Helpful Information Sharing for Partially Informed Planning Agents

Sarah Keren[1] , **David Wies**[2] and **Sara Bernardini**[3]
[1]The Taub Faculty of Computer Science, Technion - Israel Institute of Technology
[2] Benin School of Computer Science and Engineering, The Hebrew University of Jerusalem
[3]Department of Computer Science, Royal Holloway University of London

Abstract

In many real-world settings, an autonomous agent may not have sufficient information or sensory capabilities to accomplish its goals, even when they are achievable. In some cases, the needed information can be provided by another agent, but information sharing might be costly due to limited communication bandwidth and other constraints. We address the problem of *Helpful Information Sharing* (HIS), which focuses on selecting minimal information to reveal to a partially informed agent in order to guarantee it can achieve its goal. We offer a novel compilation of HIS to a classical planning problem, which can be solved efficiently by any off-the-shelf planner. We provide guarantees of optimality for our approach and describe its extensions to maximize robustness and support settings in which the agent needs to decide which sensors to deploy in the environment. We demonstrate the power of our approaches on a set of standard benchmarks as well as on a novel benchmark.

1 Introduction

An agent, or *actor*, tasked with reaching a goal, may have only partial information about its surroundings and limited sensors to acquire new information. In situations where safety constraints are in place and failures are to be avoided, the lack of information may paralyze the actor, which might fail to reach its goal even when the goal would be achievable with more information. In a collaborative multi-agent setting, another agent, the *helper*, may possess information that could make it possible for the actor to achieve its goal. At the same time, the communication bandwidth between the actor and the helper might be limited, forcing the helper to choose which information to transfer to the actor. We explore the question of *Helpful Information Sharing* (HIS) that aims at determining the minimal information the helper needs to share with the actor to guarantee its goal is achievable.

HIS is relevant to a broad variety of applications in which one may seek to minimize communication because it is expensive, unreliable, or prone to interception. Consider, for example, underwater autonomous operations with squads of robots (e.g. Camilli et al. 2019). Communication is typically highly constrained in these missions. Underwater, the communication bandwidth is limited, hence the members of the squad need to decide carefully what information to pass to each other. Resurfacing facilitates communication, but this option can only be used sparsely as it consumes resources that could be devoted to performing the mission. Also in this case, it is crucial that the agents exchange only information that is vital to reach the goal. Similar challenges would be faced by a human operator involved in the mission. Consider a controller on a surface vehicle tasked with assisting the robots in reaching their goals. Exploiting a different point of view and a different set of instruments, the controller might gain useful information to help the squad. However, being the bandwidth of the communication channel limited and sporadic, the operator needs to cautiously select which information to transmit and also decide which UAVs in the team to assist.

We formalize HIS as a two-agent setting. The actor uses a partially informed planner that computes a plan to achieve the goal by using its knowledge and available sensors. The helper is assumed to have additional information about the environment, as well as knowledge of the actor, its goal, and the approach the actor takes for planning and acting. The helper's objective is to find minimal information to share with the actor so that it can achieve its goal.

HIS is challenging because there may be a large number of possible choices in regard to what information to share, and evaluating the effect of each information sharing option may be costly. To mitigate this challenge, we offer novel compilations from the two-agent HIS problem to a single-agent planning problem, which includes both information-sharing actions and actions in the environment. After providing guarantees of the optimality of our approach, we discuss how our solution can be applied to additional settings. In particular, we show how our approach can be used in single-agent settings in which the actor needs to decide which information to acquire (e.g., by deciding which sensors to deploy), in settings with alternative performance guarantees, and in other sequential decision-making under uncertainty settings, in which information may be costly or limited.

We test our approaches on a set of standard benchmarks as well as a novel benchmark of an Escape Room that we developed. Our experiments demonstrate the power of our strategy in solving a variety of HIS problems.

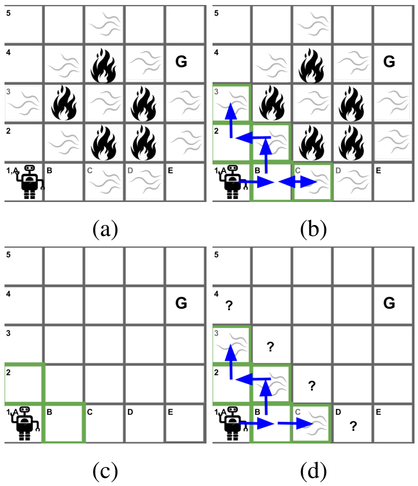

Figure 1: Helpful Information Sharing (HIS) example.

