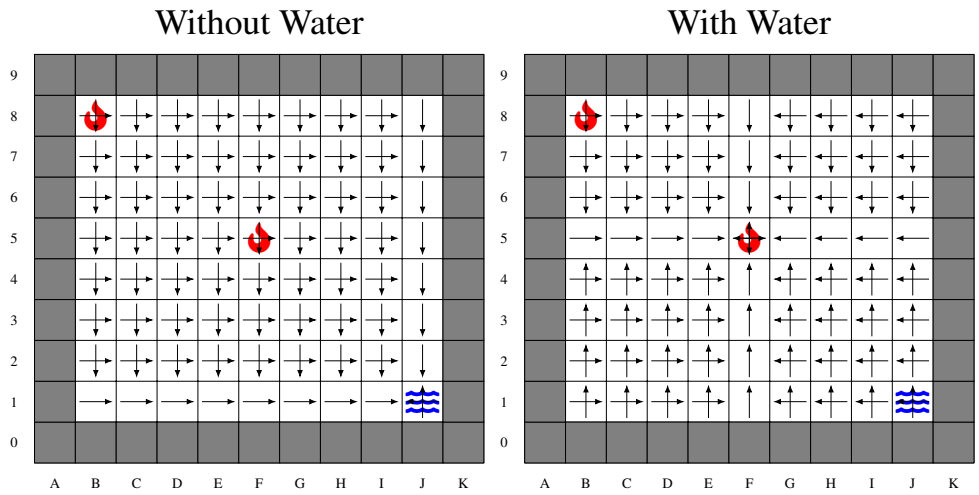
The objective of pOAMDP solution policies is to make it easier for an observer to predict actions or states. Of particular interest is the case of human observers. An experiment was thus conducted to confront actual human participants with stochastic and biased MDP policies ($\pi_{\text{MDP-S}}$ and $\pi_{\text{MDP-B}}$), and with pOAMDP policies ($\pi^{\mathcal{A}}_{\text{pred}}$). We were interested in particular in:

- assessing how predictable each type of policy was for humans, by measuring the number of prediction errors;
- assessing whether predictions were easy to make, by measuring their response times; and
- knowing how the various agent behaviors were perceived by humans.

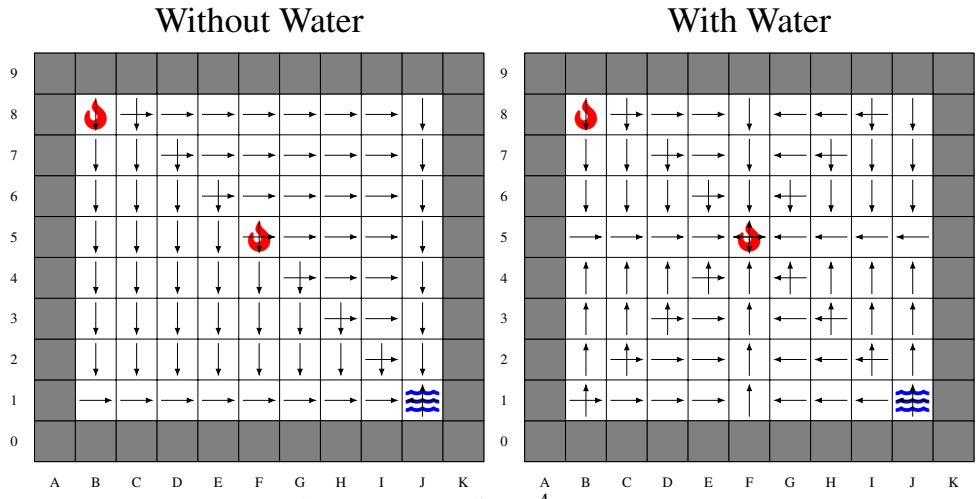
5.1 Protocol

Participants Experiments have been conducted with 20 human participants (4 women; aged 28.9 ± 7.7 years) to assess the actual predictability of the 3 policies at hand on mazes M_1–M_7 (Figure 4). All participants provided written consent prior to their participation.

Task Participants were seated in front of a computer displaying a maze containing a robot and the robot's goal. For each position of the robot, at each time step, the participant had to indicate the next action by pressing one of the four arrow keys. The robot then moved to the next position according to the policy, independently of the participant's response, and the participant had to indicate the next action, and so on along the trajectory to the goal.



(a) Stochastic policy $\pi_{\text{MDP-S}}$ ($\gamma = 0.99$)



(b) pOAMDP policy $\pi^{\mathcal{A}}_{\text{pred}}$ ($\gamma = 0.99$)

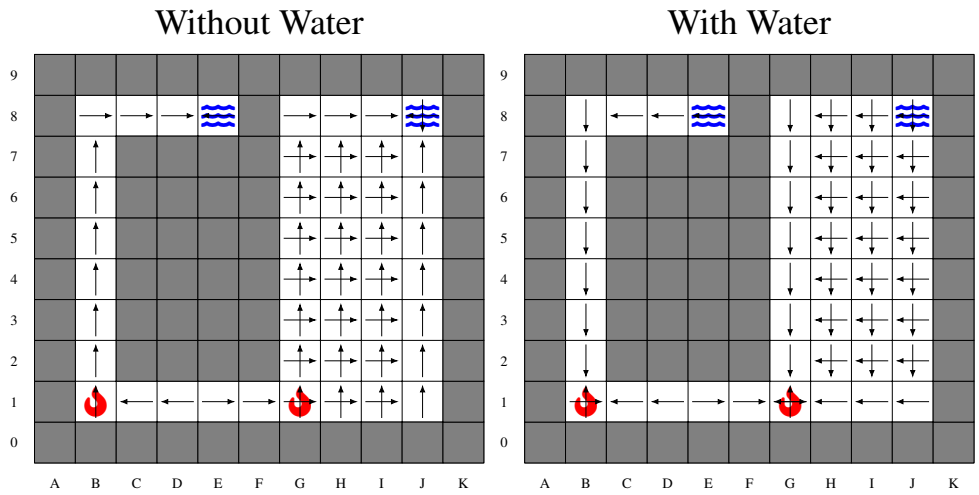
Figure 7: Results for firefighter problem F2

Experimental process Participants began with a learning phase lasting about one minute, consisting of a maze with a random policy. Then came the test phase, consisting of 3 sequences of 7 mazes each, each sequence associated with a policy. Participants were told that the robot behavior was going to change at each sequence. Each robot was identified by a color. The ordering of policies was randomized, as well as the ordering of mazes within a sequence, with the exception that M_6, the largest maze, was always presented in 4th position. For $\pi_{\text{MDP-B}}$, 4 different orderings over actions were used as biases (out of $4! = 24$ possibilities), and randomly sampled before each trajectory. All previously mentionned randomizations were controlled (hand-written) to prevent unwanted regularities.

