Proceedings of the 22nd International Conference on Principles of Knowledge Representation and Reasoning
KR in Planning and Scheduling

Probabilistic Active Goal Recognition

Chenyuan Zhang , **Cristian Rojas Cardenas** , **Hamid Rezatofighi** , **Mor Vered** , **Buser Say**
Monash University
{chenyuan.zhang, cristian.rojascardenas, hamid.rezatofighi, mor.vered, buser.say}@monash.edu

Abstract

In multi-agent environments, effective interaction hinges on understanding the beliefs and intentions of other agents. While prior work on goal recognition has largely treated the observer as a passive reasoner, Active Goal Recognition (AGR) focuses on strategically gathering information to reduce uncertainty. We adopt a probabilistic framework for AGR and propose an integrated solution that combines a joint belief update mechanism with a Monte Carlo Tree Search (MCTS) algorithm, allowing the observer to plan efficiently and infer the actor's hidden goal without requiring domain-specific knowledge. Through comprehensive empirical evaluation in a grid-based domain, we show that our joint belief update significantly outperforms passive goal recognition, and that our domain-independent MCTS performs comparably to our strong domain-specific greedy baseline. These results establish our solution as a practical and robust framework for goal inference, advancing the field toward more interactive and adaptive multi-agent systems.

1 Introduction

Imagine a robot assistant working alongside a human in a manufacturing environment. The human may be assembling one of several different products, each requiring a distinct assembly sequence. To provide effective assistance, the robot must infer the human's intended step as early as possible. This scenario highlights a broader challenge in human–robot collaboration: successful interaction often depends on accurately understanding the intentions and beliefs of other agents (Demiris, 2007; Dann et al., 2023).

While crucial for effective multi-agent interaction, the problem of recognizing other agents' goals and modeling their beliefs has received relatively limited attention in much of the multi-agent systems literature. Many studies in this area rely on model-free or learning-based approaches that do not explicitly reason about other agents (Zhang, Yang, and Başar, 2021; Canese et al., 2021), or defer such reasoning to large language models (Li et al., 2023; Shi et al., 2025). While these methods can perform well in reactive or end-to-end tasks, they often lack interpretability and struggle in situations that require anticipating others' intentions or long-term behavior. This limitation highlights the need for multi-agent frameworks that explicitly incorporate goal and belief modeling into decision-making—particularly in domains where understanding other agents is key to effective interaction.

In contrast, the goal recognition community typically employs symbolic models and planning-based approaches to reason about agent behavior (Masters and Sardina, 2019; Vered et al., 2018; Ramírez and Geffner, 2010). While these studies differentiate between a range of observed agents, be it unaware of being observed or even deceptive, they uniformly assume that the observer is a passive entity that cannot affect the environment and focuses strictly on inferring goals from a sequence of observations. However, in many real-world scenarios the observer is not merely a passive reasoner but an agent capable of taking actions to shape its own information state (Fitzpatrick et al., 2021). This aligns with the field of active information gathering, which studies how to select actions that reduce uncertainty (Shah, 2014; Veiga and Renoux, 2023).

By unifying passive goal recognition with active information gathering, Amato and Baisero (2019) introduced the problem of *Active Goal Recognition* (AGR). In their formulation, the observer is tasked with both recognizing the actor's goal and completing its own planning objective, requiring a balance between task execution and information gathering. Although they model the problem as a Partially Observable Markov Decision Process (POMDP), they manually design the reward function instead of deriving it from the formulation. Around the same time, Shvo and McIlraith (2020) also proposed an AGR formulation using the STRIPS-like language. Their approach leverages a landmark-based planning algorithm to actively collect observations for goal recognition. This method closely aligns with traditional goal recognition as planning approaches and does not incorporate a probabilistic formulation.

In this work, we focus on a setting where the observer's sole objective is to recognize the actor's hidden goal, without pursuing any independent task. To model this, we introduce a Probabilistic Active Goal Recognition (PAGR) framework based on a POMDP formulation. This framework leverages structured knowledge representation and belief-based rewards, enabling the observer to reason and act under uncertainty. Our formulation provides a unified probabilistic and decision-theoretic perspective to address a central question: how should an observer act in the environment to actively uncover the actor's goal?

2 Related Work

In this section, we provide an overview of the previous works that have modeled and solved related problems.

2.1 Goal Recognition

Two prevalent approaches of goal recognition are either using plan libraries or plan recognition as planning (PRP) (Meneguzzi and Fraga Pereira, 2021). Algorithms based on plan libraries, also known as plan recognition as parsing, present plans as a hierarchy of simpler actions. The main task becomes aligning the observed actions with these structured plans. Hierarchical Task Networks (HTNs) and grammars are typical methods for representing knowledge in plan libraries (Stuart and Norvig, 2016). HTN outlines tasks using a set of subtasks and their constraints, either separately or in relation to each other. Meanwhile, grammars describe the structure of plans through a set of production rules. These algorithms are useful in domains where the set of possible plans is known in advance, such as in video game AI and robotics (Van-Horenbeke and Peer, 2021).

On the other hand, PRP approaches use standard planning algorithms to create potential plan hypotheses for the observed agent (Ramírez and Geffner, 2010; Vered, Kaminka, and Biham, 2016; Vered and Kaminka, 2017; Zhang, Kemp, and Lipovetzky, 2023; Kaminka, Vered, and Agmon, 2018; Masters and Sardina, 2019). These planning algorithms are typically formulated using planning languages like STRIPS or PDDL, enabling them to outline the state of the environment and the impacts of applicable actions. In these approaches planners are used to calculate potential plans, as needed, that could achieve specified goals from varied initial states. The plan recognition system then assigns weights to these candidate plans by matching them against incoming observations, and the most likely plan or goal is chosen based on these weights.

In their survey, Masters and Vered (2021) identified the implicit and explicit assumption made by goal recognition researchers. A common limitation that emerges across all prior works is the implicit assumption that the observer is a static agent and unable to change its state to improve its goal recognition capability.

2.2 Active Goal Recognition

Active Goal Recognition, unlike standard goal recognition, involves an observer that can influence the observation process through its own actions. The observations of the observing agent depend on the actions it takes within its own domain, which may differ from the domain of the target. This allows the observer to strategically gather information so as to improve the accuracy and efficiency of goal inference (Shvo and McIlraith, 2020).

