Leveraging Counterfactual Paths for Contrastive Explanations of POMDP Policies

Benjamin Kraske[*]
Benjamin.Kraske@colorado.edu
University of Colorado Boulder
Boulder, CO, USA

Zakariya Laouar[*]
Zakariya.Laouar@colorado.edu
University of Colorado Boulder
Boulder, CO, USA

Zachary Sunberg
Zachary.Sunberg@colorado.edu
University of Colorado Boulder
Boulder, CO, USA

ABSTRACT

As humans come to rely on autonomous systems more, ensuring the transparency of such systems is important to their continued adoption. Explainable Artificial Intelligence (XAI) aims to reduce confusion and foster trust in systems by providing explanations of agent behavior. Partially observable Markov decision processes (POMDPs) provide a flexible framework capable of reasoning over transition and state uncertainty, while also being amenable to explanation. This work investigates the use of user-provided counterfactuals to generate contrastive explanations of POMDP policies. Feature expectations are used as a means of contrasting the performance of these policies. We demonstrate our approach in a Search and Rescue (SAR) setting. We analyze and discuss the associated challenges through two case studies.

CCS CONCEPTS

• **Human-centered computing → Collaborative interaction**; **HCI theory, concepts and models**.

KEYWORDS

Explainable Artificial Intelligence (XAI), Explainable Planning, Partially Observable Markov Decision Processes (POMDPs), Contrastive Explanations

ACM Reference Format:
Benjamin Kraske, Zakariya Laouar, and Zachary Sunberg. 2024. Leveraging Counterfactual Paths for Contrastive Explanations of POMDP Policies. In *HRI '24: Explainability for Human-Robot Collaboration (X-HRI) March 11, 2024 Boulder, CO, USA.* ACM, New York, NY, USA, 5 pages.

1 INTRODUCTION

As artificial intelligence is increasingly adopted in settings that involve human supervision, it is increasingly important that end users are able to understand the reasoning behind the decisions made by such systems. This is especially important when artificial intelligence is used in mission-critical roles, such as search and rescue [17]. The ability of an expert to ask questions and resolve confusion about the methods and results of such a system may make the difference between the adoption and appropriate trust of a system, and the mistrust and disuse of a system. Explainable Artificial Intelligence (XAI) seeks to enhance trust and enable transparency in these systems. Ideally, these autonomous systems should not only perform at a high level but also maintain sufficient transparency such that clear explanations of system behavior can be provided to end users.

The partially observable Markov decision process (POMDP) provides a flexible framework for reasoning over state and transition uncertainty. POMDPs have been applied to problems ranging from air collision avoidance [10] to cancer screening [3]. POMDPs are capable of capturing complex domains with millions of states [21] while accounting for uncertainty over these states. (PO)MDPs also lend themselves well to explanations, with more inherent transparency than black box methods. While much of XAI focuses on black box methods [16], explanations for model-based methods, specifically explainable planning, are increasingly a focus [5].

When interfacing autonomy with an end user, structured interaction can be useful to establish transparency in and trust of the system [14]. Specifically, contrastive explanations such as *"A performs better than B because C"* can be very useful in gaining the trust of end users [5]. *Counterfactuals*, or alternatives, (e.g., to executed actions or policies) provide insight into what could have happened under modified components of the system and can help make systems more interpretable [4]. However, the application of counterfactuals to different classes of problems is not always immediately intuitive. In this work, we tackle this challenge and explore the use of user-given counterfactuals to provide contrastive explanations for POMDP policies. We demonstrate the approach in the context of a Search and Rescue (SAR) POMDP example and discuss associated challenges.

We first provide a brief overview of explainable planning. We then propose a methodology for counterfactual path explanations for a SAR POMDP domain, and conclude with illustrative examples and a following discussion.

1.1 Explainable Planning

Planning explanations can be organized into model-based algorithm-agnostic and algorithm-specific explanations [5]. Model-based explanations assume that an algorithm has solved for an optimal policy, and that any user confusion would be as a result of mistranslations of the user's preferences and not a result of algorithm limitations or misunderstanding of algorithm reasoning. However, explaining the characteristics of the policy can give more insight into the quantitative reasoning of the algorithm and enhance the interpretability of the system. Interpretability amounts to understanding the outcome of an algorithm in terms of the quantitative flow of information [7]. Comprehensibility entails comprehending

[*]Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
HRI '24: X-HRI Workshop, March 11, 2024, Boulder, CO, USA
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

the outcome when explained using symbols such as environment landmarks and characteristics [7].