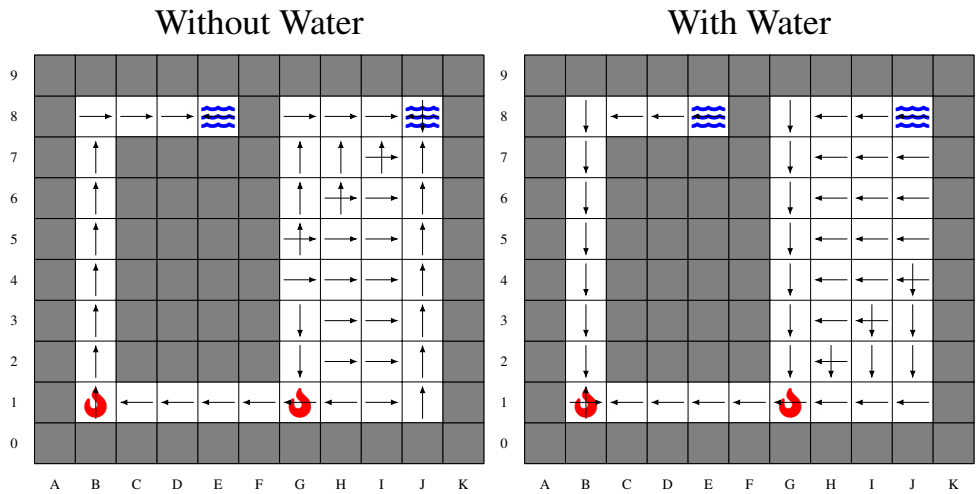
At the end of each sequence, the participant completed a 3-item questionnaire. For each item, the participant answered on a 7-point Lickert scale from "strongly disagree" to "strongly agree". The 3 items related to the policy they had just seen, and were as follows: 1. this robot was easy to anticipate (*Anticipation*); 2. its decisions seemed generally logical (*Logic*); 3. some of its decisions surprised me (*Surprise*). Each participant completed this questionnaire 3 times, once per policy. Once the test phase was over, they completed a questionnaire including: socio-demographic questions; a request to rank the three policies from easiest to most difficult, and another from most logical to most unexpected.

On average, the experiment lasted 30 minutes.

Data analysis The data recorded during each maze were: the number of errors, *i.e.* the number of times the next move predicted by the participant did not correspond to the move subsequently chosen by the robot ; the response time (in ms), *i.e.* from the instant when the robot finished a move to the instant when the participant indicated the position he thought would be the next. Each maze began with a two-square corridor to control the start of each trajectory, and the first square (the first response given by the participants) of each maze was removed from



(a) Stochastic policy $\pi_{\text{MDP-S}}$ ($\gamma = 0.99$)



(b) pOAMDP policy $\pi^{\mathcal{A}}_{\text{pred}}$ ($\gamma = 0.99$)

Figure 8: Results for firefighter problem F3

the analyses. One participant was removed from the analyses, as his response times were more than 3 standard deviations above the overall mean. Data processing on errors and response times was therefore carried out on 19 participants.

For the questionnaire, each of the 3 items was analyzed (Anticipation, Logic and Surprise).

To determine whether there were any significant differences between the three policies, standard errors were calculated for the quantitative variables, as well as for the questionnaire.

5.2 Results

5.2.1 Numbers of Errors and Response Times

The main quantitative results are presented in Table 1, page 10, as well as in Figure 9 for errors and in Figure 10 for response times, for each policy-maze combination, plus a fake maze $\bigoplus_i M_i$ whose results are obtained by assuming that the other mazes have been concatenated. The 1st column shows the expected number of errors per trajectory according to our model ($-V^{\pi}_{R^{\mathcal{A}}_{\text{pred}}}(s_0)$), which can be compared with the measured values with human observers in the 2nd column. Values are rather similar for $\pi_{\text{MDP-S}}$, with typically a few more errors made by humans. Human scores are notably better than anticipated for $\pi_{\text{MDP-B}}$ (and also better than human scores with $\pi_{\text{MDP-S}}$), because humans very quickly learn the agent's bias, which facilitates predictions in large rooms. The benefit of learning is very limited in complex mazes with many small rooms as M_6. Human scores with $\pi^{\mathcal{A}}_{\text{pred}}$ are worse than with $\pi_{\text{MDP-B}}$ on simple mazes (where learning biases helps), but notably better on complex mazes M_6+M_7.