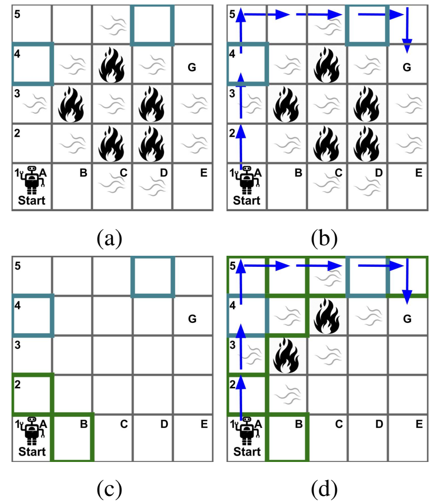

Figure 2: HIS solution

1.1 Running Example

A robot is tasked with reaching a destination in a simplified indoor environment. The robot maintains an *occupancy map* where each cell is marked as 'free' (traversable), 'occupied' (non-traversable), or 'unknown' (the robot doesn't know whether or not the cell is traversable). Initially, all cells are marked as 'unknown'. The map is updated dynamically as the robot gains more information. For simplicity, the world is static, so the value of each cell can only change from 'unknown' to 'occupied' or 'free' 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 the sensor does not indicate which specific cell is occupied. If a robot tries to move into an occupied cell, it may get stuck without the possibility of recovering. Therefore, if the robot's sensors indicate the presence of a nearby obstacle, the robot will only move to an adjacent cell if it can infer that it is free and will backtrack otherwise (based on the Wumpus domain by [Russell and Norvig, 2016]).

Example 1. *Figure 1a depicts a simple room with a single entry point and a goal destination G. Cells with obstacles (e.g., cell $(B, 3)$) are marked by flames that emit signals, depicted as smoke waves. Such a signal indicates to the robot the existence of an obstacle in one or more adjacent cells. Figure 1c depicts the robot's initial belief, which corresponds to its initial knowledge. 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 willing to follow a plan that is not guaranteed to succeed and replan its behavior if needed, it will compute a plan based on the assumption that some cells are free. As shown in Figure 1b and the corresponding robot belief in Figure 1d, without additional information, the robot will backtrack to a cell it knows is free every time it senses an obstacle. In this case, the robot will abort execution of its task after examining all possible paths.*

In Figure 2a, we show a minimal set of information-sharing interventions that guarantee that a replanning agent can achieve its goal. In this case, an optimal solution is for the helper to reveal that cells $(A, 4)$ and $(D, 5)$ are free (the cells marked blue are those whose true value is revealed). Figure 2c shows the robot's initial belief. The path that successfully leads the robot to its goal after the additional information is provided is depicted in Figure 2b, and Figure 2d shows the robot's path along with information that is acquired during execution. Note that this information-sharing intervention is minimal in terms of the number of information items shared. In order to guarantee that a minimal cost plan is achievable, more information needs to be revealed.

2 Planning Under Partial Observability

We follow Bonet and Geffner [2011; 2014] in modeling a partially informed actor.

Definition 1 (Planning Under Partial Observability). *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 (referred to as* facts*) 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 as preconditions, and a set $eff(a)$ of conditional effects $C \rightarrow L$, 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, which we assume always includes the actual state. For unknown fluents, states for both values are included in the belief until one of the values is refuted. 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 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 is the set of 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 $\mathcal{C}_a(h)$, is the accumulated cost of the performed actions (equivalent to the path length when action cost is uniform): $\mathcal{C}_a(h) = \Sigma_i \mathcal{C}(a_i)$. Each history corresponds to a path $\pi = a_0, a_1, \ldots, a_n$, which is the sequence of actions the agent performs. A path is a *plan* if the agent performing the path reaches a goal belief.

We are interested in bridging the gap between what is achievable in the environment and what can be accomplished by a partially informed agent. Given a PPO problem $\mathcal{P}$, we let $S_{\mathcal{P}}$ be the set of possible world states in $\mathcal{P}$. Similarly, the set $\mathcal{B}_{\mathcal{P}}$ is the set of possible beliefs in $\mathcal{P}$, and b_0 is the initial belief. We refer to a path as *executable* in the environment if it is executable in the real-world state. An *executable plan* is an executable path that achieves the goal.

Definition 2 (Executable Path). *Given a PPO problem $\mathcal{P}$ and state $s \in S_{\mathcal{P}}$, a path $\pi = \langle a_0, a_1, \ldots, a_n \rangle$ is* executable *in s if a_0 is applicable in s and, for any $0 < i \leq n$, a_i is applicable in $a_{i-1}(\ldots(a_0(s)))$.*

Definition 3 (Executable Plan). *Given a PPO problem $\mathcal{P}$ and a state $s \in S_{\mathcal{P}}$, a path $\pi = \langle a_0, a_1, \ldots, a_n \rangle$ is an* executable plan *in s if it is an executable path in s and $a_n(\ldots(a_0(s)))$ satisfies G.*

The above definitions account for what an agent (with full information) can achieve in the environment. To account for what is achievable with limited information and sensing capabilities, we define a *PO-executable path* and *PO-executable plan* below.

Definition 4 (PO-executable Path). *Given a PPO problem $\mathcal{P}$ and belief $b \in \mathcal{B}_{\mathcal{P}}$, a path $\pi = \langle a_0, a_1, \ldots, a_n \rangle$ is a* PO-executable path *in b if a_0 is applicable in b and, for any $0 < i \leq n$, a_i is applicable in $a_{i-1}(\ldots(a_0(b)))$.*

Definition 5 (PO-executable Plan). *Given a PPO problem $\mathcal{P}$ and belief $b \in \mathcal{B}_{\mathcal{P}}$, a plan $\pi = \langle a_0, a_1, \ldots, a_n \rangle$ is a* PO-executable plan *in b if it is a PO-executable path in b and $a_n(\ldots(a_0(b)))$ satisfies G.*

Note that, since the actor is partially informed, there is a difference between goal states, in which the goal is achieved, and goal beliefs, in which the actor *knows* the goal has been achieved. It's easy to show that, since an action is applicable in a belief only if it is applicable in every state in the belief (including the actual state), every PO-executable plan is a plan that is guaranteed to be executable in the actual state. Our objective is to find an efficient way to make sure that at least one executable plan is PO-executable by the actor. Since we assume our conservative actor only performs actions it knows the outcome of, the set of executable plans that can be sequentially applied to achieve the goal subsumes the set of PO-executable plans.