Many works have sought to increase the explainability of planning, particularly in the context of MDPs. The objective of MDPs is to generate a policy that maximizes the (discounted) expected reward. Offering contrastive explanations in terms of the expected reward may be interpretable to users but may not enhance comprehensibility. Instead, it can be intuitive to evaluate a policy with respect to the expected feature occupancy, where features symbolize abstract components of the reward function [8, 9]. This can enhance comprehensibility for the end user. Luebbers et al. [12] investigate the timing of contrastive justifications of paths generated by solving MPDs. Soni et al. [18] utilize counterfactual queries to first build user profiles and then provide tailored explanations but do not provide contrastive explanations after inferring a given user.

POMDPs introduce an added layer of complexity by reasoning over uncertainty in the state. The true state is hidden and is only partially observable. *Optimal* (i.e., with respect to reward) POMDP planning consists of branching on not only state transitions but also observations. This type of planning has the potential to seem unintuitive to an end user. As a result, many works have attempted to make POMDP reasoning more interpretable and comprehensible.

Several works explored interactions between the autonomy and the end user to enhance trust, yet do not provide explanations for agent behavior [2, 15].

Other works provide explanations in terms of the POMDP components. In [19, 20], the agent behavior is explained with respect to the POMDP's beliefs, rewards, transitions, and observations. However, the interaction between the autonomy and the user only occurs one way. The user is not able to provide counterfactuals to further understand the behavior.

To assess counterfactuals for a POMDP (e.g. a user-defined open-loop path), contrastive explanations need to present expectations with respect to state transitions, observations, and initial belief. Mazzi et al. [13] analyze traces of a POMCP tree to reason about safety-critical belief-dependent decisions. This was conducted by representing undesired actions using rule templates and shields in the POMCP tree to avoid actions that violate belief thresholds.

We aim to provide a straightforward method of explaining policies by contrasting them against user-proposed counterfactuals via feature expectations. The following section discusses our methodology for generating these explanations for a SAR POMDP.

2 METHODOLOGY

The goal of this work is to provide an intuitive means for the user to better understand why a given path is executed in the SAR POMDP domain. Here, we focus on addressing differences between user and algorithm reasoning. Consider the following motivating example:

Example 2.1 (Search and Rescue). Consider a search and rescue scenario with a human rescuer and an autonomous search agent (e.g. a UAV) collaborating to find a missing person. To accomplish this, the human expresses an objective for the agent in terms of regions of interest and the agent plans policies according to this objective and the primary objective of locating the missing person. The agent is also constrained by a limited battery life which demands the agent to return to home before depleting.

Our approach leverages the visual nature of this problem to acquire user feedback which informs explanations. We propose the following workflow for explanations of the POMDP policy given counterfactual user paths.

Given an executed POMDP policy, the user may have questions related to why the specific path was chosen, generally in contrast to another path. This provides an opportunity to explain the optimal POMDP policy in contrast to this alternative path. The user may express this counterfactual path by drawing it on a user interface or another means. This user path can then be translated into a sequence of actions, forming an open-loop policy, which is not dependent on observations or state transitions but is dependent on the horizon of the problem only. The performance of this policy is then compared to the optimal policy, forming the basis of an explanation. More specifically, an explanation describes why the optimal policy outperforms the user-proposed alternative and hence why the realized path differs from the user's expectations.

2.1 Leveraging Feature Expectations

A straightforward approach to policy explanation/justification is to convey that the optimal policy achieves the same or higher maximum expected reward than all other policies (that is, that the optimal policy is indeed optimal). In order to demonstrate that the policy is optimal, the expected reward of any proposed alternative policy could be compared against that of the optimal policy, demonstrating that the user could do no better than the optimal policy. However, while this style of explanation justifies the actions taken under the optimal policy, it does not provide any insight into why this policy accumulates the maximum expected reward.

To enhance comprehensibility, we leverage features and weights to represent the components that contribute to the reward, giving insight into which problem objectives the algorithm satisfies. This is an extension of explainable planning literature, in which feature expectations and factored rewards are used to provide explanations for MDPs [8, 9].