Shvo and McIlraith (2020) was the first to formalize the AGR problem and proposed a landmark-based approach for solving it. Their method performs hypothesis elimination within a partially observable planning framework grounded in STRIPS. Around the same time, Amato and Baisero (2019) introduced a more general framework based on POMDPs, enabling the handling of stochastic actions and transitions. Their solution relies on a linear approximation using the SARSOP solver and requires a manually designed reward structure, which depends heavily on domain knowledge. In a related line of work, Gall, Ruml, and Keren (2021) studied Goal Recognition Design (GRD) in an active setting, where the observer can interact with and modify the environment to induce the actor to reveal their true goal earlier. However, their formulation assumes full observability, and the observer's actions aim at breaking path symmetries rather than gathering information.

In this work, we propose Probabilistic Active Goal Recognition (PAGR) to address the limitations of prior approaches. Our formulation considers a partially observable setting where the observer must act to collect information and reduce uncertainty over the actor's goal, within a unified probabilistic and decision-theoretic framework.

2.3 Active Information Gathering and Online POMDP Solvers

To effectively perform AGR in partially observable environments, the observer must continuously update its beliefs and select informative actions in real time. This aligns with the broader literature on active information gathering and online POMDP solving. In this section, we review key approaches and solvers relevant to this approach.

Traditional information gathering has mostly been viewed as a means to an end in planning problems under uncertainty. In contrast, *active* information gathering refers to scenarios in which acquiring information about the environment or other agents is an integral part of the system's objective (Bajcsy, 1988). While standard reactive sensing relies on decisions driven by observed data, active information gathering, also known as active sensing, addresses this challenge by developing strategies that incorporate reasoning, decision-making, and control to maximize the value of the information collected (Veiga and Renoux, 2023).

These problems are usually modeled as POMDPs. Under the extension of uncertainty, the model is modified with a belief-based reward rather than a state-based one, resulting in what is known as a p-POMDP. However, p-POMDPs are computationally more expensive than normal POMDPs due to the introduction of the belief space (Araya et al., 2010).

Due to the intractability of solving POMDPs offline in complex domains, online POMDP solvers have been developed over the past two decades to enable scalable planning. A prominent example is POMCP (Silver and Veness, 2010), which applies Monte Carlo Tree Search (MCTS) to sample and evaluate future trajectories from the current state without modeling the belief space. Building on this approach, many variants have been proposed, differing in components such as backup strategies and sampling mechanisms (Sunberg and Kochenderfer, 2018; Thomas Vincent, Hutin Gérémy, and Buffet Olivier, 2020). To further improve scalability, especially in continuous state or action spaces, Sunberg and Kochenderfer (2018) introduced an extension of POMCP that supports planning in continuous domains, greatly expanding the applicability of online solvers to p-POMDPs. These advances provide a practical foundation for active decision-making in our setting.

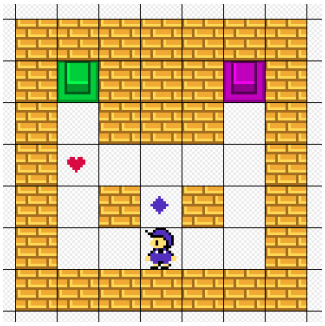

Figure 1: Illustrated example for Active Goal Recognition.

3 The Probabilistic Active Goal Recognition Problem

To motivate the problem, consider a simplified scenario where a single actor navigates an environment to reach one of two possible goals. An observer, aiming to identify the actor's true goal as efficiently as possible, is allowed to place a monitor in just one location. As shown in Figure 1, the two potential goals are marked by green and purple boxes. The observer knows the actor's starting position and that the true goal is one of the two possibilities, but receives no information about the actor's movements once the task begins except through the monitor, mimicking the limited observability available in real-world settings. The monitor provides a single observation by triggering if the actor passes through the chosen location. The challenge for the observer is to strategically place the monitor so as to maximize the chance of correctly inferring the actor's goal.

Let us examine two potential monitor placements, indicated by the red heart and blue diamond. Although the actor is very likely to pass through the blue diamond cell, this observation may offer little value: it lies along the optimal path to both potential goals, and thus provides minimal disambiguation. In contrast, placing the monitor on the red heart cell will yield a more informative observation. If the actor passes through it, this strongly indicates the green goal is the intended destination. Conversely, if the actor does not trigger the monitor at the red cell, this absence of evidence counts as negative evidence against the green goal, increasing the likelihood that the purple goal is the true target. This example shows the importance of selecting observer actions that maximize informativeness for goal disambiguation.

3.1 Formal Definition

We consider an environment shared by two agents: an *actor* and an *observer*. The actor is engaged in solving a planning problem to achieve a hidden goal, while the observer aims to infer this goal, as early as possible, while potentially interacting with the environment. We assume keyhole GR whereby the actor is unaware of, and unaffected by, being observed (Masters and Vered, 2021); that is, the actor's policy and state transitions are independent of the observer's state or actions. This assumption simplifies the formulation and is consistent with prior work in passive goal recognition. Extending the framework to model interactive actors is an important direction for future research, see Section 6.

We build on the Active Goal Recognition (AGR) formulation introduced by Amato and Baisero (2019), which allows for various types of observer actions, while the deterministic observation model is limited to specific types of observations that directly relate to the actor. In this work, we adapt and generalize their formulation to create a more concise and broadly applicable formulation. In particular, we redefine the observation function to depend jointly on the states of both the actor and the observer. This modification enables a more flexible and expressive model of perceptual uncertainty, accommodating a wider range of observation scenarios. We note that our formulation falls within the factored DEC-POMDP framework (Oliehoek, Amato, and others, 2016), which captures structured multi-agent decision making under partial observability.

In the environment, the actor solves the following planning problem $\mathcal{P}_{actor} = \langle s_0, \mathcal{E}_{actor}, g^* \rangle$, where the environment dynamics $\mathcal{E}_{actor} = \langle \mathcal{S}, \mathcal{A}^A, f^A \rangle$ are determined by the actor's state space $\mathcal{S}$, the initial state $s_0 \in \mathcal{S}$, the actor's action space $\mathcal{A}^A$, the transition function $f^A(s, a^A, s') = P(s' \mid s, a^A)$ and the goal state $g^* \in \mathcal{S}$. Given the actor planning problem $\mathcal{P}_{actor}$, the observer performs AGR $\mathcal{P}_{PAGR} = \langle \mathcal{E}_{actor}, \mathcal{U}, u_0, \mathcal{A}^O, f^O, \mathcal{O}, f_{\text{obs}}, \mathcal{G} \rangle$ where:

- $\mathcal{U}$ is the state space of the observer.
- $u_0 \in \mathcal{U}$ is the initial state of the observer.
- $\mathcal{A}^O$ is the observer's action space.
- $f^O(u, a^O, u') : \mathcal{U} \times \mathcal{A}^O \times \mathcal{U} \to [0, 1]$ be the observer's transition probability function, representing the probability of transitioning from state u to u' given an observer action a^O.
- $\mathcal{O}$ is the observation space.
- $f_{\text{obs}}(u, s, o) : \mathcal{U} \times \mathcal{S} \times \mathcal{O} \to [0, 1]$ denote the observation function, specifying the probability of observing $o \in \mathcal{O}$ given observer state u and actor state s. For simplicity, we assume that it depends only on the current states.
- $\mathcal{G} \subseteq \mathcal{S}$ is the candidate goal set such that $g^* \in \mathcal{G}$.