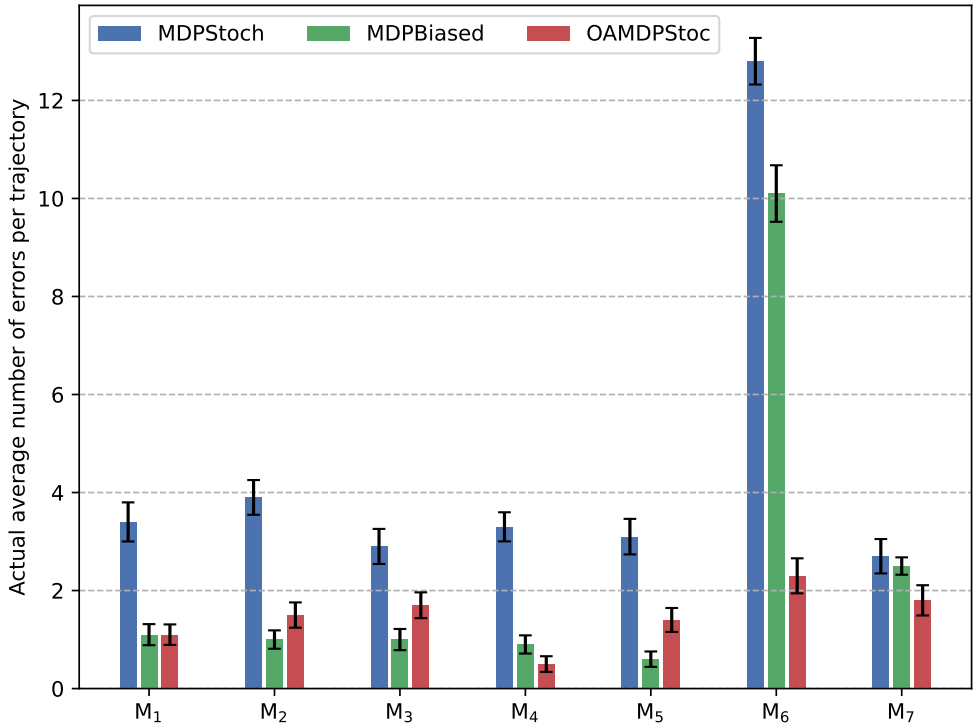


Figure 9: Graph representing the average number of errors made by the participants for each maze

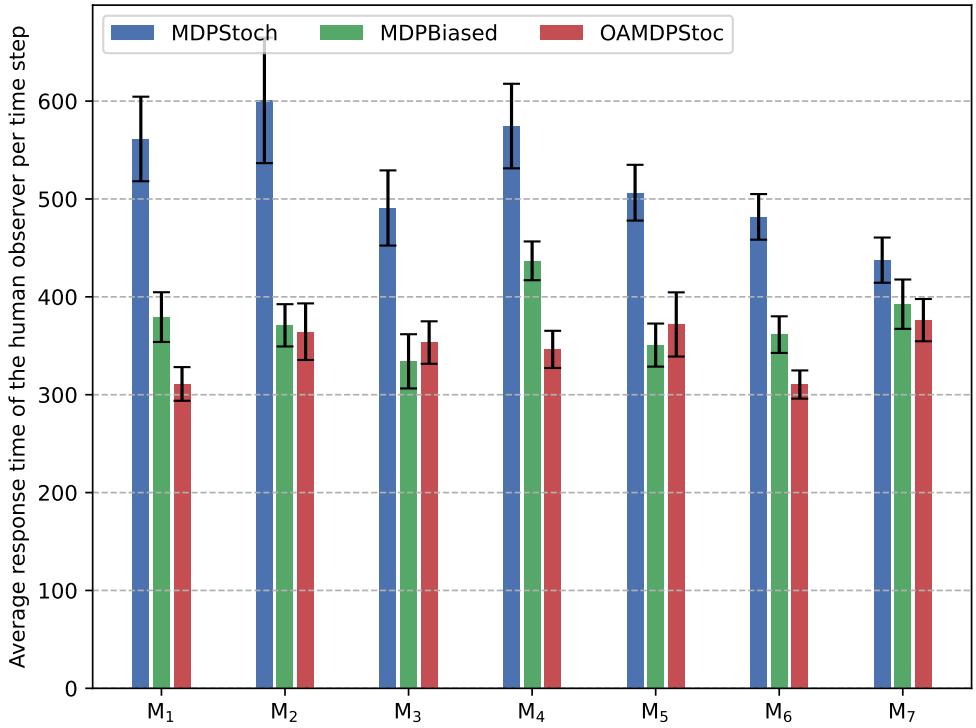


Figure 10: Graph representing the average response time of the participants for each maze

As complementary information, the 3rd column provides the (constant) lengths of trajectories in each case as an indicator of the problem size. As anticipated, $\pi_{\text{MDP-S}}$ and $\pi_{\text{MDP-B}}$ generate minimal-length trajectories, while $\pi^{\mathcal{A}}_{\text{pred}}$ generates slightly longer ones in some cases (M_1+M_6) to follow more predictable paths.

The 4th column indicates the average response time (in ms) per cell, which appears to be inversely related to the difficulty to make predictions. These average response times are lower for $\pi_{\text{MDP-B}}$ and $\pi^{\mathcal{A}}_{\text{pred}}$ than for $\pi_{\text{MDP-S}}$. An important difference between response times of $\pi_{\text{MDP-B}}$ and $\pi^{\mathcal{A}}_{\text{pred}}$ can be observed for M_6. In this maze, it is harder for the human to learn the agent's bias of $\pi_{\text{MDP-B}}$, while $\pi^{\mathcal{A}}_{\text{pred}}$ plans its actions to go through states with reduced action ambiguity.

5.2.2 Heatmaps

Heatmaps allow us to visualize where in the maze the participants made mistakes or were faster/ slower to take a decision.

Error Rate Heatmaps Error rate heatmaps for each maze in each policy are shown Figure 11. They are defined as follows:

- the blue color represents the average number of visits computed as $\frac{\#visits}{\#participants}$, and
- the red color represents the average error rate computed as $\frac{\#errors}{\#visits}$.

Note that, in cells with a low number of visits (light blue background color), the error rate estimate is poor compared to often-visited cells (dark blue background color). In unvisited cells (white background color), there is no error rate to estimate, hence the lack of inner square.

The results given by the error rate heatmaps show that:

- as expected, participants made many mistakes in open areas with the stochastic MDP policy;
- in the OAMDP policy, mistakes were mostly made in the beginning when participants needed to make a choice for example in maze M_1, cell $(C, 3)$;
- in some cases, as in maze M_1, cell $(B, 3)$, participants make mistake even if there is no ambiguity over the robot next action. one hypothesis to explain this observation is that the participants try to anticipate the robot's next action so that, when they make a mistake, they are likely to also make another one for the next action. This is coherent with some remarks made by the participants afterward: they anticipate the robot behavior over several time steps and, when the robot behavior did not match their expectation, they were unable to correct their predictions;
- for the OAMDP policy, participants tend to make mistakes whenever the robot changes direction. It seems that, for a human observer, changing direction is more costly than going forward;
- except for maze M_6 (which was designed specifically to minimize the usefulness of the bias), participants perform well with the biased policy. The chosen bias for the maze problem probably facilitates a lot the human prediction.

