Proceedings of the Sixteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE-20)

Narrative Planning for Belief and Intention Recognition

Rachelyn Farrell, Stephen G. Ware
Narrative Intelligence Lab, University of Kentucky
Lexington, KY 40506
rachelyn.farrell@uky.edu, sgware@cs.uky.edu

Abstract

Planning algorithms generate sequences of actions that achieve a goal, but they can also be used in reverse: to infer the goals that led to a sequence of actions. Traditional plan-based goal recognition assumes agents are rational and the environment is fully observable. Recent narrative planning models represent agents as *believable* rather than perfectly rational, meaning their actions need to be justified by their goals, but they may act in ways that are not optimal, and they may possess incorrect beliefs about the environment. In this work we propose a technique for inferring the goals and beliefs of agents in this context, where rationality and omniscience are not assumed. We present two evaluations that investigate the effectiveness of this approach. The first uses partial observation sequences and shows how this impacts the algorithm's accuracy. The second uses human data and compares the algorithm's inferences to those made by humans.

Introduction

Planning is finding a sequence of actions for an agent to achieve a goal. Plan recognition is the opposite problem—given a partial sequence of actions we have observed an agent taking, infer the unobserved actions. Goal recognition is closely related—given a partial sequence of actions and some candidate goals, infer which is the agent's true goal.

Ramírez and Geffner's Plan Recognition as Planning (PRP) framework (2009) showed planning algorithms can be used to solve all three problems. For plan and goal recognition they constrain a planner so that it must include all the agent's observed actions in its solutions. Then they generate one or more solutions, thus filling in any missing actions. Assuming the agent is rational, more optimal plans are more likely to be the agent's plan, and goal candidates that have more optimal solutions are more likely to be the agent's goal.

PRP demonstrates how unobservable qualities of an agent's mind, such as its goals and plans, can be automatically inferred through the agent's actions in a plan-based environment. PRP is designed for inferences about rational agents and assumes full observability, but we often want to reason about the minds of more human-like agents who are not perfectly rational and have limited knowledge. In this paper, we apply narrative planning algorithms to infer the beliefs and intentions of agents who are constrained not by rationality, but by *believability* (Mateas 1999). We demonstrate our technique by inferring the beliefs and intentions of human users in a virtual reality police training simulation.

Narrative planning (Young et al. 2013) is a form of multi-agent BDI planning (Meneguzzi and De Silva 2015) often used to generate interactive stories. It focuses on agent believability over rationality or optimality. The system user, or "author", specifies a goal for the narrative (called the author's goal) and individual beliefs and intentions for each agent. A centralized planner finds a sequence of actions that achieves the author's goal using only actions that can be explained by the individual beliefs and intentions of the agents who take those actions. We apply PRP's central idea: given candidate sets of beliefs and intentions, the better we can build a plan that (1) contains all the observed actions and (2) explains the actions of every agent, the more likely those beliefs and intentions are to be correct. Put more simply, narrative plan recognition asks what beliefs and intentions would be required to explain the actions we observe agents taking.

This method assumes the recognizing algorithm has access to the same planning domain operators that generated the observed actions. Our approach is useful when agent rationality and omniscience cannot be assumed, like understanding and predicting human behavior in games and simulations where players do not form optimal plans, or to simulate believable agents inferring each other's plans. Reasoning about the minds of believable agents is challenging and interesting. For example, classical plan recognition is trivial if the agent's whole plan has been observed (there is nothing to be inferred). This is not so in our case. An agent may form a plan, execute part of it, then change it based on new information or fail to complete it due to a conflict with another agent. Their earlier actions may only be explained by considering hypothetical future actions they never took, so even knowing all executed actions does not guarantee accurate inferences. Furthermore, beliefs and intentions both contribute to explaining actions and may do so in conflicting ways, so the task of inferring them is often difficult even for humans.

We give results from two evaluations. The first is artificial and assumes ground truth is known and that the domain simulates the world perfectly. It demonstrates accuracy with different numbers of observations. The second uses real player data and compares our automated process to how humans recognize beliefs and intentions.