Here, the observation space $\mathcal{O}$ (e.g. detect, not detect in the motivated example) captures the fact that the observer never sees s directly but must infer it via these partial, state-dependent signals from the observation function f_{obs}. Unlike prior AGR formulations that treat sensing/observing as a distinct action class, our framework assumes a more general setting in which every observer action may influence the distribution of its subsequent observations. Note that while the observer has access to the actor's environment dynamics $\mathcal{E}_{actor}$, it does not know the actor's specific goal g^* or the initial state s_0. Meanwhile, it is assumed to know the set of candidate goals $\mathcal{G}$.

At each time step t, the actor is in state $s_t \in \mathcal{S}$. The actor's state evolves stochastically according to the transition dynamic $P(s_{t+1} \mid s_t, a_t^A)$, where actions $a_t^A \in \mathcal{A}^A$ are selected based on a (possibly stochastic) policy $\pi^A(a_t^A \mid s_t, g^*)$ that aims to achieve the goal g^*. This policy arises from solving a planning problem conditioned on the goal. Importantly, the observer does not have access to the actor's

policy π^A, and cannot observe the actor's states or actions directly.

The observer maintains its own observer state $u_t \in \mathcal{U}$, which evolves according to its own transition dynamic f^O. While the observer does not have access to the actor's internal state or policy, it receives a noisy observation $o_t \in \mathcal{O}$ that provides partial information about the actor's state. This process is characterized by the observation function f_{obs}.

For notational simplicity, we will use a_t to denote the observer's action at time t when no ambiguity arises. In *Probabilistic Active Goal Recognition (PAGR)*, the observer aims to infer the actor's hidden goal $g^* \in \mathcal{G}$ through a sequence of interleaved actions and observations, which involves two key components. First, at each time step t, the observer maintains a belief distribution over all candidate goals, denoted $b_t(g)$ for each $g \in \mathcal{G}$. Second, it must execute a behavior policy to gather informative observations that support belief update.

Quantifying the effectiveness of the observer's algorithm is non-trivial. In many applications, it is important not only to identify the correct goal but to do so both early and confidently. To formalize this, we adopt the notion of *convergence* (CV), following the formulation introduced by Vered et al. (2018), which captures both the timeliness and certainty of goal inference:

$$\text{CV}(g^*) = \begin{cases} \dfrac{T - \tau(g^*)}{T}, & \text{if } \exists\, \tau(g^*) \leq T \\ 0, & \text{otherwise} \end{cases}$$

where $\tau(g^*) = \min\{t : \forall t' \in [t, T],\ b_t(g^*) \geq \theta\}$, T is the total task horizon, and θ is a predefined threshold.

The objective of the PAGR problem is to find an observer policy $\pi^O(a_t \mid u_t, o_{0:t})$ that selects informative actions, along with a belief update mechanism that maintains $b_t(g)$ over time, in order to maximize the convergence $\text{CV}(g^*)$ with respect to the true goal in a stochastic environment.

4 Inference and Planning for Active Goal Recognition

To address the PAGR problem, we propose a framework that integrates probabilistic belief update with decision-theoretic planning. In this framework, the observer maintains a joint belief distribution over the actor's possible goals and states, and selects actions aimed at actively reducing this uncertainty. The belief is updated using a Bayesian inference mechanism informed by the sequence of observations obtained through interaction with the environment. To determine informative actions, we employ a Monte Carlo Tree Search (MCTS) algorithm tailored to the PAGR setting. This section introduces the joint belief update formulation and the MCTS algorithm that guides the observer's behavior.

4.1 Joint Belief Update

The observer maintains and updates a belief over both the actor's hidden goal g and its internal state s for each time step t. This is represented as a joint belief over the pair (s_t, g). Note that the update of this belief depends not only on the history of the observations $o_{0:t}$, but also on the observer's own trajectory $u_{0:t}$, since the observation function f_{obs} is conditioned on the observer's state. Intuitively, this reflects the fact that the informativeness of each observation depends on the observer's state, which determines its perspective and sensing capabilities. To formalize this, we define the joint belief distribution j_t at time t as:

$$j_t(s_t, g) = P(s_t, g \mid o_{0:t}, u_{0:t}), \tag{1}$$

which expresses the observer's probabilistic estimate of the actor being in state s_t and pursuing goal g, given the observation and observer state histories. This joint distribution forms a matrix over $\mathcal{S} \times \mathcal{G}$.

Next, we adopt a sequential update approach, where the belief j_t is recursively computed from j_{t-1}, using newly acquired information (o_t, u_t). To model the actor's behavior, we assume it follows a goal-directed policy. Under this assumption, the actor's state transition can be written as:

$$P(s_t \mid s_{t-1}, g) = \sum_{a^A} P(s_t \mid s_{t-1}, a^A)\, \hat{\pi}^A(a^A \mid s_{t-1}, g), \tag{2}$$

where $P(s_t \mid s_{t-1}, a^A)$ is the known environment dynamics f^A and $\hat{\pi}^A(a^A \mid s_{t-1}, g)$ is how observer models actor's goal-conditioned policy.

Using this, we first perform a prediction step:

$$\begin{aligned} &P(s_t, g \mid o_{0:t-1}, u_{0:t-1}) \\ &= \sum_{s_{t-1}} P(s_t, g, s_{t-1} \mid o_{0:t-1}, u_{0:t-1}) && (3) \\ &= \sum_{s_{t-1}} P(s_t \mid s_{t-1}, g) P(s_{t-1}, g \mid o_{0:t-1}, u_{0:t-1}) && (4) \\ &= \sum_{s_{t-1}} P(s_t \mid s_{t-1}, g) j_{t-1}(s_{t-1}, g). && (5) \end{aligned}$$

Equation 3 applies marginalization. Equation 4 uses local Markov property of s_t, which states that s_t is conditionally independent of past variables given s_{t-1} and g; Equation 5 is the definition of j_{t-1}.