Response-Time Heatmaps Response-Time heatmaps for each maze in each policy are shown on Figure 12. The response-time heatmap is defined as follows:

- the blue color represents the average number of visits computed as $\frac{\#visits}{\#participants}$, and
- the green color represents the average response time computed as $\frac{\#time}{\#visits}$.

To make the heatmaps more readable, we cap the values to 1000 ms, any value above 1000 ms being indicated by a black cell. The goal of these heatmaps was to see where the participants decision making is slow, if there was any kind of anticipation, and the overall response time depending on the policies.

The response-time heatmaps show that:

- in open areas, the participants take more time to make a decision;
- consistent with the error heat map, for the OAMDP policy, participants take more time to make decision in the beginning, where they need to make a choice;
- in the OAMDP policy, we can notice that, after unexpected actions, the participants take more time for the next prediction. For example in M_1, cell $(C, 3)$, the robot action surprised the participants and their response time is higher in the next few cells. These cells also match the cells that were wrongly predicted in the error rate heatmap;
- the response time in corridors is less important, which is due to the participants anticipation.

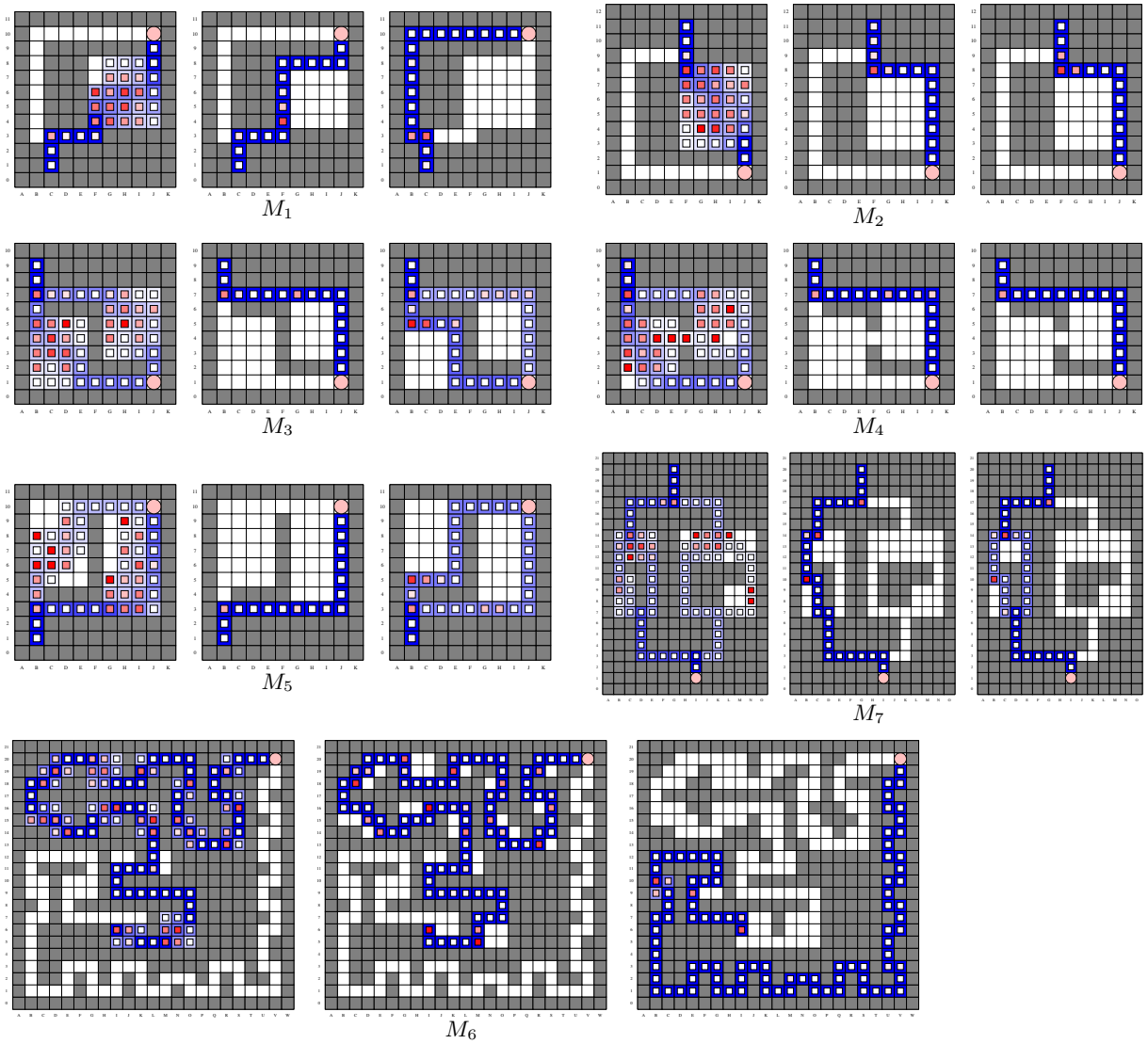


Figure 11: [Error Rate] Heatmaps showing, for each cell, (1) [in background] the probability of visit during a trajectory from dark blue ($P = 1$) to white ($P = 0$), and (2) [in the middle] the action-prediction error rate from dark red ($P = 1$) to white ($P = 0$). These heatmaps are provided for each maze and each policy, from left to right: stochastic MDP policy, biased MDP policy, and pOAMDP policy.