Related Work

Planning algorithms were originally developed for rational agent problem solving and coordination (Fikes and Nilsson 1972). Young (1999) noted that plans are an ideal representation of narratives because they provide a formal, generative computational model that captures the temporal and causal relationships between events. Since then, a branch of research has coalesced around using planning algorithms to model how people behave and reason about event sequences (Young et al. 2013). These algorithms include computational models of important narrative phenomena such as time (Sacerdoti 1975), causality (McAllester and Rosenblitt 1991), intentionality (Riedl and Young 2010), conflict (Ware et al. 2014), and belief (Teutenberg and Porteous 2015).

Our work is based on a model of narrative panning with belief and intentionality (Shirvani, Farrell, and Ware 2018) and the PRP framework (Ramírez and Geffner 2009). The relevant pieces of these models are described in detail in the next section. The PRP framework has been extended for many specific purposes relevant to games and interactive narrative (Baikadi et al. 2013; Le Guillarme et al. 2015; Sohrabi, Riabov, and Udrea 2016; Pereira and Meneguzzi 2018). Cardona-Rivera and Young (2015) demonstrated reasonable performance for a PRP-based plan recognition system that recognized player plans in an interactive narrative in real time. Although that system assumed players were playing optimally and did not attempt to recognize incorrect beliefs, we are optimistic that a similar approach could be taken to employ our framework in real-time scenarios.

Narrative Planning Framework

A classical AI planning problem defines the problem domain (a set of boolean propositions, a database of action templates, and a set of objects, or constants that can be bound to variables), as well as the initial state of the world and a goal to be achieved. A planner searches for a plan—a sequence of grounded actions that can be executed from the initial state to achieve a state where the goal is satisfied.

A narrative planner additionally defines a set of character agents and ensures that characters always have a reason for the actions they take. During planning, it builds causally linked chains of actions, called explanations, that achieve characters' goals (often called "intentions" to distinguish them from the problem goal) and that are possible according to that character's beliefs. If an action is part of any explanation for a character, that action is said to be *explained* for that character. In this section we formally describe the components of this framework that are necessary to introduce our belief and intention recognition model.

A narrative planning problem is $\langle P, A, O, C, s_0, g\rangle$ where P is a set of boolean propositional fluents, A is a set of action templates, O is a set of objects, C is a set of special constants representing characters who possess beliefs and intentions, s_0 is the initial state, and g is the goal. Each action $a \in A$ specifies preconditions $\text{PRE}(a)$ (which must hold before the action is executed), effects $\text{EFF}(a)$ (which become true after the action is executed), a set of acting characters $\text{ACT}(a) \subseteq C$ (for whom the action must be explained), and a set of observing characters $\text{OBS}(a) \subseteq C$ (whose beliefs of the action's effects are updated after it is executed).

Beliefs and intentions are represented as modal predicates: $b(c, p)$ means character c believes proposition p, and $i(c, p)$ means that c intends p. These can be nested, e.g. $b(c, b(d, p))$ means "c believes that character d believes p." They can also be combined, as in $b(c, i(d, b(c, p)))$, meaning "c believes that d intends for c to believe p."

Beliefs and intentions may be specified in the initial state. For any character c and proposition p for which there is not a belief explicitly stated, it is assumed that c believes the true value of p. Any intention not explicitly stated is assumed false. Note that $\neg b(c, p)$ is equivalent to $b(c, \neg p)$, but this is not the case for intentions: $\neg i(c, p)$ is NOT equivalent to $i(c, \neg p)$. The former means c has no intention for p to be true; the latter means c specifically intends for p to be false.

We use the notation $\alpha(a, s)$ to refer to the state that results from applying action a to state s. We use $\beta(c, s)$ to refer to the set of beliefs for character c in state s. Since this model commits each character to exactly one belief about every proposition in the domain, we can also see $\beta(c, s)$ as "the state c believes to be the case when the true state is s".