Two issues need to be addressed when planning with incomplete information: belief tracking and planning. Belief tracking considers the task of keeping track of the agent's belief as it operates in the environment and collects new information [Geffner and Bonet, 2013]. In the worst case, the computation of the belief states is exponential in the number of state variables. In this work, we focus on understanding which information is needed by a partially informed planner and assume its belief tracking is *sound* and *complete*.

There are two main approaches to planning under partial information, offline and online [Brafman and Shani, 2014], and a variety of approaches of these two types have been developed for PPO planning. A common technique for online planning is *replanning* [Zelinsky, 1992], where an agent finds a plan for its current state based on some simplification of its planning problem and on assumptions it makes about unknown variables. The agent then executes a prefix of the plan until discrepancies between the plan and the information acquired during the execution emerge and require replanning. Our focus here is on providing relevant information to a partially informed replanning agent; we are agnostic to the approach it uses for planning as long as it complies with our assumption that the agent is conservative.

3 Helpful Information-Sharing (HIS)

We consider a two-agent setting where the first agent, the *actor*, is a partially informed replanning agent that uses a PPO problem to decide how to act to achieve its goal. The second agent, the *helper*, has access to the actor's planning model and additional information in the form of facts (i.e. $\mathcal{F}$-literals), which it knows are true and can be shared with the actor. For convenience, we indicate the set of facts over $\mathcal{F}$ as $L_{\mathcal{F}}$. We assume that each fact $f \in L_{\mathcal{F}}$ is revealed and added separately to the actor's knowledge. We refer to each revelation as an *information-sharing intervention.*

Definition 6 (Information-Sharing Intervention). *Given PPO problem $\mathcal{P} = \langle \mathcal{F}, \mathcal{A}, I, G, \mathcal{O} \rangle$ and fact $f \in L_{\mathcal{F}}$,* information-sharing intervention *δ_f adds f to I, $\delta_f(I) = I \cup \{f\}$.*

We assume that the set of facts known to the actor is subsumed by the set known to the helper, but the helper is not necessarily fully informed (in Example 1, it may be aware of only a subset of the blocked cells). The helper uses its knowledge, which may include both invariant and non-invariant facts, and knowledge of the actor to find a minimal set of facts to reveal to the actor to guarantee it can achieve its goal.

Specifically, HIS consists of the helper finding a minimal set of facts to reveal to the partially informed actor to guarantee that it can execute at least one executable plan despite its partial knowledge. Given the set $\boldsymbol{\Delta}$ of information-sharing interventions the helper can perform (i.e., the facts it can reveal), we let $\mathcal{P}_0^{\Delta}$ represent the model that results from applying the set of interventions $\Delta \subseteq \boldsymbol{\Delta}$ to $\mathcal{P}_0$ such that $\mathcal{P}_0$

and $\mathcal{P}_0^\Delta$ are equivalent except for $I_{\mathcal{P}_0^\Delta} = I_{\mathcal{P}_0} \cup L_{\mathcal{F}}(\Delta)$, where $L_{\mathcal{F}}(\Delta)$ is the set of facts in Δ. Also, given a PPO problem $\mathcal{P}$, we let $\Pi_e(\mathcal{P})$ and $\Pi_{poe}(\mathcal{P})$ represent the executable plans in the initial state of $\mathcal{P}$ and the PO-executable plans in the initial belief of $\mathcal{P}$, respectively. We seek to find a minimum set of facts to reveal offline to the actor to guarantee that, if an executable plan in the initial state of $\mathcal{P}_0$ exists, then a PO-executable plan in the initial belief of $\mathcal{P}_0^\Delta$ also exists.

Definition 7 (HIS). *Given an actor's initial model $\mathcal{P}_0 = \langle \mathcal{F}, \mathcal{A}, I, G, \mathcal{O} \rangle$ and an intervention set $\mathbf{\Delta}$, the problem that HIS aims to solve is as follows:*

$$\begin{aligned} &\Delta^* = \arg \min_{\Delta \subseteq \mathbf{\Delta}} (|\Delta|) \\ &\textit{s.t. if } \Pi_e(\mathcal{P}_0) \neq \emptyset \textit{ then } \Pi_{poe}(\mathcal{P}_0^\Delta) \neq \emptyset \end{aligned} \tag{1}$$

We note that if there is no solution to the HIS problem, that does not necessarily entail there is no way for the actor to achieve the goal based on its sensors. It may mean that the helper cannot provide an offline guarantee that there is a plan that will succeed based on the information it possesses.

