Helpful Information Shaping for Planning Agents

Anonymous Author(s)
Affiliation
Address
email

Abstract

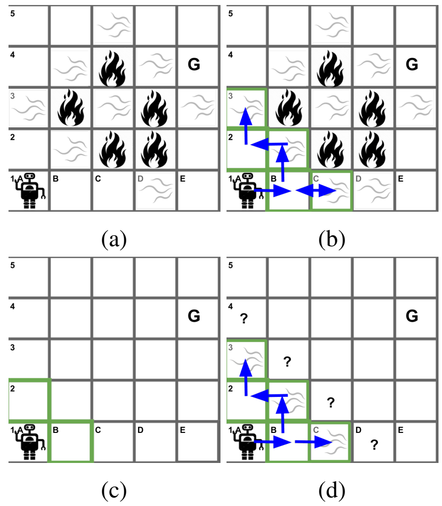
In many real-world applications, an agent may fail to reach its goal due to only partial information about its surroundings. At the same time, another *helper agent* may have information that can help with achieving this goal. We introduce *Helpful Information Shaping* (HIS), which is the problem of selecting which information to reveal to guarantee that a partially informed agent can achieve its goal. For a setting in which the partially informed actor is a planning agent, we provide a novel compilation of the HIS problem into a single-agent planning problem. The compiled problem encapsulates both the information shaping actions as well as inference about what will be possible for the agent to achieve on the basis of shared information. In this way, we achieve cooperation by solving this compilation-planning problem and can easily formulate different tradeoffs between cost of information and the cost of the resulting plan.

1 Introduction

In many applications, an agent attempting to reach a goal may have only partial information about its surroundings and limited sensors with which to acquire new information. The lack of information may cause the agent to fail to each its goal and even when the goal is achievable with more information. In a collaborative multi-agent setting, another agent, a *helper agent*, may possess information that can make it possible for the first agent to achieve its goal. The question is, given limited communication, what information should the helper agent share with the actor agent in order to help it achieve its goal? This is the problem of *Helpful Information Shaping* (HIS), which deals with determining the right information to reveal in order to ensure that the actor agent can succeed.

HIS is relevant to a variety of applications. One motivating setting is that of an under-water mission [6], where is a vessel on the surface has a noisy communication channel with an under-water autonomous vehicle (UAV). The controller needs to decide which information to share to help achieve the mission objective; e.g., revealing information about obstacles or passages that are invisible to the UAV. Assisted-cognition settings [10] provide another motivating scenario. For example, the goal of a visually impaired person may be known, say reaching the door to another room, and we need to share information to help with this goal. We would like to minimize the information shared, to avoid cognitive overload. In educational settings, a teacher might want to make sure a student receives just enough information to be able to solve a geometry problem, i.e., without providing the full solution but rather requiring the student to deduce some of what is needed to solve the problem.

In this paper we formalize a two agent HIS setting. One agent, the *actor*, is a partially informed planner that uses its partial information and available sensors to compute a plan to achieve its goal. The other agent, the *helper*, is assumed to have full information about the environment as well as knowledge about the actor and it's goal and relevant information about the kind of approach the agent takes to planning and acting. We consider two distinct kinds of actor models. One kind is a *conservative agent*, that only follows a plan if is guaranteed to achieve the goal. A second kind is a *replanning agent* that is willing to make assumptions about missing information in order move



(a) (b) (c) (d)

Figure 1: Helpful Information Shaping (HIS) environment map. The figures on the top row represent the actual environment state, with blue arrows representing the robot's behavior. Occupied cells in which a robot may get stuck are indicated by flames. Each occupied cell emits a danger signal in each adjacent cell, depicted as waves. The figures in the bottom row represent the robot's knowledge state that corresponds to each state of execution. The green cells are known to be 'free', while the question marks indicate possible locations of obstacles.

around in the environment and eventually reach its goal. This second kind of agent updates its beliefs in the event an assumption is refuted while acting and replans accordingly. The helper agent can have different kinds of objectives. The two we consider are (i) find the minimal amount of information to share so that the goal can be achieved, or (ii) find the minimal information to share so that the actor can follow a minimum cost plan to the goal.

As a running example, we consider a robot that is navigating in an environment in order to reach a destination. The robot maintains a map in which each cell is marked as 'free', in which case the cell is traversable, 'occupied', in which case the cell is non-traversable, or 'unknown', in which case the robot doesn't know whether or not the cell is traversable. Initially, only the walls are marked as 'occupied', while all other cells are marked as 'unknown'. The map is updated dynamically as the robot gains more information.

