When a Robot Reaches Out for Human Help

Ignasi Andrés[1(✉)], Leliane Nunes de Barros[1], Denis D. Mauá[1], and Thiago D. Simão[2]

[1] University of São Paulo, São Paulo, Brazil
{ignasi,leliane,ddm}@ime.usp.br
[2] Delft University of Technology, Delft, The Netherlands
T.DiasSimao@tudelft.nl

Abstract. In many realistic planning situations, any policy has a non-zero probability of reaching a dead-end. In such cases, a popular approach is to plan to maximize the probability of reaching the goal. While this strategy increases the robustness and expected autonomy of the robot, it considers that the robot gives up on the task whenever a dead-end is encountered. In this work, we consider planning for agents that proactively and autonomously resort to human help when an unavoidable dead-end is encountered (the so-called symbiotic agents). To this end, we develop a new class of Goal-Oriented Markov Decision Process that includes a set of human actions that ensures the existence of a proper policy, one that possibly resorts to human help. We discuss two different optimization criteria: *minimizing the probability to use human help* and *minimizing the expected cumulative cost with a finite penalty for using human help for the first time*. We show that for a large enough penalty both criteria are equivalent. We report on experiments with standard probabilistic planning domains for reasonably large problems.

Keywords: Probabilistic planning · Shortest stochastic path
Human-robot collaboration

1 Introduction

The next generation of robots will arguably operate surrounded and in symbiosis with humans. Autonomous robots are expected to perform tasks like driving a car, cleaning the floor, delivering products and caring for elders in the presence and under the guidance of humans. This proximity to humans provides new opportunities for building robots that are robust to failures [13].

Goal-oriented Markov Decision Processes (GMDP) are the standard framework for building mission guided planning robots [3,11,17,18]. The typical strategy in this framework is to maximize the probability of reaching the goal while minimizing the expected cumulative cost. Although this generates policies that

G. R. Simari et al. (Eds.): IBERAMIA 2018, LNAI 11238, pp. 277–289, 2018.
https://doi.org/10.1007/978-3-030-03928-8_23

are robust and cost-effective, it assumes that the robot aborts the mission whenever a dead-end (a state from which the goal cannot be attained) is encountered. An arguably better approach is to reach out for human help.

In complex environments as the ones mentioned above, it is often difficult to model all human help actions available. For example, a domestic robot designed for elder care cannot rely on the elder action of opening the door to the kitchen; nevertheless, it might still be preferable to expect that some human help is available than to simply refuse to grabbing an item from the kitchen.

In this work, we consider the problem of goal-oriented probabilistic planning with unknown human help actions. We propose a generalization of Goal-Oriented Markov Decision Processes (GMDP), called **GMDP augmented with Human Help (GMDP-HH)**, where the robot can infer from the model a distinguished set of human help actions, i.e., it can modify a single fluent (a predicate from the state description) with an uniform cost (Sect. 3). These somehow artificial actions enable robots to plan beyond dead-ends, and allow to introduce human actions into any given GMDP.

In order to increase the robustness of robots, we assume that human help actions are a scarce resource to be used only if necessary. We thus seek for a proper policy (one that reaches the goal with certainty, eventually relying on human actions) but that minimizes the probability of using human help. We call this criterion **MinHProb** (*minimizing the human-help probability*). While appealing, this criterion is difficult to obtain, as the corresponding Bellman equation has multiple non-optimal fixed points, and heuristics for probability estimation are usually inefficient [18].

We then consider an alternative class of decision problems, called **GMDPs with a Penalty on Human Help (GMDP-PHH)**, where a finite penalty is incurred only the first time a human help is used (Sect. 4). This can be seen as modeling a situation where requesting the presence of a human is expensive, but once the human is available subsequent calls for human help are cheap. An optimal policy minimizes then the expected cumulative cost (which includes the penalty for using a human help for the first time); we call this criterion **MinPCost**.

The one-time penalty leads to optimal policies that are non-Markovian. To avoid dealing with such policies, we instead operate over an augmented state space (where states are augmented with a fluent h indicating whether they were reached with the help of human); This allows us to formulate the problem as a standard Stochastic Shortest Path MDP (SSP) making the assumption that there is a proper policy from every state and all improper policies have infinite cost, and employ any of the state-of-the-art SSP solvers.