Heatmaps of "Reflex" Responses Heatmaps of "reflex" responses for each maze in each policy are shown on Figure 13. In case of a response time below 150 ms, the decision is more of a reflex, and the participant will not be able to correct his choice if needed [13, 18]. These heatmaps show which portion of participants answer in less than 150 ms in a state. They are more precisely defined as follows:

- the blue color represents the average number of visits computed as $\frac{\#visits}{\#participants}$, and
- the orange color represents the rate of response times below 150 ms, computed as $\frac{N}{\#visits}$ with N the number of times a participant answers in less than 150 ms.

To make the heatmaps more readable, we cap the values to a 50% rate of response times below 150 ms. The goal of this heatmap was to see where humans tend to predict the robot's action using anticipation or reflex responses. For example, when asked to predict the robot's move in a corridor at time t, some human participants might anticipate the prediction they will have to make at $t + 1$ (and afterwards), thus pressing keys as fast as possible along the corridor. This should result in a series of response times below 150 ms. Note that not all participants will anticipate the robot's next move at $t + 1$, and some of them might wait for the screen to redraw the robot's new position at $t + 1$ before answering.

This is also the reason why we use the 50% cap.

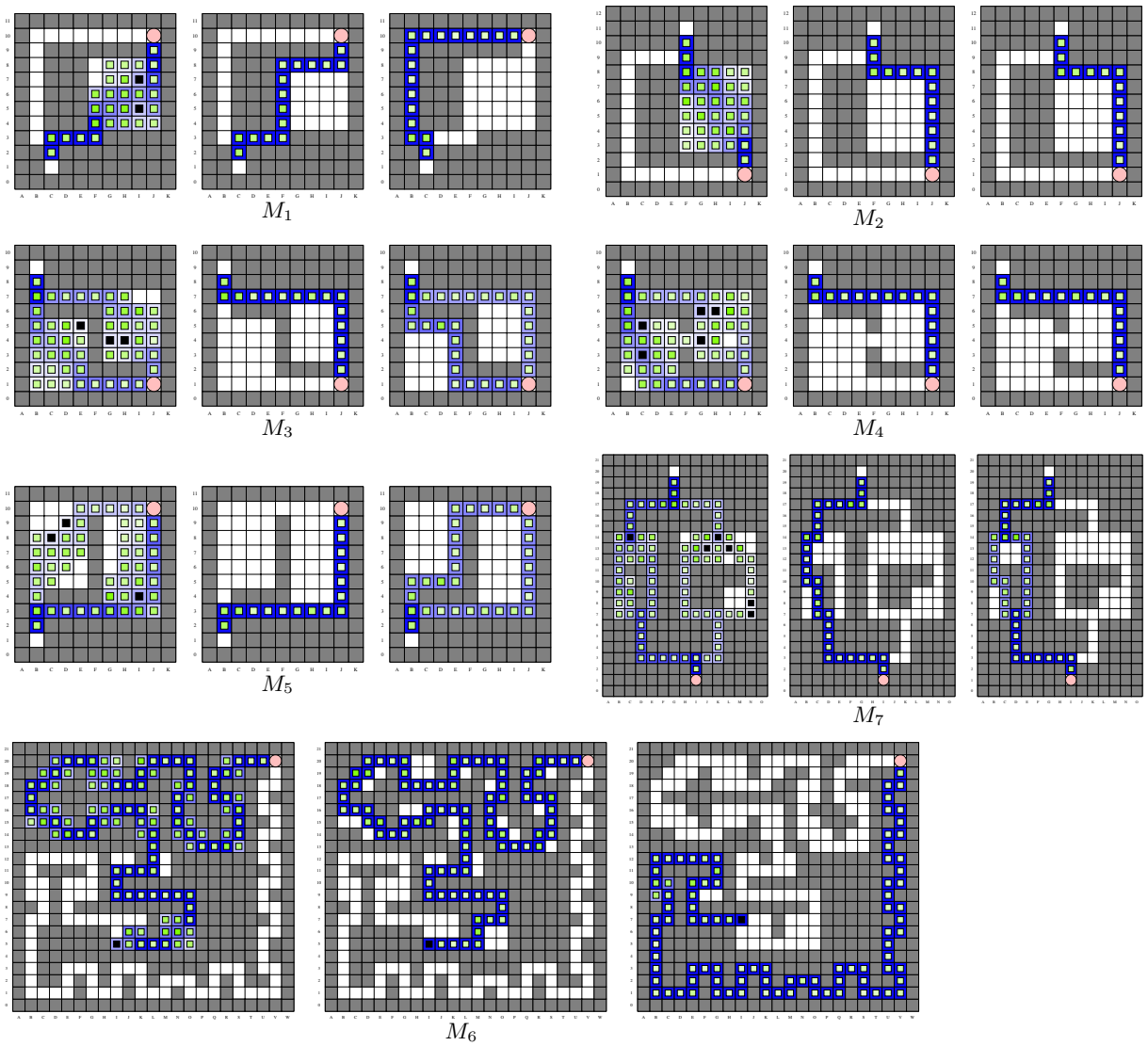


Figure 12: [Response-Time] Heatmaps showing, for each cell, (1) [in background] the probability of visit during a trajectory from dark blue ($P = 1$) to white ($P = 0$), and (2) [in the middle] the response time from green (1000 ms) to white (0 ms). These heatmaps are provided for each maze and each policy, from left to right: stochastic MDP policy, biased MDP policy, and pOAMDP policy.

These response time heatmaps show that:

- in rooms, most response times are above 150 ms, except along walls, either because following the wall is optimal, or because the agent is obviously going in a straight line. Note that the stochastic MDP policy has more chances to go through the center of rooms than to follow walls;
- in the corridors, many participants answer in less than 150 ms, as can be seen in M_1 and in M_2 for example. As explained earlier, in straight line, participants can anticipate the next action and be much faster.