Intentions are updated through action effects. Beliefs are updated in two ways: when the character observes an executed action, and when the character's beliefs are explicitly updated by the effects of an executed action (regardless of whether they observe it). Formally, when an action a occurs:

- $\forall c \in \text{OBS}(a)$: $\beta(c, \alpha(a, s)) = \alpha(a, \beta(c, s))$
- $\forall c \notin \text{OBS}(a)$: $\beta(c, \alpha(a, s)) = \beta(c, s)$
- $\forall c \in C : \exists b(c, p) \in \text{EFF}(a) \implies p \in \beta(c, \alpha(a, s))$

Solutions to the narrative planning problem are constrained by beliefs and intentions according to the following definitions. For every action a in the solution, a must be *explained*:

Definition 1. An action a is explained iff $\forall c \in \text{ACT}(a)$: a is *explained for* c in the state before a.

Definition 2. An action a is *explained for* $c \in \text{ACT}(a)$ in state s iff there exists a sequence of actions π that starts with a and meets the following criteria when taken from $\beta(c, s)$:

1. There exists a proposition p that holds at the end of π and $\forall a' \in \pi : \neg p \wedge i(c, p)$ holds in the state before a'.
2. $\forall a' \in \pi : \text{PRE}(a')$ holds in the state before a'.
3. $\forall a' \neq a \in \pi : \forall c' \in \text{ACT}(a') : a'$ is explained for c' in the state before a'.
4. π contains no sub-sequence that also meets these criteria. (This enforces π's causal coherency; it cannot be used to explain a if any step is redundant or unnecessary.)[1]

[1] Shirvani, Farrell, and Ware (2018) accomplish this through the use of causal links and provide a definition that is specific to their implementation. We offer a simpler definition.

Belief and Intention Recognition

A classical plan recognition problem (Ramírez and Geffner 2009) begins with a planning problem, a set of possible goals, and a sequence of actions an agent has been observed taking. The plan recognizer finds the set of goals for which there is an optimal solution that satisfies the observations. Satisfying the observations means that all the observed actions appear in the same order, though there may be additional actions in between. The resulting goal set represents the most likely intentions of the agent taking the actions.

PRP first transforms the original planning problem by converting the observed actions into ordered landmark events, forcing the planner to include these actions in the specified order in all solutions. The transformed problem is then augmented with each of the possible goals, producing a new problem for each one. Each of these problems is then solved by the planner. The goal that produces the lowest cost optimal solution is likely to be the agent's true goal, and the solution itself is a likely estimate of the agent's true plan.

In this work we adapt the PRP strategy to recognize character beliefs in addition to goals, and without the assumptions of rationality and omniscience. We are looking for the set of character intentions and beliefs that can best explain the observed actions for all agents, using the belief and intentionality model described in the previous section.

Our method begins with a narrative planning problem, a set of possible candidates, and an observation sequence. A candidate contains both a set of beliefs and a set of intentions, each possibly empty. Note that the planning problem may include any additional intentions or beliefs that are already known, while others are being inferred.

1. **Transform the given problem.** For each observation, add a new action to the domain with the same preconditions and effects as the observed action, but fully grounded (e.g. $walk(A, R1, R2)$ will have the same preconditions and effects as the action $walk(?char, ?from, ?to)$, but with bindings $?char = A$, $?from = R1$, and $?to = R2$). Additionally, it has a new effect[2] uniquely identifying the observation. For all but the first observation, this effect is a conditional effect whose condition is the previous observation's unique effect. The unique effect of the final observation is added to the problem goal, thereby forcing solutions to contain all observations in order.
2. **Produce a new problem for each candidate.** Create a new problem with each candidate by adding its beliefs and intentions to the initial state of the transformed problem.

Here we deviate from the PRP algorithm. Since agents may not be rational, shorter plans are not necessarily better and we cannot assume that any candidate can perfectly explain the observations. Instead of seeking optimal solutions, we use the following procedure to identify the candidates that come *closest* to perfectly explaining all the observed actions.

3. **Generate classical solutions but track explanations.** For each *candidate problem* produced in the previous step, generate a set of *classical solutions* while tracking explanations with the narrative planner. A classical solution is any sequence that is possible and achieves the problem goal, regardless of whether actions are explained. Each classical solution represents a possible way the observed actions may have occurred, and its explanations can answer whether each action therein is explained.
4. **Identify valid candidates.** An ideal solution would explain 100% of the observed actions and contain no unexplained actions. (We do not penalize solutions for containing *explained* actions that were not part of the observations.) To account for the possibility that no candidate can find such a solution, first identify the *maximum number of explained observed actions* among the classical solutions for any candidate problem. Discard all solutions (for any candidate) that explain fewer than this number of observations. Next, identify the *minimum number of unexplained actions* among all remaining solutions, and similarly discard solutions with more unexplained actions than this.