Then, we perform the update step using the current observation (o_t, u_t) and the observation model $P(o_t \mid s_t, u_t)$:

$$\begin{aligned} j_t(s_t, g) &= P(s_t, g \mid o_{0:t}, u_{0:t}) && (6) \\ &= \frac{P(o_t \mid s_t, u_t) \cdot P(s_t, g \mid o_{0:t-1}, u_{0:t-1})}{\sum_{s'_t, g'} P(o_t \mid s'_t, u_t) \cdot P(s'_t, g' \mid o_{0:t-1}, u_{0:t-1})}. && (7) \end{aligned}$$

This equation applies Bayes' rule to incorporate the new observation o_t and observer state u_t. The numerator reflects the product of the observation likelihood $P(o_t \mid s_t, u_t)$ and the predictive belief $P(s_t, g \mid o_{0:t-1}, u_{0:t-1})$ obtained from the previous step. The denominator normalizes the distribution by summing over all possible state-goal pairs (s'_t, g'), ensuring that j_t is a valid probability distribution. This step also relies on the conditional independence assumption:

$$P(o_t \mid s_t, g, o_{0:t-1}, u_{0:t}) = P(o_t \mid s_t, u_t), \tag{8}$$

which states that given the current actor state s_t and observer state u_t, the observation o_t is conditionally independent of

the goal g, the observation history $o_{0:t-1}$, and the observer's previous trajectory $u_{0:t-1}$. This follows from the structure of the observation model and is a consequence of the local Markov property in the underlying graphical model.

Combining both steps yields the full recursive update:

$$j_t(s_t, g) = \frac{P(o_t \mid s_t, u_t) \sum_{s_{t-1}} P(s_t \mid s_{t-1}, g) j_{t-1}(s_{t-1}, g)}{\sum_{s'_t, g'} P(o_t \mid s'_t, u_t) \sum_{s'_{t-1}} P(s'_t \mid s'_{t-1}, g') j_{t-1}(s'_{t-1}, g')}. \tag{9}$$

We denote this update compactly as $j_t = h(j_{t-1}, u_t, o_t)$, indicating that the current joint belief is computed from the previous belief and the latest information.

Once the joint belief distribution $j_t(s_t, g)$ is obtained, the marginal belief over goals can be computed by summing over the actor's possible states:

$$b_t(g) = \sum_{s_t} j_t(s_t, g). \tag{10}$$

4.2 Belief-Guided Action Selection

With the joint belief j_t computed at each time step, the observer must choose an observer action $a_t \in \mathcal{A}$ that maximizes its ability to infer the actor's true goal. A natural reward signal $R(j_t)$ is the marginal belief $b_t(g^*)$, which quantifies the observer's confidence in the true goal g^* under the current belief j_t. However, this formulation is not directly usable for action selection, since the observer does not know which goal g is the true goal.

Instead, the observer can aim to maximize the expected confidence across all possible goals. Conditioning on the current joint belief j_t, the expected reward $R(j_t)$ becomes:

$$R(j_t) = \mathbb{E}_{g \sim P(g|j_t)}[b_t(g)] = \sum_g b_t(g) \cdot P(g \mid j_t) \tag{11}$$

$$= \sum_g b_t^2(g) = (\sum_{s_t} j_t(s_t, g))^2. \tag{12}$$

since $P(g \mid j_t) = b_t(g)$. This squared belief reward encourages the observer to take actions that sharpen the goal distribution—i.e., to reduce uncertainty and increase confidence in a particular goal.

Given the reward signal defined from the joint belief, the observer aims to select actions that maximize the discounted cumulative expected reward. Formally, for any policy π^O, we define the value function at time t as:

$$V_t^{\pi^O}(j_t, u_t) = \mathbb{E}^{\pi^O}\Big[\sum_{k \geq t} \gamma^{k-t} R(j_k) \mid j_t, u_t\Big], \tag{13}$$

where $\gamma \in [0, 1]$ is the discount factor, and the expectation is taken over the stochastic belief transitions and observer-state dynamics induced by π^O.

Correspondingly, we define the action-value function

$$Q_t^{\pi^O}(j_t, u_t, a_t) = \mathbb{E}^{\pi^O}\Big[\sum_{k > t} \gamma^{k-t} R(j_k) \mid j_t, u_t, a_t\Big], \tag{14}$$

which gives the expected return when the observer takes action a_t in (j_t, u_t) and thereafter follows policy π^O.

The optimal policy π^{O*} is defined as the policy that selects the action maximizing the expected future value at each time step, which satisfies the Bellman relations:

$$V_t^{\pi^{O^*}}(j_t, u_t) = R(j_t) + \max_{a_t} Q_t^{\pi^{O^*}}(j_t, u_t, a_t), \tag{15}$$

$$Q_t^{\pi^{O^*}}(j_t, u_t, a_t) = \mathbb{E}[\gamma V_{t+1}^{\pi^{O^*}}(j_{t+1}, u_{t+1}) | j_t, u_t, a_t]. \tag{16}$$

$$\pi^{O*}(a_t \mid j_t, u_t) = \arg\max_a Q_t^{\pi^{O^*}}(j_t, u_t, a). \tag{17}$$

To evaluate the Q-function in Equation 16, we decompose the expectation over the possible future observer states and observations that influence the update of the joint belief. This leads to the expansion of Equation 16:

$$\mathbb{E}\left[\gamma V_{t+1}^{\pi^{O^*}}(j_{t+1}, u_{t+1}) \,\middle|\, j_t, u_t, a_t\right] = \gamma \sum_{u_{t+1}} P(u_{t+1} \mid u_t, a_t) \sum_{o_{t+1}} P(o_{t+1} \mid u_{t+1}, j_t) \cdot V_{t+1}^{\pi^{O^*}}(h(j_t, u_{t+1}, o_{t+1}), u_{t+1}), \tag{18}$$

where $h(j_t, u_{t+1}, o_{t+1})$ is the joint belief update function specified previously in Equation 9. The observation likelihood under the belief j_t is computed as:

$$P(o_{t+1} \mid u_{t+1}, j_t) = \sum_{s_{t+1}} P(o_{t+1} \mid s_{t+1}, u_{t+1}) P(s_{t+1} \mid j_t), \tag{19}$$

$$P(s_{t+1} \mid j_t) = \sum_{s_t, g} P(s_{t+1} \mid s_t, g) j_t(s_t, g). \tag{20}$$

This formulation highlights how the observer policy π^O affects the future expected reward, that is by determining the action a_t, the observer controls the transition to the next state u_{t+1}, which affects the subsequent observation o_{t+1}, and thereby influences the updated belief j_{t+1}. Because the reward $R(j_t)$ depends on the confidence in the actor's goal encoded in the belief, the observer is incentivized to choose actions that lead to informative observations. This captures the essence of the PAGR problem where the observer is not just passively reacting to observations but actively selecting actions to accelerate goal inference by driving belief updates toward greater certainty.