5.2.3 Questionnaire and ranking

After each sequence of mazes corresponding to a policy, participants were asked to rate this policy on a scale of 1 to 7 according to three dimensions: Anticipation, Logic and Surprise (see Section 5.1). The average scores obtained, as well as standard errors, are shown in Figure 14. Overlapping error bars (based on standard errors) indicate no differences, while non-overlapping error bars indicate significant differences. Concerning Anticipation, the results seem to indicate that $\pi_{\text{MDP-S}}$ is considered more difficult to anticipate than the other two policies, and that $\pi_{\text{MDP-B}}$ is tendentially a little more difficult to anticipate than $\pi^{\mathcal{A}}_{\text{pred}}$. Concerning Logic, the results seem to indicate no significant difference between the three policies, with $\pi_{\text{MDP-B}}$ tending to be judged slightly more logical than the other two. Concerning Surprise, the results seem to indicate no significant difference between the three policies.

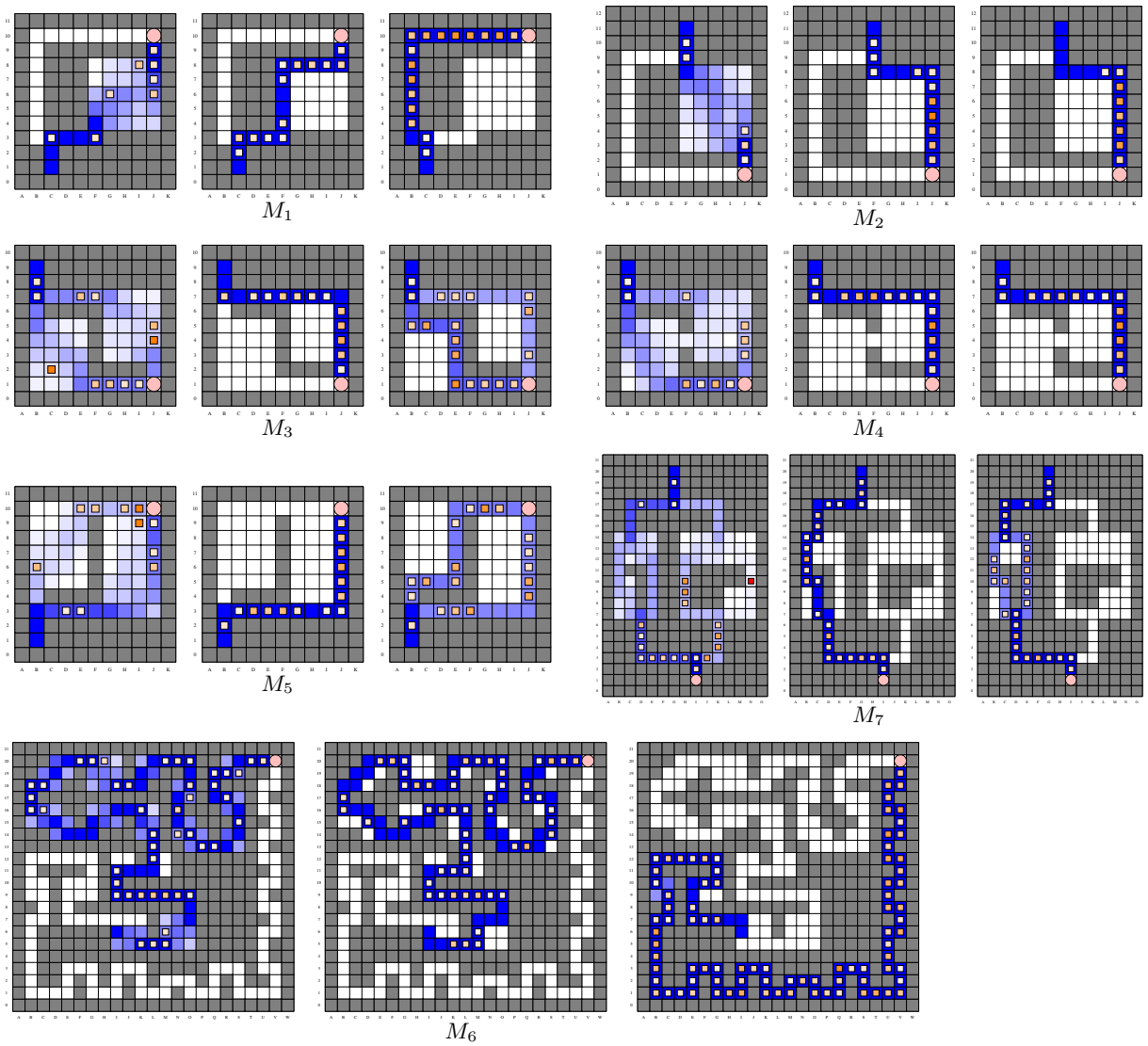


Figure 13: ["Reflex" Responses] Heatmaps showing, for each cell, (1) [in background] the probability of visit during a trajectory from dark blue ($P = 1$) to white ($P = 0$), and (2) [in the middle] the rate of < 150 ms response times from orange (max. rate: 50% or more) to white (min. rate: 0%). These heatmaps are provided for each maze and each policy, from left to right: stochastic MDP policy, biased MDP policy, and pOAMDP policy.

The results of the ranking of the three policies given at the end by the participants in terms of Anticipation and Logic are given by Table 2 (complete orderings). The two rankings show very similar patterns. For both Anticipation and Logic, as presented in Table 2 (score rankings), 1. $\pi_{\text{MDP-S}}$, which most participants consider hard to predict and even random, is typically ranked last, sometimes second, and 2. participants have a preference for $\pi^{\mathcal{A}}_{\text{pred}}$ over $\pi_{\text{MDP-B}}$.

Participants often declare that the initial choice of $\pi^{\mathcal{A}}_{\text{pred}}$ can be surprising. This is especially the case in maze M_6, and if the participants had worked with $\pi^{\mathcal{A}}_{\text{pred}}$ after $\pi_{\text{MDP-S}}$ and $\pi_{\text{MDP-B}}$. However, despite those statements, humans still performed better with $\pi^{\mathcal{A}}_{\text{pred}}$ (especially in maze M_6).