For simplicity, the world is static and the value of each cell remains the same throughout execution. The robot can move in one of the four cardinal directions, and is assumed to be able to localize itself and know which cell it occupies. It has a sensor that allows it to detect when an adjacent cell is occupied, but not the direction of the signal. If a robot tries to move into an occupied cell, it may get stuck. Therefore, if the sensor indicates the presence of a nearby obstacle while acting it will only moves to an adjacent cell if it can infer that this cell is free, and will backtrack otherwise.

Example 1.1 *Figure 1a depicts a simple HIS environment that consists of a single room with a single entry point marked by 'Start' and a goal destination* G*. The cells with obstacles (e.g., cells* $(B, 3)$*) are marked by flames that emit signals, depicted as waves (i.e. smoke) in the figure, that indicate an obstacle in one (or more) of the adjacent cells. Figure 1c depicts the robot's initial belief, that corresponds to the robot's initial knowledge about the environment. Since the robot does not sense a nearby obstacle it knows that the two adjacent cells are free (the green cells in the figure). If the robot is conservative, it will abort execution before moving, since it does not have information about a full path to the goal that is guaranteed to succeed. If the robot is a replanning actor, it will compute a plan based on assumptions that some cells are free. As shown in Figure 1b and the corresponding robot belief in Figure 1d, without additional information, the robot will abort execution of its task after examining all possible paths.*

In Figure 2 we show the minimal number of information shaping interventions that are needed to guarantee that a replanning agent can achieve its goal. In this case, an optimal solution is for a helper

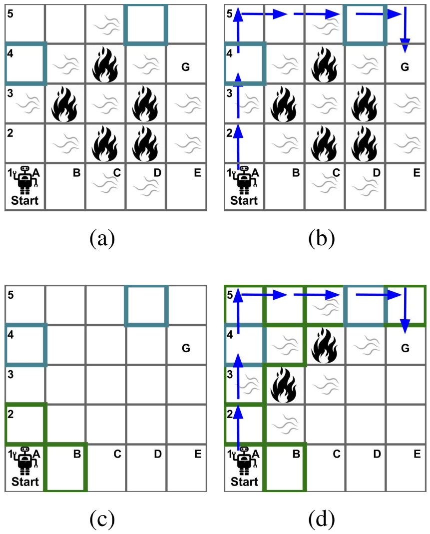


Figure 2: Using HIS to find a minimal information plan for a replaninng agent.

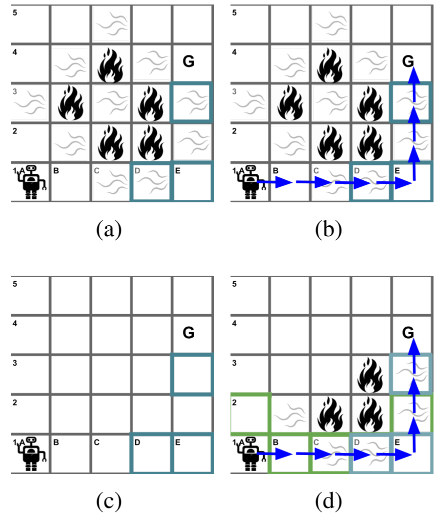


Figure 3: Using HIS to find a minimal execution plan for a replaninng agent

agent to reveal that cells $(A, 4)$ and $(D, 5)$ are free (the blue cells are those for which the true value is revealed). Figure 2c shows the robot's initial belief. The path that successfully leads the robot to it's goal after the additional information is provided is depicted in Figure 2b and Figure 2d shows the robot's path as well as the information about occupied cells that is acquired during execution.

Figures 3 depicts the solution that enables the actor to follow the cheapest path of going right from the initial state to the wall and then continuing up to the goal. Instead of revealing information about two cells, this solution would require revealing the value of three cells (i.e., $(D, 1)$, $(E, 1)$, $(E, 3)$). The path followed by the agent is depicted in blue in figures 3b and 3d. It bears emphasis that even if the information revealed enables a cost minimizing path, this does not by itself guarantee that the actor will follow that path. The decision of which path to follow depends on the planner used by the actor. The guarantee we seek is that if the agent can explore the domain, and can backtrack and replan, it will be able to eventually discover the cost minimizing path.

The problem of HIS is challenging because there may be a large number of possible choices in regard to which information to share, and because evaluating the effect of each information shaping option may be costly. To find optimal HIS solutions, we provide a compilation to a single planning problem that involves both information sharing actions as well as actions in the world and supports finding an optimal solution to HIS by a single call to an optimal, classical planner.