Choi and Kim [6] present feature expectations for POMDPs, building on Abbeel and Ng [1]. Let $\phi(s, a)$ be a feature occupancy function, where s and a are the state and action, respectively. This function returns a vector with entry i equal to 1 if a feature i is occupied, and 0 otherwise. Let α define a weighting for each feature such that the reward $R(s, a) = \alpha \cdot \phi(s, a)$. The feature expectation is defined as $\mu^{\pi}(b_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \phi\left(b_t, a_t\right) \mid \pi, b_0\right]$, where b is a belief distribution over states. Further, $\phi\left(b_t, a_t\right) = \sum_{s \in S} b(s_t)\phi(s, a_t)$, where S is the state space of the POMDP and $b(s)$ is the probability of state s in belief b. Then the value of a policy from some initial belief can be expressed as $V^{\pi}(b) = \alpha \cdot \mu(\pi)$ [6]. In this way, feature expectations provide a means of separating the valuing of certain outcomes from the frequency of these outcomes. We leverage feature expectations for POMDPs as a means of explanations in light of features (which could be thought of as multiple objectives) that compose a reward function. For the SAR domain, these features include locations of interest, battery depletion, and locating a target.

This approach gives users insight into both the frequency of visited features and the value assigned to them, such that the user can better understand the contribution of each to the expected reward. Such a method lends itself well to closed-loop user-feedback, in which a user can adjust the value assigned to different features.

2.2 Generating Explanations

Given a near-optimal policy, here calculated with SARSOP [11] (which we will consider consider optimal for the purposes of this work), a sequence of actions is executed following this policy, conditioned on observations, forming an apparent path. Given this path, a user may have questions as to why the path is optimal or an alternative path was not chosen. The user would then be prompted to provide an alternative path, represented as an open-loop action sequence in this work. Given the closed-loop optimal policy and the open-loop user policy, feature expectations can be calculated recursively using the Bellman expectation equation applied to the feature indicator function: $\mu^{\pi}(b_t) = \phi(b_t, \pi(b_t)) + \gamma\,\mathbb{E}[\mu^{\pi}(b_{t+1})]$. These feature expectations can then be translated into a plain language explanation, contrasting the outcomes expected under the two policies. The primary focus of this work is the use of feature expectations to summarize the performance of user counterfactual policies, with the specifics of translation left to future work.

It is worth noting that while this method benefits from domains in which users may readily provide counterfactuals such as the SAR POMDP domain, it can be applied to any domain in which a user provides a counterfactual policy, whether open-loop or closed-loop. This method only requires that feature expectations can be defined and calculated for a counterfactual policy and the policy requiring explanation.

3 CASE STUDIES

To demonstrate our approach, we formulate a SAR POMDP with a robot searching an *nxn* grid for a partially observable stationary target while also visiting regions of interest with a limited battery capacity. The location of the target is unknown to the robot and user and is only inferred through noisy observations. There is a uniform initial belief b over which cell the target occupies. The state space of this POMDP is made up of robot position s_{robot}, target position s_{target}, and remaining battery s_{batt}. Noisy observations of the target position are given if the robot is within one grid cell of the target. A perfect observation of the target location is provided if the robot is in the same cell as the target. Transitions are deterministic in position, with the robot moving in the direction indicated by any of the cardinal direction actions, while the battery deterministically decreases by 1 with each action. The problem terminates if the target is found (the robot and target share the same cell) or if the difference between the remaining battery and the battery required to return to the starting location is less than 1. A reward r_{target} is given for finding the target. Supplementary reward $r_{1:N}$ is given for visiting the N locations of interest $l_{1:N}$.

For this SAR POMDP, consider the following general features. Let $l_{1:N}$ denote cells of interest (which would be specified by a user in a collaborative SAR task as discussed in Example 2.1), then the feature indicator function is defined as follows:

$$\phi(s,a) = [x_1, ..., x_N, x_t, x_b] \tag{1}$$

where

$$x_i = \begin{cases} 1 \text{ if } s_{robot} = l_i \\ 0 \text{ o.w.} \end{cases} \quad \forall i \in [1...N],\; x_t = \begin{cases} 1 \text{ if } s_{robot} = s_{target} \\ 0 \text{ o.w.} \end{cases},$$

$$x_b = \begin{cases} 1 \text{ if } (batt\_to\_go - s_{batt}) \leq 1 \\ 0 \text{ o.w.} \end{cases},$$

and $batt\_to\_go$ is the battery required to return to the initial robot state from the current robot state. These features relate back to Example 2.1, representing user-specified objectives (cells of interest), a central objective (locating the target), and a constraint on the problem (preserving battery life such that the robot can return to base). Let the feature weighting be defined $\alpha = [r_1, ..., r_N, r_{target}, 0]$.