We connect both classes of problems by proving that, for a large enough penalty, the MinPCost criterion finds policies with *minimum probability of using human help*, that is, which are also optimal under the MinHProb criterion. While there is no known strategy for finding a "large enough" penalty, our empirical results show that it is often possible to efficiently find one by linear search (that

is, by solving MinPCost problems with increasingly large penalty values until the optimal policy converges).

We present experiments with extended versions of three standard planning domains (Doors, Tire World and Navigation) that suggest that solutions for the proposed models can be effectively found (Sect. 5).

2 Notation and Background

A **Goal-Oriented Markov Decision Process** (GMDP) consists of a finite set of states S, an initial state $s_0 \in S$, a set of absorbing goal states $G \subset S$, a finite set of actions A, the probabilistic transition function $T(s, a, s') \in [0, 1]$ that returns the probability of moving from s to s' after executing action a, and a cost function $C(s, a) \in \mathbb{R}_0^+$ that specifies the cost of applying action a in state s. The cost function is assumed to satisfy $C(s, a) = 0$ for all $a \in A$ and $s \in G$, and $C(s, a) > 0$ for all $a \in A, s \notin G$.

A **policy** $\pi : S \to A$ is a mapping from states to actions that prescribes the agent behavior.[1] A **history** $\sigma = \langle s_1, s_2, \ldots, s_{|\sigma|} \rangle$ is a finite sequence of non-goal states ending in a goal state $s_{|\sigma|} \in G$. We say that σ starts at s if $s_1 = s$, and write $\sigma \sim s$. The **probability of reaching the goal** when executing policy π from state s is

$$P_G^\pi(s) = \sum_{\sigma \sim s} \prod_{i=1}^{|\sigma|-1} T(s_i, \pi(s_i), s_{i+1}). \tag{1}$$

We say that a policy π is **proper** for s if $P_G^\pi(s) = 1$. The **expected cumulative cost** of a policy π in state s is

$$V^\pi(s) = \sum_{\sigma \sim s} \prod_{i=1}^{|\sigma|-1} T(s_i, \pi(s_i), s_{i+1}) \sum_{i=1}^{|\sigma|} C(s_i, \pi(s_i)), \tag{2}$$

if π is proper, else $V^\pi(s) = \infty$. We assume that a **Stochastic Shortest Path MDP (SSP)** is a GMDP such that: (Assumption I) there exists at least one proper policy π for any $s \in S$ and (Assumption II) all improper policies have infinite cost. Thus, the minimum expected cumulative cost of a policy in an SSP is the unique fixed-point solution for the following Bellman equations [2,12]:

$$V^*(s) = \begin{cases} 0, & \text{if } s \in G; \\ \min_{a \in A} C(s, a) + \sum_{s' \in S} T(s, a, s') V^*(s'), & \text{otherwise.} \end{cases} \tag{3}$$

The corresponding optimal policy π^* is any greedy policy w.r.t. V^* (i.e., one obtained by applying *argmin* instead of min in the equation above). A Value Iteration (VI) algorithm solves an SSP by applying Eq. 3 from some initialization $V^*(s)$ until a fixed-point is found [12]. To speed up convergence, the values $V^*(s)$ are usually initialized with a heuristic function, and updated asynchronously [4,5,11].

[1] We implicitly assume that every state has at least one applicable action.

A GMDP can be more concisely and conveniently specified using a **planning domain description language**, where states are described in terms of **fluents** that represent properties of the world whose truth value can be modified by the actions. So let F be a finite set of fluents. By making a Closed World Assumption (CWA), we identify any state $s \in S$ with the set of fluents that hold true in that state.

3 Goal-Oriented MDP Augmented with Human Help

For GMDPs with no proper policy, Eq. 3 might not converge, and policies might have unbounded expected cumulative cost and hence be incomparable. While several alternative optimization criteria have been proposed to cope with this issue [11,17,18], none of them have addressed the behavior of an agent that meets a dead-end, or have considered human assistance in the process.

In this work we allow the agent (e.g. a robot) to be equipped with a special set of operations called **human help actions** that can modify the truth value of any fluent at any state, thus ensuring the existence of proper policies from any state $s \in S$.