5.3 Discussion

There are few differences between the biased MDP policy and the pOAMDP policy in terms of errors, response times, and human perception, in comparison with the stochastic MDP policy, whose behavior is much harder to predict. The pOAMDP policies' response times are better on some mazes (M_1, M_4 and M_6). The pOAMDP policy induces less errors on more complex mazes (M_6 and M_7), while the biased MDP policy induces less errors on mazes M_3 and M_5, where the pOAMDP policy is not deterministic (having two possible trajectories). Subjective feedback from human participants is consistent with observed performance. The pOAMDP policy was preferred,

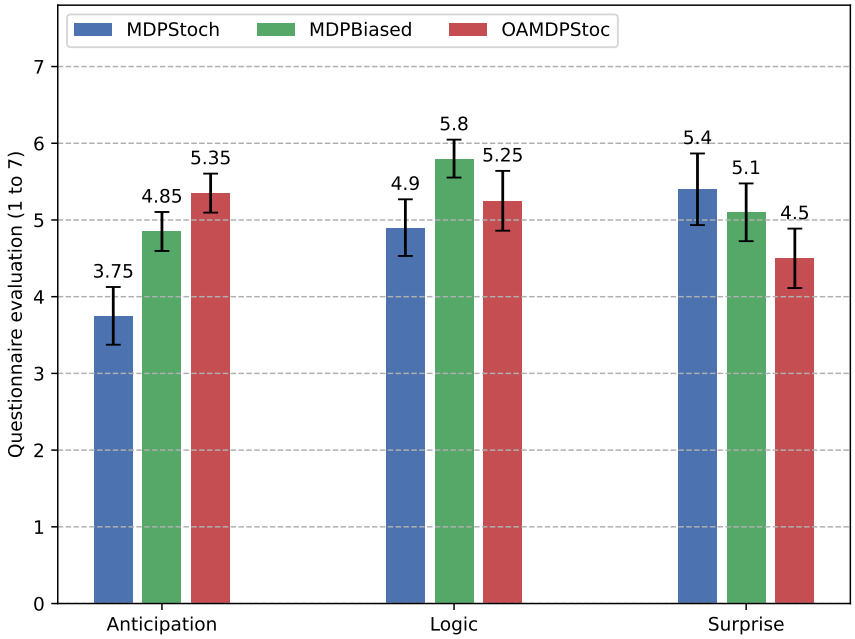


Figure 14: Graph showing questionnaire results (mean score and standard error) on a 7-point Lickert scale for each of the three policies and for the three dimensions assessed (Anticipation, Logic and Surprise).

Table 2: Human preferences over policies, where A=$\pi_{\text{MDP-S}}$, B=$\pi_{\text{MDP-B}}$, and C=π_{pred}

(a) Anticipation: Complete orderings

order	#votes
CBA	8
BCA	5
CAB	3
BAC	2
ABC	1

(b) Anticipation: Score ranking

	A	B	C
1st	1	7	11
2nd	5	9	5
3rd	14	3	3

(c) Logic: Complete orderings

order	#votes
CBA	8
BAC	4
CAB	3
BCA	3
ABC	2

(d) Logic: Score ranking

	A	B	C
1st	2	7	11
2nd	7	10	3
3rd	11	3	6

probably because it seems easier to anticipate, while the biased MDP policy is considered as slightly more logical (as it always follows shortest paths).

The pOAMDP policy is more efficient in complexe mazes such as M_6 which is interesting if we want to consider more realistic scenarios. However in less complex mazes, we do not observer significative differences between the pOAMDP policy and the biased policy. Considering more mazes such as M_6 in the experiment could improve the results and better emphasize the differences between the biased policy and the pOAMDP policy. The biased condition was added to this study to be able to compare the pOAMDP policy not only to a stochastic policy but also to a policy that could be easier to anticipate for the participants. We were expecting the differences between the pOAMDP policy and the biased policy to be less important than the differences between the stochastic MDP policy and the pOAMDP policy. However the bias used for the maze was learned really fast by the participants and facilitated their predictions to the point where in certain mazes, there was not any differences between the two.A better model of a human observer would thus account for the possible biases of the agent, *e.g.*, the bias being a hidden state variable (a *type* in the OAMDP sense) that the observer could try to infer. This being said, a scenario that could better match the human participants' model of the agent could be if the agent's state were described not only by its location (x, y), but also by a direction $d \in \{North, South, East, West\}$, and action set

$\mathcal{A} = \{forward, turnRight, turnLeft\}$, so that, without modifying the reward function, minimizing trajectory lengths will lead to prefer straight lines whenever possible.

6 Conclusion

We have introduced a new formalism, predictable observer-aware MDPs (pOAMDPs), that allows deriving policies whose next actions or next states are more predictable, and proposed accounting not only for discounted problems, but also for stochastic shortest-path problems (which requires ensuring that valid solution policies can be found). With the objective of minimizing the number of prediction errors along a trajectory in an undiscounted setting, and assuming rational observer predictions, we derived two reward functions, respectively for action and state predictability and demonstrated that they both induce valid stochastic shortest-path problems, *i.e.*, the solution predictable policies reach terminal states with probability 1. A notable property is that the solving complexity of pOAMDPs is comparable to MDPs, thus much less than OAMDPs. In some cases, the resulting policies select counter-intuitive actions early on to increase predictability later on. The interpretation of generated policies shows significant reductions in the expected number of errors when using pOAMDP solutions on some scenarios (up to fourfold), and also benefits in using biased MDP policies, which prefer following walls. Results of the experiment with human participants are consistent with these observations.

As illustrated by some benchmark problems, the proposed performance criterion can lead to poor policies in terms of the original performance criterion (here used only for the observer predictions). This can be addressed in various ways as, for instance, by linearly combining both reward functions, or, using *constrained MDPs* [1, 26], by minimizing the prediction error while constraining the value of the original criterion.

On another note, considering goal-oriented problems as we did would of course also be relevant for Miura and Zilberstein's OAMDPs, first to determine which of their scenarios result in valid SSPs. Then, to handle SSPs with traps, *i.e.*, subsets of (non-terminal) states that cannot be escaped, an interesting direction would be to extend our work to generalized SSPs [16, 25].