For the sake of simplicity, we assume that the *cost of communication* is uniform among all facts that can be revealed. Extending this approach to non-uniform cost settings is straightforward and requires minimizing the total cost of communication instead of the number of facts that are revealed. In addition, recall that our actor is a replanning agent that can only make assumptions about unknown facts that can be discovered by using one of its sensors, i.e. facts $f \in L$ in some sensor $(C, L) \in \mathcal{O}$. The actor can make new assumptions and replan if one of its previous assumptions is refuted. Since we are interested in minimizing communication, it is wasteful to reveal facts that can be sensed by the actor. The solution to HIS will, therefore, include only facts that hinder goal achievement and cannot be acquired by the actor.

Example 1 (continued). *It is redundant to reveal to the actor that cell (A,3) is 'free' since the actor can infer this when its sensors indicate no signal at adjacent cells. In contrast, its sensors are insufficient to guarantee that cell (A,4) is free given the signal detected at cell (A,3). An optimal HIS solution for this case is depicted in Figure 2.*

4 Finding Optimal Solutions for HIS

We propose two approaches to seek optimal solutions to HIS.

4.1 Lazy Breadth-First Search

A baseline approach for solving HIS is to perform a breadth-first search (BFS) in the space of interventions, computing the actor's cost to goal at each node (in the case of non-uniform communication cost, a Dijkstra search can be applied instead). The root node is the initial model $\mathcal{P}_0$ (and empty intervention set), and the operators (edges) are the interventions $\delta \in \mathbf{\Delta}$ that make transitions between models. The value of each node and the corresponding intervention set Δ represent whether the goal is reachable, i.e. whether $\Pi_{poe}(\mathcal{P}_0^\Delta) \neq \emptyset$. The search explores intervention 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, the search halts if a solution is found or if there are no more nodes to explore and returns the shortest path (smallest intervention set) to a node that achieves the optimal value. This approach is guaranteed to find an optimal solution but does not scale to large problems.

To increase efficiency over the exhaustive approach, we suggest Lazy Breadth-First Search (Lazy-BFS), which avoids the need to compute the value of nodes that are guaranteed not to represent optimal solutions. The structure of the search tree is similar to the one described for BFS. The novelty is that, instead of fully evaluating each node, we use a lazy approach to node evaluation; when a node is expanded during the search, we map it to a relaxed intervention set, which is guaranteed to overestimate the value of the node. We compute the exact value of the node only if the value for its relaxed set achieves the objective. Computational savings are achieved by storing the values of the relaxed sets and reusing them for future nodes that are mapped to the same relaxed set. To achieve a relaxation of the examined intervention set, we exploit the parameterized representation of our information-sharing interventions and use parameterized padding [Keren *et al.*, 2017], which associates each intervention set to a superset that includes all interventions that share the same value of one or more of its parameters. A full description of Lazy-BFS is given in the appendix[1].

4.2 The T_{ka} Translation

To increase efficiency, we transform HIS into a planning problem, which can be solved to obtain an optimal solution to the original HIS problem by using a single call to an off-the-shelf classical optimal planner. Our approach is an extension of the T_k translation of Bonet and Geffner [2011], which finds a plan for a partially informed agent [Bonet and Geffner, 2014]. At the core of the T_k translation is the substitution of each fluent 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 [Albore *et al.*, 2009; Palacios and Geffner, 2009]. Each original action is transformed into an equivalent action that replaces the use of every literal to the need to know its value. The agent can make assumptions about observations it can perform during execution and infer new information based on sensory input. This representation captures the underlying planning problem at the knowledge level [Petrick and Bacchus, 2002; Bonet and Geffner, 2014], accounting for the exploratory behavior of a partially informed agent. By using the T_k translation, the actor can follow a *planning under optimism* approach: it makes the most convenient assumptions about the values of hidden variables and replans its behavior if an assumption is refuted during execution.

We extend T_k by encoding the helper's knowledge into the actor's problem description and by allowing the actor to select which information it needs to acquire to reach the goal. Importantly, this approach will exhibit which information is necessary to reveal to the agent since it will not be able to acquire or deduce this information on its own.

[1] Appendix: https://github.com/sarah-keren/HIS-IJCAI-23

Our translation, denoted as T_{ka}, takes as input the actor's planning problem as well as the helper's possible information-sharing options. In addition to execution, assumption, and ramification actions, which appeared in T_k, T_{ka} also includes *knowledge acquisition actions* $\mathcal{A}^{'}_{ka}$, which are modeled as part of the planning problem and represent the sharing of information by the helper. We also add a cost function, which we use to guarantee that an optimal solution to the translation minimizes communication cost.