So let $\mathcal{M} = \langle S, s_0, G, A, T, C \rangle$ be a GMDP described in a planning domain description language with fluents F (so $s \subseteq F$, due to the CWA). To model unknown human help actions we introduce for every fluent $f \in F$ a pair of human help actions a_f and $a_{\neg f}$ that deterministically determine the value of fluent f at an uniform cost $C_H > 0$, that is

$$T(s, a_f, s \cup \{h, f\}) = 1\,, \quad T(s, a_{\neg f}, s \cup \{h\} \setminus \{f\}) = 1\,,$$

and $C(s, a_f, s') = C(s, a_{\neg f}, s') = C_H$, where h is a fluent (not yet in F) indicating that human help was used. Note that this definition is equivalent to allowing non-atomic human actions (that modify several fluents at once) at a cost proportional to the number of fluents they modify. To allow for Markovian policies while distinguishing the use of human help, we augment the state space so that for every state $s \in S$ there is a state $s_h = s \cup \{h\}$ representing that s was reached using some human help action in the past (while s now represents that it was reached without human help), and modify the transition function accordingly (i.e., we set $T(s, a, s') = 0$ if $h \in s$ and $h \notin s'$). Call S_H the set of all states reached with human help. We also distinguish goal states reached through human help as $G_H = \{s \cup \{h\} : s \in G\}$. We call the tuple $\mathcal{M}_{HH} = \langle S \cup S_H, s_0, G \cup G_H, A \cup A_H, T, C, C_H \rangle$ a **GMDP augmented with human help (GMDP-HH)**, where T and C are extended to account for human actions.

Now, we can decompose the expected cumulative cost of any policy π as the sum of expected cumulative cost of the robot actions $V_{\pi,R}(s)$ and the expected cumulative cost of the human actions $V_{\pi,H}(s), \forall s \in S \cup S_H$:

$$V^\pi(s) = V_R^\pi(s) + V_H^\pi(s), \tag{4}$$

where

$$V_R^\pi(s) = \sum_{\sigma \sim s} \prod_{i=1}^{|\sigma|-1} T(s_i, \pi(s_i), s_{i+1}) \sum_{i=1}^{|\sigma|} \Big\{ C(s_i, \pi(s_i)) : \pi(s_i) \in A \Big\} \tag{5}$$

and

$$V_H^\pi(s) = \sum_{\sigma \sim s} \prod_{i=1}^{|\sigma|-1} T(s_i, \pi(s_i), s_{i+1}) \sum_{i=1}^{|\sigma|} \Big\{ C(s_i, \pi(s_i)) : \pi(s_i) \in A_H \Big\}. \tag{6}$$

The human actions allow any goal state to be reached from any state with certainty. At the same time, improper policies remain having infinite expected costs. Hence:

Theorem 1. *A Goal-Oriented Markov Decision Process augmented with human help actions is an* SSP *with Assumption I and II.*

Our definition of GMDP-HH might lead to trivial solutions in domains where the goals have a distinguished fluent that can be modified from any state by the human. In order to avoid such trivial solutions, we remove human actions that modify fluents appearing in the goal, whenever this does not remove the existence of proper policies. This preprocessing step can be accomplished efficiently by analyzing the *causal graph* [8] of a determinized version of a planning domain description. We omit the details of this transformation due to space limitation.

3.1 Minimizing the Probability of Human Help (MinHProb)

Given that the human help is a costly resource, an intuitive criterion for solving a GMDP-HH is to find a proper policy that minimizes the probability of using human help. To this end, we define the **probability of reaching a goal using human help** when executing a policy π as:

$$P_{G_H}^\pi(s) = \sum_{\sigma \sim s : s_{|\sigma|} \in G_H} \prod_{i=1}^{|\sigma|-1} T(s_i, \pi(s_i), s_{i+1}), \tag{7}$$

where the sum is over all histories that end up in some $s_{|\sigma|} \in G_H$. The optimal policy under the **minimum human help probability** criterion, called **MinHProb**, is π_{MinHProb} such that:

$$P_{G_H}^{\pi_{\text{MinHProb}}}(s_0) = \min_\pi P_{G_H}^\pi(s_0) \text{ subject to } P_{G \cup G_H}^\pi(s_0) = 1. \tag{8}$$

In words, the optimal policy is a proper policy that minimizes the probability of using human help. The requirement of being a proper policy is necessary to avoid improper policies that e.g. do not use human help (hence have probability zero of reaching the goal with human help). This criterion has the following interesting properties:

Proposition 1. *If the original* GMDP $\mathcal{M}$ *has a proper policy* π^* *for* s_0 *then* $\pi^* \in \arg\min_\pi P^\pi_{G_H}(s_0)$. *Conversely, if* $P^{\pi_{MinHProb}}_{G_H}(s_0) = 0$ *then the original* MDP *has a proper policy* $\pi_{MinHProb}$ *for* s_0.