4.3 Monte Carlo Tree Search for Active Goal Recognition

In the previous subsection we established a principled framework for observer action selection based on maximizing the expected cumulative reward under the joint belief. This formulation, grounded in the Bellman equations, captures how observer actions influence future beliefs and ultimately the confidence in goal inference. However, computing exact value or Q-functions becomes computationally infeasible in practice due to the high-dimensional belief space

and stochastic dynamics involved in belief updates. The exponential growth of possible observation-action trajectories renders exact planning intractable in realistic settings.

To overcome this challenge, we employ *Monte Carlo Tree Search* (MCTS), which is a sample-based online solver that approximates optimal actions via forward simulation. Rather than exhaustively evaluating all possible belief trajectories, MCTS incrementally constructs a search tree rooted at the current belief j_t and observer state u_t. Through repeated simulations, it estimates the value of different action branches by sampling possible observation outcomes and belief transitions. At each time step, MCTS selects an action according to the approximation $\hat{a}_t \approx \arg\max_{a_t} Q_t^{\pi^{O^*}}(j_t, u_t, a_t)$, where the Q-values are estimated from simulations.

We now describe our adaptation of MCTS for the PAGR problem, which is similar to PFT-DPW introduced in Sunberg and Kochenderfer (2018) but without double progressive widening. To model the uncertainty in observations, our MCTS tree alternates between *decision nodes* and *chance nodes*. The root node represents the current joint belief j_t and observer state u_t, and is a decision node where the observer selects an action a_t. This node approximates the value function $V_t^{\pi^{O^*}}(j_t, u_t)$, and each child node corresponds to a possible action a_t, forming a chance node. A chance node represents the Q-value $Q_t^{\pi^{O^*}}(j_t, u_t, a_t)$. At this node, we perform the belief prediction step (Equation 3-5) and simulate forward by sampling a goal $\hat{g}$, actor state s_{t+1}, observer state u_{t+1}, and observation o_{t+1} from the updated belief. Using these, we update the joint belief to obtain $j_{t+1} = f(j_t, u_{t+1}, o_{t+1})$ as shown in Equation 9. We use *lazy expansion* for chance nodes, meaning that new decision nodes are only generated when selected for the first time to avoid full enumeration of the large belief space. During the tree traversal, decision nodes use the UCB1 algorithm for action selection. Chance nodes select the child by sampling from current subjective belief maintained by the observer. In the backpropagation stage, we backup values by averaging the values of child nodes, which improves robustness under partial observability. At decision nodes, we also incorporate the immediate reward $R(j_t)$ into the backup. To reduce computational cost, we initialize newly expanded nodes with their immediate reward $R(j_{t+1})$ rather than performing random rollouts. By combining the belief update mechanism introduced in Subsection 4.1 with MCTS described in this subsection, we obtain a complete algorithm for solving the AGR problem, which we refer to as *AGR-MCTS*.

5 Active Goal Recognition in Grid World

We present a case study of the AGR problem in a classic two dimensional grid world domain, which is widely used in both goal recognition (Masters and Sardina, 2019) and active information gathering (Varotto, Cenedese, and Cavallaro, 2021). In this environment, the actor, observer, and goals are represented as discrete positions on a grid.

While our general framework described in previous sections supports stochastic transitions and observations, we adopt a simplified deterministic setting in this section to better illustrate the core ideas. Specifically, we assume deterministic transitions for both the actor and the observer, as well as deterministic observations. As a result, the transition models (Equations 2 and f^O) and the observation function f_{obs} are treated as Dirac distributions.

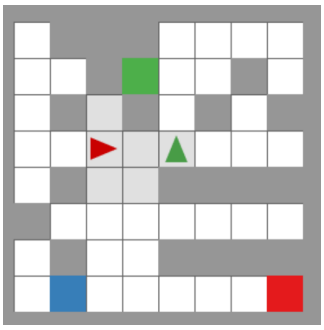

Figure 2: Illustrative experimental environment. The red cone represents the observer, and the green cone represents the actor, with each cone indicating their orientation. The light gray region denotes the observer's field of view (FoV). Colored squares mark candidate goal locations. For clarity, the grid size is scaled down to 8×8 with a FoV of 3×3.

The environment includes static obstacles that the actor cannot traverse. The actor starts from an initial location s_0 and aims to reach a fixed goal location g^*. The observer also starts from a random location u_0, but unlike the actor, it is allowed to traverse obstacles. Both agents share the same action space: move forward, turn left, turn right, and stay. The state of each agent is defined by its grid position and facing direction, which are the only relevant state variables in this domain. The observer also has a set of potential goals $\mathcal{G}$ so that $g^* \in \mathcal{G}$, which is consistent with prior work in goal recognition. An illustrative layout is shown in Figure 2.

The observer is equipped with a Field of View (FoV). In our experiments, the FoV is defined as a 5×5 grid extending in the observer's facing direction. The observer is positioned at the center of the back row of this grid, meaning the FoV extends two cells to the left and right and five cells forward relative to the observer's orientation. Cells in the FoV are excluded if a direct line of sight is blocked by an obstacle. The observation model is straightforward: the observer obtains the actor's position if it falls within the FoV; otherwise, it receives a "not observed" signal.

Most prior work in goal recognition assumes the actor behaves optimally, limiting applicability to realistic settings. In contrast, we introduce a more realistic and challenging scenario by modeling bounded rational behavior. Specifically, we generate a private cost map known only to the actor. The actor then computes an optimal path to its goal g^* under the private cost map, resulting in behavior that may appear suboptimal from the observer's perspective.

We implement this multi-agent environment using the MultiGrid Python library (Li et al., 2023).