In the remainder of this paper, we provide an overview of related work (Section 2) as well as background on planning with partial knowledge (Section 3). We define *Helpful information shaping (HIS)* in Section 4. In Section 5 we present methods we have developed to find optimal information shaping solutions. We conclude in Section 6 with a summary of the HIS framework as well as possible directions for future investigation.

2 Related Work

The idea that information is malleable is not new, and has been investigated in a variety of research disciplines including economics [9], business management [7], politics [14], and more. We focus on controlling the information of a partially informed planning actor and the effect of additional information on the actor's behavior. Our work is therefore most related to the extensive body of work on selective information revelation in multi-agent settings.

One line of work considers the communication of information in collaborative settings. For example, [8] provide a general decision-theoretic mechanism for reasoning about the utility of communicating information relevant to an agent's plans in a multi-agent collaborative setting. Similarly, several lines of work [20, 19, 18, 17] consider a multi-agent collaborative setting where communication is limited, and the decision whether to communicate some information considers the gain to effective coordination, versus the communication cost. To address this challenge, a unified framework is created where communication becomes part of the overall agent decision problem. Specifically, [17] consider a distributed mixed human-computer team, where the value of of information held by a teammate is assessed by taking into account the costs of a potentially disruptive communication action and its effect on performance.

The information shaping approach we present here considers a setting where a passive observer can perform a one-time offline intervention, in which the same information is revealed to all actors in the system. The benefit of revealing a certain information item depends on the observer's objective, and is assessed according to a model that describes how actors reason and make decisions about how to act. Another distinction from previous works such as [17] is that we do not consider the effect of interrupting an agent in order to share information. Instead, our focus is on settings where there may not be even a direct interaction with the actors. Instead, information shaping can be used to represent either direct manipulations of actors' knowledge (e.g., manipulating the map used for navigation), or indirect placing of informative signals or sensors in the environment. Actors in our model are agnostic to the intentions of an observer, and may not even be aware of the offline information shaping process. A major challenge in previous work [17, 8] is that the value of information may be expensive to compute in general. The key challenge in our setting is due to the size of the information shaping task, requiring the use of efficient search techniques for finding optimal solutions.

Information shaping can also be viewed as a special case of *environment design* [22], which provides a framework for an interested party to seek an optimal way to modify an environment to maximize some utility. Among the many ways to instantiate the general model, *policy teaching* [23, 22] enables a system to modify the reward function of a stochastic agent, in order to entice the agent to follow specific policies. We focus here on performing design via controlling the information an agent uses for planning, rather than by reward shaping, and on using information shaping to ensure that a goal is achievable by an agent.

3 Background

3.1 Planning Under Partial Observability

To model the actors, we follow the approach by [3] for contingent planning under partial observability.

Definition 1 (Planning under partial observability (PPO)) *A* planning under partial observability (PPO) problem *is a tuple $\mathcal{P} = \langle \mathcal{F}, \mathcal{A}, I, G, \mathcal{O} \rangle$ where $\mathcal{F}$ is a set of fluent symbols, $\mathcal{A}$ is a set of deterministic actions, I is a set of* clauses *over $\mathcal{F}$-literals defining the initial situation, G is a set of $\mathcal{F}$-literals defining the goal condition, and $\mathcal{O}$ represents the agent sensor model.*

An action $a \in \mathcal{A}$ has a set $prec(a)$ of $\mathcal{F}$-literals preconditions, and a set $eff(a)$ of conditional effects $C \to L$, where C is a set of $\mathcal{F}$-literals and L is an $\mathcal{F}$-literal. The sensor model $\mathcal{O}$ is a set of observations $o \in \mathcal{O}$ represented as pairs (C, L), where C is a set of $\mathcal{F}$-literals, and L is a positive fluent indicating that the value of L is observable when C is true. Each observation $o = (C, L)$ can be conceived as a sensor on the value of L that is activated when C is true.

A state s is a truth valuation over the fluents $\mathcal{F}$ ('true' or 'false'). For an agent, the value of a fluent may be known or unknown. A fluent is *hidden* if its true value is unknown. A *belief* b is a non-empty collection of states that the agent deems as possible at some point. A formula $\mathbb{F}$ *holds* in b if it holds for every state $s \in b$. An action a is *applicable* in b if the preconditions of a hold in b, and the *successor* belief $b^{'}$ is the set of states that result from applying the action a to each state s in b. When an observation $o = (C, L)$ is activated, the successor belief is the set of states in b that agree on L (i.e., the set of states where fluent L has the sensed value). The initial belief is the set of states that satisfy I, and the goal belief are those that satisfy G. A formula is *invariant* if it is true in each possible initial state and remains true in any state reachable from it. A PPO problem is *simple* if the non-unary clauses in I are all invariant, and no hidden fluent appears in the body of a conditional effect. We hereon assume our PPO problems are simple.