Finally, we had to depart from Miura and Zilberstein's original formalism and their static types [19], but an important perspective is to generalize both formalisms, making for a more unified theory of observer-aware sequential decision-making. We believe that a key point to achieve this is to restrict the observer's observability of states and actions so that the target variable, whether static or dynamic, can be a state variable, even for action predictability. What is more, this partial observability would also allow covering more real-world scenarios. In this setting, we envision looking at the continuity properties of the optimal value function to possibly propose bounding approximators and derive point-based solvers (as was done for POMDPs and related models [23, 24, 17, 20, 22, 6, 15, 14]).

References

[1] E. Altman, *Constrained Markov Decision Processes*, Chapman and Hall/CRC, 1999.

[2] C.L. Baker, R. Saxe and J.B. Tenenbaum, Action understanding as inverse planning, *Cognition* **113**(3) (2009), 329–349. doi:10.1016/j.cognition.2009.07.005.

[3] R. Bellman, A Markovian Decision Process, *Journal of Mathematics and Mechanics* **6**(5) (1957), 679–684.

[4] D.P. Bertsekas, *Dynamic Programming and Optimal Control*, Athena Scientific, 2005.

[5] T. Chakraborti, A. Kulkarni, S. Sreedharan, D.E. Smith and S. Kambhampati, Explicability? Legibility? Predictability? Transparency? Privacy? Security? The Emerging Landscape of Interpretable Agent Behavior, in: *Proceedings of the Twenty-Ninth International Conference on Automated Planning and Scheduling (ICAPS)*, AAAI Press, 2019. `https://ojs.aaai.org/index.php/ICAPS/article/view/3463`.

[6] J. Dibangoye, C. Amato, O. Buffet and F. Charpillet, Optimally Solving Dec-POMDPs as Continuous-State MDPs, *Journal of Artificial Intelligence Research* **55** (2016), 443–497. `http://www.jair.org/papers/paper4623.html`.

[7] A.D. Dragan, K.C.T. Lee and S.S. Srinivasa, Legibility and predictability of robot motion, in: *Proceedings of Eighth ACM/IEEE International Conference on Human-Robot Interaction*, 2013, pp. 301–308.

[8] J.F. Fisac, C. Liu, J.B. Hamrick, S. Sastry, J.K. Hedrick, T.L. Griffiths and A.D. Dragan, Generating plans that predict themselves, in: *Algorithmic Foundations of Robotics XII: Proceedings of the Twelfth Workshop on the Algorithmic Foundations of Robotics*, 2020.

[9] D. Fudenberg and J. Tirole, *Game Theory*, The MIT Press, 1991.

[10] E.A. Hansen, Error Bounds for Stochastic Shortest Path Problems, *Mathematical Methods of Operations Research* **86**(1) (2017), 1–27. doi:10.1007/s00186-017-0581-5.

[11] E.A. Hansen and S. Zilberstein, LAO*: A heuristic search algorithm that finds solutions with loops, *Artificial Intelligence* **129**(1–2) (2001), 35–62.

[12] J.C. Harsanyi, Games with Incomplete Information Played by "Bayesian" Players, I-III. Part I. The Basic Model, *Management Science* **14**(3) (1967), 159–182. http://www.jstor.org/stable/2628393.

[13] F.M. Henry and D.E. Rogers, Increased response latency for complicated movements and a "memory drum" theory of neuromotor reaction, *Research Quarterly. American Association for Health, Physical Education and Recreation* **31**(3) (1960), 448–458.

[14] K. Horák and B. Bošanský, Solving Partially Observable Stochastic Games with Public Observations, in: *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, 2019, pp. 2029–2036. doi:10.1609/aaai.v33i01.33012029.

[15] K. Horák, B. Bošanský and M. Pěchouček, Heuristic Search Value Iteration for One-Sided Partially Observable Stochastic Games, in: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 558–564.

[16] A. Kolobov, Mausam, D.S. Weld and H. Geffner, Heuristic Search for Generalized Stochastic Shortest Path MDPs, in: *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS'11)*, 2011.

[17] H. Kurniawati, D. Hsu and W.S. Lee, SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces, in: *Robotics: Science and Systems IV*, 2008.

[18] L. Marin and F.S. Danion, *Neurosciences comportementales: contrôle du mouvement et apprentissage moteur*, Editions Ellipses, 2019.

[19] S. Miura and S. Zilberstein, A unifying framework for observer-aware planning and its complexity, in: *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, Proceedings of Machine Learning Research, Vol. 161, PMLR, 2021, pp. 610–620. https://proceedings.mlr.press/v161/miura21a.html.

[20] J. Pineau, G. Gordon and S. Thrun, Anytime point-based approximations for large POMDPs, *Journal of Artificial Intelligence Research* **27** (2006), 335–380.

[21] B.R. Schadenberg, D. Reidsma, D.K.J. Heylen and V. Evers, "I See What You Did There": Understanding People's Social Perception of a Robot and Its Predictability, *J. Hum.-Robot Interact.* **10**(3) (2021). doi:10.1145/3461534.

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

[23] T. Smith and R.G. Simmons, Point-Based POMDP Algorithms: Improved Analysis and Implementation, in: *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, 2005, pp. 542–549.

[24] M.T.J. Spaan and N. Vlassis, Perseus: Randomized Point-based Value Iteration for POMDPs, *Journal of Artificial Intelligence Research* **24** (2005), 195–220. http://www.aaai.org/Papers/JAIR/Vol24/JAIR-2406.pdf.

[25] F.W. Trevizan, F. Teichteil-Königsbuch and S. Thiébaux, Efficient solutions for Stochastic Shortest Path Problems with Dead Ends, in: *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, (UAI-17)*, G. Elidan, K. Kersting and A. Ihler, eds, AUAI Press, 2017. http://auai.org/uai2017/proceedings/papers/280.pdf.

[26] F.W. Trevizan, S. Thiébaux, P.H. Santana and B.C. Williams, I-dual: Solving Constrained SSPs via Heuristic Search in the Dual Space, in: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, (IJCAI-17)*, C. Sierra, ed., ijcai.org, 2017, pp. 4954–4958. doi:10.24963/ijcai.2017/701.