5.1 Experiment Setup

We evaluate several algorithms on the previously described grid world domain to illustrate the inherent challenges of

AGR. To systematically vary the difficulty of the task, we manipulate two key factors: grid size and the initial distance between the actor and the observer. We consider two grid sizes: a small grid (10×10) and a large grid (20×20). For each grid size we test three levels of initial actor-observer distances: easy (3 cells), normal (5 cells), and hard (7 cells in small grid and 10 cells in large grid).

The observer's Field of View (FoV) is fixed at five cells. In the easy condition, the actor often falls within the observer's initial FoV, making it observable from the start. In contrast, under the hard condition, the actor starts far enough away that it is initially unobservable. These combinations yield a total of six unique environment configurations.

For each configuration, we generate ten random grid layouts, each with static obstacles. To simulate imperfect behavior and induce goal ambiguity, each layout includes a hidden cost map for the actor, encouraging suboptimal paths. For each layout, we then create five distinct task instances. In each instance, we randomly sample three goal candidates $\mathcal{G}$ and randomly select one as the actor's true goal g^*. The actor's starting position s_0 is randomly assigned. The observer's starting position u_0 is then placed at the specified distance from the actor according to the configuration, and its initial orientation is set to face the actor. Importantly, the observer has access only to the goal candidate set $\mathcal{G}$ and no other information about the actor's internal state or behavior. Each episode concludes once the actor reaches its goal.

Evaluation Metrics. As introduced in Section 3, we use *convergence* (CV) as the primary evaluation metric, with a threshold parameter set to $\theta = 0.5$. In addition, we report two auxiliary metrics: (1) *Final probability*: the average final belief probability assigned to the true goal $b_T(g^*)$, where T denotes the length of the episode (i.e., the number of steps the actor takes to reach the goal) and (2) *Success rate*: the percentage of instances in which the true goal g^* was correctly identified by the observer by the end of the episode (i.e. $b_T(g^*) > \theta$ was considered as correctly identified).

We compare our proposed method, *AGR-MCTS*, against several baseline approaches. Here, we introduce our own domain-specific *Belief-Greedy* algorithm, which uses joint belief updates but selects actions greedily. It is always moving toward the most likely actor position without considering long-term planning. A brief description of each approach is provided in Table 1. By default, our formulation uses the accumulated belief in the true goal as the primary reward signal, reflecting the confidence gained over time in the correct hypothesis. However, the active information gathering literature often employs entropy reduction as a reward signal to encourage uncertainty minimization (Veiga and Renoux, 2023). To assess the effectiveness of this alternative strategy, we extend *AGR-MCTS* by incorporating an entropy-based regularization term of actor state into the reward function. We set the number of MCTS iterations to 100 to balance computational efficiency and decision quality, and the algorithm assumes an ϵ-greedy actor model for belief update in Equation 2.

5.2 Passive Goal Recognition

For the passive goal recognition algorithm, we adopt the single-observation approach proposed by Masters and Sardina (2019), which has been shown to be effective in similar grid world domains. This approach is particularly well-suited to our setting, where the number of valid observations (i.e., known actor positions) may be very limited. Its ability to perform inference based on a single observation is especially advantageous under such constraints.

The original method assumes access to the actor's start position, which is not always available in our setting. To address this, we adapt the algorithm to a more general setting. Specifically, if no valid observation has been made, we default to the prior distribution over goals. Once a valid observation o is available, we compute an accumulated cost difference value for each goal g, denoted as $\text{cdiff}(o, g)$, and apply the probabilistic model from Ramírez and Geffner (2010):

$$P(g \mid o) = \alpha \cdot \frac{e^{-\beta \cdot \text{cdiff}(o,g)}}{1 + e^{-\beta \cdot \text{cdiff}(o,g)}}, \tag{21}$$

where α is a normalization constant and β is a scaling parameter.

In the formulation by Masters and Sardina (2019), this cost difference is computed using the final observation and a known start state. Instead, we extend this to operate incrementally by maintaining accumulated cost differences across observations. When a new valid observation o_t is made, we consider the last valid observation o', and update $\text{cdiff}(g)$ as follows:

$$\text{cdiff}(g) \leftarrow \text{cdiff}(g) + \text{optc}(o_t, g) + \text{step}(o', o_t) - \text{optc}(o', g), \tag{22}$$

where $\text{optc}(o, g)$ denotes the optimal cost from observation o to goal g, which can be precomputed, and $\text{step}(o', o_t)$ is the number of steps taken between the two observations, which is directly accessible.

This incremental formulation allows the method to accommodate suboptimal behavior and operate under partial observability, without requiring any additional online planning. Although this represents a novel adaptation of passive goal recognition for settings with substantial missing observations and online inference requirements, we present it here only briefly, as it is not the primary focus of this work.

5.3 Results

Table 2 presents a comparative evaluation of the four selected algorithms across the six unique environment configurations, combining variations in grid size (Small/Large) and initial distance conditions (Easy/Normal/Hard). As the three metrics exhibit similar performance trends across configurations, we focus on the Coverage (CV) metric in the following discussion for clarity and conciseness.

Among all algorithms inspected, *Belief-Greedy* consistently demonstrates strong performance across all configurations, achieving the highest scores under most conditions. *AGR-MCTS*, which incorporates a more complete planning-based approach, further improves performance in easier scenarios, where it outperforms all other methods.

Method	Goal Inference	Action Selection	Reward Design
Passive-Random	Passive Goal Recognition	Random	N/A
Search-and-Follow	Passive Goal Recognition	Actively searches for the actor, then follows its trajectory.	N/A
Belief-Greedy	Joint Belief Update	Greedy	Negative distance to most likely actor position.
AGR-MCTS	Joint Belief Update	MCTS	Accumulated Goal Probability plus actor state entropy

Table 1: Comparison of AGR-MCTS and some of ablation algorithms, in terms of goal inference, action selection, and reward design.

Algorithm	Metric	S-E	S-N	S-H	L-E	L-N	L-H
Passive-Random	CV	0.12	0.03	0.03	0.21	0.09	0.06
	SR	0.24	0.12	0.18	0.28	0.16	0.12
	FP	0.51	0.42	0.44	0.52	0.43	0.40
Search-and-Follow	CV	0.26	0.14	0.09	0.31	0.12	0.08
	SR	0.66	0.46	0.34	0.54	0.24	0.16
	FP	0.75	0.62	0.53	0.70	0.49	0.43
Belief-Greedy	CV	0.35	**0.27**	**0.24**	0.45	**0.31**	**0.22**
	SR	**0.82**	**0.76**	**0.68**	0.80	**0.70**	**0.60**
	FP	**0.87**	**0.82**	**0.78**	0.85	**0.79**	**0.69**
AGR-MCTS	CV	**0.39**	0.22	0.21	**0.51**	0.30	0.12
	SR	**0.82**	0.70	0.66	**0.86**	0.60	0.30
	FP	0.86	0.76	0.76	**0.90**	0.72	0.54

Table 2: Results across six configurations for each algorithm. Each algorithm is evaluated using three metrics: Convergence (CV), Success Rate (SR), and Final Probability (FB). The best performance for each metric in each configuration is highlighted in **bold**. S and L denote Small and Large grid sizes, respectively, while E, N, and H indicate Easy, Normal, and Hard initial distance conditions.

