Hybrid Goal Selection and Planning in a Goal Reasoning Agent Using Partially Specified Preferences

Michael W. Floyd[1], Mark Roberts[2], and David W. Aha[2]

[1]Knexus Research Corporation; Springfield, Virginia; USA
[2]Navy Center for Applied Research in AI; Naval Research Laboratory (Code 5514); Washington, DC; USA
michael.floyd@knexusresearch.com | mark.roberts@nrl.navy.mil | david.aha@nrl.navy.mil

Abstract

Goal Reasoning agents are not restricted to pursuing a static set of predefined goals but can instead reason about their goals and, if necessary, dynamically modify the set of goals that they will pursue. For a solitary agent, goal selection is guided by the agent's own internal motivations. However, an agent that is a member of a team also needs to consider its teammates' preferences when selecting goals. In this work, we propose an online approach to estimate the utility of goals based on a supervisor's partially specified preferences. Estimated goal utilities are used during a hybrid goal selection and planning process to select a subset of goals for the agent to pursue. We report evidence from an empirical study which demonstrates that our approach outperforms several baselines in scenarios drawn from a simulated human-agent teaming domain.

1. Introduction

Goal Reasoning (GR) agents are able to dynamically reason about their goals and modify them in response to unexpected events or opportunities (Aha, Cox and Muñoz-Avila 2013). Compared to traditional agents that use static, pre-defined goals, GR agents have the ability to autonomously respond to uncertain environments and changing conditions (e.g., conditions that make their previous goals unsuitable or unachievable). Although GR agents may operate largely autonomously, in practice they have at least some interaction with collaborators or teammates and need to consider the preferences of those actors when selecting goals to pursue or generating plans. However, the agent may only have a partial understanding of its teammates' preferences (e.g., because the teammates do not have time to fully specify their preferences) and will need to estimate whether its potential goals align with those preferences. Additionally, while GR agents have the ability to dynamically modify their goals, many existing agents only pursue a single goal at a time. This may be acceptable for simple scenarios, but we argue that complex, long-term scenarios may require pursuing multiple goals concurrently. For agents that do pursue multiple concurrent goals, using separate processes for selecting goals (i.e., based on teammates' preferences) and planning may be insufficient as they do not consider the interdependencies among goals when computing their expected utility or achievability.

In this paper, we describe a hybrid goal selection and planning approach for GR agents. Our approach allows for agents that are members of human-agent teams to use the partially specified preferences of their teammates to estimate the utility of goals and guide goal selection. Our work focuses on an agent that operates under the *Single Supervisor* human-agent teaming model (Molineaux et al. 2018). Under this model, a team is composed of a single human, the supervisor, and a single agent, the supervisee. Although the supervisee is able to formulate its own goals, some of its goals may come directly from being tasked by the supervisor.

This paper makes three primary contributions. First, it presents a GR agent that can commit to multiple goals concurrently. Many existing GR agents only commit to a single goal at a time (i.e., the previous goal is suspended or abandoned). Second, our GR agent is able to use the partially specified preferences of its supervisor during its reasoning process. Existing work only allows for the use of fully specified preferences. Third, we demonstrate the use of Partial Satisfaction Planning (PSP) by a Goal Reasoning agent and, to the best of our knowledge, the first use of PSP with partially specified preferences and online goal utility estimation.

We present our hybrid goal selection and planning approach in Section 2, and describe how a supervisor's partially specified preferences are used to estimate the utility of goals. In Section 3, our approach is empirically evaluated in a simulated human-agent teaming domain. Section 4 examines related work, and we conclude with a discussion of future work in Section 5.