At this point, any candidate with a nonempty solution set belongs to the set of valid candidates. Thus a candidate is valid if, and only if, no other candidate can explain more of the observations, or can explain the same number of observations while adding fewer actions that are unexplained. The output of our method is the set of valid candidates; a conservative answer to the inference question, representing all of the most plausible explanations for the observed behavior. Futher tie-breaking may be required, but we do not propose a particular method for doing so; we address our reasons for this in the Discussion section.

Theoretical Evaluation

Our first evaluation examines this technique with varying numbers of observations. We generate test sequences with known beliefs and intentions, then ablate the sequences using five different methods. For each ablated sequence, we test how accurately the algorithm identifies the true beliefs and intentions from among ten possible candidates. We expect higher accuracy when more observations are given.

The domain used for this evaluation contains 3 agents, 3 items, and 4 locations arranged in a 2x2 grid. Each agent wants to have one item, and loves one other agent (and therefore wants this agent to have the item that agent wants).

Agents can walk between rooms, pick up items (holding at most one at a time), put items down, give items to one another, and trade items. One agent can tell another what item they want (which can be a lie). Agents may want the same item and/or love the same agent. Everyone knows what they themselves want, but beliefs about what others want can be wrong. Whom everyone loves is known by all. Beliefs about the locations of items and agents can be wrong, and are updated by observation axioms: When an agent enters a room, they learn which agents and items are there. They also observe all actions that happen in the room they occupy.

[2]These effects must be ignored in causal links-based explanations, if applicable.

We designed this domain to simulate many of the qualities we encounter in narrative planning domains: movement with partial observability, relationships, wrong beliefs, learning, communication, deception, cooperation, and conflict.

We generated 20 different initial states in this domain, randomizing whom every agent loves and the initial locations of all agents and items. For each initial state, we created 10 candidates with randomly selected intentions (what each agent wants) and wrong beliefs. We then used each candidate to find one valid solution, where the problem goal is for two randomly selected agents to achieve their goals. Solutions were limited to 5 steps; if a candidate could not produce a solution with 5 steps[3] or fewer, the candidate was discarded and replaced with a new randomized candidate.

This resulted in 10 $\langle candidate, sequence\rangle$ pairs for each of the 20 initial states (200 sequences in all). We ablated each of these sequences using five methods. The first three methods kept the beginning of the sequence (100%, 66%, or 33% of the actions), removing the rest. The fourth method kept only the actions that occurred in a single room, and the fifth kept only those that occurred in the presence of a single agent. Our final data set contained 1000 ablated sequences for which the ground truth candidate is known. We later discarded results for which the ablated sequence contained zero steps (145 out of 1000, leaving 855 sequences for analysis).

With each ablated sequence as input observations, we used our algorithm to compare ten possible candidates: the same ten that were used to generate starting sequences for that initial state. Since we are guaranteed that at least one can explain all the observed actions without adding any unexplained actions (the one that originally generated the sequence), we accomplished this simply by searching for a solution to each of the other 9 candidate problems. Each candidate belongs in the set of valid candidates if and only if at least one solution can be found within the limit of 5 steps.

Results

The purpose of this analysis was to test how accurately the algorithm distinguishes the correct candidate from 9 others, and to show accuracy improving as more actions are observed. By higher accuracy we mean a smaller set of valid candidates: the most accurate trials are ones in which the candidate that generated the solution is the only one in the valid set, which was the case for 242 out of the 855 sequences. The least accurate are those in which all 10 candidates were considered valid, which occurred only 22 times.

It is worth noting that perfect accuracy is not necessarily desirable. It is reasonable that sometimes other candidates are able to explain the observations just as well as the one that generated the sequence, especially when fewer observations are given. In fact, one benefit of this method is its ability to find multiple plausible explanations for the given behavior. Different applications would use this information in different ways, e.g. to search for explanations that meet certain criteria, so it is important that the valid set contain all candidates it finds to be equally likely.