In contrast, the Passive-Random and Search-and-Follow baselines perform significantly worse, indicating that non-belief-driven strategies struggle under partial observability. Overall, the results demonstrate the advantage of combining belief-aware planning and informative rewards for AGR in partially observable environments.

Goal Inference Comparison To further evaluate the effectiveness of the proposed joint belief update mechanism, we conduct an ablation study comparing it with the passive goal recognition algorithm. Specifically, for the same sequence of observations collected under four different algorithms, we apply both the passive goal recognition and the joint belief update methods, and compare their performance, as illustrated in Figure 3.

Joint belief update outperforms the passive baseline across all configurations and action selection strategies. This improvement is primarily due to how each approach handles missing observations. In passive goal recognition, only positive observations (i.e., those where the actor is detected) are utilized for inference, while unobserved signals are treated as missing observation and thus provide no information. In contrast, the joint belief update algorithm leverages both observed and unobserved information. Returning to our motivating example, if the observer receives no signal when monitoring at the red marker, the joint belief update infers that the actor is likely pursuing an alternative goal, rather than treating this absence of evidence as uninformative.

6 Discussion

In this section, we discuss the insights from our experiment results and highlight the potential future directions in the field of AGR.

6.1 Goal Inference Mechanism

Our experimental evaluation confirms the superior performance of joint belief updating over passive goal recognition algorithms. Like we discussed in the previous section, the key advantage lies in the belief formulation's capacity to seamlessly integrate all available information. Although passive goal recognition methods could potentially exploit similar information by explicitly modeling missing observation effects, such an approach would incur exponential computational cost due to model size expansion. These results validate the theoretical advantages of belief-based goal inference approaches in POMDP settings.

6.2 Action Selection Strategy

While there is a clear advantage to the belief update and goal recognition mechanism, the complexity of designing effective online solvers in PAGR settings became apparent even

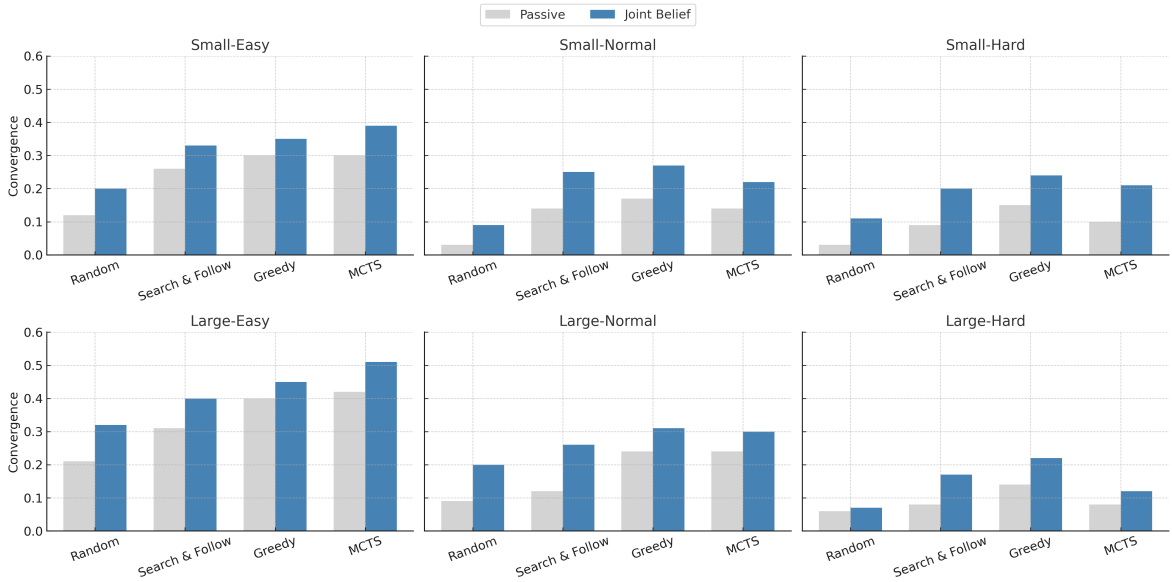

Figure 3: Comparison of CV between Joint Belief Update (active) and Passive Goal Recognition across different algorithms and scenario configurations. Each group contrasts the performance of the two inference methods under the same observation sequences.

within our constrained experimental domain. *Belief-Greedy* enhanced with domain-specific heuristics (i.e., distance to the most probable actor position) achieved strong performance through joint belief integration, but its reliance on domain knowledge limits its broader applicability. Conversely, our MCTS-based approach, though not consistently superior to the greedy algorithm, maintains complete domain independence, thereby offering greater generalizability across different domains.

For completeness, we also tested MCTS using the same heuristic-based reward signal, yet it still underperformed compared to the theoretically myopic greedy algorithm. One possible explanation is that MCTS fails to search deep enough to gather meaningful reward signals. In our domains, rewards are extremely sparse, because a strong signal only arises when the observer detects the actor. Indeed, we measured the average search depths of just four steps in large grids (and eight in small ones), explaining why MCTS only outperforms the greedy strategy in simpler scenarios where positive observations are more likely. These findings suggest enhancing online solvers to better exploit belief information for tree pruning and deeper lookahead (e.g., techniques like double progressive widening (Sunberg and Kochenderfer, 2018)) as an important avenue for future work.

6.3 Relaxing Keyhole Assumption

As mentioned previously, this work focuses exclusively on keyhole goal recognition (Masters and Vered, 2021), where the actor remains completely unaffected by the observer. While this assumption holds in certain scenarios, such as a high-altitude drone monitoring ground targets, it limits the applicability to other real-world situations. An important future direction is extending this framework to cases where the actor is aware of the observer, whether in collaborative settings (e.g., transparent planning as formulated in MacNally et al. (2018)) or adversarial contexts. This represents a crucial step toward more interactive multi-agent environments.