Proof. Note that $P^\pi_{G \cup G_H}(s_0) = P^\pi_G(s_0) + P^\pi_{G_H}(s_0) = 1$. Hence, a proper policy π for s_0 in the original GMDP $\mathcal{M}$ satisfies $P^G_\pi(s_0) = 1$, which implies that $P^\pi_{G_H}(s_0) = 0$. Conversely, any policy π with $P^\pi_{G_H}(s_0) = 0$ must satisfy $P^\pi_G(s_0) = 1$, and hence be proper for s_0 in the original problem. □

According to the proposition above, the MinHProb criterion finds a policy that uses human help only if necessary, that is, only when the robot finds itself in a dead-end.

3.2 Bellman Equation for MinHProb

One can show that $P^*_{G_H} = P^{\pi_{\text{MinHProb}}}_{G_H}$ is a fixed-point of the following Bellman equation:

$$P^*_{G_H}(s) = \begin{cases} 0, & \text{if } s \in G; \\ 1, & \text{if } s \in G_H; \\ \min\limits_{a \in A} \sum\limits_{s' \in S \cup S_H} T(s,a,s') P^*_{G_H}(s'), & \text{otherwise.} \end{cases} \tag{9}$$

However, not every fixed-point of the Eq. (9) is equal to $P^{\pi_{\text{MinHProb}}}_{G_H}$. To see this, consider the GMDP-HH in Fig. 1 (left) for which $P^{\pi_{\text{MinHProb}}}_{G_H}(x) = 0.5$ and $P^{\pi_{\text{MinHProb}}}_{G_H}(y) = P^{\pi_{\text{MinHProb}}}_{G_H}(y,z) = 1$. Any solution such that $P^*_{G_H}(y) = P^*_{G_H}(y,z) < 1$ is also a fixed-point.

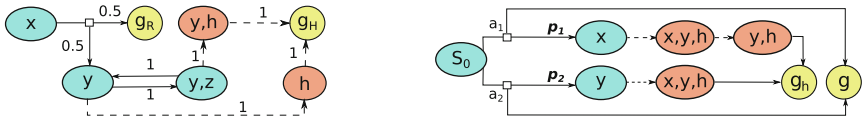

Fig. 1. Examples of GMDP-HH with fluents $F = \{x, y, z, h\}$: nodes, solid edges and dotted edges represent, resp., states, agent actions and human help actions; the numbers denote transition probability; g_H (resp., g_R) is the goal reached using (not using) human actions. Left: a GMDP-HH with multiple fixed-point solutions. Right: two actions are applicable at s_0, resulting in four different histories starting at s_0, all of them with the same $P^{\pi_{\text{MinHProb}}}_{G_H}(s_0)$.

As usual we can apply Value Iteration based algorithms to solve Eq. 9. However, since this equation has multiple fixed-points not every initialization leads to an optimal fixed-point. In particular, admissible heuristics for $P^*_{G_H}$ do not ensure convergence and hence cannot be used. One possible solution is to adapt algorithms for SSPs such as FRET and FRET-π that find and remove problematic

cycles [10,16], ensuring convergence from any initialization. Another possible approach is to use linear programming reformulations of the problem as in [18].

Another issue with the MinHProb is that two policies might achieve the same probability $P^{\pi}_{G_H}(s_0)$ while executing a very different number of human actions. For example, take the GMDP-HH in Fig. 1 (right), and assume that $p_1 = p_2 = p$. Then, selecting either action at s_0 leads to an optimal policy π_{MinHProb} with $P^{\pi_{\text{MinHProb}}}_{G_H}(s_0) = p$, while executing a different number of human actions (and obtaining different cumulative costs). Situations like these can be remedied by additionally minimizing the expected cumulative cost among policies π_{MinHProb}, that is, by adopting a lexicographic criterion that first minimizes $P^{\pi}_{G_h}(s_0)$ then minimizes $V^{\pi}(s_0)$. We show in the next Section how this two-step criterion can be more efficiently computed using a surrogate criterion that introduces a finite penalty on the first time a human action is used.