A *history* is a sequence of actions and beliefs $h = b_0, a_0, b_1, a_1, \ldots, b_n, a_n, b_{n+1}$, such that action a_i is applicable in belief b_i. The cost of history h, denoted $\mathcal{C}_a(h) = \Sigma_i \mathcal{C}(a_i)$, is the accumulated cost of the performed actions (equivalent to the path length when action cost is uniform). A history is *complete* if the agent performing the actions reaches a goal belief. Each history corresponds to a path $path = a_0, a_1, \ldots, a_n$, which is the sequence of actions performed by the agent. A solution to a PPO problem is a *policy* π, which is a partial function from beliefs to actions.

A PPO problem corresponding to the navigating robot example in Section 1 includes fluents representing whether a cell is free (e.g., $IsFree(D, 1)$), whether the agent is in a location (e.g., $AgentAtCell(B, 1)$), and whether there is a danger signal in a location (e.g. $SignalAtCell(B, 2)$). Actions represent movements between adjacent cells (e.g. $Move(B, 1, C, 1)$). The goal is for the agent to be in a particular state (e.g., $AgentAtCell(E, 4)$). The sensor model includes observations that specify the robot's ability to sense a signal in a particular location. For example, the observation $o = (C = AgentAtCell(B, 1), L = Signal(B, 1))$ indicates the robot can sense a signal in cell $(B, 1)$ if it is in the cell. Finally, the initial situation includes all fluents that are true at the beginning of execution (e.g., $AgentAtCell(A, 1)$), as well as all the invariant information about the environment. For example, $IsFree(D, 1) \vee SignalAtCell(C, 1)$ indicates the danger signal that can be sensed in cell $(C, 1)$ if cell $(D, 1)$ is occupied.

There are two main approaches to planning with partial information: offline planning and online planning [4]. In offline planning, a complete plan tree is generated to account for all the contingencies that may arise [13]. This plan tree may grow exponentially in the number of problem variables, making it an impractical approach in all but simple problems. In online planning, the agent makes local decisions on how to behave next, which can typically be generated much more quickly, but might not provide the same guarantees as the offline approach. A variety of offline and online approaches have been developed for PPO planning [2, 16, 3, 4, 13, 5, 1, 12, 15]. A common technique for online planning is *replanning* [21], where an agent finds a plan for its current state based on some simplification of its planning problem and executes a prefix of the plan until discrepancies between the plan and the information acquired during the execution emerge and require replanning.

4 Helpful Information Shaping

Helpful information shaping (HIS) focuses on finding information to reveal to a partially informed actor so it can achieve its goal. We assume that communication is limited, and that an actor computes its own plan by considering all the information it has available and the assumptions it can make about missing information.

Definition 2 *A* **helpful information shaping** *(HIS) model is a tuple $M = \langle R_0, \mathbf{\Delta} \rangle$ where:*

- $R_0 = \langle \mathcal{F}, \mathcal{A}, I, G, \mathcal{O} \rangle$ *is the initial model, and*
- *Set $\mathbf{\Delta}$ denotes the set of information items that the helper can communicate to the actor.*

We are assuming the helper agent has full knowledge of the the actual world state defined as a truth valuation over the fluents $\mathcal{F}$. This means we have one intervention $\delta_f \in \mathbf{\Delta}$ that corresponds to revealing the true value of each fluent $f \in \mathcal{F}$. Each intervention may be associated with a *information cost* $\mathcal{C}_{is} : \mathcal{M} \rightarrow \mathbb{R}^+$. We overload the notation for a single intervention by using $\mathcal{C}_{is}(\Delta) = \sum_{\delta \in \Delta} \mathcal{C}_{is}(\delta)$ to denote the total cost of a set Δ of interventions. Another cost we consider is the cost of executing a plan. We let $\mathcal{C}^{min}_{exe}(R)$ represent the minimal cost of a plan to the goal in R and $\mathcal{C}^*_{exe}(R)$ is the minimal cost under full information. Finally, we let R_0^{Δ} denote the model that results from applying the set $\Delta \subseteq \mathbf{\Delta}$ to the initial model R_0.