Figure 1 shows the average size of the valid candidate set for trials with each possible number of observations. This is shown for each of the five ablation methods as well as for all of them combined. Note that two lines appear to be incomplete; this is due to the ablation method. For example, the First 33% method always removes 67% of the observed actions, of which there can be up to 5, so this method can never create a sequence with more than 2 observations.

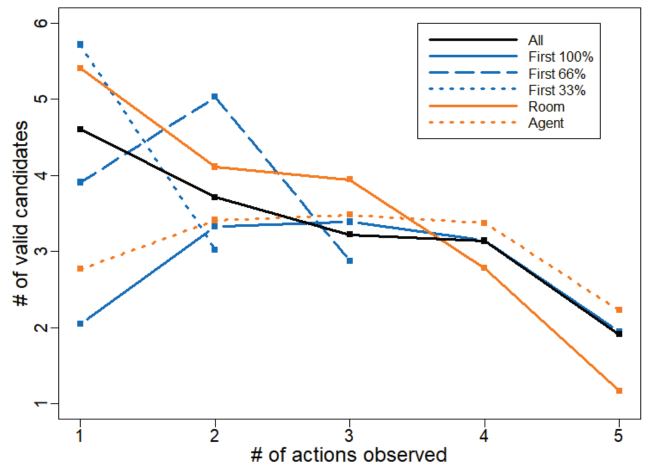

Figure 1: Valid candidates by number of observations

We can see that around 2 candidates were identified on average when the full sequence was observed. Furthermore, even when only one action was observed, the algorithm usually rejected more than half the candidates. We can also verify that in general, the more observations given, the better the algorithm can hone in on just a few candidates.

It is interesting that for some ablation methods—First 100%, First 66%, and Agent—the algorithm was more accurate for 1-step sequences than for 2-step ones. We suspect this is due to the simplicity of those sequences, but more investigation is required to fully explain this.

Also note that even with 100% of the observations present, this does not necessarily account for agents' entire plans, since their plans may have contained actions that were intended but not executed. On average, the ablated sequences represented about 71% of the executed actions, and about 69% of the full plans (including non-executed actions) required to explain the sequence.

Practical Evaluation

Our second evaluation compares the algorithm's inferences to those made by humans performing the same task. For this experiment we did not generate the test sequences ourselves, so the ground truth beliefs and intentions were not known. We began with 33 different sequences generated by humans playing a short simulation. We displayed these sequences to a group of trained raters and asked them to identify the player's beliefs and intentions. We then compared their responses to those produced by the algorithm.

[3]The 5 step limit was imposed because the planner often fails, and a failed complete search of all possible sequences and explanations may require visiting tens of thousands of nodes. These 10,000 plan recognition problems represent about 3 weeks of CPU time.

Since the sequences were generated by humans, we could not assume that they could be fully explained by any candidate, as we could in the previous experiment. Therefore we were required to use the full method involving generating a set of classical solutions for each candidate and tracking the explained and unexplained actions.

For this evaluation we utilized a short police training simulation developed by Garcia, Ware, and Baker (2019). It uses a narrative planning backend, keeps action-level gameplay logs, and explicitly models multiple incorrect beliefs and alternative goals in order to teach certain rules advocated by the Police Executive Research Forum.

The simulation (depicted in Figure 2) was designed to teach the concept "distance + cover = time", which states that by keeping distance and cover between the officer and the suspect, the officer can buy more time for a peaceful resolution (Police Executive Research Forum 2012). The simulation rewards the player (the officer) for achieving the peaceful resolution and not killing the suspect.