4 Goal-Oriented MDP with a Penalty on Human Help

An alternative criterion to find a policy that minimizes human help is to minimize the expected cumulative cost while severely penalizing any history that uses a human help. Intuitively, this criterion assumes that the cost of human help is amortized if used repeatedly. This is a realistic scenario when there is a high cost of requesting human presence, but a small cost for actually using human help. Thus, we define the **Goal-Oriented MDP with a Penalty on Human Help (GMDP-PHH)** as the tuple $M_{HP} = \langle S \cup S_H, s_0, G \cup G_H, A \cup A_H, T, C, D_H \rangle$, where all terms are defined as in a GMDP-HH, and $D_H > 0$ is a finite value denoting the penalty incurred the first time a human action is used.

4.1 Minimizing Expected Cumulative Cost with a Penalty on Human Action

Solving a GMDP-PHH is akin to solving GMDPs with a give-up action that takes the agent from any state directly into a goal state and incurs a (usually large) finite penalty [11]. Conceptually however a GMPDP-PHH differs from a GMPD with a give-up action (a.k.a. fSSPUDE) since in the former the agent resumes planning after paying the penalty D_H.

We can solve a GMDP-PHH efficiently by using any off-the-shelf SSP solver by modifying the cost function $\mathcal{C}(s, a)$ so that it returns $C_H + D_H$ if $s \in S$ and $a \in A_H$, and otherwise remains unchanged. We call this criterion of minimizing the expected cumulative cost increased with the penalty D_H the **MinPCost** criterion. It is easy to prove that a GMDP-PHH with this modified cost function is still an SSP.

4.2 MinHProb Versus MinPCost

Theorem 2 shows that one can use MinPCost as a solution to MinHProb.

Theorem 2. *There exists a value $D_{MinHProb}$ such that for all $D_H > D_{MinHProb}$ a $\pi_{MinPCost}$ policy with D_H is also a $\pi_{MinHProb}$ policy. Additionally, any $\pi_{MinPCost}$ policy with D_H minimizes the unpenalized expected cumulative cost among all $\pi_{MinHProb}$ policies (i.e., it optimizes the two-step criterion),*

Proof. Given a policy π, we can decompose $V^\pi(s)$ as the sum of expected cumulative costs of robot actions, human actions and the one-time penalty:

$$V^\pi(s) = V_R^\pi(s) + V_H^\pi(s) + P_{G_H}^\pi(s) \cdot D_H, \tag{10}$$

where $P_{G_H}^\pi(s)$ is given by Eq. 7. For large enough D_H, a policy that uses a human help action in a given state has a higher expected cumulative cost than a policy that differs only by the choice of agent action in that same state. Hence, an optimal policy will use a human action only if no agent action can lead the agent out of a dead-end. The same argument shows that MinPCost breaks ties by selecting a policy that minimizes the expected cost of robot and human actions, thus satisfying the lexicographic criterion. □

According to Theorem 2, for large enough D_H the optimal policy π_{MinPCost} also optimizes MinHProb while minimizing the unpenalized expected cumulative cost, that is, $P_{G_H}^{\pi_{\text{MinPCost}}}(s_0) = P_{G_H}^{\pi_{\text{MinHProb}}}(s_0)$ and π_{MinPCost} minimizes $V_R^\pi(s) + V_H^\pi(s)$. However, there is no known procedure for finding the value D_{MinHProb} or even for verifying if a given value satisfies the condition on the Theorem 2. In our experiments we observed that by guessing a sufficiently large value D_H and verifying whether increasing this value changes the optimal policy provides an effective means for finding D_{MinHProb} in practice.

5 Empirical Analysis

We performed experiments with the objective to: (i) *analyze the soundness and performance time of solving* GMDP-HH *problems under the MinPCost criterion using state-of-the-art* SSP *planners*, and (ii) **investigate the effectiveness of finding the $\pi_{MinHProb}$ by solving GMDP-HH problems under the MinPCost criterion with increasingly large penalties**. Our tests show that directly solving GMDP-HH problems under the MinHProb criterion using state-of-the-art SSP planners was highly inefficient in nearly all instances; for this reason we omit this analysis here.