We consider two objectives. The first objective, demonstrated in Figure 2 and formally defined in Equation 1, is to the set of interventions with minimal information cost needed to guarantee that the goal is achievable:

$$\begin{aligned} \Delta^*(M) &= \arg \min_{\Delta \subseteq \mathbf{\Delta}} (\mathcal{C}_{is}(\Delta)) \\ \text{s.t. } & \mathcal{C}^{min}_{exe}(R_0^{\Delta}) < \infty \end{aligned} \tag{1}$$

The second objective, demonstrated in Figure 3 and formally defined in Equation 2, is to find the minimal set of interventions that guarantee that a plan with minimal cost under full information is executable by the partially informed actor:

$$\Delta^*(M) = \arg\min_{\Delta \subseteq \mathbf{\Delta}} (\mathcal{C}_{is}(\Delta))$$
$$\textbf{s.t. } \mathcal{C}^{min}_{exe}(R^{\Delta}_0) = \mathcal{C}^*_{exe}(R_0) \tag{2}$$

The above objectives may be relevant to different settings. In applications in which communication is costly, such as the under water mission setting described in Section 1, the helper's objective may be described by Equation 1. In applications in which physical movements are expensive, Equation 2 may be more suitable.

5 Finding Optimal Solutions for HIS

A baseline approach for HIS is to perform a breadth first search (BFS) in the space of sensor extensions, computing the actor's cost to goal at each node (in the case of non-uniform information cost, a Dijkstra search is applied instead). The search explores modification sets of increasing size, using a closed-list to avoid the computation of pre-computed sets. To find a solution that complies with Equation 1 (a solition that minimizes information cost), the search halts if a solution is found, or if there are no more nodes to explore, and returns the shortest path (smallest sensor extension set) to a node that achieves the optimal value. For solutions that comply with Equation 2 (solutions that guarantee a minimum cost plan is executable), the search needs to continue until all possible information shaping options are explored. This approach is guaranteed to find an optimal solution, but does not scale to large problems. We suggest instead a compilation to classical planning that allows finding a solution to a HIS problem with a single call to an off-the-shelf planner.

5.1 Knowledge Acquisition Compilation (KAC)

To generate a plan for a partially-informed agent, [3] suggest the $K(\mathcal{P})$ translation that transforms the PPO problem into a classical planning problem. This substitutes each literal L in the original problem with a pair of fluents KL and $K\neg L$, representing whether L is known to be true or false, respectively [1, 16]. Each original action $a \in \mathcal{A}$ is transformed into an equivalent *execution action* that replaces the use of every literal L ($\neg L$), with its corresponding fluent KL ($K\neg L$). Each observation $o = (C, L)$ is translated into two deterministic *assumption actions*. These allow the solver to compute a plan while choosing to make assumptions about unknown variables. For example, the actor can assume that a cell is free. Each invariant clause is translated into a set of *ramification actions*. These actions can be used to set the truth value of some variable as new sensing information is collected. For example, a ramification action can be activated to infer that a cell is safe when no obstacle signal is sensed in an adjacent cell. This representation captures the underlying planning problem at the knowledge level, accounting for the exploratory behavior of a partially informed agent. A plan can be found using any off-the-shelf classical planner.

We suggest the KAC translation that takes as input the actor's PPO description, and the informer's possible information shaping options and creates a single planning problem. Like Bonet and Geffner's $K(\mathcal{P})$ translation [3], the translated problem includes execution actions $\mathcal{A}^{'}_{exe}$, that represent actual behavior in the environment, $\mathcal{A}^{'}_{as}$ assumption actions, that allow the actor's planner to choose the value of variables, and ramification actions $\mathcal{A}^{'}_{ram}$, that correspond to the invariant information that allows the actor to infer new information. The novelty here is that KAC includes *knowledge acquisition actions* $\mathcal{A}^{'}_{ka}$, modeled as part of the planning problem and that represent the sharing of information by the helper (via an 'acquisition' of the actor). While the $K(\mathcal{P})$ translation associates the same cost to all actions in the compiled problem, KAC includes a cost function, associating a cost for each type of action and giving flexibility to the formulation.