Definition 8 (T_{ka} **Translation).** *For any HIS problem* $M = \langle \mathcal{P}_0, \mathbf{\Delta} \rangle$, $T_{ka}(M) = \langle \mathcal{F}', I', G', \mathcal{A}', \mathcal{C}' \rangle$ *is the fully observable planning problem where*

- $\mathcal{F}' = \{KL, K\neg L : L \in \mathcal{F}\}$
- $I' = \{KL : L \in I\}$
- $G' = \{KL : L \in G\}$
- $\mathcal{A}' = \mathcal{A}^{'}_{exe} \cup \mathcal{A}^{'}_{ram} \cup \mathcal{A}^{'}_{as} \cup \mathcal{A}^{'}_{ka}$ *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* $\neg K\neg C \rightarrow \neg K\neg L$.
  - $\mathcal{A}^{'}_{ram} = \{a_{ram} |$ *for invariants* $\neg C \vee L$ *in* $I\}$ *where*
    * *prec*$(a_{ram}) = \{KC\}$ *and*
    * *eff*$(a_{ram}) = \{KL\}$
  - $\mathcal{A}^{'}_{as} = \mathcal{A}^{'+}_{as} \cup \mathcal{A}^{'-}_{as}$ *where*
    * $\mathcal{A}^{'+}_{as} = \{a_{(C,L)} | o = (C, L) \in \mathcal{O}, L \in L_{\Delta}\}$ *where*
      · *prec*$(a_{(C,L)}) = \{KC, \neg KL, \neg K\neg L, L\}$ *and eff*$(a_{(C,L)}) = \{KL\})$
    * $\mathcal{A}^{'-}_{as} = \{a_{(C,\neg L)} | o = (C, L) \in \mathcal{O}, \neg L \in L_{\Delta}\}$ *where*
      · *prec*$(a_{(C,\neg L)}) = \{KC, \neg KL, \neg K\neg L, \neg L\}$ *and eff*$(a_{(C,\neg L)}) = \{K\neg L\}$
  - $\mathcal{A}^{'}_{ka} = \mathcal{A}^{'+}_{ka} \cup \mathcal{A}^{'-}_{ka}$ *where*
    * $\mathcal{A}^{'+}_{ka} = \{a_L | L \in L_{\Delta}\}$ *where*
      · *prec*$(a_L) = \{\neg KL, \neg K\neg L\}$ *and eff*$(a_L) = \{KL\})$
    * $\mathcal{A}^{'-}_{ka} = \{a_{\neg L} | \neg L \in L_{\Delta}\}$ *where*
      · *prec*$(a_{\neg L}) = \{\neg KL, \neg K\neg L\}$ *and eff*$(a_{\neg L}) = \{K\neg L\})$
- $\mathcal{C}'(a) = \begin{cases} 0 \leq C^{'}_{exe} < \infty & \textit{if } a \in \mathcal{A}^{'}_{exe} \\ 0 \leq \mathcal{C}^{'}_{as} < \infty & \textit{if } a \in \mathcal{A}^{'}_{as} \\ 0 \leq \mathcal{C}^{'}_{ram} < \infty & \textit{if } a \in \mathcal{A}^{'}_{ram} \\ 0 \leq \mathcal{C}^{'}_{ka} < \infty & \textit{if } a \in \mathcal{A}^{'}_{ka} \end{cases}$

In the definition above, $\mathcal{F}'$, I' and G' are the same as in the T_k translation. The key difference is in the definition of $\mathcal{A}'$ and the addition of the cost function $\mathcal{C}'$. T_{ka} includes an action in $\mathcal{A}^{'}_{ka}$ for every fact in L_{Δ} that is known by the helper and that can be shared with the actor. In addition, instead of allowing the planner to make any assumption that is part of the actor's sensor model, in T_{ka}, we only allow making assumptions that are known to be true by the helper. This is because T_{ka} takes the point of view of the helper, which uses the translation to find facts to reveal to the actor that are guaranteed not to be refuted during execution and that ensure the goal is achievable by the actor.

A solution π' to the T_{ka} translation is a sequence of actions in $\mathcal{A}'$, which includes execution actions ($\mathcal{A}^{'}_{exe}$), ramification actions ($\mathcal{A}^{'}_{ram}$), assumptions ($\mathcal{A}^{'}_{as}$) and knowledge acquisition actions ($\mathcal{A}^{'}_{ka}$). The solution to the HIS problem is the set $\Pi_{ka}(\pi')$ of knowledge acquisition actions in π', which represents the information the helper should share with the actor to guarantee goal reachability. We denote the sequence of execution action in π' as $\Pi_{exe}(\pi')$.

4.3 Theoretical Analysis

We start our formal analysis of the T_{ka} translation by showing that the output of the translation is a valid solution for HIS, i.e., it finds a set of facts to share with the actor to guarantee the goal is achievable. Then, we provide conditions under which the set is minimal. All proofs are in the appendix. We first show that, under the assumption that the actor's belief tracking is sound and complete, at every execution step, the belief of the actor in the original model M is subsumed by the belief in the translated problem $T_{ka}(M)$.

Lemma 1. *Given a HIS problem* $M = \langle \mathcal{P}_0, \mathbf{\Delta} \rangle$*, and a solution* π' *to* $T_{ka}(M)$*, we let* b_i *and* b'_i *represent the actor's belief directly before applying action* a_i *in* $\Pi_{exe}(\pi')$ *in* M *and* $T_{ka}(M)$*, respectively. If the actor's belief tracking is sound and complete, then for any* i*,* $b_i \subseteq b'_i$.

We will now show that the solution produced by the translation represents a valid solution to the HIS problem.