7 Conclusion

In this paper, we introduced Probabilistic Active Goal Recognition (PAGR), a novel probabilistic formulation of the Active Goal Recognition problem. Our main contributions are twofold: (1) a formal definition of PAGR grounded in the Partially Observable Markov Decision Process (POMDP) framework, which enables principled reasoning under uncertainty; and (2) an integrated solution framework that combines joint belief update with Monte Carlo Tree Search (MCTS) to efficiently solve the problem without relying on domain-specific knowledge.

We developed a comprehensive set of baselines to empirically evaluate the effectiveness of our approach. The joint belief update was shown to significantly outperform passive goal recognition methods by making more effective use of available information. Additionally, our domain-independent MCTS approach performed comparable to that of our strong domain-specific greedy algorithm, suggesting a promising direction for future work on developing more effective domain-independent methods for PAGR.

In summary, the presented contributions push the boundaries of AGR and offer a rigorous platform for continued exploration in multi-agent reasoning, goal inference, and decision-making under uncertainty.

Acknowledgments

This work is supported by the DARPA Assured Neuro Symbolic Learning and Reasoning (ANSR) program under award number FA8750-23-2-1016.

References

Amato, C., and Baisero, A. 2019. Active goal recognition. *arXiv preprint arXiv:1909.11173*.

Araya, M.; Buffet, O.; Thomas, V.; and Charpillet, F. 2010. A pomdp extension with belief-dependent rewards. In Lafferty, J.; Williams, C.; Shawe-Taylor, J.; Zemel, R.; and Culotta, A., eds., *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc.

Bajcsy, R. 1988. Active perception. *Proceedings of the IEEE* 76(8):966–1005.

Canese, L.; Cardarilli, G. C.; Di Nunzio, L.; Fazzolari, R.; Giardino, D.; Re, M.; and Spanò, S. 2021. Multi-agent reinforcement learning: A review of challenges and applications. *Applied Sciences* 11(11):4948.

Dann, M.; Yao, Y.; Alechina, N.; Logan, B.; Meneguzzi, F.; and Thangarajah, J. 2023. Multi-agent intention recognition and progression. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence, IJCAI 2023*, 91–99. IJCAI Organization.

Demiris, Y. 2007. Prediction of intent in robotics and multi-agent systems. *Cognitive processing* 8(3):151–158.

Fitzpatrick, G.; Lipovetzky, N.; Papasimeon, M.; Ramirez, M.; and Vered, M. 2021. Behaviour recognition with kinodynamic planning over continuous domains. *Frontiers in Artificial Intelligence* 4:717003.

Gall, K. C.; Ruml, W.; and Keren, S. 2021. Active goal recognition design. In Zhou, Z.-H., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, 4062–4068. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Kaminka, G.; Vered, M.; and Agmon, N. 2018. Plan recognition in continuous domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Li, H.; Chong, Y.; Stepputtis, S.; Campbell, J. P.; Hughes, D.; Lewis, C.; and Sycara, K. 2023. Theory of mind for multi-agent collaboration via large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 180–192.

MacNally, A. M.; Lipovetzky, N.; Ramirez, M.; and Pearce, A. R. 2018. Action selection for transparent planning. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 1327–1335.

Masters, P., and Sardina, S. 2019. Cost-based goal recognition in navigational domains. *Journal of Artificial Intelligence Research* 64:197–242.

Masters, P., and Vered, M. 2021. What's the context? implicit and explicit assumptions in model-based goal recognition. In Zhou, Z.-H., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, 4516–4523. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Meneguzzi, F., and Fraga Pereira, R. 2021. A Survey on Goal Recognition as Planning. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 4524–4532. Montreal, Canada: International Joint Conferences on Artificial Intelligence Organization.

Oliehoek, F. A.; Amato, C.; et al. 2016. *A concise introduction to decentralized POMDPs*, volume 1. Springer.

Ramírez, M., and Geffner, H. 2010. Probabilistic plan recognition using off-the-shelf classical planners. In *Proceedings of the AAAI conference on artificial intelligence*, volume 24, 1121–1126.

Shah, C. 2014. Collaborative information seeking. *Journal of the Association for Information Science and Technology* 65(2):215–236.

Shi, H.; Ye, S.; Fang, X.; Jin, C.; Isik, L.; Kuo, Y.-L.; and Shu, T. 2025. Muma-tom: Multi-modal multi-agent theory of mind. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 1510–1519.

Shvo, M., and McIlraith, S. A. 2020. Active goal recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 9957–9966.

Silver, D., and Veness, J. 2010. Monte-carlo planning in large pomdps. *Advances in neural information processing systems* 23.

Stuart, R., and Norvig, P. 2016. Artificial intelligence: a modern approach (global edition). *Harlow: Pearson*.

Sunberg, Z., and Kochenderfer, M. 2018. Online algorithms for pomdps with continuous state, action, and observation spaces. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 28, 259–263.

Thomas Vincent; Hutin Gérémy; and Buffet Olivier. 2020. Monte Carlo Information-Oriented Planning. In *Frontiers in Artificial Intelligence and Applications*. IOS Press.

Van-Horenbeke, F. A., and Peer, A. 2021. Activity, Plan, and Goal Recognition: A Review. *Frontiers in Robotics and AI* 8:643010.

Varotto, L.; Cenedese, A.; and Cavallaro, A. 2021. Active sensing for search and tracking: A review. *arXiv preprint arXiv:2112.02381*.

Veiga, T., and Renoux, J. 2023. From Reactive to Active Sensing: A Survey on Information Gathering in Decision-theoretic Planning. *ACM Computing Surveys* 55(13s):1–22.

Vered, M., and Kaminka, G. A. 2017. Heuristic online goal recognition in continuous domains. *arXiv preprint arXiv:1709.09839*.

Vered, M.; Pereira, R. F.; Kaminka, G.; and Meneguzzi, F. R. 2018. Towards online goal recognition combining goal mirroring and landmarks. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, 2018, Suécia.*

Vered, M.; Kaminka, G. A.; and Biham, S. 2016. Online goal recognition through mirroring: Humans and agents. In *Annual Conference on Advances in Cognitive Systems 2016*. Cognitive Systems Foundation.

Zhang, C.; Kemp, C.; and Lipovetzky, N. 2023. Goal recognition with timing information. In *Proceedings of the international conference on automated planning and scheduling*, volume 33, 443–451.

Zhang, K.; Yang, Z.; and Başar, T. 2021. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control* 321–384.