2. Hybrid Goal Selection and Planning

A Goal Reasoning agent does not rely on a predefined set of static goals but can instead reason over and dynamically modify its goals (i.e., those that it will pursue). The agent maintains a set G_s of selected goals that represent the goals it is currently attempting to achieve ($G_s \subseteq G$, where G is the set of all goals). During agent plan execution, at any time the agent can add a goal $g' \in G$ ($G_s \leftarrow G_s \cup g'$), remove a goal ($G_s \leftarrow G_s \setminus g'$), or abandon all previous goals and commit to only a single goal ($G_s \leftarrow g'$). The decision to modify its goals may result from an unexpected event occurrence (e.g., damaged hardware, the appearance of a hostile agent), an opportunistic situation (e.g., discovering the existence of a cache of resources), or further deliberation (e.g., determining that the expected benefit of achieving a goal differs from an initial estimate).

We assume that each goal has a *type* that defines higher-level properties of the goal ($type: G \rightarrow T$, where T is the set of all types). A type $\tau \in T$ contains a label l and the expected influence on each of the n factors of interest $f_1, \dots, f_n$:

$$\tau = \langle l, f_1, \dots, f_n \rangle$$

These factors represent quantifiable properties that may be important to the agent or its teammates (i.e., may potentially influence how the success of the agent is measured). Although the expected influence on each factor can be defined as precisely as required, we use a coarse representation that encodes only whether a factor is expected to be positively impacted, negatively impacted, or unaffected by achieving a goal of that type ($f_i \in \{-1,0,1\}$), thereby reducing the knowledge engineering required to define goal types. For example, consider a ground goal g' which has a type τ_{ic} that represents *information collection*. If there are three factors of interest, *time*, *knowledge*, and *safety*, then information collection goals can be encoded to negatively influence time (i.e., time needs to be spent collecting information), positively influence knowledge, and have no impact on safety ($\tau_{ic} = \langle "information\ collection", -1, 1, 0 \rangle$). Thus, before committing to an *information collection* goal, the agent can use the goal's type to estimate the potential impact of achieving that goal. It should be emphasized that due to environment uncertainty, the information contained in the goal's type is not a guarantee of the true influence on the factors of interest. Returning to the *information collection* example, it is possible that while collecting information the agent would accidentally damage itself, thereby actually having a negative influence on safety (i.e., an influence of -1 rather than the expected influence of 0). However, the goal's type is assumed to contain reasonable assumptions about how committing to the goal will impact the factors of influence, in general.

In addition to its type, each goal is either *hard* or *soft* ($hardness: G \rightarrow \{hard, soft\}$). Hard goals must be achieved by the agent, whereas soft goals may be achieved but are not required to be. In the *Single Supervisor* teaming model, hard goals include orders from the supervisor (e.g., "*Investigate the fire*") or self-preservation goals of the agent (e.g., *"Don't drive off the cliff"*). Soft goals could include optional events that occur during a mission such as helping a vehicle in distress, investigating a potential cache of resources, or interacting with another agent.

2.1 Teammate Preferences

A fully autonomous agent can select goals to pursue based on its own internal preferences. For example, these could include preferences that are hard-coded by the agent's designer or preferences that adapt over time. However, an agent that is a member of a team should also take into account the preferences of its teammates. We define an actor's preferences $pref$ to be the weights $w_1, \dots, w_n$ ($w_1, \dots, w_n \in \mathbb{R} \mid 0 \leq w_1, \dots, w_n \leq 1$) the actor places on each of the n factors of interest ($pref = \langle w_1, \dots, w_n \rangle$).

These weights, which represent an actor's *true preferences*, are internal to the actor and may not be fully available to other actors. For example, while a supervisor will have preferences over each of the factors of interest, the agent will not have complete knowledge of those preferences unless the supervisor fully specifies them to the agent and updates the agent when the preferences change. In a human-agent teaming context, there are a number of reasons why the supervisor's true preferences would not be fully known by the agent:

- *Only providing relevant preferences*: The supervisor may only provide the subset of its preferences that it believes are relevant in the current context. For example, if the supervisor does not foresee any potential for injury, it may provide preferences for *time* and *knowledge* but omit *safety.*
- *Inexact preferences*: The supervisor may provide inexact preferences rather than giving its precise preference values. For example, the supervisor may tell the agent to *"drive slowly"* rather than giving its exact preference that the agent keeps its speed under 40 km/h.
- *Unknown preferences*: The supervisor may not be aware of a particular factor or never have taken the time to generate a preference. For example, if the supervisor was unaware that there was relevant information in the environment, it may not have a preference for whether the agent collects such information. However, once it is made aware of the previously unknown factor it may quickly generate a preference for it (e.g., it would have preferred if the agent would have collected information).
- *Adversarial supervisor*: The supervisor could intentionally omit preferences or provide incorrect

values. For example, if the supervisor and agent were engaged in negotiations, the supervisor may choose to hide information from the agent. Similarly, if the agent was a new member of the team the supervisor may not have adequate trust in the agent to share its complete preferences.

Since it may be impractical to have the supervisor's true preferences, the agent will instead have access to the *provided preferences* $pref^*$ that contain the provided values $p_1, \dots, p_n$ for each of the true preference weights ($pref^* = \langle p_1, \dots, p_n \rangle$). We assume that the actor providing the preferences uses a function $provide$ to map its true preference weights to provided preferences ($provide: W \rightarrow P$, where W is the set of all weights and P is the set of all provided preferences). However, unlike the actor's true preferences, the provided preferences may also be unspecified ($p_1, \dots, p_n \in \mathbb{R} \cup \{\text{"unknown"}\}$). The $provide$ function can be thought of as the decision making process by which the actor selects what information to share with the agent (i.e., what subset of preferences to provide) and how to format that information (e.g., discretize or obfuscate the preference values).

2.2 Goal Utility Estimation

When a member of a team considers whether to pursue a goal, it needs consider the potential impact of the goal as well as the preferences of its team. In the general case where the team is composed of the agent and m teammates, the utility of goal g' is a function of the expected influence on each factor of interest $f_1, \dots, f_n$, the true preferences of the agent $pref_{self}$, and the provided preferences $pref_1^*, \dots, pref_m^*$ of each teammate ($utility(f_1, \dots, f_n, pref_{self}, pref_1^*, \dots, pref_m^*) \rightarrow \mathbb{R}$). In practice, the specific method for calculating utility will depend on the composition and organization of the team. For example, an agent that is a team leader would put more weight on its own preferences, an agent with a single team leader would put more weight on that teammate's preferences, or an agent with teammates that are peers may weigh all teammates' preferences equally.

In this work, since we are focused on the *Single Supervisor* team formation, we consider the specific case of goal utility estimation where the agent is attempting to satisfy its supervisor. Thus, the utility of a goal is a function of only the expected influence on each factor of interest and the supervisor's provided preferences ($utility(f_1, \dots, f_n, pref_1^*) \rightarrow \mathbb{R}$). A simple form of the utility estimation is a linear combination of the expected influence the goal will have on each factor of interest weighted by the provided preference for that factor:

$$utility = C_1 \times f_1 \times p_1 + \cdots + C_n \times f_n \times p_n$$

where $C_1, \dots, C_n$ are constant values, and a preprocessing step can be used to account for any unspecified preferences (e.g., ignoring that factor of interest, giving unspecified preferences a fixed value). This assumes that the agent is motivated to align its behavior with the preferences of its supervisor and therefore uses the supervisor's preferences as its own. However, such an assumption may not hold true for rebellious agents (Coman, Gillespie and Muñoz-Avila 2015).

2.3 Goal Selection and Planning

Most Goal Reasoning agents treat *goal selection* and *planning* as two separate processes, with the output of goal selection (i.e., the subset of goals the agent will attempt to achieve) being used during planning (i.e., to find a sequence of actions that are expected to achieve the selected goals). For example, the goal selection process might estimate the expected utility of each goal and then select a subset of goals with the highest utility. However, a primary limitation of separating goal selection and planning is that it may be difficult to know whether a subset of goals can be achieved together (or even if a single goal is achievable). It may not be until the agent attempts to generate a plan to achieve the goals that it realizes that some goals cannot be achieved together (e.g., they are logically conflicting, there are insufficient resources to complete both, they both require using a single-use action). Since most existing Goal Reasoning agents select and plan with only a single goal (i.e., the goal with the highest utility is selected and replaces the previous goal), separating goal selection and planning has not proven to be an issue. However, we argue that Goal Reasoning agents which operate in more complex environments will be required to commit to multiple concurrent goals. To allow for this, we propose a hybrid goal selection and planning process for a Goal Reasoning agent based on a teammate's partially specified preferences.