3.1 Case Study 1: Observable and Unobservable Objectives

The purpose of this example is to demonstrate contrastive explanations of paths in a context where there is one readily observable objective (a cell of interest) and one partially observable objective (the hidden target), the location of which is unknown initially.

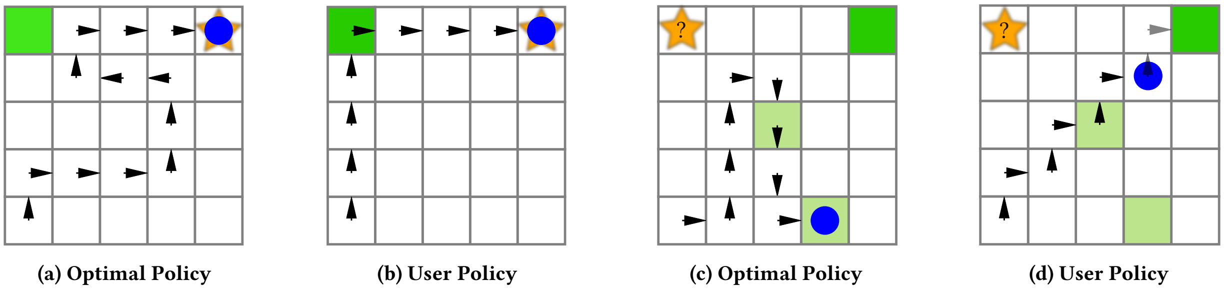
3.1.1 Model and Features. For this example, there is one cell of interest $l_1 = [1,5]$ with reward $r_1 = 3.0$ and a partially observable target located at $s_{target} = [5,5]$ with reward $r_{target} = 500.0$. The available battery is $s_{batt} = 25.0$.

3.1.2 Contrasting Path Outcomes. In this domain, the optimal policy executes the path shown in Fig. (1a) and finds the target, which is now visible to the user. With this hindsight knowledge of the location of the target, the user may wonder why the optimal policy did not simply go immediately up to collect the observable reward, and then to the target, as in Fig. (1b). In fact, for this particular simulation, the user policy (with hindsight knowledge of the target location) achieves greater discounted reward ($r^{\pi_{hu}} = 334.154$) than the optimal policy ($r^{\pi^*} = 270.180$). This further underscores the need for explanation. The feature expectations from the open-loop policy based on the suggested path π_{hu} and the optimal SARSOP policy π^* are shown in Table (1).

Table 1: Feature expectations for optimal and open-loop user policies for Case Study 1.

	l_1	$target$	$battery$
μ^{π^*}	0.036	**0.731**	0.0
$\mu^{\pi_{hu}}$	**0.684**	0.296	0.0

While in this case, where the target is in cell [5, 5], both policies find the target and the user policy achieves greater discounted reward than the optimal policy, the user policy does not account for the uncertainty in target location. In expectation over all possible target states (i.e. the initial belief), the closed-loop optimal policy outperforms the open-loop user policy in terms of the expected frequency of locating the target. This results in the optimal policy having a higher value than the user policy, as locating the target has a much higher weight. The open-loop user policy does outperform the optimal policy in terms of the frequency with which it reaches the cell of interest, but this feature has a much lower weight.


(a) Optimal Policy

(b) User Policy

(c) Optimal Policy

(d) User Policy

Figure 1: Case studies. (a), (b) An example in which the readily observable objective and the more valuable, partially observable, objective do not align. Note the target location (orange star) is unknown initially and only discovered by the robot (blue circle) after the optimal policy is executed. (c), (d) An example in which constraints restrict the feasibility of a proposed user policy. The *black* arrows represent the executed actions while the *gray* arrows represent the remaining actions of the user counterfactual path that were not executed due to the agent reaching a terminal battery state.

Because the target could be in any one of the cells and the problem terminates when the robot and target share a cell, there is some probability of terminating in every cell from the initial belief. This is likely why frequencies less than 1 are observed.