Definition 3 (**KAC Translation**) *Given a HIS problem $M = \langle R_0, \mathbf{\Delta} \rangle$ and a cost function $\mathcal{C} \in \Re^+ \cup \infty$, $KAC(M) = \langle \mathcal{F}', I', G', \mathcal{A}', \mathcal{C}' \rangle$ is the fully observable problem where*

- $\mathcal{F}' = \{KL, K\neg L : L \in \mathcal{F}\}$
- $I' = \{KL : L \in I\}$

- $G' = \{KL : L \in G\}$
- $\mathcal{A}' = \{\mathcal{A}^{'}_{exe} \cup \mathcal{A}^{'}_{as} \cup \mathcal{A}^{'}_{ram} \cup \mathcal{A}^{'}_{ka}\}$ *where*
  - $\mathcal{A}^{'}_{exe}$ *includes all actions $a \in \mathcal{A}$, but with each precondition L replaced by KL, and each conditional effect $C \rightarrow L$ replaced by $KC \rightarrow KL$ and $\neg K \neg C \rightarrow \neg K \neg L$.*
  - $\mathcal{A}^{'}_{as} = \{a_{(C,L)}, a_{(C,\neg L)} | o = (C, L) \in \mathcal{O}\}$ *where*
    - $prec(a_{(C,L)}) = \{KC, \neg KL, \neg K\neg L, L\}$ *and* $eff(a_{(C,L)}) = \{KL\})$
    - $prec(a_{(C,\neg L)}) = \{KC, \neg KL, \neg K\neg L, \neg L\}$ *and* $eff(a_{(C,\neg L)}) = \{K\neg L\}$
  - $\mathcal{A}^{'}_{ram} = \{a_{ram} |$ *for invariants* $\neg C \vee L$ *in* $I\}$ *where*
    - $prec(a_{ram}) = \{KC\}$ *and*
    - $eff(a_{ram}) = \{KL\}$
  - $\mathcal{A}^{'}_{ka} = \{\mathcal{A}^{'+}_{ka} \cup \mathcal{A}^{'-}_{ka}\}$ *where*
    - $\mathcal{A}^{'+}_{ka} = \{a_{(C,L)} | o = (C, L) \in \mathcal{O}^o\}$ *where*
      - $prec(a_{(C,L)}) = \{KC, \neg KL, \neg K\neg L\}$ *and* $eff(a_{(C,L)}) = \{KL\})$
    - $\mathcal{A}^{'-}_{ka} = \{a_{(C,\neg L)} | o = (C, L) \in \mathcal{O}^o\}$ *where*
      - $prec(a_{(C,\neg L)}) = \{KC, \neg KL, \neg K\neg L\}$ *and* $eff(a_{(C,\neg L)}) = \{K\neg L\}$
- $\mathcal{C}'(a) = \begin{cases} \mathcal{C}^{'}_{ex} & \textit{if } a \in \mathcal{A}^{'}_{exe} \\ \mathcal{C}^{'}_{as} & \textit{if } a \in \mathcal{A}^{'}_{as} \\ \mathcal{C}^{'}_{ram} & \textit{if } a \in \mathcal{A}^{'}_{ram} \\ \mathcal{C}^{'}_{ka} & \textit{if } a \in \mathcal{A}^{'}_{ka} \end{cases}$

Given a plan π that is a solution to the KAC translation, we let $\Pi^{'}_{exe}(\pi)$, $AS(\pi)$ and $\Pi^{'}_{ka}(\pi)$ represent the sequence of execution actions, assumption actions and knowledge acquisition actions in π, respectively. We use $|\Pi^{'}_{exe}(\pi)|$, $|\Pi^{'}_{ka}(\pi)|$ and $|AS(\pi)|$ to represent the respective sizes of these sequences. The cost function $\mathcal{C}'$ dictates the plan that is a solving to the translation.

In [11] the cost of assumptions is used to guarantee a level of *robustness* of the generated plan, representing the plan's ability to avoid failure. We use the cost function to support solutions that minimize either the execution cost or information cost. In the following, we show how these two objectives can be supported for conservative and replanning agents.

5.2 Finding HIS Solutions for Conservative Agents

Conservative agents only follow plans that are guaranteed to succeed and do not rely on any assumptions. To support such agents, we disable the planner's ability to make assumptions (which is equivalent to setting the cost of assumptions $\mathcal{C}^{'}_{as}$ to be infinite). Execution actions are assumed to have a uniform cost $\mathcal{C}^{'}_{ex}$ of 1 for all actions. Under the assumption that reasoning is done with no computational overheard, we assign zero cost to ramification actions $\mathcal{A}^{'}_{ram}$, (i.e., $\mathcal{C}^{'}_{ram} = 0$). The cost of knowledge acquisition actions $\mathcal{C}^{'}_{ka}$ is set according to the helper's objective.