We use Partial Satisfaction Planning (PSP), as it allows for the generation of plans that achieve only a subset of the specified goals (van den Briel et al. 2004; Benton, Do, and Kambhampati 2009). PSP planners evaluate the quality of a plan based on its *net benefit*. The net benefit of a plan is the difference between the *utility* of all goals achieved by the plan and the *cost* of taking the actions in the plan. Additionally, PSP planners allow for both hard goals (i.e., a valid plan must achieve that goal) and soft goals (i.e., a plan may not achieve the goal yet still be considered valid). Thus, if a PSP planner can generate a valid plan, that plan will achieve all hard goals and a subset of soft goals (i.e., those soft goals for which their utility outweighs the cost of achieving them).

Our proposed use of PSP differs in a key way from traditional PSP. We do not assume that the utility of each goal is known in advance (e.g., using a fixed goal utility

function) but instead assume it must be estimated by the agent based on the partially specified preferences of its supervisor. These preferences can be teammate-dependent, mission-dependent, or time-dependent, so it may not be possible to know the utility of all goals in advance. For example, two different supervisors may have different preferences. Similarly, the same supervisor may have different preferences depending on the importance of the mission or how much experience it has in the mission type.

Our hybrid goal selection and planning approach operates using the following cycle:

1. **Receive Goal**: A new goal g' is received by the agent and will be considered. Goals can be received from either *external* sources or *internal* sources. External sources would include the supervisor providing the agent with g'. Internal sources would self-generate new goals for the agent to achieve in response to external events or opportunities.
2. **Determine Hardness**: The agent determines whether g' is a hard or soft goal. In practice, we consider goals provided by the agent's supervisor to be hard goals (unless the supervisor specifies they are optional), whereas goals the agent provides itself are soft goals. This is primarily based on the supervisory relationship of the team, with the assumption that goals provided by the supervisor are more important than goals generated by the agent.
3. **Estimate Goal Utility**: The utility of g' is estimated using the goal's type and the agent's knowledge of its supervisor's preferences (i.e., $pref_1^*$).
4. **Add Goal**: If g' is a hard goal (from Step 2) or a goal with an estimated utility greater than zero (from Step 3), it is added to the set of selected goals: $G_s \leftarrow G_s \cup g'$. Otherwise, g' is ignored (i.e., it is a soft goal with a zero or negative estimated utility).
5. **Plan**: If g' was added to G_s (from Step 4), PSP is used to generate a plan π to achieve G_s. No replanning is necessary if g' was ignored, since the active plan is assumed to be valid.
6. **Act**: Perform the actions in π (from Step 5). During this step, if any noteworthy events occur that cause the agent to generate a new goal, or a new goal is provided by the supervisor, the agent will pause its current plan and return to Step 1.

This process continues until the agent achieves all of its goals, achieves all the goals it can complete, or it is instructed to stop by its supervisor.

3. Evaluation

Our empirical evaluation assesses the ability of the agent to respond to unexpected events and opportunities when using our hybrid goal selection and planning approach. Our experiments concern the following hypotheses:

H1: The agent will obtain reasonable mission performance when the supervisor's preferences are partially specified.

H2: The agent will achieve higher mission performance than if it attempted to achieve all goals.

H3: The agent will achieve higher mission performance that if it only attempted to achieve hard goals.

H4: The agent will achieve higher mission performance than if it performed a separate goal selection process.