We find optimal policies under the *MinPCost* criterion using a modified version of the LRTDP algorithm [4] implemented on the MGPT Framework [6]. This modification was done to deal with the augmented state space and includes a function to verify if a state s satisfies the fluent h. All experiments were performed in a Linux machine with a 2.4 GHz processor and 213 GB RAM, with a time limit of 1 h per instance.

To perform our tests, we considered several instances of the following modified versions of three standard planning domains:

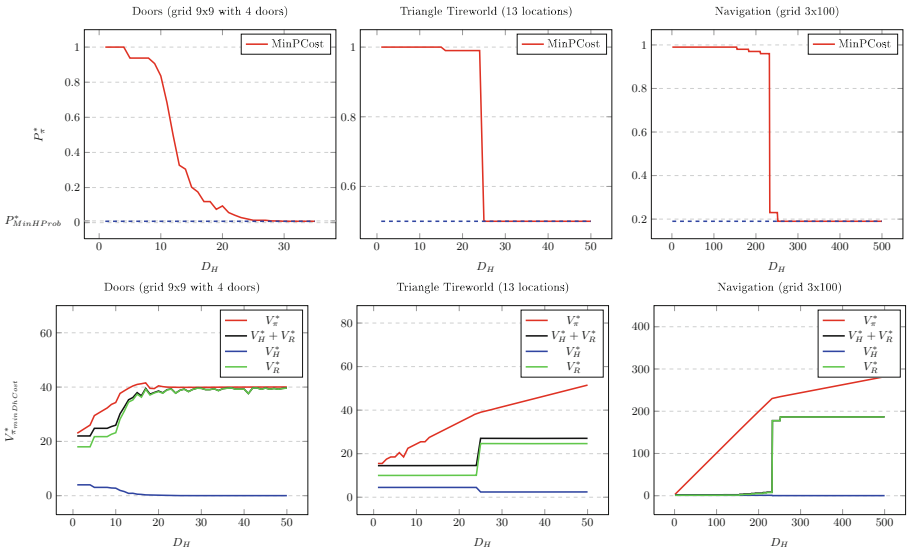

Fig. 2. Characteristics of the optimal policy for three large instances of the tested domains. Top: $P^*(s_0) = P^{\pi_{\text{MinPCost}}}_{G_H}(s_0)$ for increasing values of the penalty D_H; $P^*(s_0) = P_{MinHProb}(s_0)$ is the minimal probability. Bottom: $V_{MinPcost}(s_0)$, $V^*_R(s_0)$, $V^*_H(s_0)$ and $V^*_R(s_0) + V^*_H(s_0)$ for increasing values of the penalty D_H.

- **Doors**, where a robot must navigate in a grid world to reach a goal location, while passing through a sequence of locked doors; and for each door the robot needs to find the key in order to open it; in each step there is a 0.5 probability of finding the key to the next door in the current cell; alternatively, the robot can ask for a human to open the door;
- **Navigation**, where a robot navigates in a grid world to reach a goal location; in every cell there is a certain probability that the robot gets stuck (and thus reaches a dead-end). A human can take the robot to any location (other than the goal) and thus escape dead-ends; and
- **Triangle Tireworld**, where a car moves through connected locations and has to reach a goal location; in each movement between locations the car can have a flat tire with non-zero probability; some locations contain a spare; the agent would be in a dead-end if it has a flat tire and no spare; a human can deliver a spare tire or take the car to any location.

As discussed in the introduction, the large number of human actions leads to a large branching factor in the search, which makes heuristic search less efficient. To overcome this issue, we select a subset $A'_H \subseteq A_H$ involving only the set of *relevant* fluents [7,8], that is, the fluents that are relevant to lead the agent to the goal which was automatically extracted from the domains description in PDDL [19]. For the largest Triangle Tireworld instance, from a total of 92 fluents, we only used 46 relevant fluents to create the set of human help actions; for the largest Navigation instance, from a total of 309 fluents, only 154 were

the relevant fluents; for the largest Doors instance, from the total of 417 fluents, we only consider 146 relevant fluents.

Table 1. Optimal values and exec. time in secs for a given D_H.