Theorem 1. *For a HIS problem* $M = \langle \mathcal{P}_0, \mathbf{\Delta} \rangle$*, assuming belief tracking is sound and complete, there exists a solution to* $T_{ka}(M)$ *if and only if there is a solution to* M.

Corollary 1. *For a HIS problem* $M = \langle \mathcal{P}_0, \mathbf{\Delta} \rangle$ *and a solution* π' *of* $T_{ka}(M)$*,* $\Pi_{exe}(\pi')$ *is a PO-executable plan of* M.

The intuition behind the proofs is that the translation represents all possible behaviors of the PPO actor because it captures all possible changes that can be applied to the actual state as well as the agent's belief. Also, based on Lemma 1, we know that the solution to the translation is applicable in the original PPO problem, and therefore it represents a PO-executable plan.

So far, we have shown that the T_{ka} translation reveals a set of interventions that guarantee the goal is achievable. We now specify conditions that guarantee the solution minimizes communication cost. This is achieved by manipulating the cost function so that a single information-acquisition action is more costly than all other actions. This computational artifice is used to guarantee the conditions under which the translation will find the minimal set of facts to reveal to achieve the objective and does not correspond to actual cost.

Theorem 2. *For any HIS problem* $M = \langle \mathcal{P}_0 \mathbf{\Delta} \rangle$ *if*

$$C^{'}_{exe} \cdot |\mathcal{A}^{'}_{exe}| + \mathcal{C}^{'}_{as} \cdot |\mathcal{A}^{'}_{as}| + \mathcal{C}^{'}_{ram} \cdot |\mathcal{A}^{'}_{ram}| < \mathcal{C}^{'}_{ka}$$

then, for any optimal solution π'^{*} *to* $T_{ka}(M)$*,* $\Pi_{ka}(\pi'^{*})$ *represents a minimal set of information-sharing interventions that guarantee the actor can achieve its goal.*

5 T_{ka} Extensions

In the HIS setting described above, we have a helper agent that can reveal known facts to the actor. In many real-world settings, the actor itself can choose to perform information acquisition actions but needs to decide which of the possible costly actions to perform. Such actions can correspond to placing or activating sensors in the environment, performing information-gathering actions, or proactively requesting information from other agents. The key difference of such settings is that, as opposed to the two-agent setting described above in which the actual values of the revealed information items are known to the actor, here the actual outcome of the information acquisition actions is not known. To support such settings, we extend our approach by letting the actor itself apply T_{ka} to its planning problem such that $\boldsymbol{\Delta}$ now represents the information the actor can request (e.g., which sensors to deploy). Since the real values of the facts in L_Δ are not known, we have knowledge acquisition actions for every possible value of each fact (i.e., for each fact $L \in L_\Delta$, the agent can acquire L or $\neg L$). Actions for which the agent decides to acquire a specific value are those which represent valuable information acquisition actions.

Our approach can also be used in settings in which the observer may not have enough knowledge to guarantee the actor can accomplish its goal but may be willing to settle for operating under a *robustness guarantee*, which quantifies the plan's ability to avoid failure. For this purpose, we rely on Keren et al. [2020a], where plan robustness is defined as the number of states in the belief for which the plan is guaranteed to succeed. Thus, we can find a minimal intervention set that yields a solution that complies with a robustness threshold. This is achieved by allowing the solution to include a specified number of assumptions in $\mathcal{A}'_{as}$ for which the value is not known (the larger the number, the lower the robustness guarantee).

Finally, Theorem 2 provides conditions on the cost function under which T_{ka} finds a minimal set of interventions that guarantee goal achievement, thus focusing on settings in which communication is costly. Applying this idea to settings in which execution actions or ramification actions should be minimized is straightforward. Using the cost function, we can impose any ordering by which actions of each type are performed. For example, by creating a strict ordering between execution, information-acquisition, assumptions (on values that are known to be true) and ramification actions, we can find the minimal information that is needed to guarantee an optimal plan is achievable by the partially informed actor.

6 Empirical Evaluation

The objective of the evaluation is to examine our approaches in terms of the computational resources used and their ability to find helpful interventions in a variety of domains.

Dataset. We use seven partially observable planning domains: **WUMPUS**, **TRAIL**, **COLOR-BALLS**, **COLOR-BALLS-E**, **LOGISTICS**, and **UNIX** [Bonet and Geffner, 2011; Albore *et al.*, 2009]. The adaptation to HIS involves limiting the actor's sensors and specifying the information that can be revealed. We also introduce a new domain, **ESCAPE-ROOM**, in which the helper seeks a minimal set of clues that guarantee a puzzle is solvable. For each domain, we generated at least 100 benchmarks by randomly selecting the initial state and the set of items that are known to the helper. The Appendix includes a description of all the domains and the complete dataset and code.

	BFS		Lazy-BFS		T_{ka}	
	sol	time	sol	time	sol	time
WUMPUS	0.81	27.12	0.81	18.33	0.71	**13.18**
TRAIL	1	21.24	1	21.17	0.69	**15.76**
COLOR-BALLS	0	–	0	–	0.77	**16.69**
COLOR-BALLS-EX	0	–	0	–	0.45	**21.07**
UNIX	0.56	16.35	0.56	15.39	0.99	**3.01**
LOGISTICS	0.54	36.59	0.52	32.08	1	**0.29**
ESCAPE-ROOM	0.84	0.84	0.84	1.25	0.84	**0.53**

Table 1: Comparing performance of the different HIS approaches.