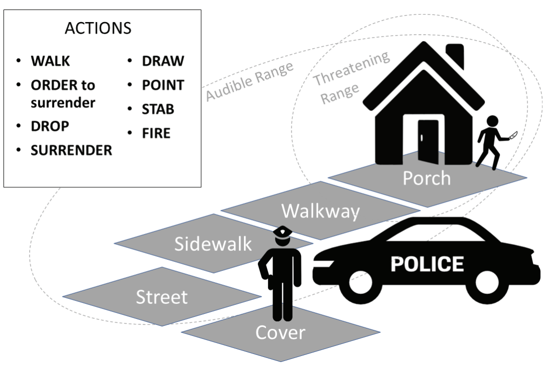

Figure 2: Police Use of Force Domain

In the simulation, a man referred to as the suspect is on a porch wielding a knife. The officer can order him to surrender, which makes him want to surrender if he is within audible range. However the suspect will not surrender as long as it is possible to attack the officer, i.e. not until the officer backs up and uses the police car as cover. The following facts about the domain are not explicitly stated and must be revealed by playing:

- $cover$: the location behind the police car is safe (the officer cannot be harmed there)
- $distance$: approaching makes the suspect want to attack
- $point$: pointing the gun makes the suspect want to attack

These are facts players may not have known, so we use their negation as possible beliefs for inference. For example, the belief $\neg point$ means the player does not realize that pointing the gun at the suspect makes him want to attack. Players may have possessed any number of these beliefs, including none (if they fully understood the domain). Our set of possible belief combinations is the power set of the three beliefs plus the empty set (8 sets of beliefs in total).

The best ending is for the suspect to surrender without ever threatening the officer's life. The simulation rewards the officer for achieving this by displaying a score based on the following three properties:

- $surrender$: suspect has surrendered (ends the simulation)
- $threatened$: suspect has raised his knife at the officer
- $kill$: officer has killed the suspect (ends the simulation)

We use these as the basis for the set of possible intentions. We assume players were either trying to achieve $surrender$ or $kill$, since these are the only two ways to end the scenario without the officer dying. Additionally, players may have wanted $threatened$ to be either true or false (meaning they explicitly wanted to be threatened, or to avoid it, respectively), or may have had no intention related to that property. Therefore there are 6 possible intention combinations.

Recall that a candidate has two components: a set of intentions and a set of beliefs. Our full set of possible candidates is the combination of each possible intention set (6) with each possible belief set (8); a total of 48 candidates.

We obtained gameplay logs from a previous experiment in which participants (mostly students and university staff) played the simulation between two and ten times each, thus displaying varying amounts of domain knowledge. We recreated first-person play-through videos from all available logs of sequences with 8 or fewer steps (33 logs).

To establish the correct answers according to humans, we first described the domain in detail to three raters and ensured that they understood how the simulation worked. Each rater then watched the 33 videos conveying different observation sequences. After each video, they answered six questions: For each of the three goals, they answered whether the player possessed that goal (e.g. "the officer wanted the suspect to surrender"), the negation of that goal (e.g. "the officer wanted the suspect NOT to surrender"), or neither ("the officer did not care whether or not the suspect surrendered"). Similarly, for each of the three beliefs, they answered whether the officer believed it to be true, believed it to be false, or did not indicate a belief about it.

We expected that when the majority of raters agreed on the officer's beliefs or goals, our algorithm would produce the same answer. We compared all 48 candidates for each of the 33 sequences using our method. To obtain the classical solutions for each candidate problem we performed a complete breadth-first search to a depth of 8. (An alternative and faster approach would be to use a heuristic search and find as many classical solutions as possible given a time or memory limit. We opted for a depth limit to guarantee that we found *all* the classical solutions within the specified length.)

Results

We observed moderate agreement between the three raters overall (Krippendorff's $\alpha = 0.5460$). We compared our results to the raters' responses in two ways. First we used only the sequences for which at least 2 raters agreed on a value for all six features (i.e. each intention and belief is agreed to be either true, false, or absent/unknown; identifying a single candidate as the correct answer). This happened 12 times out of 33. Considering our algorithm to be correct if that candidate is among the set of valid candidates returned for that sequence, the algorithm was correct 8 out of 12 times (66.7% accuracy). As a baseline for comparison, consider a randomized approach to the same task: The average size of the candidate set returned by the algorithm was 8. If we were to randomly select a set of 8 candidates from the 48 possible, we would have about a 16.7% chance of succeeding.