In order to produce an explanation, these feature occupancies must be translated into plain language, as in [9]. The translation approach itself is left to future work. A plausible explanation based on the feature expectations could be:

> *"Over all possible target locations, **the optimal policy finds the target about twice as often** as the user policy. The optimal policy will visit the cell of interest almost never. Since **the target has a much higher weighting than the cell of interest**, the optimal policy will outperform the user policy."*

3.2 Case Study 2: Resource Constraints

The purpose of this example is to demonstrate the effectiveness of contrastive feature expectation explanations as they relate to constraints on the problem, in this case the finite battery available in the SAR POMDP.

3.2.1 Model and Features. For this example, there are three cells of interest $l_1 = [5, 5]$, $l_2 = [4, 1]$, and $l_3 = [3, 3]$ with reward $r_1 = 3.0$, $r_2 = 1.0$, and $r_3 = 1.0$, respectively. A partially observable target is located at $s_{target} = [1, 5]$ with reward $r_{target} = 100.0$. The available battery is $s_{batt} = 12.0$.

3.2.2 Contrasting Path Outcomes. Given the path generated by the optimal policy (Fig. (1c)), the user may wonder why the path did not reach the higher-reward cell of interest in the upper right (l_1) and propose a path to that cell. However, the battery constraint does not allow for that cell to be reached and the shortened, feasible path shown Fig. (1d) is used as the basis for comparing outcomes. The corresponding feature expectations are shown in Table (2).

From these feature expectations, it is apparent that neither policy successfully reaches the higher-reward cell (l_1) and that the optimal policy is about twice as likely to locate the target when compared to the open-loop user policy. Likewise, the optimal policy avoids the battery terminal criteria more often. The optimal policy will visit the other cells of interest (l_2, l_3) in aggregate slightly less often than open-loop user path.

Table 2: Feature expectations for optimal and open-loop user policies for Case Study 2.

	l_1	l_2	l_3	*target*	*battery*
μ^{π^*}	0.0	**0.202**	0.354	**0.550**	**0.346**
$\mu^{\pi_{hu}}$	0.0	0.0	**0.684**	0.241	0.559

With these feature expectations and domain-knowledge about the limited battery, an explanation of the following form could be provided:

> *"The **battery constraint** makes it **impossible for either policy to reach** l_1. Over all possible locations of the target, **the optimal policy will find the target more often** leading to a higher reward (since the target is valued higher than any location of interest)."*

4 CONCLUSION AND FUTURE DIRECTION

In this work, we present an approach to explaining paths generated by optimal solutions to POMDP search and rescue problems. This initial approach takes a counterfactual user path as the basis for an open-loop policy which is contrasted against an optimal policy through the use of feature expectations.

While this work presents one means of providing contrastive explanations of optimal POMDP solutions, there are shortcomings to this approach. One substantial assumption is that the user maintains an open-loop policy that does not change with new information. However, accounting for policy changes due to new information likely better captures user reasoning. In particular, accounting for the influence of observations on user's reasoning will likely make for more effective POMDP explanations which better capture user behavior. Ideally, such methods could account for this closed-loop reasoning while still requiring limited user input.

Additionally, providing proactive explanations of executed paths which can be provided automatically in anticipation of user confusion will be valuable in reducing user workload. This would also reduce dependence on a domain-specific means of user feedback.

REFERENCES

[1] Pieter Abbeel and Andrew Y. Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the Twenty-First International Conference on Machine Learning* (Banff, Alberta, Canada) *(ICML '04)*. Association for Computing Machinery, New York, NY, USA, 1. https://doi.org/10.1145/1015330.1015430

[2] Dylan Asmar and Mykel J Kochenderfer. 2022. Collaborative Decision Making Using Action Suggestions. *Advances in Neural Information Processing Systems* 35 (2022), 33457–33468.

[3] Turgay Ayer, Oguzhan Alagoz, and Natasha K Stout. 2012. OR Forum—A POMDP approach to personalize mammography screening decisions. *Operations Research* 60, 5 (2012), 1019–1034.