Setup. We compare three HIS solution approaches: BFS, Lazy-BFS (Section 4.1), and the T_{ka} transformation (Section 4.2). We use PDDL [McDermott *et al.*, 1998] to specify the domains and the available information-sharing interventions. The BFS approaches are implemented using an adaptation of *pyperplan* (https://github.com/aibasel/pyperplan) to parse the intervention file and provide the set of applicable information-sharing interventions for each node in the search and the model that results from applying each intervention. While our method is agnostic to the planner used by the actor, we use the K-planner [Bonet and Geffner, 2011] with Fast-Downward [Helmert, 2006] classical planner (FD) and the Lm-cut heuristic [Helmert and Domshlak, 2009].

We run each instance on the three approaches with a time limit of 10 minutes, a memory limit of 2548 MB, and 1000 explored nodes. All evaluations were run on an 11th Gen Intel® Core™ i9-11900F @ 2.50GHz $\times 16$.

We evaluate the following information-sharing settings:

- MININF: HIS setting described in Definition 7 in which the helper reveals a minimal set of facts that guarantees goal achievement.
- OPOP (Section 5): the helper reveals a minimal set of facts that guarantee an optimal plan is PO-executable.
- ROB (Section 5): the helper reveals a minimal set of facts to achieve the required robustness. We examine settings with decreasing robustness guarantees by allowing 1, 2 and 3 assumptions to be made by the agent, denoting them as ROB_1, ROB_2, and ROB_3.

Results. Table 1 compares the computational resources used by BFS, Lazy-BFS and T_{ka}. For each approach and domain, we measure 'sol' as the ratio of solved instances, i.e., instances for which the approach finishes before the allocated time bound is reached (best result is underlined), and 'time' as the average computation time in seconds for instances solved by all methods (best result in bold). For COLOR-BALLS(-EX), we indicate the average computation time for T_{ka}, since BFS and Lazy-BFS fail on all instances.

Results show that for WUMPUS and TRAIL, BFS solved a higher percentage of instances (with lower computation time

	MININF	OPOP
WUMPUS	1.3	1.52
TRAIL	1	1
COLOR-BALLS	1.23	1.23
COLOR-BALLS-EX	2.09	2.09
UNIX	1.82	1.82
LOGISTICS	1.84	4.17
ESCAPE-ROOM	1.1	1.3

Table 2: T_{ka} extensions: Comparing the effect of information sharing in MININF and OPOP.

	ROB_1	ROB_2	ROB_3
WUMPUS	2.67	2.39	2.28
TRAIL	3.31	2.73	2.38
COLOR-BALLS	2.33	2.33	2.33
COLOR-BALLS-EX	4.18	4.18	4.18
UNIX	2.19	2.19	2.19
LOGISTICS	1.84	1.84	1.84
ESCAPE-ROOM	2.79	2.5	1.5

Table 3: T_{ka} extensions: Comparing the effect of information sharing interventions under different robustness requirements.

for Lazy-BFS for all but ESCAPE-ROOM), while T_{ka} solves more instances on the other domains. In addition, T_{ka} substantially reduces computational time for all domains (most notably for the LOGISTICS domain).

An investigation of the instances for which T_{ka} did not complete, but BFS did, reveals instances with large spaces of information-sharing interventions for which the grounded version of the compiled problem exhausted the allocated memory. A direction for future investigation would be to explore ways to prune the space of grounded operators.

In Tables 2 and 3, we present the performance of the extensions of T_{ka} presented in Section 5. Table 2 examines the average number of information-sharing interventions needed to achieve the requirement in MININF and OPOP settings in which the observer has sufficient information for achieving the requirement (which is that some path is PO-executable for MININF and an optimal one is achievable for OPOP). As expected, the average number of information-sharing interventions increases as the requirements are more restrictive, i.e. more (or an equal number of) interventions are needed for OPOP than for MININF. In Table 3, we compare the number of interventions needed to achieve a required level of robustness when the observer does not necessarily have the needed information to guarantee goal achievement, but the actor can make $1-3$ assumptions. We note that when we compromise robustness (i.e. the agent can make more assumptions), fewer interventions are needed.

7 Related Work

The idea that information is malleable has been investigated in a variety of research disciplines, including economics [Kamenica, 2019], business management [Drucker, 2012], politics [Murphy and Shleifer, 2004], 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 most related to the extensive body of work on selective information revelation in multi-agent AI.

One relevant line of work considers communication in collaborative settings. Kamar et al. [2009] provide a mechanism for reasoning about the utility of communicating information relevant to an agent's plans in a collaborative setting. Similarly, several lines of work [Xuan *et al.*, 2001; Wu *et al.*, 2011; Unhelkar and Shah, 2016; Sarne and Grosz, 2007; Macke *et al.*, 2021] consider limited communication settings and investigate the gain to effective coordination versus communication cost. To address this challenge, a unified framework is created where communication becomes part of the overall agent decision problem. In [Marcotte *et al.*, 2020] a team of robots can share observations to improve team performance, but communication bandwidth is limited. The decision of what to communicate is based on forward simulations and a bandit-based combinatorial optimization algorithm. Our technique is different as it focuses on planning agents, ensuring the helper shares minimal information with the actor so that it can achieve its goal.