3.1 Domain

Our experiments use a simulated human-agent teaming domain involving one agent and one supervisor. Each mission operates as follows:

- *Initial Interactions*: At the start of each mission, the agent and supervisor have an initial interaction where the supervisor provides the agent with an initial hard goal and partially provides its preferences.
- *Autonomous Behavior*: After the initial interaction, the agent generates a plan to achieve its initially provided goal and executes that plan. While acting autonomously, it may encounter unexpected events that cause it to generate new goals and replan (i.e., as described in Section 2.3). It does not interact with the supervisor again until the mission is complete.
- *Debriefing*: After completing its selected goals, the agent performs a debriefing with its supervisor.

The environment is represented as a 2-dimensional grid that the agent can move and act in, with various obstacles in its path. There are four factors of interest used for preferences (and encoded in goal types):

- **Time**: The importance of completing the mission quickly.
- **Agent Safety**: The importance of the agent remaining safe and avoiding any damage while completing the mission.
- **Information Retrieval**: The importance of collecting relevant information during the mission.
- **Humanitarian Assistance**: The importance of helping distressed individuals while completing the mission.

During the *Autonomous Behavior* part of the mission, several unexpected events or opportunities can occur than may provide the agent with new goals. They are:

- **Dangerous Device**: The agent learns of a dangerous device at a location on the map. This can result in the agent creating a new *device disposal* goal that will negatively impact time, but positively impact agent safety and humanitarian assistance (i.e., the device will not hurt the agent or others). It has no impact on information retrieval. The device disposal goal type is: $\tau_{dd} = \langle "device\ disposal", -1, 1, 0, 1 \rangle$.
- **Knowledgeable Actor**: The agent learns of an actor at a specified location that has potentially valuable information. This can result in a new *conversation* goal that will negatively impact time and positively impact information retrieval. It has no impact on humanitarian assistance or agent safety. The conversation goal type is: $\tau_c = \langle "conversation", -1, 0, 1, 0 \rangle$.
- **Suspicious Location**: The agent learns of a location that seems suspicious. This can result in a new *investigation* goal that will negatively impact time and positively impact information retrieval (e.g., if there is important information there) and humanitarian assistance (e.g., if the agent learns information that can keep others safe). It has no impact on agent safety. The investigation goal type is: $\tau_i = \langle "investigation", -1, 0, 1, 1 \rangle$.
- **Damaged Path**: The agent comes across damage to its path that will make it more difficult to navigate. This can result in a *detour* goal that will negatively impact time but positively impact agent safety (e.g., it will not be injured by debris). It has no impact on humanitarian assistance or information retrieval. The detour goal type is: $\tau_d = \langle "detour", -1, 1, 0, 0 \rangle$.

3.2 Experimental Conditions

Our evaluation involves a number of experimental trials, and each trial uses randomly selected initial conditions (all random values are selected using a uniform distribution). At the start of each trial, the environment map is randomly populated with obstacles and the robot is placed at a random initial location (from among the non-obstructed locations on the map). Additionally, the supervisor's true preferences are randomly generated. During their initial interaction, the supervisor randomly selects an initial goal location for the agent to navigate to and randomly selects between 0 and 4 (inclusive) preferences to provide to the agent. For the preferences that are provided to the agent, the values are discretized from their true values to *LOW* (< 0.25), *MEDIUM* (≥ 0.25 and ≤ 0.75), or *HIGH* (> 0.75). A list of eight random events are created for use during the trial. The random events are created before the trial to ensure that all variants of the agent that we test use identical trial conditions (i.e., map, initial goal, provided preferences, and encountered events).

The four variants of our agent that we use are:

- **Hybrid**: Uses the hybrid goal selection and planning approach that we present in this paper. The PSP planner that we use is SapaReplan (Talamadupula et al. 2010). This variant of the agent can dynamically add new goals, prioritize supervisor-provided goals, and estimate goal utilities using partially specified preferences.
- **Initial Only:** The agent only attempts to achieve the initial goal provided by its supervisor. It treats that goal as a hard goal and ignores all others. This variant does not reason about or modify its goals.
- **All Hard:** The agent treats all goals as hard goals and adds a new goal after each external event occurs. Thus, the agent does not differentiate between supervisor-provided goals and self-generated goals. Additionally, the agent does not attempt to estimate goal utility, so a new goal is added whenever an unexpected event occurs.
- **Only High Utility:** The agent performs a separate goal selection process to filter out any goals that are not expected to have a high utility based on the supervisor's provided preferences. The remaining goals are treated as hard goals. This represents a Goal Reasoning agent that separates goal selection and planning using a cautious goal selection process.