Problem instance	D_H	$P_{G_H}^{\pi_{MinPCost}}(s_0)$	$V^{\pi_{MinPCost}}(s_0)$	Time (sec)
Doors-7	30	0.03	31.6	3.4
Doors-9	30	0.01	39.4	5.4
Triangle Tireworld-4	50	0.50	49.6	0.6
Triangle Tireworld-5	50	0.50	57.6	3.0
Triangle Tireworld-6	70	0.50	74.0	65.4
Triangle Tireworld-7	90	0.50	90.5	757.0
Navigation 3×103	500	0.19	321.3	15.4
Navigation 4×103	500	0.27	381.9	19.5
Navigation 5×103	500	0.34	435.7	32.7

Solving* GMDP-HH *Problems Under the MinPCost Criterion. Table 1 shows the values of $P_{G_H}^{\pi_{MinPCost}}(s_0)$ and $V_{G_H}^{\pi_{MinPCost}}(s_0)$ for large enough values of D_H that allow convergence to MinHProb policies; and time in seconds for finding the optimal policies for 9 instances of the tested domains. For each instance, the $P_{G_H}^{\pi_{MinPCost}}(s_0)$ and $V_{G_H}^{\pi_{MinPCost}}(s_0)$ values were confirmed to be equal to the analytically computed value proving the soundness of our solution. We also see from Table 1, that for all Doors instances and the two small Triangle Tireworld instances, the optimal solution was found in few seconds; while for the Navigation domain and the larger instances of the Triangle Tireworld domain the time was one order of magnitude larger, with the exceptions being the largest Triangle Tireworld instance, which are considerably larger than the other instances.

***Finding* $P^*(s_0)$ *with Increasingly Large Penalties* D_H.** Figure 2 shows the results of our experiments using the *MinPCost* criterion on large instances of selected domains (a 9×9 grid with 4 doors for the Doors, a triangle with 11 locations at each side for the Triangle Tireworld, and a 3×100 grid for Navigation). In all domains the probability of using human actions to reach the goal from s_0 decreases as the penalty increases until it reaches the minimal probability $P_{G_H}^*(s_0)$ (analytically computed as 0.0078 for Doors, 0.5 for Triangle Tireworld and 0.19 for Navigation). This decreasing is smoother for the Doors instance (as the probability of using human help is very small) and somewhat abrupt for the other two domains, showing that the optimal policies are very sensitive to the penalty value. We also see that as predicted, the optimal expected cumulative cost $V^*(s_0)$ increases as the penalty D_H increases, with an inflection point near the steepest descent of the probability (note however that the cost $V_\pi^*(s_0)$ still grows linear with D_H even after the policy has converged). We also see a similar behavior to the probability $P_{G_H}^\pi(s_0)$ in the factors V_{H*} and V_{R*}, that is, they

have a clear inflection point when the policy converges to the MinHProb, and eventually converge to their values (again, this change is smoother for Doors and more abrupt for the other domains). These experiment suggest that a reasonable value for the penalty can often be found with some experimentation and analysis of the domain which proves that this is a reliable solutions to find an optimal policy for GMDP-HHs problems under the criteria proposed in this paper.

6 Related Work

Most of the work on human-robot interaction is based on POMDPs (*Partially Observable Markov Decision Process*) [1,9,14,15], augmented with a set of given human observations and actions, with negative reward and whose objective is to find a policy that maximizes the expected reward over a given horizon, not explicitly treating goal and dead-end states. In this work we consider GMDP with fully observability and the presence of dead-ends.

In all previous approaches, the human observations are known a priori, while in our work we automatically generate human actions from the GMDP problem description. The goal of this work is to maximize agent autonomy in domains where assessing the cost and specially the type of human intervention is difficult, costly or simply undesirable.

7 Conclusions

Algorithms that solve GMDPs assume that when an agent encounters a dead-end its only action is abort the mission. However, robots operating in the presence and under the guidance of humans can often reach out for help in order to resume its mission. Still, in many complex environments, it is unrealistic to assume that the available human help actions are known a priori.

In this work we develop two new classes of Goal-Oriented Markov Decision Processes that allow for planning in uncertain environments and with unknown human actions. The first class, called goal-oriented Markov Decision Problem augmented with Human Help (GMDP-HH), assumes that human actions can modify the state of any fluent, and thus ensure that a goal is reached from any state. To avoid trivializing the problem, we then seek for the optimal policy that reaches the goal with certainty while minimizing the probability of using human help. While this criterion is appealing as it uses human help only if necessary, it leads to inefficient optimization problems.