To comply with the objective of finding a minimal set of information items to reveal to guarantee the goal is achievable (Equation 1), we set the cost to be high enough so that a single knowledge acquisition action is higher than the cost of all other actions the actor can perform. This will guarantee such actions are included in a solution only if they are necessary to accomplish the task. Specifically, if $\mathcal{C}^{'}_{ka} > |\mathcal{A}^{'}_{exe}|$, then $\Pi^{'}_{ka}(\pi)$ represents a minimal set that achieves the objective of making sure $\mathcal{C}^{min}_{exe}(R_0^{\Delta}) < \infty$. This HIS formulation is well suited for settings in which communication is costly. For settings in which movement in the environment is costly, we would instead set the cost of $\mathcal{C}^{'}_{ka}$ so that the total cost of acquisition actions is lower than a single execution action. This guarantees that a cost minimal plan to the goal is available. Specifically, if $\mathcal{C}^{'}_{ka} < \frac{1}{|\mathcal{A}^{'}_{ka}|}$, the accumulated cost of knowledge acquisition actions is smaller than the cost of a single execution action (equal to 1) and $\mathcal{C}^{min}_{exe}(R_0^{\Pi^{'}_{ka}}) = \mathcal{C}^{*}_{exe}(R_0)$.

5.3 Finding HIS Solutions for Replanning Agents

In contrast to conservative agents, *replanning agents* are willing to make assumptions about missing information to compute plans. Even under the assumptions that backtracking is always possible, a replanning actor may fail to reach an achievable goal due to its limited sensors. In Figure 1, even though there is a path to the goal, the robot fails because it's sensor could not detect it. The objective of HIS is to automatically find the minimal information that needs to be communicated to the actor to guarantee that at least one plan (Equation 1) or a minimal cost plan (Equation 2) is achievable, in spite of the actor's limited sensors.

The actor's ability to make assumptions are modeled in the KAC translation via the assumption actions $\mathcal{A}^{'}_{as}$, which set the value of variables during the planning. It is important take such assumptions into account since we are considering cases in which communication is costly. This means that if an actor can make the correct assumption about some variable, it would be wasteful for the helper to convey that information.

In Example 1.1, an optimal solution in the case of conservative actors that don't make assumptions is to reveal that cells $(C, 1)$, $(D, 1)$, $(E, 1)$, $(E, 2)$ and $(E, 3)$ are free. In the case of replanning actors, some of this communication is redundant, since the helper can rely on the actor's willingness to make assumptions and its ability to infer some of this information during execution. In Figure 2, the helper knows the actor will be able to infer that cell $(C, 1)$ is free when reaching cell $(B, 1)$ (and sensing no danger signal). In contrast, because of the signal in cell $(C, 1)$, the actor has no way to infer that cell $(D, 1)$ is safe based only on it's sensors, requiring the helper to communicate that the cell is free.

A replanning agent can make assumptions and revise them if they are refuted during execution. Since our objective is to find a plan that the actor will be able to follow, we restrict the planner to assumption actions that correspond to the actual values of variables. As before, we assign a cost of 1 to execution actions and a cost of 0 to ramifications.

First we consider the case in which the helper is interested in minimizing information cost (Equation 1). In this case, the cost of assumptions is set such that the maximal total cost of assumptions is smaller than the cost of a single action, i.e., $\mathcal{C}^{'}_{as} < \frac{1}{\mathcal{A}'}$. To guarantee knowledge acquisition actions are applied only if necessary, we assign them with a cost $\mathcal{C}^{'}_{ka}$ s.t. $|\mathcal{A}'| < \mathcal{C}^{'}_{ka}$. This guarantees that a single knowledge acquisition action is higher than the cost of the most expensive plan. To account for cases in which the helper aims to minimize execution cost (Equation 2), we set the cost of assumptions to be negligible. Specifically, we want to guarantee that the maximal accumulated cost of assumptions is smaller than the cost of a single knowledge acquisition action. In turn, the maximal accumulated cost of information acquisition actions is set to be smaller than the cost of a single execution action. Formally, $\mathcal{C}^{'}_{as} \cdot |\mathcal{A}'| < \mathcal{C}^{'}_{ka}$ and $\mathcal{C}^{'}_{ka} \cdot |\mathcal{A}^{'}_{ka}| < \mathcal{C}^{'}_{ex} = 1$.

6 Conclusions

We have introduced the problem of helpful information shaping (HIS), which considers a fully informed helper agent, who can share some information with a partially informed actor, the helper seeking to help the acting agent with achieving its goal. To solve HIS, we suggest the KAC compilation that represents in a single planning problem both the actor's planning problem and the helper's ability to share information. We then show how the compilation can account for different objectives the helper may have and produce a solution with a single call to an off-the-shelf classical planner.