During a trial, each agent acts in the environment, encounters unexpected events, responds to the unexpected events, and may add new goals (depending on the agent variant). A trial terminates when the agent completes all of the goals it is attempting to achieve or when it fails (i.e., it cannot generate a plan to achieve its remaining goals). After a trial concludes, the performance of the agent is measured as the sum of the true utility of all the soft goals it achieved. The utility function is similar to the one used by the agent to estimate goal utilities, but differs in that it uses the supervisor's true preferences rather than provided preferences: $utility = C_1 \times f_1 \times p_1 + \cdots + C_n \times f_n \times p_n$. Additionally, the evaluation of each agent's performance includes a penalty for each action it performed (identical penalty for all actions) and a penalty if the agent fails to achieve the initial hard goal provided by its supervisor (i.e., if it failed to generate a valid plan). Thus, the best performing variant will be the one that maximizes this metric by completing its hard goal and the highest utility soft goals in as few actions as possible. Each agent variant performed a total of 1000 experimental trials.

3.3 Results

Figure 1 shows the cumulative performance of the four variants over 1000 experimental trials. The results show that our hybrid approach for goal selection and planning achieves the best results, showing a steady increase in performance and outperforming the other three variants, providing support for **H1**. Similarly, *Only High Utility* has performance that increases over time, but at a slower rate than *Hybrid*. The primary reason for this is that *Only High Utility* is overly cautious when selecting goals and only selects goals it knows will have high utility. Since *Hybrid* performs goal selection during planning, it is able to more accurately determine the cost of achieving goals and achieve goals with less certain utility if they are low-cost (e.g., a lower-utility goal is nearby a high-utility goal and can be inexpensively achieved). *Initial Only* shows a slow decrease in performance. This is expected since a cumulative performance of 0 would result if the agent achieved only the hard goals at no cost (i.e., without any actions), whereas *Initial Only* achieved only the hard goals but required actions to do so. The largest variance in performance is from *All Hard*. This variant tended to result in either high-performance or low-performance trials. High performance trials occurred when all goals were achievable and most had a positive utility. Low-performance trials occurred when there was no plan that could achieve all goals or many goals had a negative utility.

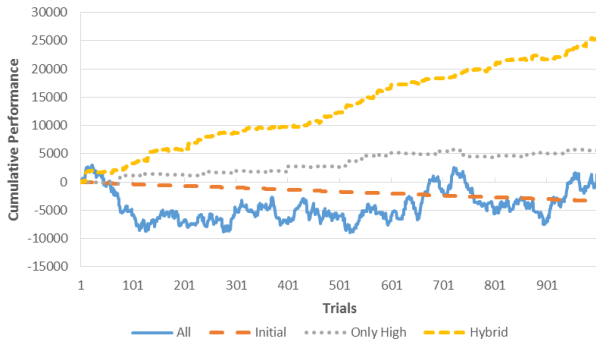

Figure 1: The cumulative performance of each variant over 1000 trials

We repeated our experiment 25 times (i.e., 25 experiments of 1000 trials) and compared the mean performance of each variant across those experiments. Over all experiments, *Hybrid* was a statistically significant improvement over all other variants (using a single-tailed t-test with $p < 0.01$). These results were consistent with our initial experiment (i.e., Figure 1), with *Hybrid* having the highest mean performance (30,693), followed by *Only High Utility* (4,136), *All Hard* (2,375), and *Initial Only* (−3,442). This provides support for **H2**, **H3**, and **H4**.