Our second class of problems, called Goal-Oriented Markov Decision Problems with a Penalty on Human Help (GMDP-PHH), assumes that an additional finite penalty is incurred the first time a human action is used. An optimal policy simply minimizes the expected cumulative cost (including the finite penalty) and can take advantage of standard solvers. Importantly, we show that for a large enough penalty, the optimal policy also minimizes the probability of using human help, thus providing an efficient solution to the first class of problems but also guaranteeing minimal costs.

The atomic human actions that we considered in this work can be interpreted as possible explanations for a mission failure in a standard Goal-Oriented Markov Decision Process, as in [7]; it also can be used to provide some guidance in modifying the domain so as to ensure that the goal is always met (i.e., to transform the problem into an Stochastic Shortest Path MDP).

An open question is how to find the minimum value of the finite penalty that ensures that the probability of reaching the goal using human help is minimized.

As future work we intend to compute the minimum human help probability by adapting the algorithm FRET for the *MinHProb* criterion [10] and using linear programming reformulations of a GMDP-HH problem as in [18].

Acknowledgments. Authors received financial support from CAPES, FAPESP (grants #2015/01587-0 and #2016/01055-1) and CNPq (grants #303920/2016-5 and #420669/2016-7).

References

1. Armstrong Crews, N., Veloso, M.: Oracular partially observable markov decision processes: a very special case. In: Proceedings of the IEEE ICRA (2007)
2. Bellman, R.: Dynamic Programming. Princeton University Press, Princeton (1957)
3. Bertsekas, D.P., Tsitsiklis, J.N.: An analysis of stochastic shortest path problems. Math. Oper. Res. **16**(3), 580–595 (1991). INFORMS
4. Bonet, B.: Labeled RTDP: improving the convergence of real-time dynamic programming. In: Proceedings ICAPS-03 (2003)
5. Bonet, B., Geffner, H.: Faster heuristic search algorithms for planning with uncertainty and full feedback. In: Proceedings of the IJCAI (2003)
6. Bonet, B., Geffner, H.: mGPT: a probabilistic planner based on heuristic search. J. Artif. Intell. Res. **24**, 933–944 (2005)
7. Göbelbecker, M., Keller, T., Eyerich, P., Brenner, M., Nebel, B.: Coming up with good excuses: what to do when no plan can be found. In: ICAPS (2010)
8. Helmert, M.: The fast downward planning system. J. Artif. Intell. Res. **26**, 191–246 (2006)
9. Karami, A.B., Jeanpierre, L., Mouaddib, A.I.: Partially observable markov decision process for managing robot collaboration with human. In: Proceedings of the 21st IEEE ICTAI (2009)
10. Kolobov, A., Daniel, M., Weld, S., Geffner, H.: Heuristic search for generalized stochastic shortest path MDPs. In: Proceedings of the ICAPS (2011)
11. Kolobov, A., Mausam, M., Weld, D.: A theory of goal-oriented MDPs with dead ends. In: Proceedings of the 28th Conference on UAI (2012)
12. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley (2014)
13. Rosenthal, S., Biswas, J., Veloso, M.: An effective personal mobile robot agent through symbiotic human-robot interaction. In: Proceedings of the AAMAS (2010)
14. Rosenthal, S., Veloso, M., Dey, A.K.: Learning accuracy and availability of humans who help mobile robots. In: Proceedings of the AAAI (2011)
15. Schmidt-Rohr, S.R., Knoop, S., Lösch, M., Dillmann, R.: Reasoning for a multimodal service robot considering uncertainty in human-robot interaction. In: Proceedings of the 3rd HRI (2008)

16. Steinmetz, M., Hoffmann, J., Buffet, O.: Revisiting goal probability analysis in probabilistic planning. In: Proceedings of the 26th ICAPS (2016)
17. Teichteil-Königsbuch, F.: Stochastic safest and shortest path problems. In: Proceedings of the NCAI (2012)
18. Trevizan, F., Teichteil-Königsbuch, F., Thiébaux, S.: Efficient solutions for stochastic shortest path problems with dead ends. In: Proceedings of 33rd Conference on UAI (2017)
19. Younes, H.L., Littman, M.L.: PPDDL1.0: an extension to PDDL for expressing planning domains with probabilistic effects. Technical report CMU-CS-04-162 (2004)