There are many ways to extend the HIS framework. First, while our work used a qualitative non-deterministic representation of the uncertainty of a world state, it will be interesting to use probabilistic models to represent the actor's belief. Future work could also consider settings where information shaping can be applied at execution time, based on actors' actual progress. It is also interesting to consider how our approach can be extended to support reinforcement learning agents, and to support multi-agent settings, where the helper agent uses a limited communication channel to support a diverse set of actors in the system. This brings about normative questions about how to balance in a fair way possibly competing considerations.

References

[1] Alexandre Albore, Héctor Palacios, and Héctor Geffner. A translation-based approach to contingent planning. In *IJCAI*, 2009.

[2] Blai Bonet and Hector Geffner. Planning with incomplete information as heuristic search in belief space. In *Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems (AIPS)*, pages 52–61. AAAI Press, 2000.

[3] Blai Bonet and Hector Geffner. Planning under partial observability by classical replanning: Theory and experiments. In *IJCAI*, 2011.

[4] Ronen Brafman and Guy Shani. Replanning in domains with partial information and sensing actions. *Journal of Artificial Intelligence Research*, 45, 2014.

[5] Michael Brenner and Bernhard Nebel. Continual planning and acting in dynamic multiagent environments. *Autonomous Agents and Multi-Agent Systems*, 19(3):297–331, 2009.

[6] Richard Camilli, Eric Timmons, Benjamin Ayton, Zachary Duguid, Aggelos Mallios, David Pingal, Christopher DeCollibus, Peter Vrolijk, Paraskevi Nomikou, Lori L Summa, et al. Simultaneous localization and mapping of seafloor cold seeps with autonomous underwater gliders. *AGUFM*, 2019:OS23C–1799, 2019.

[7] Peter Drucker. *The practice of management*. Routledge, 2012.

[8] Ece Kamar, Ya'akov Gal, and Barbara J Grosz. Incorporating helpful behavior into collaborative planning. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. Springer Verlag, 2009.

[9] Emir Kamenica. Bayesian persuasion and information design. *Annual Review of Economics*, 11(1):249–272, 2019.

[10] H. Kautz, O. Etzioni, D. Fox, D. Weld, and L. Shastri. Foundations of assisted cognition systems. Technical report, University of Washington, 2003.

[11] Sarah Keren, Sara Sara Bernardini, Kofi Kwapong, and David C. Parkes. Reasoning about plan robustness versus plan cost for partially informed agents. In *International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 2020.

[12] Shlomi Maliah, Ronen Brafman, Erez Karpas, and Guy Shani. Partially observable online contingent planning using landmark heuristics. In *Twenty-Fourth International Conference on Automated Planning and Scheduling*, 2014.

[13] Christian J Muise, Vaishak Belle, and Sheila A McIlraith. Computing contingent plans via fully observable non-deterministic planning. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2014.

[14] Kevin M Murphy and Andrei Shleifer. Persuasion in politics. *American Economic Review*, 94(2):435–439, 2004.

[15] Tuan Nguyen, Sarath Sreedharan, and Subbarao Kambhampati. Robust planning with incomplete domain models. *Artificial Intelligence*, 245:134–161, 2017.

[16] Hector Palacios and Hector Geffner. Compiling uncertainty away in conformant planning problems with bounded width. *Journal of Artificial Intelligence Research*, 35:623–675, 2009.

[17] David Sarne and Barbara J Grosz. Estimating information value in collaborative multi-agent planning systems. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pages 1–8, 2007.

[18] Vaibhav V Unhelkar and Julie A Shah. Contact: Deciding to communicate during time-critical collaborative tasks in unknown, deterministic domains. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2016.

[19] Feng Wu, Shlomo Zilberstein, and Xiaoping Chen. Online planning for multi-agent systems with bounded communication. *Artificial Intelligence*, 175(2), 2011.

[20] Ping Xuan, Victor Lesser, and Shlomo Zilberstein. Communication decisions in multi-agent cooperation: Model and experiments. In *Proceedings of the fifth international conference on Autonomous agents*, 2001.

[21] Alexander Zelinsky. A mobile robot exploration algorithm. *IEEE transactions on Robotics and Automation*, 8(6):707–717, 1992.

[22] Haoqi Zhang, Yiling Chen, and David C Parkes. A general approach to environment design with one agent. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2009.

[23] Haoqi Zhang and David C Parkes. Value-based policy teaching with active indirect elicitation. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2008.