4. Related Work

The inclusion of Goal Reasoning agents as members of human-agent (or human-robot) teams has seen increased interest in recent years as a result of GR agents' ability to intelligently respond to dynamic environments and provide explanations for their changing behavior (Molineaux et al. 2018). The Autonomous Squad Member (ASM) agent recognizes the plans and goals of its teammates and modifies its own goal in order to align its behavior with its teammates (Gillespie et al. 2015). This is similar to our work in that the agent implicitly estimates when teammates' preferences change over time (i.e., resulting in changing goals). However, the ASM agent requires its teammates to be visible to it so it can observe their actions. The Tactical Battle Manager (TBM) represents goals as the degree to which environment states satisfy high-level desires (Floyd et al. 2017). The preferences of teammates can be specified during a pre-mission briefing and used as the desires the agent pursues. However, this differs from our own work in that it requires fully specified preferences. Additionally, both the ASM and TBM agents only pursue a single goal.

Goal Reasoning agent design frameworks, like the Goal Lifecycle (Roberts et al. 2014), allow agents to manage and pursue multiple concurrent goals but, like the ASM and TBM agents, existing GR agents pursue only a single goal at a time. Goal motivators (Wilson, Molineaux and Aha 2013) base goal selection on a set of criteria to evaluate each goal. The urgency of each motivator (i.e., how important the motivator is at the current time) and its fitness (i.e., how well the future states will satisfy the motivator) are used to select a single goal with the highest overall fitness. This is similar to our work in that it uses a metric to quantify the value of each goal. However, it differs in that it performs goal selection before planning (i.e., does not consider the interdependency of goals), only selects a single goal to pursue, requires more complete knowledge of a teammate's preferences, and requires pre-defined knowledge to tune the fitness calculation.

Considering a user's preferences during planning is central to preference-based planning (Baier and McIlraith 2008), and the ability to define preferences has been included in recent versions of the Planning Domain Definition Language (PDDL). Preference-based planning allows for both hard and soft preferences to be encoded and used during planning. However, these are preferences for which plan to select for achieving a specific goal, rather than preferences for both goal selection and plan generation. Preferences can be provided for goals (Brafman and Chernyavsky 2005), but that work focuses more on how the planner can modify a goal if it cannot be achieved (i.e., preferences over acceptable states that are similar to the goal state), rather than providing preferences over a set of distinct goals. Similarly, the preferences we use are less explicit than

those used in PDDL (e.g., representing preferences as goal descriptors).

As we mentioned previously, Partial Satisfaction Planning (van den Briel et al. 2004; Benton, Do, and Kambhampati 2009) has the advantage of combining goal selection and planning into a single process. However, to the best of our knowledge there have not been any uses of PSP where an agent is directly responsible for creating its own goals, dynamically computing the utility of its goals, or incorporating a teammate's partially specified preferences into goal selection. Goal selection and replanning using PSP in response to a changing environment has been performed in search-and-rescue scenarios (Talamadupula et al. 2011). This is similar to our work in that it uses PSP to perform goal selection and replan, but differs in that the human teammate is always in the loop and explicitly provides the agent with its goals (and other updated information). Thus, in addition to all of its goals being externally provided, the agent does not need to estimate its teammate's preferences or the utility of goals.

5. Conclusion

We described a hybrid approach for goal selection and planning in Goal Reasoning agents that are members of human-agent teams. Our approach uses a supervisor's partially specified preferences to estimate the utility of goals, prioritizes goals based on their source (i.e., provided by the supervisor or formulated by the agent), and generates a plan to achieve the subset of goals with the highest expected utility. In our empirical study in a simulated human-agent teaming domain, our approach demonstrated strong mission performance and outperformed three variants. Even though the supervisor's preferences were partially specified and discretized (i.e., on average only half the preferences were provided), the agent's mission performance aligned closely with how the supervisor evaluated the agent.