[4] Ruth M. J. Byrne. 2019. Counterfactuals in Explainable Artificial Intelligence (XAI): Evidence from Human Reasoning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 6276–6282. https://doi.org/10.24963/ijcai.2019/876

[5] Tathagata Chakraborti, Sarath Sreedharan, and Subbarao Kambhampati. 2020. The Emerging Landscape of Explainable Automated Planning & Decision Making. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, Christian Bessiere (Ed.). International Joint Conferences on Artificial Intelligence Organization, 4803–4811. https://doi.org/10.24963/ijcai.2020/669 Survey track.

[6] Jaedeug Choi and Kee-Eung Kim. 2011. Inverse Reinforcement Learning in Partially Observable Environments. *Journal of Machine Learning Research* 12, 21 (2011), 691–730. http://jmlr.org/papers/v12/choi11a.html

[7] Derek Doran, Sarah Schulz, and Tarek R Besold. 2017. What does explainable AI really mean? A new conceptualization of perspectives. *arXiv preprint arXiv:1710.00794* (2017).

[8] Francisco Elizalde, Enrique Sucar, Julieta Noguez, and Alberto Reyes. 2009. Generating explanations based on Markov decision processes. In *MICAI 2009: Advances in Artificial Intelligence: 8th Mexican International Conference on Artificial Intelligence, Guanajuato, México, November 9-13, 2009. Proceedings 8*. Springer, 51–62.

[9] Omar Khan, Pascal Poupart, and James Black. 2009. Minimal sufficient explanations for factored markov decision processes. In *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 19. 194–200.

[10] Mykel J Kochenderfer, Jessica E Holland, and James P Chryssanthacopoulos. 2012. Next generation airborne collision avoidance system. *Lincoln Laboratory Journal* 19, 1 (2012), 17–33.

[11] Hanna Kurniawati, David Hsu, and Wee Sun Lee. 2008. SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *In Proc. Robotics: Science and Systems*.

[12] Matthew B Luebbers, Aaquib Tabrez, Kyler Ruvane, and Bradley Hayes. 2023. Autonomous Justification for Enabling Explainable Decision Support in Human-Robot Teaming. *Proceedings of Robotics: Science and Systems. Daegu, Republic of Korea. https://doi. org/10.15607/RSS* (2023).

[13] Giulio Mazzi, Alberto Castellini, and Alessandro Farinelli. 2023. Risk-aware shielding of Partially Observable Monte Carlo Planning policies. *Artificial Intelligence* 324 (2023), 103987. https://doi.org/10.1016/j.artint.2023.103987

[14] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267 (2019), 1–38. https://doi.org/10.1016/j.artint.2018.07.007

[15] Oriana Peltzer, Dylan M Asmar, Mac Schwager, and Mykel J Kochenderfer. 2023. Incorporating Human Path Preferences in Robot Navigation with Minimal Interventions. *arXiv preprint arXiv:2303.03530* (2023).

[16] Gabrielle Ras, Ning Xie, Marcel Van Gerven, and Derek Doran. 2022. Explainable deep learning: A field guide for the uninitiated. *Journal of Artificial Intelligence Research* 73 (2022), 329–396.

[17] Hunter M Ray, Zakariya Laouar, Zachary Sunberg, and Nisar Ahmed. 2023. Human-Centered Autonomy for Autonomous sUAS Target Searching. *arXiv preprint arXiv:2309.06395* (2023).

[18] Utkarsh Soni, Sarath Sreedharan, and Subbarao Kambhampati. 2021. Not all users are the same: Providing personalized explanations for sequential decision making problems. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 6240–6247.

[19] Ning Wang, David V. Pynadath, and Susan G. Hill. 2016. The Impact of POMDP-Generated Explanations on Trust and Performance in Human-Robot Teams. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems* (Singapore, Singapore) *(AAMAS '16)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 997–1005.

[20] Ning Wang, David V. Pynadath, Ericka Rovira, Michael J. Barnes, and Susan G. Hill. 2018. Is It My Looks? Or Something I Said? The Impact of Explanations, Embodiment, and Expectations on Trust and Performance in Human-Robot Teams. In *Persuasive Technology*, Jaap Ham, Evangelos Karapanos, Plinio P. Morita, and Catherine M. Burns (Eds.). Springer International Publishing, Cham, 56–69.

[21] Nan Ye, Adhiraj Somani, David Hsu, and Wee Sun Lee. 2017. DESPOT: Online POMDP planning with regularization. *Journal of Artificial Intelligence Research* 58 (2017), 231–266.

Received 6 March 2024