Our second comparison considers all 33 sequences, not just the ones where raters agreed on all features. This time we determined the *set* of candidates deemed plausible by the raters, using the following procedure: For each feature (belief or intention) that the majority of raters agreed on, we discarded all candidates that did not have that value for that feature. When raters either did not agree on a feature, or agreed that the feature did not matter, we allowed candidates with any value for that feature to remain in the set. The average size of these plausible candidate sets was 4. In this case, we considered our algorithm to be correct if the majority of the candidates in this set are among those it returns, which happened 20 out of 33 times (60.6% accuracy). By comparison, a random selection of candidates would have only about a 1.2% chance of succeeding at this task.

The algorithm was generally successful for "straightforward" sequences in which the officer pursued a goal and achieved it without taking extra actions. It was also successful for some of the more complex sequences like the following, in which the officer approaches the suspect (which makes him start trying to attack the officer) then draws the gun, walks around a bit more, and eventually shoots:

> 1) Officer walks to Street. 2) Officer walks to Sidewalk. 3) Officer walks to Walkway. 4) Officer draws. 5) Officer walks to Sidewalk. 6) Officer walks to Walkway. 7) Officer points gun at Suspect. 8) Officer fires gun.

The algorithm identified a set of four valid candidates for this sequence, all of which contained the intention $surrender$ and the wrong beliefs $\neg distance$ and $\neg cover$. This means the officer wanted the suspect to surrender but did not realize that keeping distance and finding cover would help. Different values for the $threatened$ intention and the $point$ belief appeared in the valid set, meaning the algorithm made no claim about these. The raters were more specific: they identified the candidate that also included the intention $\neg threatened$ and the belief $\neg point$. Since that candidate was one of the four in the set, we say our result was consistent with their answer (though our algorithm considered three other candidates equally likely).

We find these results somewhat encouraging, but we believe higher accuracy should have been attainable. The problem may lie in how the simulation and the experiment were constructed, not with the algorithm or the concept itself.

One complication is revealed by sequences in which the player ordered the suspect to surrender multiple times, perhaps simply because the simulation does not give clear feedback when that action is executed. The raters agreed that these players wanted the suspect to surrender, even for sequences that ended with the officer killing the suspect (e.g. because the player didn't know that cover was safe, and could see no other way to end the scenario.) However, the domain assumes the effect of the action is observed, so a second *order* action cannot be explained (because the suspect already intends to surrender). As a result, candidates with the intention $kill$ were often able to explain more actions in these situations than those with the intention $surrender$.

There were also limitations to the experimental procedure. Much of the error may be due to the small number of raters, and the algorithm failing to identify features that were only accidentally agreed upon. Additionally, we allowed sequences of length 8 when our search limit was also 8. This constrains the planner to only finding one classical solution for those sequences—the observed sequence itself. Although the algorithm still behaves correctly in these cases, we expect better performance when we have a larger set of classical solutions to evaluate. We believe we would have achieved higher accuracy had we searched one depth higher than the maximum sequence length, but due to time constraints we were unable to complete that search.

Discussion

We have presented an adaptation of the PRP formalism for inferring the beliefs and intentions of believable (not perfectly rational) agents in a partially observable environment, based on observations of their actions. We accomplish this using a narrative planning framework that explicitly tracks agent beliefs and intentions and uses them to determine whether actions make sense for the agents who take them.

As mentioned earlier, our algorithm produces a set of plausible candidates as its final answer if there is not a single best explanation for the observed behavior. Some applications will require a single candidate to be identified, but we do not propose a specific tie-breaking method here because this decision is likely to be system-dependent.

For example, one might use an "Occam's razor" approach and say the simplest plausible explanation (e.g. with the fewest wrong beliefs and/or the fewest intentions) is most likely. On the other hand, for a tutoring system it may be best to assume players have the *most* possible wrong beliefs, so as to avoid skipping material they have not yet learned. In the police domain, it may be best to assume the officer had good intentions (wanted the suspect to surrender), and to only conclude otherwise if there is no plausible candidate that would allow this. Alternatively, when simulating one agent's inferences about other agents, we might base this decision on the biases of the agent doing the inference; e.g. tending to assume others are working for or against them.