Several areas of future work remain. First, our work was focused on the *Single Supervisor* team composition, where the agent is teamed with a supervisory agent. In future work, we plan to extend our goal utility estimation and prioritization to allow for teams of arbitrary composition. This will allow the agent to consider the preferences of multiple teammates concurrently and operate as a member of larger teams. For example, the agent may have a supervisor, several peers, and several subordinates. Different weights will be necessary for the preferences of each of these types of teammates and the relative priority of any goals they request. Additionally, we considered only a supervisor-supervisee relationship where the supervisor's preferences are the only ones considered. Future work will investigate the ability of the agent to consider its own preferences and motivations when selecting goals, and allow the agent to generate its own hard goals. This will be important for teams where the agent is a supervisor or peer of other teammates, rather than a subordinate. We will also examine the agent's ability to learn a teammate's true preferences over time. Our evaluations used different preferences for each run, but in a real team there would likely be less variance in preferences between runs. By combining partial preferences between runs and using the supervisor's evaluation of the agent, the agent could learn a better estimate of the supervisor's true preferences and improve its utility estimation. This would also allow the agent to detect changes in the supervisor's preferences over time and ask questions about why the changes occurred.

Acknowledgements

Thanks to the Office of Naval Research for supporting this research.

References

Aha, D.W., Cox, M.T., and Muñoz-Avila, H. (Eds.). 2013. *Goal Reasoning: Papers from the ACS Workshop (Technical Report CS-TR-5029)*. University of Maryland, Department of Computer Science.

Baier, J.A., and McIlraith, S.A. 2008. Planning with preferences. *AI Magazine*, 29(4): 25-36.

Benton, J., Do, M., and Kambhampati, S. 2009. Anytime heuristic search for partial satisfaction planning. *Artificial Intelligence*, 173(5-6): 562-592.

Brafman, R.I., and Chernyavsky, Y. 2005. Planning with goal preferences and constraints. In *Proceedings of the Fifteenth International Conference on Automated Planning and Scheduling*, 182-191. AAAI Press.

Coman, A.; Gillespie, K.; and Muñoz-Avila, H. 2015. Case-based local and global percept processing for rebel agents. In *Case-Based Agents: Papers from the ICCBR 2015 Workshops*, 23–32.

Floyd, M.W., Karneeb, J., Moore, P., and Aha, D.W. 2017. A Goal Reasoning agent for controlling UAVs in beyond-visual-range air combat. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 4714-4721. AAAI Press.

Gillespie, K., Molineaux, M., Floyd, M.W., Vattam, S.S., and Aha, D.W. 2015. Goal reasoning for an autonomous squad member. In *Goal Reasoning: Papers from the ACS Workshop*, 52-67.

Molineaux, M., Floyd, M.W., Dannenhauer, D., and Aha, D.W. 2018. Human-agent teaming as a common problem for Goal Reasoning. AAAI Symposium on Integrating Representation, Reasoning, Learning, and Execution for Goal Directed Autonomy.

Roberts, M., Vattam, S.S., Alford, R., Auslander, B., Karneeb, J., Molineaux, M., Apker, T., Wilson, M., McMahon, J., and Aha, D.W. 2014. Iterative goal refinement for robotics. In *Planning and Robotics: Papers from the ICAPS Workshop*.

Talamadupula, K., Benton, J., Kambhampati, S., Schermerhorn, P.W., and Scheutz, M. 2010. Planning for human-robot teaming in

open worlds. *ACM Transactions on Intelligent Systems and Technology*, 1(2): 14:1-14:24.

Talamadupula, K., Schermerhorn, P., Benton, J., Kambhampati, S., and Scheutz, M. 2011. Planning for agents with changing goals. In *Proceedings of the International Conference on Automated Planning and Scheduling Systems Demos and Exhibits*.

van den Briel, M., Sanchez Nigenda, R., Do, M.B., and Kambhampati, S. 2004. Effective Approaches for Partial Satisfaction (Over-Subscription) Planning. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence*, 562-569. San Jose, USA: AAAI Press.

Wilson, M., Molineaux, M., and Aha, D.W. 2013. Domain-independent heuristics for goal formulation. In *Proceedings of the Twenty-Sixth Florida Artificial Intelligence Research Society Conference*, 160-165. AAAI Press.