Information sharing can also be viewed as a special case of *environment design* [Zhang *et al.*, 2009], 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* [Zhang and Parkes, 2008; Zhang *et al.*, 2009] enables modifying the reward function of a stochastic agent to entice the agent to follow specific policies. We focus here on performing design by controlling the information an agent uses for planning, rather than by reward shaping, and on using information sharing to ensure that the goal is achievable.

Recent work [Shmaryahu *et al.*, 2019; Nguyen *et al.*, 2017; Keren *et al.*, 2020b] provides various comparative criteria for plans and policies for partially informed planning and ways to compute plans that comply with those criteria, i.e., plans that maximize robustness. Specifically, Keren et al. [2020b] offer a translation of the problem to classical planning that accounts for a user-specified level of robustness by associating a corresponding cost to making assumptions. Here, we extend this approach by allowing the actor to 'buy' the information it needs and exploit the cost function to guarantee that the generated solution minimizes communication costs.

8 Conclusion

We presented HIS as the problem of finding a minimal set of information-sharing interventions that guarantee that a partially informed actor can achieve its goal. To find optimal solutions, we offer a variation of BFS and a transformation of HIS to a single-agent planning problem that can be solved using any off-the-shelf classical planner. Our evaluation, based on a set of domains adapted from the literature, shows the computational benefit of our suggested approaches.

Possible extensions include an account of information-sharing settings with stochastic sensor models and faulty communication. Another extension considers the cost of sensor placement in settings in which the helper's knowledge is not sufficient to guarantee the goal is achievable.

Acknowledgements

Sarah Keren acknowledges the support of the Technion Autonomous Systems Program as well as the helpful discussions with Lucy Liu at the preliminary stages of this work.

References

[Albore *et al.*, 2009] Alexandre Albore, Héctor Palacios, and Héctor Geffner. A translation-based approach to contingent planning. In *The International Joint Conference on Artificial Intelligence (IJCAI)*, 2009.

[Bonet and Geffner, 2011] Blai Bonet and Hector Geffner. Planning under partial observability by classical replanning: Theory and experiments. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2011.

[Bonet and Geffner, 2014] Blai Bonet and Hector Geffner. Flexible and scalable partially observable planning with linear translations. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

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

[Camilli *et al.*, 2019] 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.

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

[Geffner and Bonet, 2013] Hector Geffner and Blai Bonet. *A concise introduction to models and methods for automated planning*, volume 8. Morgan & Claypool Publishers, 2013.

[Helmert and Domshlak, 2009] Malte Helmert and Carmel Domshlak. Landmarks, critical paths and abstractions: What's the difference anyway? In *International Conference on Automated Planning and Scheduling (ICAPS)*, 2009.

[Helmert, 2006] Malte Helmert. The fast downward planning system. *Journal of Artificial Intelligence Research(JAIR)*, 26, 2006.

[Kamar *et al.*, 2009] 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.

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

[Keren *et al.*, 2017] Sarah Keren, Luis Pineda, Avigdor Gal, Erez Karpas, and Shlomo Zilberstein. Equi-reward utility maximizing design in stochastic environments. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[Keren *et al.*, 2020a] Sarah Keren, 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.

[Keren *et al.*, 2020b] Sarah Keren, Haifeng Xu, Kofi Kwapong, David Parkes, and Barbara Grosz. Information shaping for enhanced goal recognition of partially-informed agents. In *Proceedings of the Conference of the Association for the Advancement of Artificial Intelligence (AAAI)*, February 2020.

[Macke *et al.*, 2021] William Macke, Reuth Mirsky, and Peter Stone. Expected value of communication for planning in ad hoc teamwork. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11290–11298, 2021.

[Marcotte *et al.*, 2020] Ryan J Marcotte, Xipeng Wang, Dhanvin Mehta, and Edwin Olson. Optimizing multi-robot communication under bandwidth constraints. *Autonomous Robots*, 44(1):43–55, 2020.

[McDermott *et al.*, 1998] Drew McDermott, Malik Ghallab, Adele Howe, Craig Knoblock, Ashwin Ram, Manuela Veloso, Daniel Weld, and David Wilkins. Pddl-the planning domain definition language. Technical report, Yale Center for Computational Vision and Control, 1998.

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

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

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

[Petrick and Bacchus, 2002] Ronald PA Petrick and Fahiem Bacchus. A knowledge-based approach to planning with incomplete information and sensing. In *ICAPS*, 2002.

[Russell and Norvig, 2016] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Pearson Education, 2016.

[Sarne and Grosz, 2007] 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.

[Shmaryahu *et al.*, 2019] Dorin Shmaryahu, Guy Shani, and Jörg Hoffmann. Comparative criteria for partially observable contingent planning. *Autonomous Agents and Multi-Agent Systems*, 33(5):481–517, 2019.

[Unhelkar and Shah, 2016] 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.

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

[Xuan *et al.*, 2001] 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.

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

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

[Zhang *et al.*, 2009] 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.