Two preliminary evaluations demonstrate that our technique can be successful even with very few observations, that its accuracy improves as more information is provided, and that it is possible to use the technique to infer the beliefs and intentions of humans. Although the results presented in the second evaluation were somewhat underwhelming, we believe this was due to the limitations of the experiment and not the algorithm itself. Both evaluations were also limited by time constraints, particularly because we were using breadth-first search for completeness. Ideally we would scale up both experiments with more sequences and a higher search depth, and the latter with more participants. However our results do suggest that the technique is promising, and we believe it could be employed effectively in various types of planning-based interactive narrative scenarios.

Acknowledgement

This work was funded in part by the Department of Defense.

References

Baikadi, A.; Rowe, J. P.; Mott, B. W.; and Lester, J. C. 2013. Improving goal recognition in interactive narratives with models of narrative discovery events. In *Intelligent Narrative Technologies*.

Cardona-Rivera, R. E., and Young, R. M. 2015. Symbolic plan recognition in interactive narrative environments. In *Intelligent Narrative Technologies and Social Believability in Games*.

Fikes, R. E., and Nilsson, N. J. 1972. STRIPS: a new approach to the application of theorem proving to problem solving. *Artificial Intelligence* 2(3):189–208.

Garcia, E. T.; Ware, S. G.; and Baker, L. J. 2019. Measuring presence and performance in an intelligent virtual reality police use of force training simulation prototype. In *Proceedings of the 32nd AAAI international conference of the Florida Artificial Intelligence Research Society*, 276–281.

Le Guillarme, N.; Mouaddib, A.-I.; Lerouvreur, X.; and Gatepaille, S. 2015. A generative game-theoretic framework for adversarial plan recognition. In *10es Journées Francophones sur la Planification, la Décision et l'Apprentissage (JFPDA 2015)*.

Mateas, M. 1999. *An Oz-Centric Review of Interactive Drama and Believable Agents*. Berlin, Heidelberg: Springer-Verlag. 297–328.

McAllester, D., and Rosenblitt, D. 1991. Systematic nonlinear planning. Technical report, Massachusetts Institute of Technology Artificial Intelligence Laboratory.

Meneguzzi, F., and De Silva, L. 2015. Planning in bdi agents: a survey of the integration of planning algorithms and agent reasoning. *Knowledge Engineering Review* 30(1):1–44.

Pereira, R., and Meneguzzi, F. 2018. Goal recognition in incomplete domain models. In *Proceedings of the Thirty-Second Association for the Advancement of Artificial Intelligence (AAAI)*, 8127–8128.

Police Executive Research Forum. 2012. An integrated approach to de-escalation and minimizing use of force. In *Critical Issues in Policing Series*.

Ramírez, M., and Geffner, H. 2009. Plan recognition as planning. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*.

Riedl, M. O., and Young, R. M. 2010. Narrative planning: balancing plot and character. *Journal of Artificial Intelligence Research* 39(1):217–268.

Sacerdoti, E. D. 1975. The nonlinear nature of plans. Technical report, Stanford Research Institute.

Shirvani, A.; Farrell, R.; and Ware, S. G. 2018. Combining intentionality and belief: revisiting believable character plans. In *Proceedings of the 14th AAAI international conference on Artificial Intelligence and Interactive Digital Entertainment*, 222–228.

Sohrabi, S.; Riabov, A. V.; and Udrea, O. 2016. Plan recognition as planning revisited. In *IJCAI*, 3258–3264.

Teutenberg, J., and Porteous, J. 2015. Incorporating global and local knowledge in intentional narrative planning. In *Proceedings of the 2015 international conference on Autonomous Agents and Multiagent Systems*, 1539–1546.

Ware, S. G.; Young, R. M.; Harrison, B.; and Roberts, D. L. 2014. A computational model of narrative conflict at the fabula level. *IEEE Transactions on Computational Intelligence and Artificial Intelligence in Games* 6(3):271–288.

Young, R. M.; Ware, S. G.; Cassell, B. A.; and Robertson, J. 2013. Plans and planning in narrative generation: a review of plan-based approaches to the generation of story, discourse and interactivity in narratives. *Sprache und Datenverarbeitung, Special Issue on Formal and Computational Models of Narrative* 37(1-2):41–64.

Young, R. M. 1999. Notes on the use of plan structures in the creation of interactive plot. In *Proceedings of the AAAI Fall Symposium on Narrative Intelligence*, 164–167.